A STUDY OF THE APPLICATIONS OF A SPRING-ASSISTED MODULAR AND RECONFIGURABLE ROBOT FOR SAFE ROBOT OPERATION

by

Harshita Patel

Bachelor of Engineering, Ryerson University (2014)

A thesis

presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Master of Applied Science

in the Program of

Aerospace Engineering

Toronto, Ontario, Canada, 2017

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my thesis may be made electronically available to the public.

A STUDY OF THE APPLICATIONS OF A SPRING-ASSISTED MODULAR AND RECONFIGURABLE ROBOT FOR SAFE ROBOT OPERATION

Harshita Patel

Master of Applied Science, Aerospace Engineering, Ryerson University (2017)

Abstract

A novel spring-assisted modular and reconfigurable robot (SA-MRR) has been recently developed at our laboratory to reinforce its performance, and to enable safe and dexterous operations in human environments. A power spring is inserted between the brake rotor and the motor shaft through a decoupling bearing. With the spring engaged, the working range of the joint is mechanically limited for safe operations, and such a limited working range can be established at any joint position. The safety aspect of the SA-MRR is investigated by operating the robot in a limited workspace created by activating the spring. The trajectory tracking capability of the SA-MRR is explored by comparing trajectories followed by a conventional MRR and SA-MRR in a restricted workspace, while lifting a heavy payload. Trajectory tracking is performed with various payloads to demonstrate the SA-MRR's superior payload handling capacity performance due to addition of the spring-generated moment. These algorithms have been implemented on a 3-DOF SA-MRR and numerical simulations have been carried out to investigate the improved tracking accuracy and safety features due to addition of the spring-brake system.

Acknowledgements

I would like to thank Dr. Guangjun Liu for his continuous support and guidance through each stage of the program, and for always being there with words of encouragement and giving sound advice whenever I needed it. I would also like to thank all members of the Systems and Controls Lab at Ryerson University for being my support system at school, helping me with every aspect of graduate studies. I am very grateful to everyone who edited and reviewed my work. This thesis would not have been possible without your constructive criticism and suggestions.

I want to thank my family for their unconditional support and never-ending love. A special thank you to my parents, Dr. Hemlata and Dr. Dharmendra Patel, for their sacrifices to bring me where I am today. I could not have achieved this goal without their unwavering belief in me. Last, but not the least, thanks to Jigar for the countless phone calls, words of inspiration, love and support throughout this journey.

Dedication

- To Mummy and Papa

Table of Contents

List of Tables

List of Figures

List of Appendices

1. Introduction

Most industrial robots used in a manufacturing environment have their workspace isolated from humans, for the safety of human workers. With the development of robot technologies, human-robot collaboration in manufacturing environments has been made possible. The use of robots in human environments for various tasks has been increasing in the last few years and such robots are expected to become more common in the future. This thesis presents a novel approach to safe robot operation in human presence by using a modular and reconfigurable robot (MRR) with a brake and an embedded spring at each modular joint.

Robots used for physical assistance should reduce fatigue and stress, increase human capabilities in terms of force, speed, and precision, and improve the quality of work done. In return, the human operator brings experience, knowledge and understanding for the correct methods to execute a task [1]. This raises the safety issues related to physical human-robot interaction. To guarantee the safety of the human operator, and to avoid damage to robots, unexpected collisions should be avoided whenever possible. Safe human-robot coexistence can be achieved through an active or passive compliance system. An actively compliant system detects an obstacle by using data from various sensors, and modifies the robot's control software to create a scheme to avoid a collision. On the other hand, a robot based on passive compliance has mechanical elements that absorb excessive collision force [2].

Elements such as a spring or flexible link are used for robots based on passive compliance to provide fast and reliable response and avoid collisions. Robot joints with springs or similar mechanisms have been developed to limit the robot's working envelope for safe operation in human presence [3]-[6]. These approaches, discussed in detail in later sections, are effective in limited scenarios. Most of these designs detect collision and react accordingly to make it a safe

collision. More research needs to be conducted for robots that avoid collisions while continuing normal operation.

The spring-assisted MRR (SA-MRR) developed at The Systems and Controls Laboratory at Ryerson University provides strengthened manipulation capability and a distributed control method based on torque sensing is utilized to control the modules with the spring activated or deactivated. Spring-assisted motion is used to control the robot's working range and velocity while operating in the presence of a human operator, hence providing a safe human-robot collaboration environment. The working range of the robot can be limited by activating the power spring at any joint position. The working range of each joint depends on the manipulator's weight, spring characteristics and payload. When the brake is engaged, the additional torque generated by the spring, along with the distributed control system, can be used to boost the robot's tracking accuracy and payload handling capability. The objective of the present work is to explore the application potentials of the spring assisted modular and reconfigurable robot, including safe operation and improved trajectory tracking capability for a 3-DOF manipulator in a restricted workspace.

1.1 Modified Joints for Positioning Accuracy and Safety

Conventional robot joints have been modified to create robots that remain safe even in unexpected collisions. Over the years, research has been conducted to develop robots that avoid collisions by actively monitoring its environment during operation to detect obstacles, or to reduce damage due to collisions by using a variety of passive mechanisms to deflect the joint upon collision and protect the robot from harm. Some robots have been modified with passive mechanisms to absorb the impact force upon collision [3] or to deflect the joint when a collision is detected [4, 5]. Other robots have also been developed to provide additional manipulation capability for gravity compensation or positioning accuracy [6]-[9].

A variable stiffness unit (VSU) that uses two concentric rings with magnets to vary the joint stiffness and avoid collisions by stopping the joint or decelerating it was presented in [3]. The VSU shown in Fig. 1.1 was designed for a manipulator with proximity sensing capabilities to sense and avoid collisions. In case of failure of the proximity sensing capability, the VSU has been designed to absorb initial impact upon collision and avoid further damage to humans sharing the workspace with the manipulator. The simulations and experiment for this joint use the Head Injury Criteria (HIC) [10] to determine the safety envelope during operation. The safety envelope is determined such that the HIC is lower than the maximum allowable value and it is achieved by lowering the joint stiffness or the operating speed using the concentric magnetic rings.
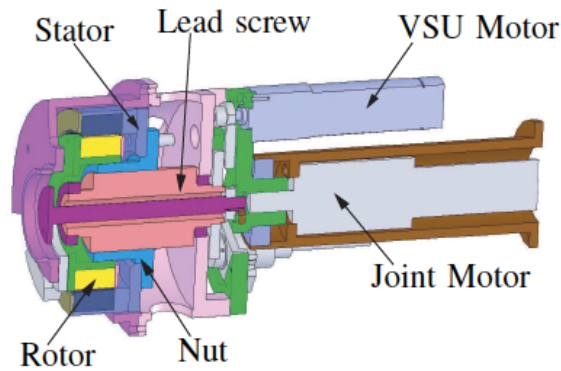
Fig. 1.1: A variable stiffness joint with the joint motor, two concentric rings composed of arc-shaped magnets and spacers, and a linear guide to make displacement [3]

A similar concept was introduced in [4], and developed further in [5], where a spring-clutch mechanism is added to each manipulator joint to ensure the safe use of a manipulator. When the external torque at a joint goes beyond the threshold torque due to a collision, the spring-clutch mechanism operates as a revolute joint to retract the joint from the collision. It reinitializes itself when the external torque is removed. This mechanism was tested on a 4-DOF robot arm to demonstrate its safety and reliability while satisfying the safety index requirements set by the HIC.
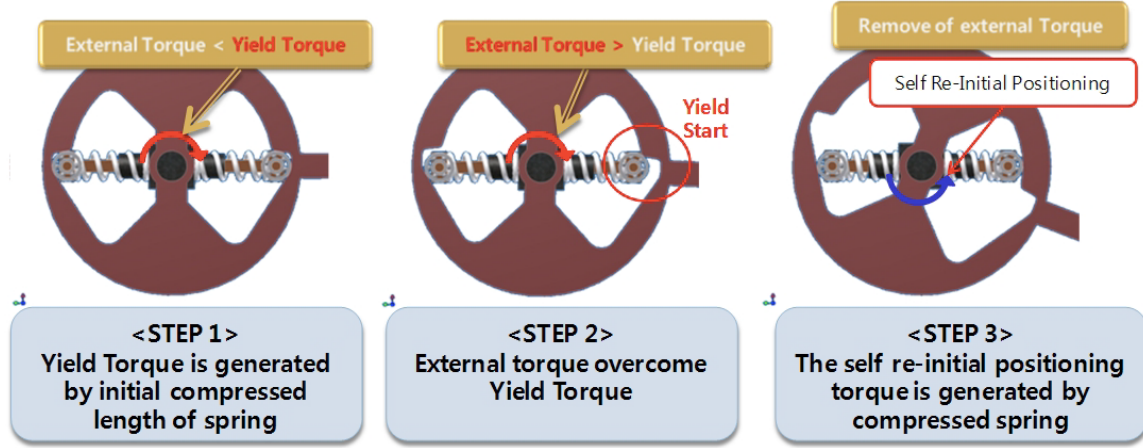
Fig. 1.2: Working concept of a spring-clutch mechanism [4]

Passive mechanisms have been used at the joint level to balance robots, boost joint performance and provide better manipulation capability. In [6], a parallelogram linkage configuration was developed for gravity compensation using torsional springs at each joint of a robot actuated in the vertical plane. This mechanism statically balances the joint at a pre-defined position by choosing a spring configuration such that it produces a torsional moment to cancel out the gravity torque experienced at that specific joint position. This property is used in a small range of joint position such that the spring deflection can be implemented to partially cancel the gravity torque, thus reducing the peak torque required by the motor to balance the joint during operation. In addition to reducing the motor peak torque, this configuration also ensures equilibrium positions for all joints, creating a safe configuration on power-off and eliminating the need for fail-safe brakes by preventing them from colliding with one another or with other objects.

An example of boosted manipulation capability along with collision safety is shown by studies conducted by Park and Song in [7]. The safe joint mechanism (SJM-III) presented in [7] uses an inclined-link, a slider with rollers, and linear springs to achieve non-linear stiffness capability for simultaneously providing collision safety and positioning accuracy. The prototype model of SJM-III is illustrated in Fig. 1.3. When a collision occurs, the external torque detected by the SJM-III exceeds a predetermined threshold torque. This leads to activation of the SJM-III

and an abrupt drop in stiffness of the robot manipulator ensures collision safety. If the robot arm stiffness is maintained to be high enough such that it remains below the threshold torque, it can be used for positioning accuracy during normal operation.
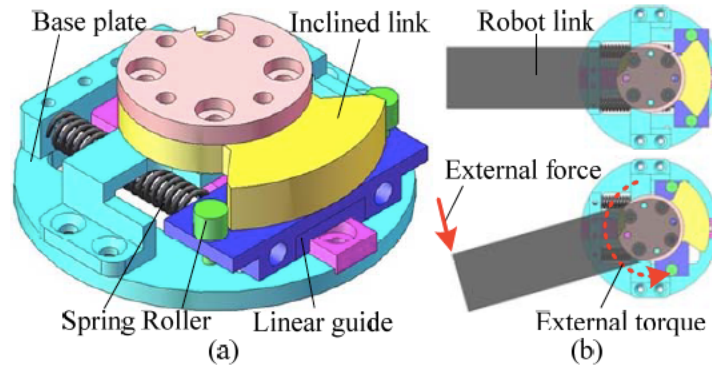
Fig. 1.3: Prototype model of SJM-III. (a) 3-D CAD model, and (b) operation of SJM-III [7]

The 1-joint spring-motor coupling system designed in [8] introduces a novel approach to use the motor inertia and a specific range of the robot link to study the velocity increasing effect due to addition of elastic elements to the joint. A small range of the link's movable area where the angular velocity of the link peaks is used to increase the velocity of the link for an instant and use it for specific applications that require boosted velocity. This range is selected such that it is about half of the total movable range of the joint. The link angle where velocity is at its peak is determined first and then the movable range is selected to allow for enough time for acceleration to reach the maximum possible velocity required for the application.

1.2 Robots for Safe Human-Robot Coexistence

Workspace sharing between humans and robot manipulators is often necessary in an industrial work environment during calibration and repair processes of the industrial manipulators. With the advancement of robot control and programming, and as hardware and sensors mature over time, significant research has been conducted in designing control laws for safe human-robot coexistence. Most robot manipulators are designed to use an obstacle avoidance algorithm based

on a pre-defined work cell where it is safe for humans to be in contact with an operating robot. The general control architecture of a robot with workspace sensing and sensor-based reactive control is illustrated in Fig. 1.4.
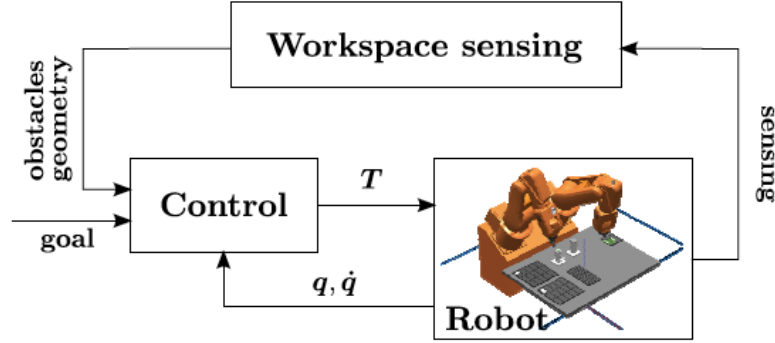
Fig. 1.4: General control architecture [11]

A passivity-based control law is introduced in [11] that provides a solution for the occurrence of local minima and collision-free motion. A danger field is calculated based on the robot position and velocity from an obstacle in its workspace and a PD controller with gravity compensation is used to control the manipulator's trajectory. A similar concept was introduced by Bicchi et al. in [12] where they integrated active and passive compliance control for safe human-robot coexistence. A *C-Obstacle* mapping method is developed to detect the collision configurations of the manipulator and is then used to calculate a *C-Force*, which is a repulsive force/torque generated by the free motion controller to avoid collisions. This force/torque has a module related to the distance between the actual configuration q and the considered C-Obstacle cell, and a direction given by the straight line passing through them. Fig. 1.5 shows the basic concept of C-Obstacle mapping.
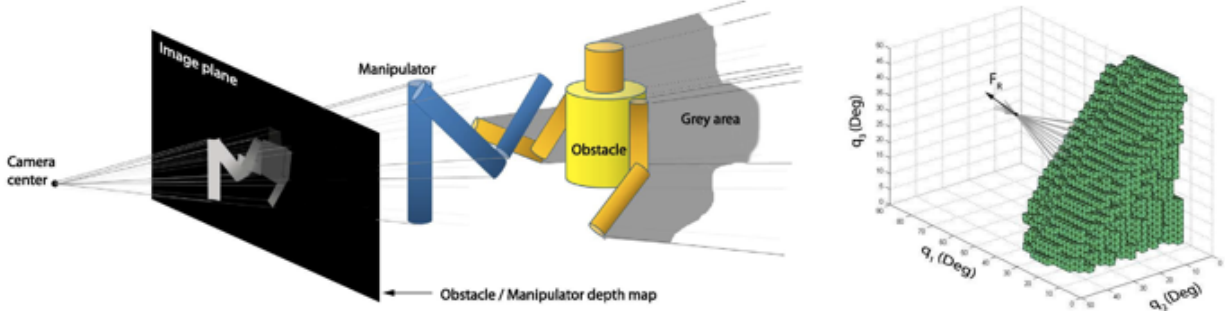
Fig. 1.5: C-obstacle mapping for safe human-robot coexistence [12]

For safe human-robot interaction, the human working near the robot must be aware of the robot's capabilities and its working limits. At the same time, the robot manipulator's trajectory must be modified to accommodate human presence in its workspace. One such example is the path-consistent safe motion planning introduced in [13]. The robot configuration and velocity is used to modify the traversing velocity along a given pre-planned path when a human enters the robot's workspace. The robot trajectory obeys the following constraint

$$distance \geq velocity \cdot braking\ time \tag{1.1}$$

where the braking time depends on the robot payload. This simple expression in equation (1.1) is developed further with more specifications and constraints to create a safety envelope around the manipulator where the workspace environment is constantly monitored for human presence. Human-centered robot navigation is primarily used in industrial work environment where safety of the human operator is a priority. In [14], Lam et al. proposed a framework that allows robots to coexist with humans and other robots harmoniously following the human-centered sensitive navigation (HCSN) that considers various sensitive fields to create the most acceptable movements in human presence. Fig. 1.6 provides the overall structure and HCSN design proposed in [14]. The robot's working envelope is modified depending on the pre-defined criteria set in the system model.
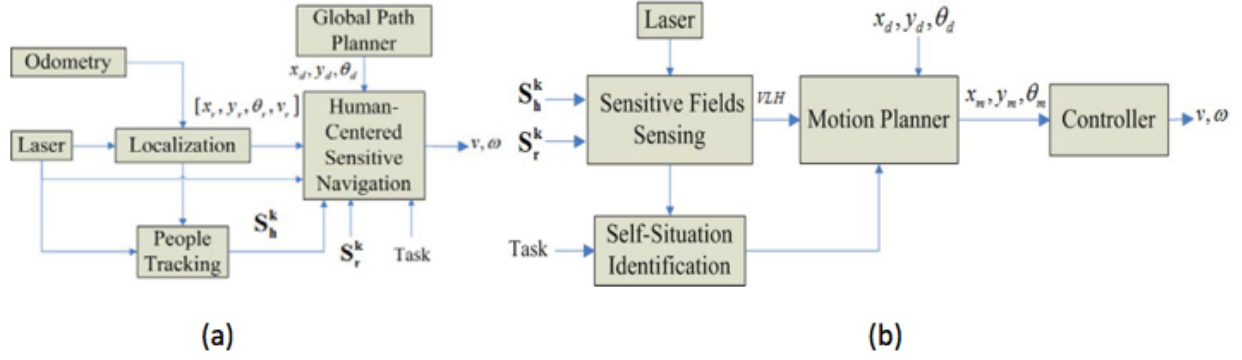
Fig. 1.6: Human-centered sensitive navigation system architecture. (a) System architecture, (b) HCSN design [14]

1.3 Modular and Reconfigurable Robots (MRRs)

The development of modular and reconfigurable robots (MRRs) is one of the most promising research areas with substantial application potential, especially in the aerospace sector [15]. Three types of MRRs have been reported in literature: (i) self-assembly, (ii) self-configuring, and (iii) manual configuring. Self-assembly robots possess the highest level of reconfigurability because they can easily assemble into a connected structure, and also disassemble into a group of unconnected units [16]-[20]. Self-configuring robots require assistance with assembly, but they can perform reconfiguration with some manual assistance. Manual-configuring robots are modular robots that can be reconfigure with manual assistance [21]. A brief survey of MRR systems is conducted by Setchi and Lagos in [22] where the authors discuss reconfigurable manufacturing systems in detail.

In the last decade, a series of MRRs have been developed at the Systems and Controls Laboratory at Ryerson University. The MRRs were developed to provide instantaneous adaptation to robot reconfigurations through the novel distributed control technique based on joint torque sensing [23]. Under this new control system architecture, the robot is stabilized joint by joint, and modules can be added or removed without the need to adjust control parameters. Over the years, the MRRs have been modified with new system models and joint designs to accommodate the

ever-changing demands from the industry. Recently, a power spring was added to the MRR modules to enhance actuator output torque by utilizing the moment generated by the power spring as it rotates along with the motor shaft. This new design was introduced in [24] and experiments were conducted with one MRR module embedded with the spring to prove the effectiveness of the proposed design.

1.4 Research Contributions

The following contributions were made from this research:

- Applications of spring-assisted modular and reconfigurable robot for safe robot operation in a human robot collaboration environment were investigated.

- A static model to determine the working range of an SA-MRR module once the spring-brake system is activated was developed.

- Simulation model of an SA-MRR was developed for system parameters from the MRR3, and test runs were conducted for a 3-DOF robot manipulator in a restricted workspace.

- Trajectory tracking accuracy, power consumption and payload handling of the SA-MRR were compared for conventional MRR and SA-MRR modules for a small working range.

1.5 Outline of Thesis

Chapter 2 starts with a description of the mechanical design of the MRR module with an embedded power spring and magnetic brake, followed by a detailed explanation of the integration of the spring-brake system into a conventional MRR module joint dynamics and its contribution to performance of the joint.

Chapter 3 provides the kinematics model of a 3-DOF robot manipulator, followed by determination of the working range of the manipulator for various joint limits and robot configurations.

Chapter 4 presents the system static and dynamic models, and the distributed and robust control system utilized for operation of the SA-MRR in various configurations.

Chapter 5 is a detailed discussion of the simulation studies of the applications of the SA-MRR. A system setup is described in detail along with the system parameters used in the test runs. Trajectory tracking accuracy and the control torque inputs for each joint with the spring-brake system engaged and disengaged are compared to prove the significance of adding the power spring for better joint performance. Payload handling capabilities of the SA-MRR are also tested for different cases to showcase the superiority of the SA-MRR over conventional MRR modules.

Chapter 6 concludes the thesis by summarizing the research contributions and conclusions from the simulation test results. A brief discussion of possible future work is also included in this chapter.

2. MRR Module with Embedded Spring

2.1 SA-MRR Design

A conventional MRR module consists of components such as motor, encoder, speed reducer, and a brake. The MRR modules developed in our laboratory also have a joint torque sensor in each module. The modified module presented by Liu et al. in [19] contains a power spring inserted between the brake armature and the motor shaft through a decoupling bearing. The inner side of the spring is connected to the motor shaft and the outer end to the armature of the brake. When the brake is not activated, the spring and brake armature rotate along with the motor shaft. Once the brake is engaged, the brake armature stops and the motor shaft continues to rotate, hence energizing the spring. The power spring generates a moment that grows as the motor shaft rotates from the position when the brake is activated. The spring-generated moment (SGM) can be used to assist the motor in tasks such as balancing the robot and payload shared by the module.
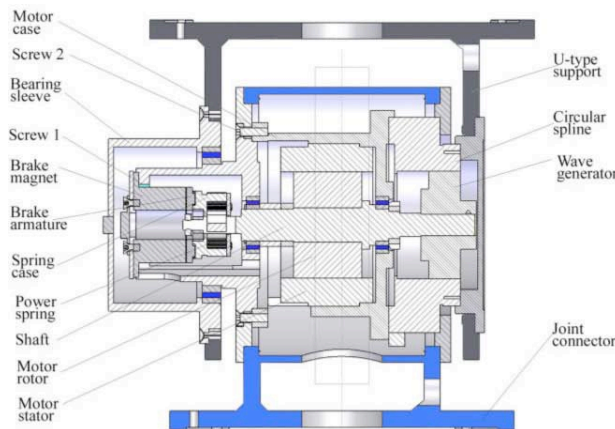
Fig. 2.1. Section view of the MRR joint mechanism with an embedded spring and brake *[19]*

As illustrated in Fig. 2.1, the power spring is inserted between the brake armature and motor shaft. The torque versus deflection relationship for the power spring can be approximated by [20, 21]

$$\frac{\tau_k}{T_{k\max}} \approx 1 - \left(\frac{u}{U_{\max}} - 1\right)^2 \tag{2.1}$$

where τ_k is the SGM, u is the deformation of the spring, $T_{k\max}$ is the maximum SGM and $U_{\max}$ represents the maximum spring deformation within an operating range. More information on the difference between power springs, and the selection of springs based on applications, can be found in [20].

2.2 Integration of Spring-Brake System into Joint Dynamics

Since the spring is inserted between the existing brake armature and motor shaft, a new complex or bulky mechanism is not required. Hence, the overall robot structure and working envelope are not affected when the spring is released. Once the spring is engaged by activating the magnetic brake, the working range of the joint is limited. The proposed design is intended for applications that do not require a large working range and spring-assisted motion at the same time.

The selection of the spring and brake is driven by the design specifications of the module and its applications. For the proposed design, the SGM is amplified by the gear ratio of the harmonic drive. To determine the maximum SGM required, the potential payload has to be projected on the motor side by dividing the gear ratio. Similarly, the maximum spring deformation is determined by multiplying the desired working range with the gear ratio.

The torque/deflection relationship shown in equation (2.1) can be used for the proposed MRR design. The deformation of the spring, u, is calculated by multiplying the working range by the gear ratio of the harmonic drive. The working range of the joint is defined as the difference between the any joint position and the locking position. Once the brake is activated at the locking position, q_l, the spring generates a moment that grows with the rotation of the actuator shaft and change in position of the shaft from the locking position. The SGM at any joint position q can be calculated as

$$\tau_k = T_{k\max}\left\{1 - \left[\frac{\gamma(q_l - q)}{U_{\max}}\right]\right\} \tag{2.2}$$

where γ is the reduction ratio of the speed reducer. The locking position q_l and joint position q are the link-side positions, i.e., the positions at the output of the speed reducer. The working range and payload capacity of the module upon spring activation depend on the gear ratio. A smaller gear ratio will increase the working range of the module, at the cost of a smaller payload. Hence, the gear ratio should be selected according to the application of the module.
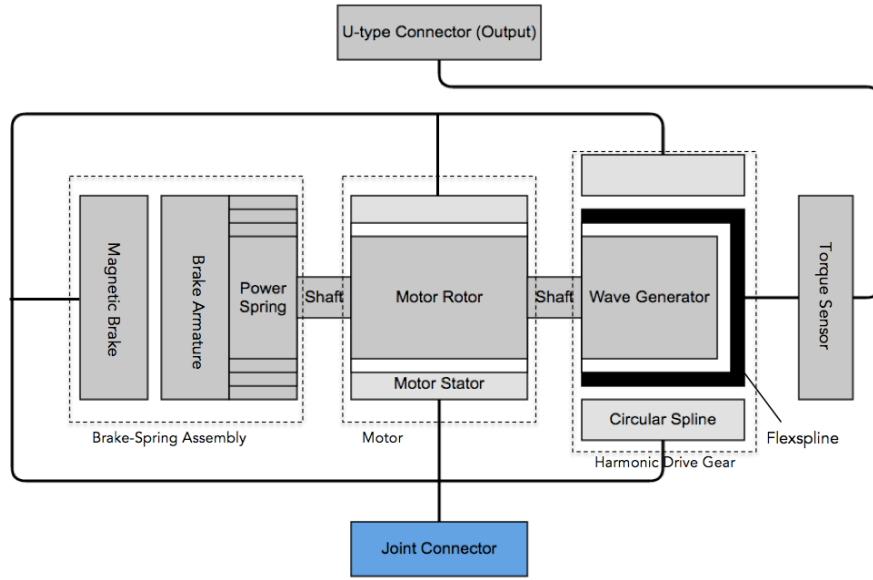
Fig. 2.2. Schematic diagram of the MRR module

Fig. 2.2 provides a schematic diagram of the MRR module with a clearer view of the different components in the system. With the motor, encoder, speed reducer, joint torque sensor, power spring and magnetic brake installed on the MRR module, the dynamics equation of one module with the embedded spring can be described by

$$I_m \gamma \ddot{q} + f(q, \dot{q}) + \frac{\tau_s}{\gamma} = \tau + \tau_k \tag{2.3}$$

where I_m is the moment of inertia of the rotor about its axis of rotation, $f(q, \dot{q})$ is the joint friction modeled as a function of position and velocity [22], τ_s is the torque at the torque sensor location and τ is the motor output torque.

From (2.3), it is clear that the SGM works together with the motor output torque to provide the total output torque of the joint. The SGM can help with robot operation in several ways, including statically balancing the robot and reinforcing the motor output. With the proposed module with an embedded spring, the robot can be balanced at any desirable configuration by activating the brakes at properly selected joint positions. The SGM can also be used to help actuate the robot joint. Once the brake is engaged, the spring can be charged to generate additional actuation torque, which can then be used to control the acceleration of the joint. In this work, applications of the SA-MRR will be explored for safe operation in a human-robot collaboration environment as discussed in the following chapters.

3. MRR Kinematics Model Formulation and Workspace Mapping

3.1 Kinematic Model of a 3-DOF SA-MRR

The SA-MRR used for this thesis is a 3-DOF robot manipulator with joints set-up as shown in Fig. 3.1 that illustrates the structure of the arm. The Denavit-Hartenberg method is used to compute the forward kinematics equation for the SA-MRR [2].
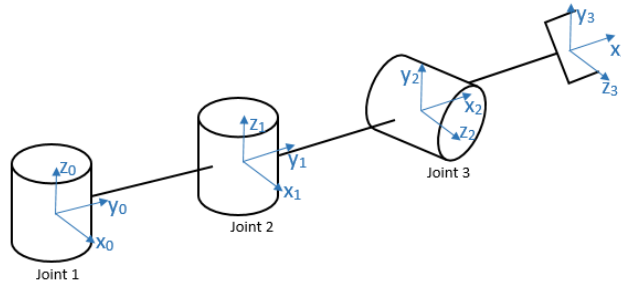
Fig. 3.1. 3-DOF Modular Reconfigurable Robot

During normal operations, the robot is assumed to operate in the workspace defined by joint limits shown in Table 3.1. When the magnetic spring-brake system is activated, the robot workspace is restricted to a small area, depending on the maximum spring deformation.

Table 3.1: MRR joint and link parameters

Joint (i)	Link Length (m)	Joint Limits (°)
1	0.406	±60°
2	0.400	±90°
3	0.400	±90°

According to the D-H coordinate system shown in Fig. 3.1, the D-H parameters for the robot are listed in Table 3.2.

Table 3.2: D-H parameters for 3-DOF SA-MRR

Joint (i)	a_i (m)	d_i (m)	α_i	θ_i
1	l_1	0	0°	q_1
2	l_2	0	90°	q_2
3	l_3	0	0°	q_3

The transformation matrices for adjacent joints are as follows:

$$H_1^0 = \begin{bmatrix} C_1 & -S_1 & 0 & l_1 C_1 \\ S_1 & C_1 & 0 & l_1 S_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.1}$$

$$H_2^1 = \begin{bmatrix} C_2 & 0 & S_2 & l_2 C_2 \\ S_2 & 0 & -C_2 & l_2 S_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.2}$$

$$H_3^2 = \begin{bmatrix} C_3 & -S_3 & 0 & l_3 C_3 \\ S_3 & C_3 & 0 & l_3 S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.3}$$

The forward kinematics is then given by

$$H_3^0 = \begin{bmatrix} C_3 C_{12} & -S_3 C_{12} & S_{12} & l_1 C_1 + l_2 C_{12} + l_3 C_3 C_{12} \\ C_3 S_{12} & -S_3 S_{12} & -C_{12} & l_1 S_1 + l_2 S_{12} + l_3 C_3 S_{12} \\ S_3 & C_3 & 0 & l_3 S_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

The following three equations determine the end-effector position at any given joint positions:

$$p_x = l_1 C_1 + l_2 C_{12} + l_3 C_3 C_{12} \tag{3.5}$$
$$p_y = l_1 S_1 + l_2 S_{12} + l_3 C_3 S_{12} \tag{3.6}$$
$$p_z = l_3 S_3 \tag{3.7}$$

3.2 Workspace Plots

The workspace of a manipulator is a space that the end of the robot arm can reach in all-possible configurations for different joint positions. Knowing the workspace of a manipulator allows the robot operator to plan an effective trajectory to fulfill a give task. The following figures illustrate the SA-MRR's workspace for when the spring-brake system is not engaged. The joint limits in Table 3.1 were used to determine this working range for the SA-MRR.
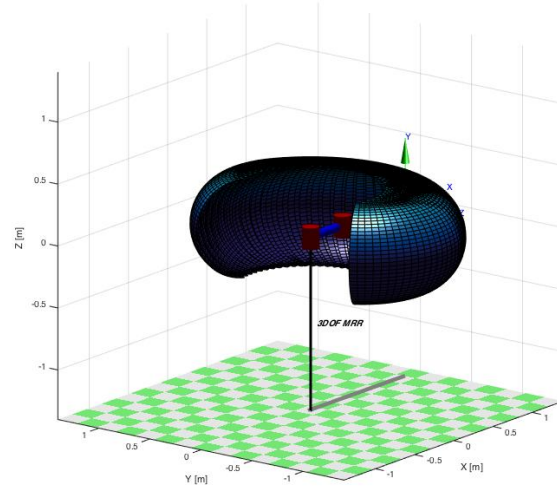
Fig. 3.2. 3-D workspace of the SA-MRR for normal operation

When the magnetic spring-brake system is activated, the robot workspace is limited to a small area, which depends on the embedded spring's maximum deformation. Fig. 3.3 below shows the working range of the robot with the spring-brake system activated in the third joint of the SA-MRR. In the example in Fig. 3.3, the motion of the third joint is limited to ±1.2°. Once the spring-brake system is activated at any joint position, the robot can be used to perform tasks that require precise motion in a small area, or to perform tasks involving a heavy payload. Applications of the SA-MRR are discussed in detail in Chapter 5.
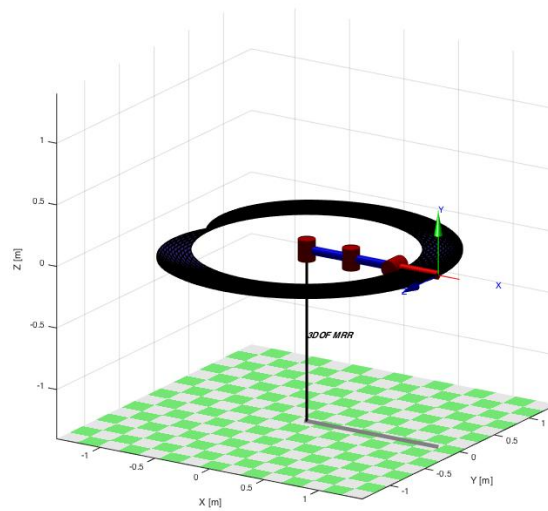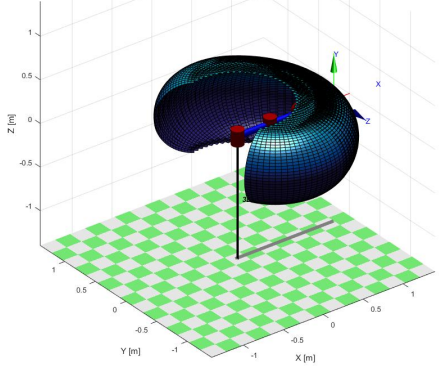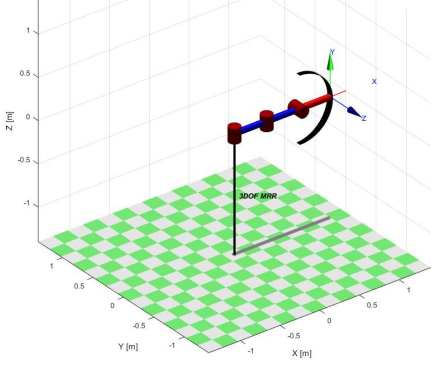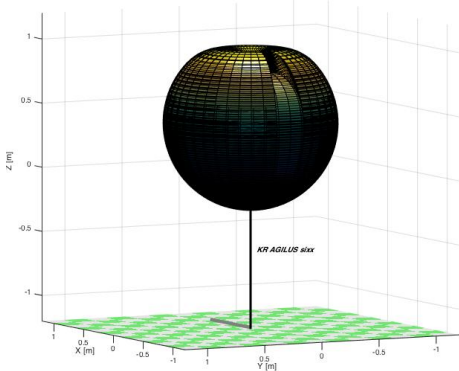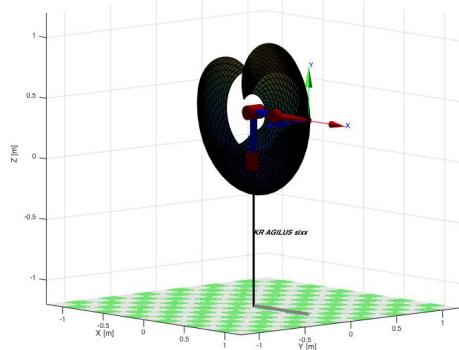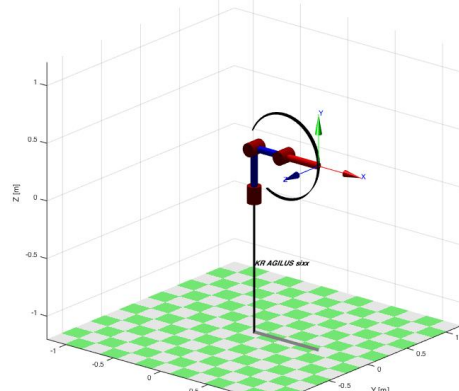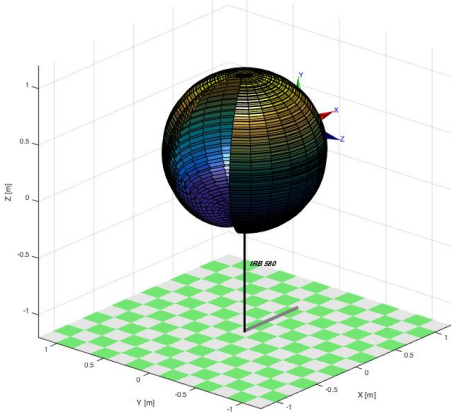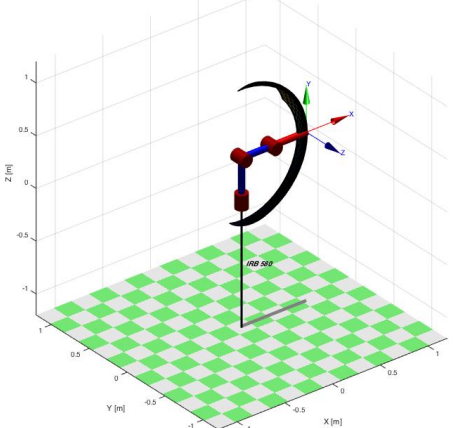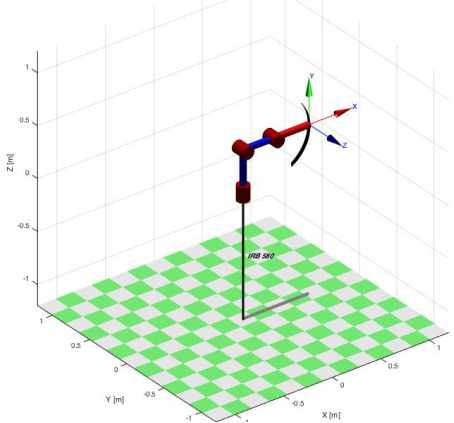
Fig. 3.3. 3-D workspace of the 3-DOF MRR with spring-brake system activated in the 3rd joint

17

As discussed in Section II, the spring-brake system can be activated at any joint position. This adaptable property of the SA-MRR is demonstrated in the plots in Table 3 that illustrates various configurations for 3-DOF robot manipulators with the spring-brake system activated at different joints. Once activated, the robot's motion is limited to joint positions determined by the characteristics of the power spring embedded in each joint. This working range can be determined by manipulating equation (2.3) explained in Chapter 2. The developed method is discussed in detail in the next chapter.

Table 3.3: Workspace plots for different manipulator configurations

Robot Information	Working Range	Workspace Plot
3-DOF MRR with-out spring-brake system activated	Joint 1: ±60° Joint 2: ±90° Joint 3: ±90°	 Fig. 3.4. MRR workspace with all joints working within normal joint limits
3-DOF MRR with spring activated at Joints 1 & 2	Joint 1: ±1.2° Joint 2: ±1.2° Joint 3: ±90°	 Fig. 3.5. MRR workspace with spring-brake system activated at Joints 1 & 2

KR AGILUS sixx with the first 3 joints in normal working range [23]	Joint 1: ±170° Joint 2: +45°/-190° Joint 3: +156°/-120°	Fig. 3.6. Workspace of KR AGILUS with all joints working within normal joint limits
KR AGILUS sixx with spring activated at Joint 1	Joint 1: ±1.2° Joint 2: +45°/-190° Joint 3: +156°/-120°	Fig. 3.7. Workspace of KR AGILUS with spring-brake system activated at Joint 1
KR AGILUS sixx with spring activated at Joints 1 & 2	Joint 1: ±1.2° Joint 2: ±1.2° Joint 3: +156°/-120°	Fig. 3.8. Workspace of KR AGILUS with spring-brake system activated at Joint 1 & 2

IRB 580 with the first 3 joints in normal working range [24]	Joint 1: ±150° Joint 2: +75°/-70° Joint 3: +35°/-60°	 Fig. 3.9. Workspace of IRB 580 with all joints operating within normal joint limits
IRB 580 with spring activated at Joint 1	Joint 1: ±1.2° Joint 2: +75°/-70° Joint 3: +35°/-60°	 Fig. 3.10. Workspace of IRB 580 with spring-brake system activated at Joint 1
IRB 580 with spring activated at Joints 1 & 2	Joint 1: ±1.2° Joint 2: ±1.2° Joint 3: +35°/-60°	 Fig. 3.11. Workspace of IRB 580 with spring-brake system activated at Joint 1 & 2

As shown in the plots in Table 3.3, the spring-brake system can be activated at any joint to achieve a configuration required for a given task. If the robot is operating in the presence of a human, the spring is activated at configurations that make it safe for the human operator to work alongside the robot. The plots in the table above show configurations of the three robots with the spring activated at different joints. A suitable configuration can be chosen for the SA-MRR to work in for a given task and its trajectory can be planned accordingly without modifying its physical structure or control architecture. As seen in Figs. 3.5, 3.7, 3.8, 3.10, 3.11, the workspace can be limited to a small range in the xy-plane with more reach in the z-direction, providing a good working envelope for the robot to carry out tasks that require minimal movement in the xy-plane. Such a restricted envelope can be useful when a human operator has to be in close proximity of the robot for a task. Similarly, if the robot has to carry out a task that requires motion in the xy-plane and minimal movement in the z-direction, it can be achieved as illustrated in Fig. 3.3.

The SA-MRR modules also provide an additional safety feature compared to conventional robot joint modules that have a simple stop switch to prevent collisions. In a conventional robot manipulator, a task requiring minimal movement along certain axes can be performed by utilizing the excellent controllers and actuators onboard but in case a safety issue arises, the joint modules will lock in one position to present any accidents in its work environment. It is possible that the robot may lose control and move in an undesirable position while trying to come to a standstill. In the SA-MRR, once the spring is activated at a particular joint and the manipulator performs its task, the magnetic brakes ensure the robot does not lose control and harm the operator. In the case of an emergency, the magnetic brakes prevent the robot from moving any further than the allowable range determined by the power spring's maximum deformation.

The SA-MRR modules also provide an additional safety feature compared to conventional robot joint modules. In a conventional robot manipulator, a task with minimal movement along certain axes can be achieved by controlling the joint actuators and moving along the desired trajectory. In the case of a system malfunction, if the robot joints lose control while following a given trajectory, it is possible for the manipulator cause sudden motion over a large range. This can cause potential injury to the human operator working alongside the manipulator or damage objects in its environment. The SA-MRR is equipped with the spring-brake system which limits the joint's working range to a restricted area as discussed earlier in this section. With the spring-brake system activated, the SA-MRR is safer than conventional robot manipulators because even if the joint actuator loses control while following a given trajectory, the joint is still limited to the small area defined by the maximum spring-deformation. This property of the SA-MRR is demonstrated and discussed in Section VII in experiments by forcing the manipulators out of their given trajectory at a random point while performing a given task.

The allowable range after spring-brake system is activated can be calculated from the maximum spring deformation and the gear ratio of speed reducer. A power spring with a maximum spring deformation of 120° was chosen as an example for the plots in Table 3; taking the gear ratio of 100:1 into account, this translates to 1.2° in joint position on the link-side. A detailed explanation on calculation of the working range is given in Section IV. Section IV also presents the dynamic model of the SA-MRR, which is later used to design a control system for the manipulator.

4. System Dynamic Model Formulation
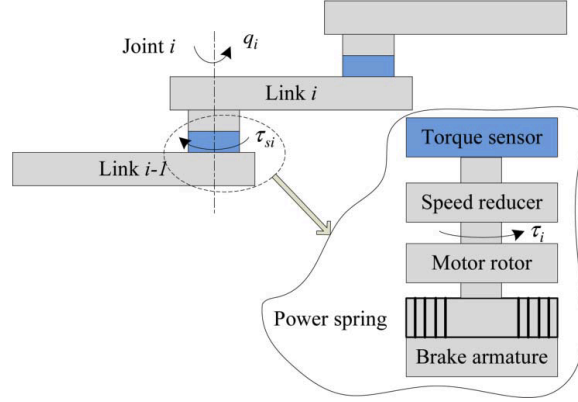
Fig. 4.1. Schematic Diagram of the MRR *[19]*

Consider a modular and reconfigurable robot with n modules. The joint modules with an embedded power spring and brake are installed in series, as illustrated in Fig. 4.1. To derive the dynamics equations, the following assumptions were made [19]:

A1. The rotor is symmetric with respect to the axis of rotation.

A2. The flexibility of the joint shaft and speed reducer is negligible.

A3. The torque transmission does not fail at the speed reducer, and the inertia between the torque sensor and the speed reducer is negligible.

A4. Inertial moment of the spring and brake armature is negligible in comparison to that of the motor rotor and the shaft.

As discussed in the previous chapter, the working range of the robot joint upon spring activation can be altered for different applications by changing the gear ratio of the harmonic drive. A smaller gear ratio will result in a larger working range, but at the cost of a smaller payload capacity. Hence, all necessary trade-offs must be conducted for each application before setting the gear ratio of the speed reducer. This characteristic of the SA-MRR modules is useful to setup the robot for a wide range of applications.

Once the spring is activated, the working range of the joint is restricted to a certain area depending on the spring characteristics, gravity torque and the actuator torque output at the locking position. This restricted working range, along with the robust distributed control algorithm introduced in [19] can be used to perform safe operation in a human-robot collaboration environment. When the robot is required to work in a small area with precision, it can combine the boosted actuator output from the spring and the robustness of the controller to perform a task successfully. This property can also be used for safe robot operation in human presence by engaging the brake when a human operator is detected in the robot's working area and to keep it out of the way of harming the operator. A static model of the robot is discussed in the following section to determine the joint torque in a static condition and its restricted working range upon activation of the spring-brake system.

The static conditions are studied first to determine how the spring-brake system can be integrated into the system dynamics for normal operation. The SGM can also be used to accelerate or decelerate the joint in a controlled manner with the boosted actuator output. The spring can be charged at any joint position by engaging the brake and then using the SGM to control the acceleration of the joint without consuming more power. This additional torque can also be used to decelerate the robot smoothly when the robot needs to slow down or stop to avoid collision with a human operator. To resume normal operation from a static position, a large motor torque is required. The SGM can help achieve boosted acceleration required to go from zero speed to the required operating speed.

4.1 Static Model

The SGM can be used to operate the robot in a restricted working range. With the availability of joint torque sensing, the working range of the robot can be determined at any joint

position by combining (2.2) and (2.3) and solving it for $(q_l - q)$. We get the expression in equation (4.1), which can later be rearranged to determine the working range for the module. Equation (4.1) is the expression for joint torque in a static condition:

$$\tau_s = T_{kmax}\gamma \left\{ 1 - \left[\frac{\gamma(q_l - q)}{U_{max}} - 1 \right]^2 \right\} \tag{4.1}$$

$$q_l - q = \frac{U_{max}}{\gamma} \left\{ 1 \pm \sqrt{1 - \frac{\tau_s}{T_{kmax}\gamma}} \right\} \tag{4.2}$$

Since there is a limit on the maximum spring deformation, once the brake is engaged, the joint can only move until U_{max} is reached. Hence only the negative term of the square root in equation (4.2) should be considered. Thus, the final for that gives the working range of the joint, once the brake is engaged, is shown in equation (4.3):

$$q_l - q = \frac{U_{max}}{\gamma} \left\{ 1 - \sqrt{1 - \frac{\tau_s}{T_{kmax}\gamma}} \right\} \tag{4.3}$$

The joint torque changes as the joint rotates within the restricted working range and is continuously sensed by the joint torque sensor on the link side. Equation (4.3) gives the working range for any joint position q. This can be used to determine the SGM of the joint at any position q. Once the locking position is known, the actual output torque for the module with the brake engaged can be calculated at any joint position using equation (2.3).

4.2 Dynamic Model

As shown in Fig. 4.1, the base module is denoted as the first module. Modules close to the first module are called lower modules, and those close to the end effector are named upper modules. Based on the dynamic equations of a rigid robot manipulator with n rotary joints and joint torque sensing, the dynamic equations of MRRs with embedded spring can be formulated as follows [19]:

For the base module, $i = 1$, the dynamic equation can be obtained by rearranging equation (2.3):

$$I_{m1}\gamma_1\ddot{q}_1 - \tau_{k1} + f_1(q_1, \dot{q}_1) + \frac{\tau_{s1}}{\gamma_1} = \tau_1 \tag{4.4}$$

For the second module from the base, $i = 2$,

$$I_{m2}\gamma_2\ddot{q}_2 - \tau_{k2} + f_2(q_2, \dot{q}_2) + I_{m2}z_2^T z_1\ddot{q}_1 + \frac{\tau_{s2}}{\gamma_2} = \tau_2 \tag{4.5}$$

For upper modules, $i \geq 3$,

$$I_{mi}\gamma_i\ddot{q}_i - \tau_{ki} + f_i(q_i, \dot{q}_i) + I_{mi}\sum_{j=1}^{i-1} z_i^T z_j\ddot{q}_j + \frac{\tau_{si}}{\gamma_i}$$
$$+ I_{mi}\sum_{j=2}^{i-1}\sum_{k=1}^{j-1} z_i^T(z_k \times z_j)\dot{q}_k\dot{q}_j = \tau_i \tag{4.6}$$

In Equations (4.4)-(4.6), $I_{mi}, \gamma_i, \tau_{ki}, \tau_{si}$ and τ_i are the same as those in equation (2.3), with the subscript "i" representing the ith module. The SGM for the ith module can be derived as shown

$$\tau_{ki} = \begin{cases} k_{i1}\gamma_i(q_{li} - q_i) + k_{i2}\gamma_i^2(q_{li} - q_i)^2 & , \qquad \text{brake engaged} \\ 0 & , \qquad \text{brake released} \end{cases} \tag{4.7}$$

where $k_{i1} = 2T_{kimax}/U_{imax}$ and $k_{i2} = -T_{kimax}/U_{imax}^2$ are the spring constants.

As mentioned earlier, the joint friction $f_i(q_i, \dot{q}_i)$ is assumed to be a function of the joint position and velocity [22, 25, 26]. The mathematical model of $f_i(q_i, \dot{q}_i)$ is of the form:

$$f_i(q_i, \dot{q}_i) = \left[f_{ci} + f_{si}\,exp(-f_{\tau i}\dot{q}_i^2)\right]sgn(\dot{q}_i) + b_i\dot{q}_i + f_{qi}(q_i, \dot{q}_i) \tag{4.8}$$

where f_{ci} is the Coulomb friction-related parameter, f_{si} is the static friction-related parameter, $f_{\tau i}$ is the positive parameter corresponding to the Stribeck effect, b_i is the viscous friction coefficient, $f_{qi}(q_i, \dot{q}_i)$ is the position dependency of friction and other friction modeling errors, and $sgn(\dot{q}_i)$ is the sign function defined as

$$sgn(\dot{q}_i) = \begin{cases} 1 & for \quad \dot{q}_i > 0 \\ 0 & for \quad \dot{q}_i = 0 \\ -1 & for \quad \dot{q}_i < 0 \end{cases} \tag{4.9}$$

Let $F_i = [b_i\ f_{ci}\ f_{si}\ f_{\tau i}]^T$, $\Theta_{ijk} = z_i^T(z_k \times z_j)$. According to the model uncertainty decomposition scheme proposed by Liu and Goldenberg [27], $\theta_{ij}, \Theta_{ijk}, F_i$ can be decomposed into a constant and variable part as

$$\begin{aligned}
\theta_{ij} &= \theta_{ij}^c + \theta_{ij}^v \\
\Theta_{ijk} &= \Theta_{ijk}^c + \Theta_{ijk}^v \\
F_i &= F_i^c + F_i^v
\end{aligned}$$

(4.10)

where the superscripts "c" and "v" denote the constant and variable parts, respectively.

Let $\hat{b}_i^c$, $\hat{f}_{ci}^c$, $\hat{f}_{si}^c$ and $\hat{f}_{\tau i}^c$ represent the estimated constant friction parameters. Using the linearization scheme proposed by Liu in [22], the friction model in equation (4.8) can be approximated by:

$$f_i(q_i, \dot{q}_i) \approx \hat{f}_i^c(\dot{q}_i) + Y(\dot{q}_i)(\tilde{F}_i^c + F_i^v) + \check{f}_i(\dot{q}_i) + f_{qi}(q_i, \dot{q}_i) \tag{4.11}$$

where $\tilde{F}_i^c = F_i^c - \hat{F}_i^c$ and $\hat{F}_i^c = [\hat{b}_i^c\ \hat{f}_{ci}^c\ \hat{f}_{si}^c\ \hat{f}_{\tau i}^c]^T$. $\hat{f}_i^c(\dot{q}_i), \check{f}_i(\dot{q}_i)$ and $Y(\dot{q}_i)$ are described in detail as follows:

$$\begin{aligned}
\hat{f}_i^c(\dot{q}_i) &= [\hat{f}_{ci}^c + \hat{f}_{si}^c exp(-\hat{f}_{\tau i}^c \dot{q}_i^2)]sat(\dot{q}_i, \epsilon_{\dot{q}_i}) + \hat{b}_i^c \dot{q}_i \\
\check{f}_i(\dot{q}_i) &= [f_{ci} + f_{si} exp(-f_{\tau i}\dot{q}_i^2)][sgn(\dot{q}_i) - sat(\dot{q}_i, \epsilon_{\dot{q}_i})] \\
Y(\dot{q}_i) &= [0\ 1\ exp(-\hat{f}_{\tau i}^c\dot{q}_i^2) - \hat{f}_{si}^c\dot{q}_i^2\ exp(-\hat{f}_{\tau i}^c\dot{q}_i^2)] \cdot sat(\dot{q}_i, \epsilon_{\dot{q}_i}) + [\dot{q}_i\ 0\ 0\ 0]
\end{aligned}$$

(4.12)

where the saturation function is defined as follows:

$$sat(\dot{q}_i, \epsilon_{\dot{q}_i}) = \begin{cases} \dfrac{\dot{q}_i}{|\dot{q}_i|}, & |\dot{q}_i| > \epsilon_{\dot{q}_i} \\ \dfrac{\dot{q}_i}{\epsilon_{\dot{q}_i}}, & |\dot{q}_i| \le \epsilon_{\dot{q}_i} \end{cases} \tag{4.13}$$

where $\epsilon_{\dot{q}_i}$ is a positive constant.

From the parameters and variables in equations (4.4)-(4.6) with consideration of (4.10), model uncertainties can be classified into three categories: i) constant parametric uncertainties in $k_{i1}, k_{i2}, \theta_{ij}^c, \Theta_{ijk}^c$, and F_i^c, which do not vary with the robot reconfiguration; ii) variable parametric uncertainties in the terms of $I_{mi}\sum_{j=1}^{i-1}\theta_{ij}^v\ddot{q}_j$ and $I_{mi}\sum_{j=2}^{i-1}\sum_{k=1}^{j-1}\Theta_{ijk}^v\dot{q}_k\dot{q}_j$ can be results of the robot reconfiguration due to reasons such as misalignment of axes; furthermore, F_i^v may vary as the

manipulator moves to different positions; iii) non-parametric model uncertainties include torque sensor inaccuracy and noise as well as the friction term $f_{qi}(q, \dot{q})$.

The constant parametric uncertainties can be compensated with adaptive control schemes, and the variable parametric uncertainties and non-parametric uncertainties can be controlled with the decomposition-based robust control algorithms developed by Liu and Goldenberg [27, 28, 29]. The following properties outlines the bounds of each model uncertainty:

Property 1: The variable parts in (4.10) are bounded as

$$\left|\theta_{ij}^v\right| < \rho_{\theta ij} \qquad \left|\Theta_{ijk}^v\right| < \rho_{\Theta ijk} \qquad \left|F_{ij}^v\right| < \rho_{Fij} \qquad , j = 1, \dots, 4 \tag{4.14}$$

where $\rho_{\theta ij}, \rho_{\Theta ijk}$, and ρ_{Fij} are known constant bounds.

Property 2: The nonparametric friction term is bounded as

$$\left|f_{qi}(q_i, \dot{q}_i)\right| < \rho_{fqi} \tag{4.15}$$

where ρ_{fqi} is a known constand bound for any position q_i and velocity $\dot{q}_i$.

Property 3: The joint torque sensor inaccuracy and noise is bounded,

$$\left|\tau_{si} - \hat{\tau}_{si}\right| < \rho_{\tau si} \tag{4.16}$$

where $\hat{\tau}_{si}$ denotes the measured coupling torque at the torque sensor location, and $\rho_{\tau si}$ is a known constant bound.

Property 4: As the joint is stabilized, the acceleration and velocity must be bounded as well. Assuming that the desired acceleration and velocity are always bounded, it can be concluded that the acceleration and velocity errors of a stabilized joint are bounded. For a stabilized joint i,

$$\left|\ddot{e}_i\right| < \rho_{eai} \qquad \left|\dot{e}_i\right| < \rho_{evi} \tag{4.17}$$

where ρ_{eai} and ρ_{evi} are known constant bounds. The analytical determination of these bounds is complicated as they depend on the various model uncertainties of the MRR, including those associated with the joint actuators. In theory, these bounds can be selected conservatively to be large enough. In practice, they need to be tuned with other control parameters.

Property 5: Since $z_i, z_j,$ and z_k are the unit vectors along the direction of rotation of the corresponding joints, their vector products are bounded as

$$\left|\theta_{ij}\right| = \left|z_i^T z_j\right| \leq 1 \qquad \left|\Theta_{ijk}\right| = \left|z_i^T(z_k \times z)j\right| \leq 1 \tag{4.18}$$

Property 6: The last term in (4.11) $\check{f}_i(\dot{q}_i)$ is bounded as

$$\left|\check{f}_i(\dot{q}_i)\right| < \rho_{fi} \tag{4.19}$$

This dynamic model was used as the basis to develop a robust distributed control system for the SA-MRR. The controller is presented in the following section with detailed explanation on how the tracking errors and model uncertainties are bounded within limits introduced in this section.

4.3 Control System Design

The distributed control system presented here was developed by Liu et al in [19]. This distributed control algorithm can be used to control MRRs with any number of modules, and with any configurations.

The overall control of each joint is defined as

$$\tau_i = \frac{\hat{\tau}_{si}}{\gamma_i} - \hat{\tau}_{ki} + u_i, \qquad i = 1,2, \dots n \tag{4.20}$$

where u_i is the control input to be determined for the ith joint, and the estimated spring-generated moment $\hat{\tau}_{ki}$ can be calculated using equation (4.7).

For the base joint $i = 1$, combining (4.20) with (4.4) yields

$$I_{m1}\gamma_1\ddot{q}_1 + \frac{\tilde{\tau}_{s1}}{\gamma_1} - \tilde{\tau}_{k1} + f_1(q_1, \dot{q}_1) = u_1 \tag{4.21}$$

where $\tilde{\tau}_{s1} = \tau_{s1} - \hat{\tau}_{s1}$ and $\tilde{\tau}_{k1} = \tau_{k1} - \hat{\tau}_{k1}$. For this joint, since there is no coupling term, the control input u_1 can be calculated by using control design techniques for a single joint.

For the second joint i = 2, combining (4.20) with (4.5) yields

$$I_{m2}\gamma_2\ddot{q}_2 + \frac{\tilde{\tau}_{s2}}{\gamma_2} - \tilde{\tau}_{k2} + f_2(q_2,\dot{q}_2) + I_{m2}\theta_{21}\ddot{q}_1 = u_2 \qquad (4.22)$$

where $\tilde{\tau}_{s2} = \tau_{s2} - \hat{\tau}_{s2}$ and $\tilde{\tau}_{k2} = \tau_{k2} - \hat{\tau}_{k2}$. The inertial force associated with acceleration of the base joint is involved in the control of the second joint, and a model uncertainty exists in θ_{21} as a result of the reconfiguration.

For any additional joints $i \geq 3$, substituting (4.20) into (4.6) yields

$$I_{mi}\gamma_i\ddot{q}_i + \frac{\tilde{\tau}_{si}}{\gamma_i} - \tilde{\tau}_{ki} + I_{mi}\sum_{j=1}^{i-1}\theta_{ij}\ddot{q}_j + f_i(q_i,\dot{q}_i) + I_{mi}\sum_{j=2}^{i-1}\sum_{k=1}^{j-1}\Theta_{ijk}\dot{q}_k\dot{q}_j = u_i \qquad (4.23)$$

where $\tilde{\tau}_{si} = \tau_{si} - \hat{\tau}_{si}$ and $\tilde{\tau}_{ki} = \tau_{ki} - \hat{\tau}_{ki}$. The inertial and Coriolis forces associated with the motion of the lower joints are involved in the motion of the robot as shown in equation (4.23), and reconfiguration can result in model uncertainties in the θ_{ij} and Θ_{ijk} terms. However, as the lower joints have been stabilized, the acceleration and velocities of the lower joints must be bounded. Thus, the uncertainties in the $I_{mi}\sum_{j=1}^{i-1}\theta_{ij}\ddot{q}_j$ and $I_{mi}\sum_{j=2}^{i-1}\sum_{k=1}^{j-1}\Theta_{ijk}\dot{q}_k\dot{q}_j$ terms are also bounded. From these observations, it can be concluded that the control input u_i can be designed for each joint independently. The model uncertainties are all bounded with consideration of the bounded model uncertainty due to the motion of the lower joints, as well as Properties 1-6 discussed in the previous section.

The system errors are defined as:

$$\begin{aligned}e_i &= q_i - q_{id} \\ r_i &= \dot{e}_i + \lambda_i e_i \\ a_i &= \ddot{q}_{id} - 2\lambda_i\dot{e}_i - \lambda_i^2 e_i\end{aligned} \qquad (4.24)$$

To stabilize the base joint, the control law is designed as

$$\tau_1 = I_{m1}\gamma_1 a_1 + \frac{\hat{\tau}_{s1}}{\gamma_1} - \hat{\tau}_{k1} + \hat{f}_1^c(\dot{q}_1) - k_{I1}\int_0^t r_1(t)dt + \tau_{r1} \qquad (4.25)$$

where $k_{I1} > 0$ is a constant and τ_{r1} is the decomposition-based robust control term [27, 28, 29] which is defined as

$$\tau_{r1} = -\left(\rho_{f1} + \rho_{fq1} + \frac{\rho_{\tau_{s1}}}{\gamma_1}\right) sat(r_1, \epsilon_{r1})$$

$$- \sum_{j=1}^{4} \left\{\rho_{F1j} Y_j(\dot{q}_1) \, sat\left(r_1 Y_j(\dot{q}_1), \epsilon_{F1j}\right)\right\} \tag{4.26}$$

where $Y_j(\dot{q}_1)$ is the jth element of $Y(\dot{q}_1)$, and ϵ_{r1} and ϵ_{F1j} are the positive control parameters.

Using the control law given in (4.25), the stabilization of the first joint results in the boundedness of the magnitudes of $\dot{q}_1$ and $\ddot{q}_1$. Then, the control torque for the second joint is designed as follows:

$$\tau_2 = I_{m2}\gamma_2 a_2 + \frac{\hat{\tau}_{s2}}{\gamma_2} - \hat{\tau}_{k2} - k_{I2} \int_0^t r_2(t)dt + \hat{f}_2^c(\dot{q}_2) + I_{m2}\hat{\theta}_{21}^c \ddot{q}_{1d} + \tau_{r2} \tag{4.27}$$

where $k_{I2} > 0$ and τ_{r2} is defined as

$$\tau_{r2} = -\left(\rho_{f2} + \rho_{fq2} + \frac{\rho_{\tau_{s2}}}{\gamma_2} + I_{m2}\rho_{ea1}\right) sat(r_2, \epsilon_{r2})$$

$$- \sum_{j=1}^{4} \left\{\rho_{F2j} Y_j(\dot{q}_2) \, sat\left(r_2 Y_j(\dot{q}_2), \epsilon_{F2j}\right)\right\} \tag{4.28}$$

$$- I_{m2}\rho_{\theta21}\ddot{q}_{1d} sat(r_2 \ddot{q}_{1d}, \epsilon_{\theta21})$$

where ϵ_{r2}, $\epsilon_{\theta21}$ and ϵ_{F2j} are positive control parameters.

The control torque for the ith joint can be designed as follows:

$$\tau_i = I_{mi}\gamma_i a_i + \frac{\hat{\tau}_{si}}{\gamma_i} - \hat{\tau}_{ki} - k_{Ii} \int_0^t r_i(t)dt + \hat{f}_i^c(\dot{q}_i)$$

$$+ I_{mi}\left\{\sum_{j=1}^{i-1} \hat{\theta}_i^c \ddot{q}_{jd} + \sum_{j=2}^{i-1}\sum_{k=1}^{j-1} \hat{\Theta}_{ijk}^c \dot{q}_k \dot{q}_j\right\} + \tau_{ri} \tag{4.29}$$

where $k_{Ii} > 0$, and τ_{ri} is designed as follows:

$$\tau_{ri} = -\left(\rho_{fi} + \rho_{fqi} + \frac{\rho_{\tau si}}{\gamma_i} + I_{mi}\sum_{j=1}^{i-1}\rho_{eaj}\right)sat(r_i, \epsilon_{ri})$$

$$-\sum_{j=1}^{4}\left\{\rho_{Fij}\,Y_j(\dot{q}_i)\,sat\left(r_iY_j(\dot{q}_i), \epsilon_{Fij}\right)\right\}$$

$$-I_{mi}\sum_{j=1}^{i-1}\left\{\rho_{\theta ij}\,\ddot{q}_{jd}\,sat\left(r_i\ddot{q}_{jd}, \epsilon_{\theta ij}\right)\right\} \tag{4.30}$$

$$-I_{mi}\sum_{j=2}^{i-1}\sum_{k=1}^{j-1}\left\{\rho_{\Theta ijk}\,\dot{q}_k\dot{q}_j\,sat\left(r_i\dot{q}_k\dot{q}_j, \epsilon_{\Theta ijk}\right)\right\}$$

where ϵ_{ri}, ϵ_{Fij}, $\epsilon_{\theta ij}$, and $\epsilon_{\theta ijk}$ are positive control parameters.

The closed-loop expression for the ith joint can be determined by substituting (4.10) and (4.29) into (4.6) as follows:

$$I_{mi}\gamma_i\dot{r}_i + K_{Ii}\int_0^t r_i(t)dt$$

$$= -I_{mi}\gamma_i\lambda_i r_i + \tau_{ri} + \tilde{\tau}_{ki} - Y(\dot{q}_i)\tilde{F}_i^c - Y(\dot{q}_i)F_i^v - f_{qi}(q_i, \dot{q}_i)$$

$$-\check{f}_i(\dot{q}_i) - I_{mi}\sum_{j=2}^{i-1}\sum_{k=1}^{j-1}\left(\theta_{ij}\ddot{e}_j + \theta_{ij}^v\ddot{q}_{jd} + \tilde{\theta}_{ij}^c\ddot{q}_{jd}\right) - \frac{\tilde{\tau}_{si}}{\gamma_i} \tag{4.31}$$

$$-I_{mi}\sum_{j=2}^{i-1}\sum_{k=1}^{j-1}\left(\Theta_{ijk}^v\dot{q}_k\dot{q}_j - \tilde{\Theta}_{ijk}^c\dot{q}_k\dot{q}_j\right)$$

The adaptation law is given as follows:

$$\dot{\hat{k}}_{i1} = \begin{cases} \mu_{ki1}\gamma_i(q_{li} - q_i)r_i, & brake\ engaged \\ 0, & brake\ released \end{cases}$$

$$\dot{\hat{k}}_{i2} = \begin{cases} \mu_{ki2}\gamma_i^2(q_{li} - q_i)^2 r_i, & brake\ engaged \\ 0, & brake\ released \end{cases} \tag{4.32}$$

$$\dot{\hat{F}}_i^c = -\mu_{Fic}[Y(\dot{q}_i)]^T r_i$$

$$\dot{\hat{\theta}}_{ij}^c = -\mu_{\theta ij}\ddot{q}_{jd}r_i$$

$$\dot{\hat{\Theta}}_{ijk} = -\mu_{\theta ijk}\dot{q}_k\dot{q}_j r_i$$

where $\mu_{ki1}, \mu_{ki2}, \mu_{Fic}, \mu_{\theta ij}$ and $\mu_{\theta ijk}$ are positive constants.

In [19] Liu et al. proved that for an n-DOF MRR with the joint dynamics given by equations (4.4)-(4.6), and model uncertainty defined by (4.10), the tracking error for each joint is uniformly

32

ultimately bounded under the control law given by (4.29) and the adaptation shown in (4.32). The ultimate bound of tracking error is determined by the variable parametric uncertainties, nonparametric uncertainties and the control parameters, but it is not affected by the constant parametric uncertainties.

The control of MRRs with joint torque sensing discussed in this section is one of the many control approaches that can be applied to the SA-MRR, as long as the spring dynamics are taken into account in the control system design. The following chapter discusses the application of this control system on spring-assisted MRRs for motion in a restricted workspace once the brake is engaged.

5. Simulation Studies

5.1 Applications of the SA-MRR for Safe Human-Robot Collaboration

In this section, the importance of the spring-embedded MRR modules for a safe, restricted working range during human-robot collaboration is discussed in detail. For a regular MRR without spring-assisted motion, a major portion of the joint torque is used to balance the robot arm against gravity. This, in turn, reduces the actuator torque available to control the joint's motion. In order to operate the robot in a restricted working range, while balancing the robot at a desired configuration, a robust controller and high joint actuator output torque is required. With the spring embedded in each MRR module, the spring generated moment can be used for static balancing of the robot, allowing the entire motor torque to be used for robot operation at any joint position once the brake is engaged. The static balancing property of a spring-assisted robot is useful on mobile robots, where low power consumption is a priority.

In addition to simple static balancing, the spring-brake system can also be used for precise operation within a small range by applying the distributed robust control method developed in [36]. During operation in a human-robot collaboration environment, the robot's working range can be restricted to a small area where it can perform its task without causing harm to the human operator. A demonstration of the SA-MRR's capabilities compared to those of a regular MRR without a spring shall be conducted to show the enhanced performance of the SA-MRR.

The following section describes the robot setup and discuss simulation results in detail.

5.2 Simulations

5.2.1 Simulation Control System Architecture

The Simulink model created for the 3-DOF SA-MRR manipulator takes the desired joint trajectories as the input and calculates the control torque to be sent to the SimMechanics model for the 3-DOF arm. Once the control input is computed by the distributed control algorithm discussed in Chapter 4, the spring generated moment and friction terms are added to the control input before sending the control signal to the SimMechanics module. Fig. 5.1 illustrates the basic control system architecture for the simulation models used to obtain results for the different test cases. The Simulink models for each subsystem have been included in Appendix A.
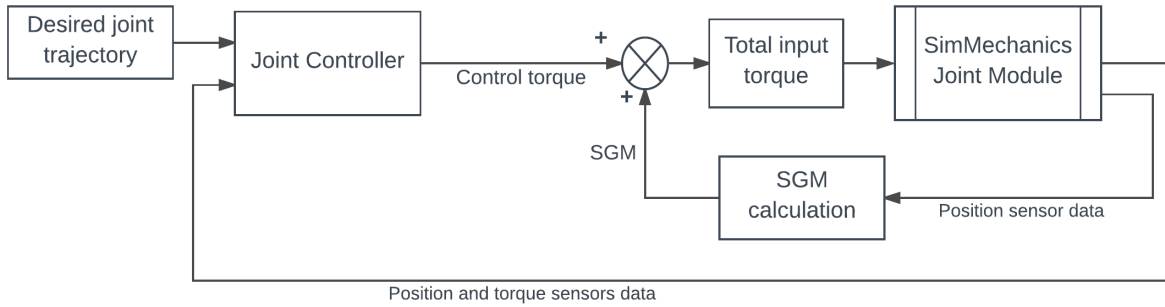

Fig. 5.1: Basic control system architecture

The Joint Controller block in Fig. 5.1 includes all error calculations based on sensor data, and the estimation of friction and other coupling terms and the spring-generated moment based on the system errors, using the adaptation law. The control torque calculated by the joint controller is then combined with the system friction model described in the next section and the spring torque calculated for the current joint position. The resultant torque is then sent to the joint in the SimMechanics model used to represent the 3-DOF SA-MRR.

5.2.2 Simulation Setup

The robot setup consists of three MRR modules with the spring-brake system in each module. Fig. 5.2 shows the arrangement of the three modules in the robot manipulator. All simulations are conducted in two different configurations: (i) without the spring-brake system activated and (ii) with the spring-brake system activated at $q_l = 0°$. The simulations focus on studying the difference in performance between the two robot configurations.
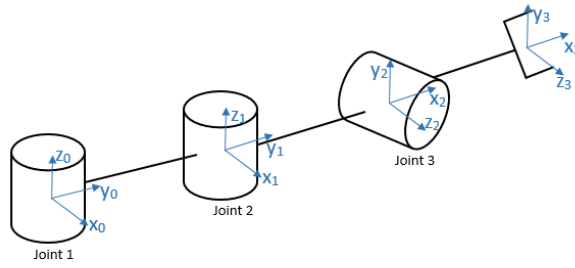
Fig. 5.2. 3-DOF-robot model for simulations

To validate the SA-MRR design's system models and algorithms for the proposed applications, simulations were conducted on a 3-DOF SA-MRR manipulator with various working ranges. The simulation studies in this chapter provide a detailed investigation of the effect of spring-generated moment and the distributed robust control algorithm on the performance of the robot in a shared workspace. The following physical and control parameters were used for the plant simulation models used in these studies:

Table 5.1: Design and control parameters, and parametric uncertainty bounds

I_{mI}	0.0192 kg.m²	$\mu_{\theta ij}$	0.01	$\epsilon_{\theta ij}$	0.01
I_{m2}	0.0192 kg.m²	$\mu_{\Theta ijk}$	0.01	$\epsilon_{\Theta ijk}$	0.01
I_{m3}	0.0192 kg.m²	μ_{ki}	1.00	ϵ_{ri}	10^{-3}
m_{mi}	6.0 kg	μ_{Fic}	0.10	ϱ_{fqi}	0.5 (Nm)
T_{kmaxi}	10.0 N.m	ϵ_{Fi1}	10^{-6}	$\varrho_{\theta ij}$	1.0
U_{maxi}	1.5°	ϵ_{Fi2}	10^{-4}	$\varrho_{\Theta ijk}$	1.0
γ_i	100	ϵ_{Fi3}	10^{-4}	$\varrho_{\tau si}$	10.0 (Nm)
λ_i	160	ϵ_{Fi4}	0.01	ϱ_{evi}	0.01 (rad/s)
k_{Ii}	15.0	$\epsilon_{\dot{q}_\iota}$	10^{-3}	ϱ_{eai}	0.01 (rad/s²)

For simulations, the variable parametric uncertainty parameters were eliminated as no data was available to perform simulations with the variable terms in the friction and joint rotation terms. In addition to this, the friction model for simulations was simplified to reduce heavy computations by eliminating the last term in equation (4.8), resulting in the following expression:

$$f_i(q_i, \dot{q}_i) = \left[f_{ci} + f_{si}\, exp(-f_{\tau i}\dot{q}_\iota{}^2) \right] sgn(\dot{q}_\iota) + b_i \dot{q}_\iota \tag{5.1}$$

The friction parameters obtained from experimental data are listed below:

Table 5.2: Friction model parameters

$f_{ci} = 4.95\ N \cdot m,$	$f_{si} = 5.403\ N \cdot m,$
$f_{\tau i} = 1.567\ (s/deg)^2,$	$b_i = 0.177\ N \cdot m \cdot s/deg$

The nominal values of the friction model parameters and uncertainty bounds for the SA-MRR modules were estimated based on the experimental data obtained from the MRR3 modules in the lab, using a procedure similar to that in [33]. The estimated parameters are as follows:

Table 5.3: Nominal friction parameters

|	
$\hat{f}_{ci} = 3.96\ N \cdot m,$	$\hat{f}_{si} = 4.322\ N \cdot m,$
$\hat{f}_{\tau i} = 1.254\ (s/deg)^2,$	$\hat{b}_i = 0.142\ N \cdot m \cdot s/deg$

Table 5.4: Parametric uncertainty bounds for friction

$\rho_{fci} = 0.99\ N \cdot m,$	$\rho_{fsi} = 1.081\ N \cdot m,$
$\rho_{f\tau i} = 0.313\ (s/deg)^2,$	$\rho_{bi} = 0.035\ N \cdot m \cdot s/deg$

### 5.2.3 Simulations for trajectory tracking with spring-assisted motion

The SA-MRRs can be used to enhance trajectory tracking performance of a robot, while consuming less power as mentioned in Section 5.1. The SA-MRR's tracking capability compared to that of a conventional MRR is tested for a restricted trajectory based on the working range defined by the maximum spring deformation. When the spring is engaged, the additional torque generated by the spring is used to balance the robot joints and the distributed controller utilizes the motor output torque to contribute to better trajectory tracking.

To verify the robot's tracking accuracy, simulation tests were conducted for two cases. In *Case 1*, the first two joints have the spring activated and the MRR's performance is tested for tracking a sinusoidal trajectory with a 2 kg payload attached at the end-effector. For *Case 2*, the spring is activated at all three joints, and a 2 kg payload is attached at the end-effector. The robot follows a sinusoidal trajectory for this case as well. For each case, tests were conducted in the limited working range with and without the spring activated. The desired and controlled trajectories, position errors, and the control torques for the tests are compared in Fig. 5.3-Fig. 5.5.

The desired and controlled trajectories, position errors and control inputs for Joints 1 and 3 of the 3-DOF MRR for *Case 1* are shown in Fig. 5.3 and Fig. 5.4, respectively. To compare the results with and without the brake engaged and to confirm the effectiveness of the SA-MRR, the tracking errors and control inputs at each joint are compared for the two configurations and the results are shown in Table 5.5.
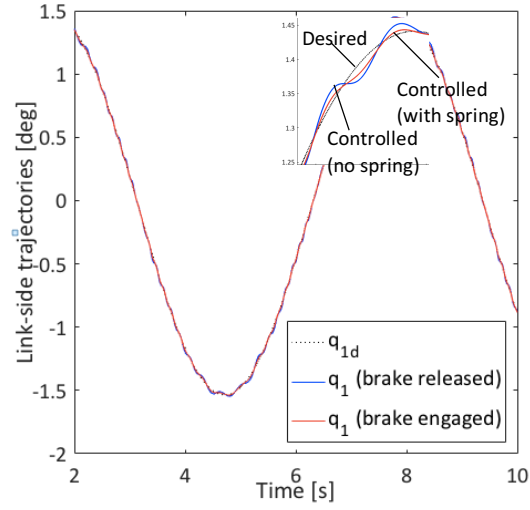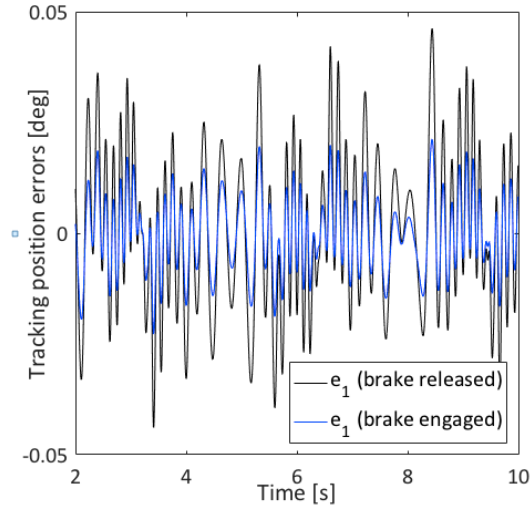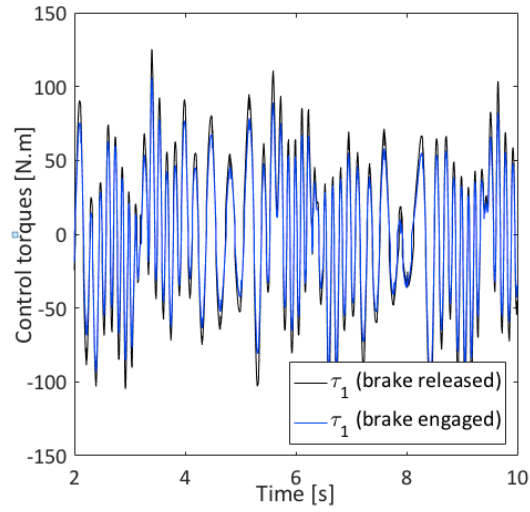
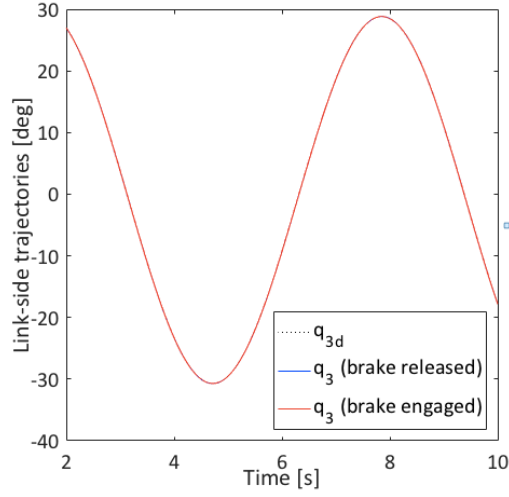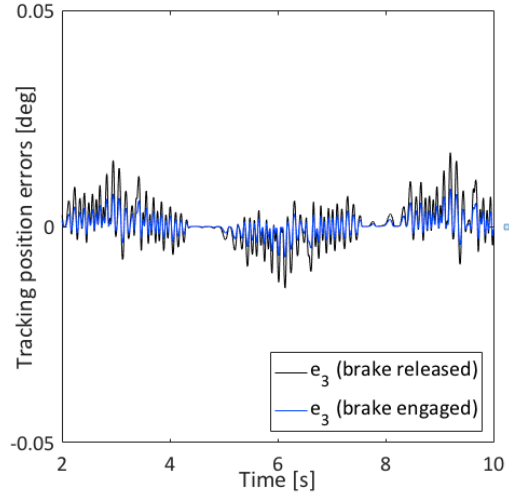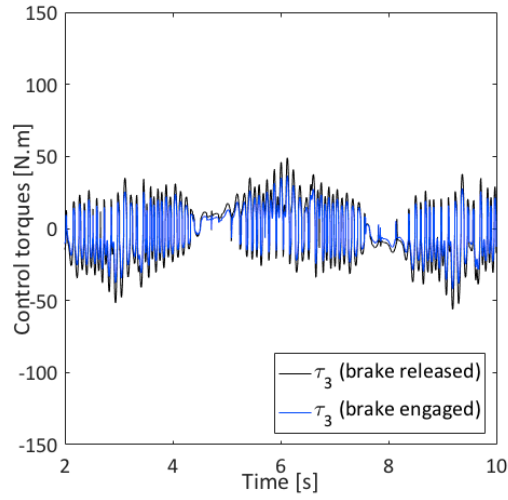Fig. 5.3: *Case 1:* Simulation results for Joint 1. (a) Link-side trajectories, (b) Position errors, (c) Control inputs.
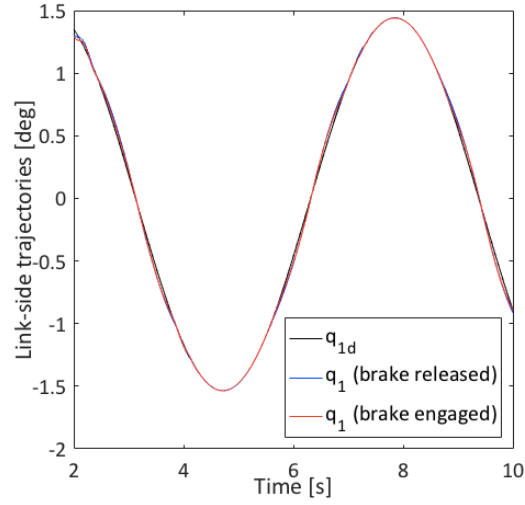
Fig. 5.4: *Case 1:* Simulation results for Joint 3. (a) Link-side trajectories, (b) Position errors, (c) Control inputs.

Table 5.5: Comparison of results for SA-MRR with restricted motion at Joints 1 & 2, and a 2 kg payload

Brake	Joint 1		Joint 2		Joint 3	
	$\max\{|e_1|\}$	$\max\{|\tau_1|\}$	$\max\{|e_2|\}$	$\max\{|\tau_2|\}$	$\max\{|e_3|\}$	$\max\{|\tau_3|\}$
Released	0.046°	125.15 Nm	0.072°	173.56 Nm	0.017°	56.23 Nm
Engaged	0.023°	105.88 Nm	0.030°	135.33 Nm	0.009°	41.99 Nm

In Table 5.5, the maximum tracking errors and control inputs for each joint with and without the brake engaged are listed. Fig. 5.3(a) illustrates a comparison of the trajectory tracking for the base joint with and without the brake engaged. When the brake is activated at the base joint, with the assistance of the spring, the required motor torque is reduced by 13.0% and the tracking error is 50.0% smaller. This is a vast difference in tracking error, even though the control parameters remain the same for both tests. For Joint 2, the maximum control input is reduced by 22.03% and the maximum tracking error is reduced by 58.33%. The third joint did not have a spring in the MRR module and its motion was not restricted to a small area. Yet, the maximum control input was reduced by 25.32% and the tracking error went down by 47.06%.

Comparing the trajectory tracking results for the three joints for *Case 1*, the joint positions have higher tracking errors for the first two joints and the third joint has almost perfect trajectory when the brake is engaged at the first two joints. From these results, it is clear that the SA-MRR improves the manipulator's trajectory tracking performance significantly while keeping it restrained in a limited working range. The trajectories followed by all joints in *Case 2* further reiterate the improved tracking accuracy provided by the additional torque from the spring when the brake is activated.
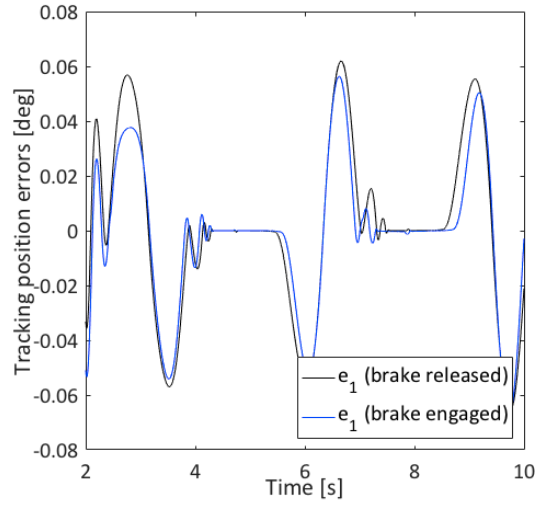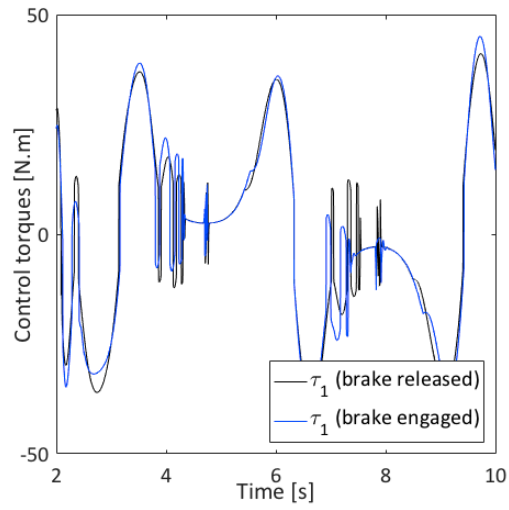
Fig. 5.5: *Case 2:* Simulation results for Joint 1. (a) Link-side position, (b) Position errors, (c) Control inputs.

Fig. 5.5 shows the performance of the base joint of a 3-DOF SA-MRR with spring-brake system activated at all joints. In *Case 2*, the trajectory of each joint was limited to $\pm 1.5°$ and a cosinusoidal trajectory was followed. The control inputs in Fig. 5.5(c) are about a third of the required control torques in *Case 1*. It should be noted that the motion of the third joint in *Case 1* is 20 times higher than the motion of the third joint in *Case 2*. Since the trajectory in *Case 1* is different for the third joint, it affects the control inputs required for best performance from the robot manipulator. Comparing the outputs of *Case 1 & 2*, it can be concluded that the SA-MRR provides robust performance in terms of trajectory tracking over a wide working envelope. The SA-MRR will perform well with spring-assistance as long as the maximum motor output torque is not reached.

When the spring-brake system is activated at a joint, the mechanical lock ensures that the joint position does not exceed the limit set by the maximum spring deformation. Since the robot is limited to a small working range, it is possible for a human operator to work in its vicinity without getting negatively affected by its presence. Thus, the addition of the spring-brake system to a conventional MRR module not only improves its trajectory tracking ability and lowers power consumption, it also creates a safe working environment for humans to work alongside the robot manipulator.

The joint trajectories in this section have the same joint limits but the joint limits of the SA-MRR modules can be modified according to the workspace required for a specific application. Since the power spring is simply inserted in the spring case as shown in Fig. 2.1, it can be exchanged for another spring with different spring characteristics without making any major changes to the joint design. Power springs with different stiffness can be used in an SA-MRR module to generate a variety of working envelopes. This property of the SA-MRR modules make

them an excellent choice for a wide range of applications that require movement in a restricted range.

5.2.4 Simulations for handling heavy payload with spring-assisted motion

To demonstrate the SA-MRR's enhanced actuator output and heavy payload handling capabilities, two cases are studied with three different payloads attached to the end-effector. A sinusoidal trajectory is followed by the SA-MRR modules in all simulation tests.

In *Case 1*, the control torques for Joint 3 of the SA-MRR with restricted motion at the first two joints is compared for handling heavy payloads. Fig. 5.6 shows the control inputs for the third joint while carrying a different payload in each case. From these plots, one can see that the required motor torque is significantly reduced for test runs with 2 kg and 3 kg payloads when the brake is engaged. For a payload of 5 kg, the control torque does not show a big difference in the control inputs with and without the brake engaged. As the payload is increased, the required control input increases. A similar pattern is observed in Fig. 5.7 for *Case 2,* which compared the payload handling capacity of the SA-MRR with restricted motion at all three joints.
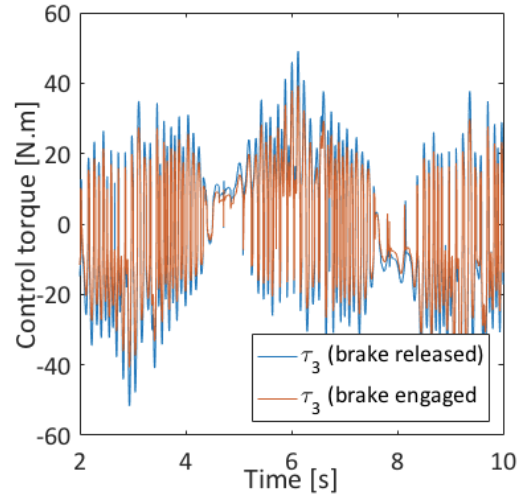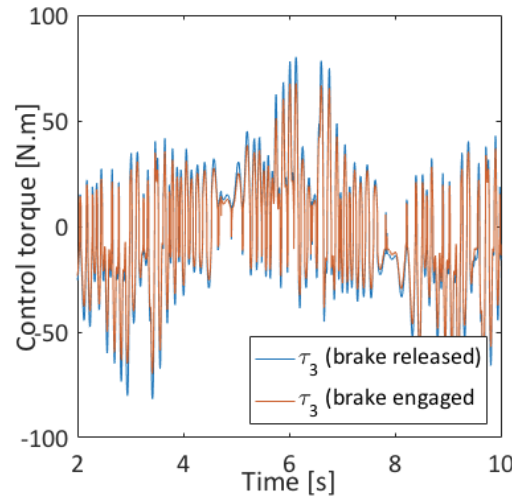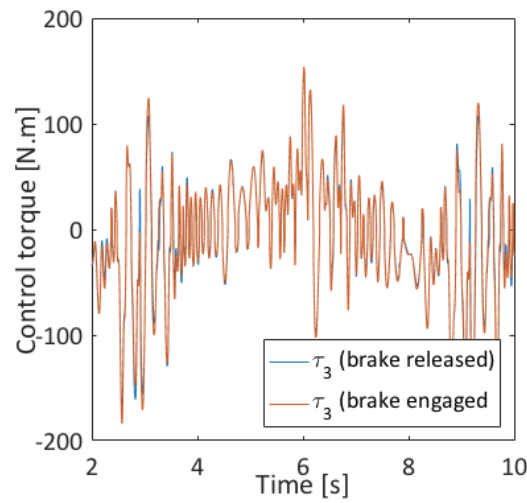
Fig. 5.6: *Case 1*: Control inputs required at Joint 3 to handle a payload at the end-effector. (a) 2kg, (b) 3 kg, (c) 5 kg.

(a)

(b)

(c)

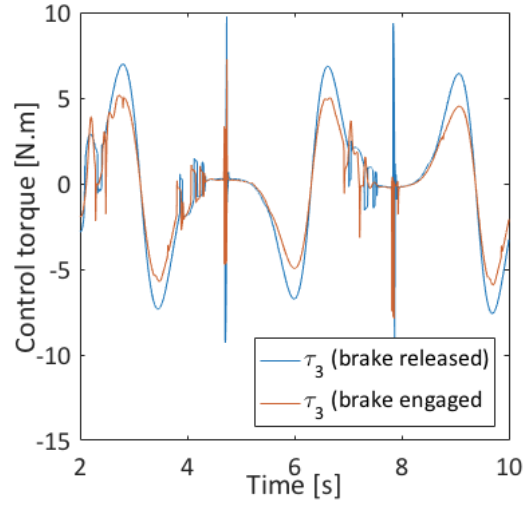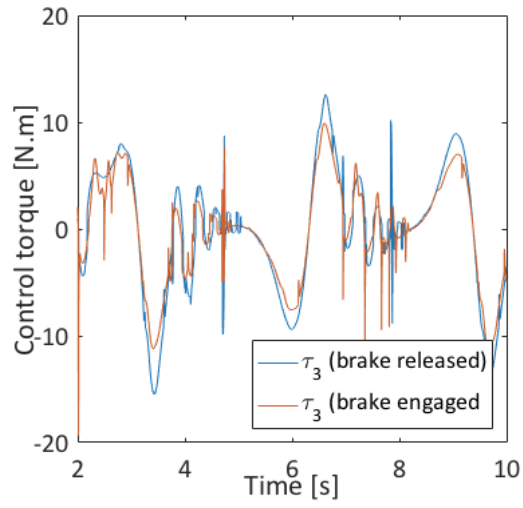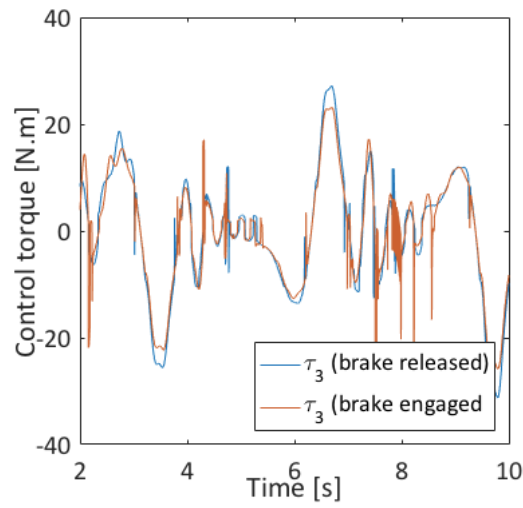Fig. 5.7: *Case 2:* Control inputs required as Joint 3 to handle a payload at the end-effector. (a) 2 kg, (b) 3 kg, (c) 5kg.

For *Case 2*, the control torques are lower compared to the inputs required in *Case 1* to handle the same payloads. The required control torque is lower in this case because all joints have restricted movement. The change in control inputs follows the same pattern as *Case 1* as the joint controller is still the same and the control torque is calculated to compensate for the system errors, which are much lower for a smaller working envelope and lower velocity. With the assistance of the spring-generated moment, the required motor torque is further reduced for the test runs with 3 kg and 5 kg payloads. Thus, the addition of the spring-brake system relieves the motor of the additional stress of handling heavy payloads, allowing the actuator to control the joint trajectories with higher tracking accuracy with the distributed robust controller.

From the simulation studies conducted in this thesis, it is clear that the addition of the spring-brake system, along with the distributed robust controller, to a conventional MRR module increases the manipulator's capability to track a trajectory and handle heavy payloads. From the tests conducted for trajectory tracking, it can be concluded that the robot performs better when all joints operate in a restricted working range compared to when one of the joints has a wider working envelope.

6. Conclusions and Future Work

6.1 Conclusions

This thesis presents a study of the applications of the spring-assisted modular and reconfigurable robots, which is equipped with a power spring and magnetic brake at each joint. The trajectory tracking and payload handling capabilities of the SA-MRR were studied with the safety aspect for human-robot collaboration as the main focus. Simulation tests were conducted with different robot working envelopes with and without the spring-brake system activated for the MRR modules. It was concluded that the SA-MRR provided excellent tracking accuracy in a relatively small working area while balancing the robot using the additional torque provided by the spring-brake system. With the distributed control method, it was easy to add SA-MRR modules to the simulations without retuning the control parameters for each additional joint. The SA-MRR, along with the distributed robust control method [36], provides excellent tracking accuracy and payload handling capability in a small working envelope, that can be easily defined based on the robot's applications and the level of interaction with humans. This has been verified by simulations and will be demonstrated with experiments in the future.

6.2 Future work

Simulation cases studied in this thesis validated the benefits of having the spring-brake system added to a conventional MRR module, the next step is to implement this design in an actual MRR module. After obtaining required data from the joint sensors, the simulation can be further improved by including the variable control parameters and variable parametric uncertainties to the Simulink models.

The applications discussed in this thesis are only limited to restricting the robot's working range for tasks that require human presence in the robot's workspace. Once the SA-MRR is successfully redesigned and tested, more experiments can be designed where data from joint sensors and other visual systems to sense the robot's environment are integrated for applications that require human-robot collaboration at all times in an unstructured work environment.

Appendix A – Simulink models for 3-DOF SA-MRR Manipulator

A1. Simulink Model of the 3-DOF SA-MRR
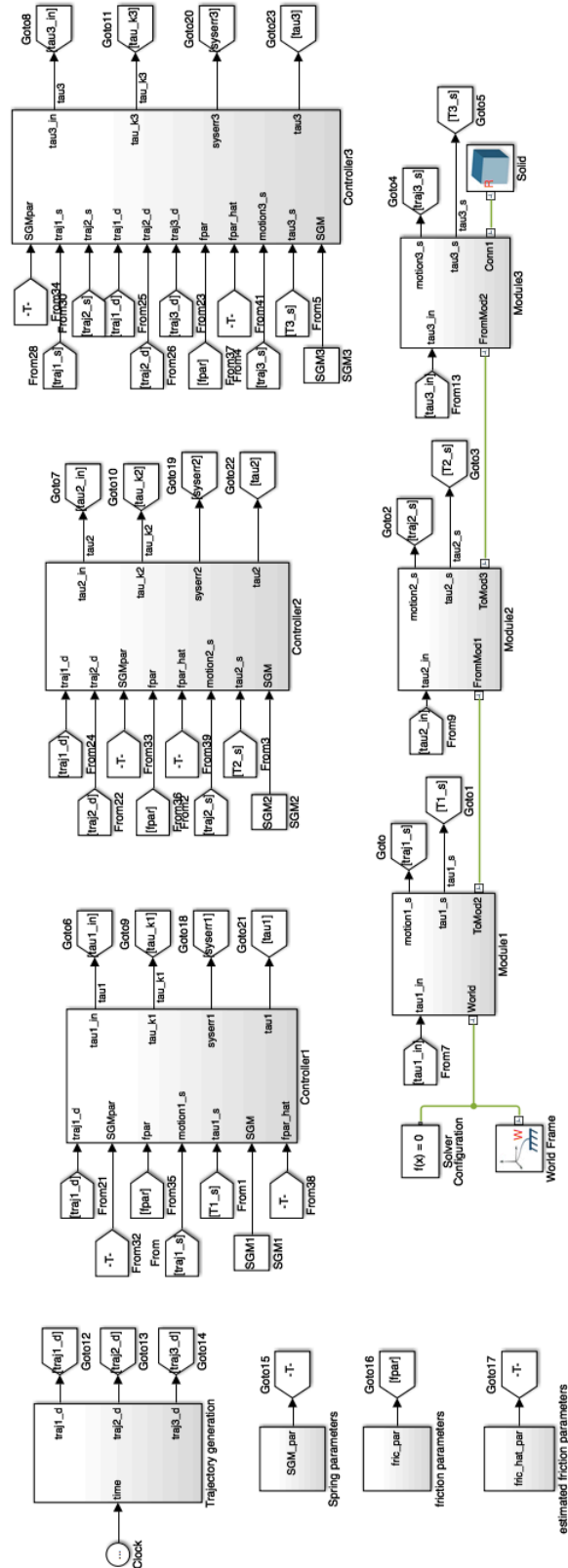
A2. Model for 3-DOF SA-MRR using 2nd Generation SimMechanics Blocks
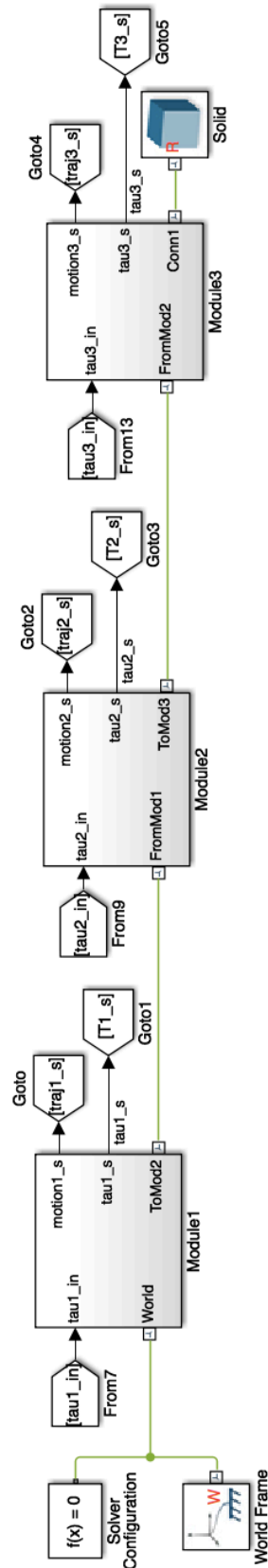
A3. SimMechanics Model for One Joint
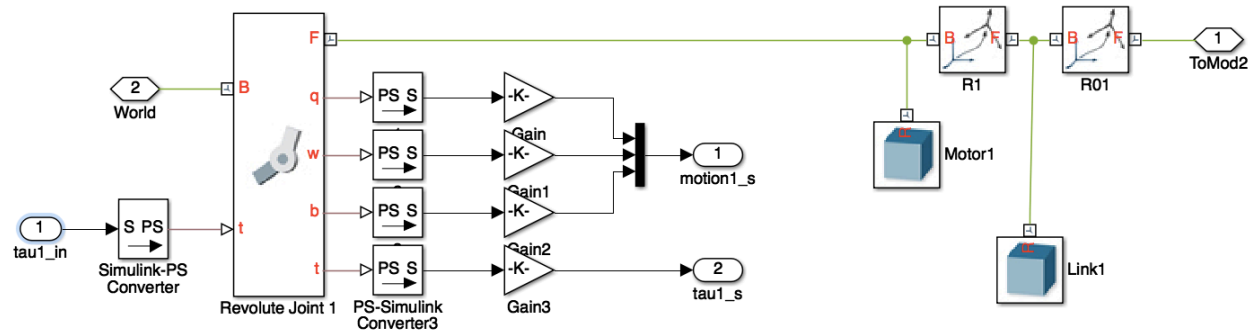
A4. Controller Subsystem for a Joint Module
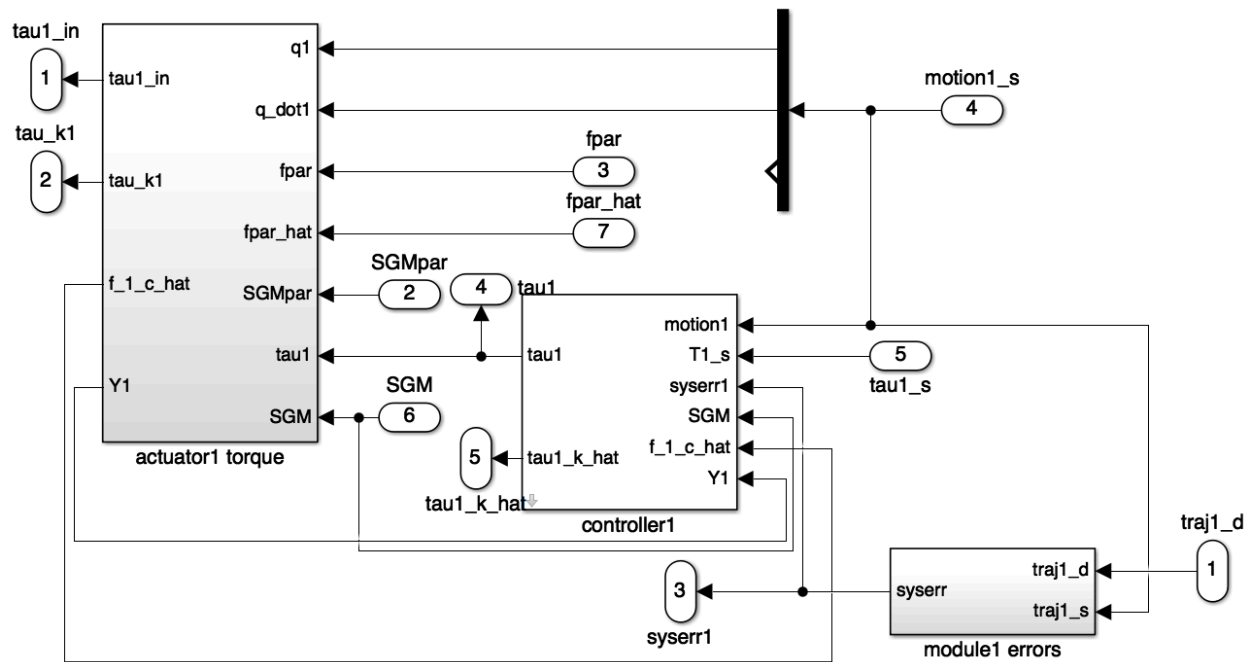
A5. Input Torque Calculation Subsystem Model
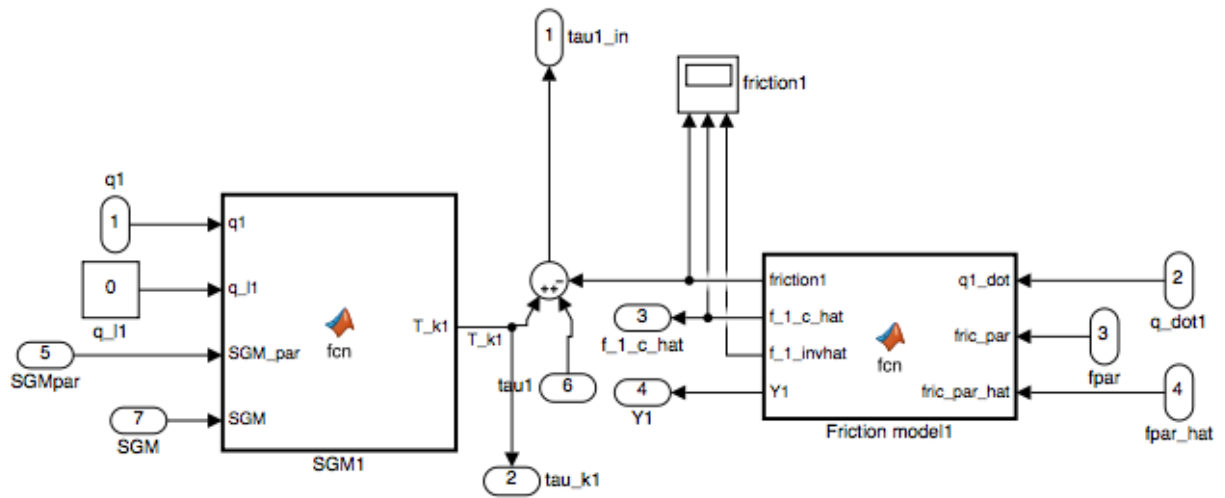
Appendix B – Matlab/Simulink Code for Simulations

B1. Matlab Script to Plot Workspace of a 3-DOF Robot Manipulator

```matlab
clear all
clc


%% Declare variables
syms q1 q2 q3 a1 a2 a3 l1 l2 l3
% q1,q2,q3 are joint variables,
% d1 is distance from o0 to b1 measured along z0
% l1,l2,l3 are the locations of C.M. of each link

a1 = 0.406; % l1
a2 = 0.4;   % l1
a3 = 0.4;   % l3
d1 = 0;
d2 = 0;
d3 = 0;
alpha1 = 0;
alpha2 = 90;
alpha3 = 0;
theta1 = q1;
theta2 = q2;
theta3 = q3;


%% Compute H-matrices
H01 = Hmat(a1,alpha1,d1,theta1);
H12 = Hmat(a2,alpha2,d2,theta2);
H23 = Hmat(a3,alpha3,d3,theta3);

H02 = H01*H12;
H03 = H01*H12*H23;


%% Extract joint position vectors from H-matrices
p00 = [0;0;0];
p01 = H01([1 2 3],[4]);
p02 = H02([1 2 3],[4]);
p03 = H03([1 2 3],[4]);


%% Extract rotation matrices from H-matrices
R00 = eye(3);
R01 = H01([1 2 3],[1 2 3]);
R02 = H02([1 2 3],[1 2 3]);
R03 = H03([1 2 3],[1 2 3]);


%% Plot 3D Workspace
```

```matlab
subs(H03, {theta1, theta2, theta3}, {0, 0, 0});

%%% Build the robot using the SerialLink function
L(1) = Link([0 0 0.406 0]);
L(2) = Link([0 0 0.4 pi/2]);
L(3) = Link([0 0 0.4 0]);

RobotArm =  SerialLink(L);
RobotArm.name = '3DOF MRR';
RobotArm.plot([0 0 0]);

RobotArm.fkine([q1 q2 q3]);

%%% To obtain the workspace, we can fix a1 and vary a2 and a3 to
%%% calculate the end-effector's position and plot it using the
%%% surf function.
RobotArm.plot([0 0 0]);
xlabel('X [m]');
ylabel('Y [m]');
zlabel('Z [m]');
title('Workspace of 3-DOF MRR');
hold on;

N = 20;

% First, choose the k or l value for the normal or limited working
% range option
% To plot the workspace for each quadrant, use the four TR terms one
% by one along with the two options for k to get a full 3-D workspace
% plot


%% for normal, unrestricted working range
for k = 0:0.1:(pi/3)
% for k = (-pi/3):0.1:0


%% for limited working range
% for l = 0:0.1:(2*pi/3)
% for l = (-2*pi/3):0.1:0


% For normal, unrestricted working range
    for i = 1:N+1
        for j = 1:N+1
            TR = RobotArm.fkine([k pi/2*((i-1)/N) pi/2*((j-1)/N)]);
%            TR = RobotArm.fkine([k -pi/2*((i-1)/N) pi/2*((j-1)/N)]);
%            TR = RobotArm.fkine([k pi/2*((i-1)/N) -pi/2*((j-1)/N)]);
%            TR = RobotArm.fkine([k -pi/2*((i-1)/N) -pi/2*((j-1)/N)]);
```

56

```matlab
%% For limited working range
%         TR = RobotArm.fkine([pi/3*((i-1)/N) l/101 l/101]);
%         TR = RobotArm.fkine([pi/3*((i-1)/N) -l/101 l/101]);
%         TR = RobotArm.fkine([-pi/3*((i-1)/N) l/101 -l/101]);
%         TR = RobotArm.fkine([-pi/3*((i-1)/N) -l/101 -l/101]);

        abc(i,j,:) = TR(:,4);
    end
  end

surf(abc(:,:,1), abc(:,:,2), abc(:,:,3));

end
```

B2. Matlab Script to Setup SA-MRR's Physical Parameters, Control Parameters and Uncertainty Bounds

```matlab
clear all
clc


%% set SGM = 1 to activate brakes
SGM1 = 0;
SGM2 = 0;
SGM3 = 0;


%% Link basic inertial information
% Link length in [m]
l1 = 0.406;
l2 = 0.4;
l3 = 0.4;


% Sim dimensions in [m]
d1m = 0.164;
h1m = 0.160;

d1l = 0.084;
h1l = 0.240;

d2m = 0.160;
h2m = 0.160;

d2l = 0.084;
h2l = 0.240;

d3m = 0.160;
h3m = 0.160;

d3l = 0.084;
h3l = 0.240;


% Link parameters
m_l1 = 0.64869; % [kg]
I_l1 = [0.00459583068,0, 0; 0, 0.00464928047, 0; 0, 0, 0.00130317108]; % [kg*m^2] taken at center of
mass

m_l2 = 0.64869; % [kg]
I_l2 = [0.00459583068,0, 0; 0, 0.00464928047, 0; 0, 0, 0.00130317108]; % [kg*m^2] taken at center of
mass

m_l3 = 0.64869; % [kg]
```

```matlab
I_l3 = [0.00459583068,0, 0; 0, 0.00464928047, 0; 0, 0, 0.00130317108]; % [kg*m^2] taken at center of
mass


% Motor parameters
m_m1 = 6; % [kg]
I_m1 = [0,0,0;0,0,0;0,0,0.0192]; % [kg*m^2] taken at center of mass

m_m2 = 6; % [kg]
I_m2 = [0,0,0;0,0,0;0,0,0.0192]; % [kg*m^2] taken at center of mass

m_m3 = 6; % [kg]
I_m3 = [0,0,0;0,0,0;0,0,0.0192]; % [kg*m^2] taken at center of mass

gammai = 100;   % gear ratio


%% Spring characteristics
T_kimax = 50.0;
U_imax = 1.5*pi/180;

ki1 = 2*T_kimax/U_imax;
ki2 = -T_kimax/(U_imax^2);


%% Control parameters

% initial value for integral term
global integ
integ = 0;

lambdai = 160;

mu_theta_ij = 0.01;
mu_Theta_ijk = 0.01;
mu_ki1 = 1.00;
mu_ki2 = 1.00;
mu_Fic = 0.10;

E_F = [1e-6 1e-4 1e-4 1e-2];
E_qdot = 0.001;
E_thetaij = 0.010;
E_Thetaijk = 0.010;
Er = 0.010;

% Gains for noSGM and wSGM
k_I1 = 15.0;
k_I2 = 15.0;
k_I3 = 15.0;

% Constant friction model parameters
```

```matlab
b_c = 10.16588;
f_cc = 4.95;
f_sc = 5.403;
f_Tc = 5144.1576;
F_c = [b_c; f_cc; f_sc; f_Tc];

% Estimated friction parameters
b_c_hat = b_c*0.8;
f_cc_hat = f_cc*0.8;
f_sc_hat = f_sc*0.8;
f_Tc_hat = f_Tc*0.8;
F_c_hat = [b_c_hat; f_cc_hat; f_sc_hat; f_Tc_hat];

F_c_tilde = F_c - F_c_hat;

% Constant uncertainty bounds for friction
rho_bi = 0.2*b_c;
rho_fci = 0.2*f_cc;
rho_fsi = 0.2*f_sc;
rho_fTi = 0.2*f_Tc;

rho_F = [rho_bi rho_fci rho_fsi rho_fTi];

% Other system uncertainty bounds
rho_fqi = 0.5;
rho_thetaij = 1.0;
rho_Thetaijk = 1.0;
rho_Tsi = 10.0;
rho_eai = 0.01;
```

B3. Matlab Function to Calculate Friction

```matlab
function [friction1, f_1_c_hat, f_1_invhat, Y1] = fcn(q1_dot, fric_par, fric_par_hat)

b_1_c = fric_par(1);
f_c1_c = fric_par(2);
f_s1_c = fric_par(3);
f_T1_c = fric_par(4);

b_1_c_hat  = fric_par_hat(1);
f_c1_c_hat = fric_par_hat(2);
f_s1_c_hat = fric_par_hat(3);
f_T1_c_hat = fric_par_hat(4);
E_q1dot = fric_par_hat(5);

% Sign function
if q1_dot > 0
    sgn_q1dot = 1;
elseif q1_dot < 0
    sgn_q1dot = -1;
else
    sgn_q1dot = 0;
end

% Saturation function for q1_dot and E_q1dot
if (abs(q1_dot) > E_q1dot)
    sat_q1dot_Eq1dot = q1_dot/abs(q1_dot);
elseif (abs(q1_dot) <= E_q1dot)
    sat_q1dot_Eq1dot = q1_dot/E_q1dot;
else
    sat_q1dot_Eq1dot = 1;
end

% Friction calculations
F_1_c = [b_1_c; f_c1_c; f_s1_c; f_T1_c];
F_1_c_hat = [b_1_c_hat; f_c1_c_hat; f_s1_c_hat; f_T1_c_hat];

F_1_c_tilde = F_1_c - F_1_c_hat;

f_1_c_hat = (f_c1_c_hat + (f_s1_c_hat*exp(-f_T1_c_hat*q1_dot^2)))*sat_q1dot_Eq1dot + b_1_c_hat*q1_dot;

f_1_invhat = (f_c1_c + (f_s1_c*exp(-f_T1_c*q1_dot^2))) * (sgn_q1dot - sat_q1dot_Eq1dot);

Y1 = [0 1 exp(-f_T1_c_hat*q1_dot^2) -f_s1_c_hat*q1_dot^2*exp(-f_T1_c_hat*q1_dot^2)]*sat_q1dot_Eq1dot + [q1_dot 0 0 0];

friction1 = f_1_c_hat + Y1 * F_1_c_tilde + f_1_invhat;
```

B4. Matlab Function to Calculate SGM

```matlab
function T_ki = fcn(qi,q_li,SGM_par, SGM)
%#codegen
% q_l can determined by using a logic that detects the brake engage/release
% switch signal and record the link position when the brake is engaged

% max spring deformation is 150 degrees for the simulations
% max SGM is considered to be the same as the maximum actuator moment

% for now, we use a constant value for q_l
%q_l = 0;

ki1 = SGM_par(1);
ki2 = SGM_par(2);
gammai = SGM_par(3);
U_imax = SGM_par(4);
T_kimax = SGM_par(5);

if SGM == 1

    if -U_imax < (q_li-qi) <= U_imax
      T_ki = T_kimax * (1-(((q_li-qi)/U_imax)-1)^2);
    else
        msg = 'cannot go over the maximum spring deformation! STOP.'
        error(msg)
    end

else
    T_ki = 0;
end
```

B5. Matlab Function to Calculate Estimated Parameters using Adaptation Law

- Code to calculate spring constants

```
ki1_hat_dot = mu_ki1*(qi_l-qi)*ri;
ki2_hat_dot = mu_ki2 * (qi_l-qi)^2 *ri;
```

- Code to calculate estimated SGM

```
function Ti_k_hat = fcn(gammai, qi, qi_l, ki1_hat, ki2_hat, SGM)

if SGM ==1
    Ti_k_hat = ki1_hat*(qi_l-qi) + ki2_hat*((qi_l-qi)^2);
else
    Ti_k_hat = 0;
end

end
```

- Code to calculate theta_cij

```
function theta_cij_hat_dot = fcn(syserri, trajj_d, mu_thetaij)
ri = syserri(2);
qj_d_ddot = trajj_d(3);

theta_cij_hat_dot = -mu_thetaij*qj_d_ddot*r3;
```

- Code to calculate Theta_cijk

```
function Theta_cijk_hat_dot = fcn(syserri,motionk,motionj,mu_Thetaijk)
qk_dot = motionk(2);
qj_dot = motionj(2);
ri = syserri(2);

Theta_cijk_hat_dot = -mu_Thetaijk*qk_dot*qj_dot*r3;
```

B6. Matlab Function to Calculate Control Torque

```matlab
function [tau3,tau_r3,r3] = fcn(motion1, motion2, motion3, tau_s3_hat, err3, tau_k3_hat, theta_c31_hat, theta_c32_hat, Theta_c321_hat, traj1_d, traj2_d, sys_par, fric_par, f_3_c_hat, Y3)

q1_dot = motion1(2);
q2_dot = motion2(2);

q3 = motion3(1);
q3_dot = motion3(2);
q3_ddot = motion3(3);

% System errors
e3 = err3(1);
r3 = err3(2);
a3 = err3(3);

% Desired Trajectory
q1_d_ddot = traj1_d(3);
q2_d_ddot = traj2_d(3);

% Control parameters and bounds
I_m3 = sys_par(1);
gamma3 = sys_par(2);
kI3 = sys_par(3);
Er3 = sys_par(4);
E_q3dot = sys_par(5);
rho_Ts3 = sys_par(6);
E_theta31 = sys_par(7);
E_theta32 = sys_par(8);
rho_theta31 = sys_par(9);
rho_theta32 = sys_par(10);
E_Theta321 = sys_par(11);
rho_Theta321 = sys_par(12);
rho_ea1 = sys_par(13);
rho_ea2 = sys_par(14);

b_3_c_hat = fric_par(1);
f_c3_c_hat = fric_par(2);
f_s3_c_hat = fric_par(3);
f_T3_c_hat = fric_par(4);
rho_fq3 = fric_par(5);
rho_F3 = zeros(1,4);
rho_F3(1) = fric_par(6);
rho_F3(2) = fric_par(7);
rho_F3(3) = fric_par(8);
rho_F3(4) = fric_par(9);
E_F3 = zeros(1,4);
E_F3(1) = fric_par(10);
E_F3(2) = fric_par(11);
E_F3(3) = fric_par(12);
```

```matlab
E_F3(4) = fric_par(13);

r_prev3 = r3;

% Saturation function for r3 and Er3
if all(abs(r3) > Er3)
    sat_rEr3 = r3/abs(r3);
elseif all(abs(r3) <= Er3)
    sat_rEr3 = r3/Er3;
else
    sat_rEr3 = 1;
end

sat_r3Y_EF3 = zeros(1,4);
rho_F3Y = zeros(1,4);

for j = 1:4
    % Saturation function for r3*Y(q3_dot) and E_F3
    if abs(r3*Y3(j)) > E_F3(j)
        sat_r3Y_EF3(j) = (r3*Y3(j))/(abs(r3*Y3(j)));
    elseif abs(r3*(Y3(j))) <= E_F3(j)
        sat_r3Y_EF3(j) = (r3*Y3(j))/E_F3(j);
    else
        sat_r3Y_EF3(j) = 1;
    end

    % rhoF3Y calculation
    rho_F3Y(j) = rho_F3(j) *Y3(j)*sat_r3Y_EF3(j);
end

% Saturation function for r3*q1_d_ddot and E_theta31
if abs(r3*q1_d_ddot) > E_theta31
    sat_r3q1ddd_ET31 = (r3*q1_d_ddot)/(abs(r3*q1_d_ddot));
elseif abs(r3*q1_d_ddot) <= E_theta31
    sat_r3q1ddd_ET31 = (r3*q1_d_ddot)/E_theta31;
else
    sat_r3q1ddd_ET31 = 1;
end

% Saturation function for r3*q2_d_ddot and E_theta32
if abs(r3*q2_d_ddot) > E_theta32
    sat_r3q2ddd_ET32 = (r3*q2_d_ddot)/(abs(r3*q2_d_ddot));
elseif abs(r3*q2_d_ddot) <= E_theta32
    sat_r3q2ddd_ET32 = (r3*q2_d_ddot)/E_theta32;
else
    sat_r3q2ddd_ET32 = 1;
end

% Saturation function for r3*q1_dot*q2_dot and E_Theta321
if abs(r3*q1_dot*q2_dot) > E_Theta321
    sat_r3q1dq2d_ET321 = (r3*q1_dot*q2_dot)/(abs(r3*q1_dot*q2_dot));
```

```
elseif abs(r3*q1_dot*q2_dot) <= E_Theta321
    sat_r3q1dq2d_ET321 = (r3*q1_dot*q2_dot)/E_Theta321;
else
    sat_r3q1dq2d_ET321 = 1;
end

rho_b3  = rho_F3(1);
rho_fc3 = rho_F3(2);
rho_fs3 = rho_F3(3);
rho_fT3 = rho_F3(4);

rho_f3 = rho_fc3 + rho_fs3*exp(rho_fT3*E_q3dot^2);

% decomposition-based robust control term
tau_r3 = -(rho_f3+(rho_Ts3/gamma3)+I_m3*(rho_ea1+rho_ea2)) * sat_rEr3 - sum(rho_F3Y) -
I_m3*(rho_theta31*q1_d_ddot*sat_r3q1ddd_ET31 + rho_theta32*q2_d_ddot*sat_r3q2ddd_ET32) -
I_m3*rho_Theta321*q1_dot*q2_dot*sat_r3q1dq2d_ET321;

% Note that the integral term of the controller is computed as a simple sum by adding the error in each
iteration
integ = r3 + r_prev3;

tau3 = I_m3*gamma3*a3 - tau_s3_hat/gamma3 - tau_k3_hat - kI3*integ + f_3_c_hat +
I_m3*theta_c31_hat*q1_d_ddot + I_m3*theta_c32_hat*q2_d_ddot +
I_m3*Theta_c321_hat*q1_dot*q2_dot + tau_r3;
```

References

[1] O. Khatib, K. Yokoi, O. Brock, K. Chang and A. Casal, "Robots in human environments: Basic autonomous capabilities," *International Journal of Robotics Research,* vol. 18, no. 7, p. 684–696, 1999.

[2] B. Siciliano, L. Sciavicco, L. Villani and G. Oriolo, Robotics: Modelling, Planning and Control, London: Springer, 2010.

[3] J. Choi, S. Park, W. Lee and S.-C. Kang, "Design of a robot joint with variable stiffness," in *IEEE International Conference on Robotics and Automation*, Pasadena, California, USA, 2008.

[4] W. Lee, J. Choi and S. Kang, "Spring-Clutch: A safe torque limiter based on a spring and CAM mechanism with the ability to reinitialize its position," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, 2009.

[5] S. Hong, J. Choi, W. Lee, S. Kang and H. Lee, "Development of a safe manipulator using spring-clutches," in *13th International Conference on Control, Automation and Systems (ICCAS)*, Gwangju, 2013.

[6] A. Gopalswamy, P. Gupta and M. Vidyasagar, "A new parallelogram linkage configuration for gravity compensation using torsional springs," in *1992 IEEE International Conference on Robotics and Automation*, Nice, France, 1992.

[7] J. J. Park and J. B. Song, "Safe joint mechanism using inclined link with springs for collision safety and positioning accuracy of a robot arm," in *2010 IEEE International Conference on Robotics and Automation (ICRA)*, Anchorage, Alaska, USA, 3-7 May 2010.

[8] T. Hondo and I. Mizuuchi, "Analysis of the 1-Joint Spring-Motor Coupling System and optimization criteria focusing on the velocity increasing effect," in *2011 IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 9-13 May 2011.

[9] R. Schiavi, G. Grioli, S. Sen and A. Bicchi, "VSA-II: a novel prototype of variable stiffness actuator for safe and performing robots interacting with humans," in *IEEE International Conference on Robotics and Automation*, Pasadena, California, 2008.

[10] J. Versace, "A review of the severity index," in *15th Stapp Car Crash Conference*, 1971.

[11] A. M. Zanchettin, B. Lacevic and P. Rocco, "A novel passivity-based control law for safe human-robot coexistence," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura, Algarve, Portugal, October 7-12, 2012.

[12] R. Schiavi, A. Bicchi and F. Flacco, "Integration of Active and Passive Compliance Control for Safe Human-Robot Coexistence," in *2009 IEEE International Conference on Robotics and Automation*, Kobe, Japan, May 12-17, 2009.

[13] A. M. Zanchettin and P. Rocco, "Path-consistent safety in mixed human-robot collaborative manufacturing environments," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Tokyo, Japan, November 3-7, 2013.

[14] C.-P. Lam, C.-T. Chou, K.-H. Chiang and L.-C. Fu, "Human-Centered Robot Navigation—Towards a Harmoniously Human–Robot Coexisting Environment," *IEEE Transactions on Robotics*, vol. 27, no. 1, pp. 99-112, February 2011.

[15] M. Yim, W.-M. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, E. Klavins and G. S. Chirikjian, "Modular self-reconfigurable robot systems - challenges and opportunities for future," *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, pp. 43-52, 2007.

[16] T. Fukuda and S. Nakagawa, "Dynamic Reconfigurable Robotic System," in *IEEE Conference on Robotics and Automation*, Philadelphia, PA, USA, 1988.

[17] M. Yim, D. G. Duff and K. D. Roufas, "Polybot: a modular reconfigurable robot," in *IEEE International Conference on Robotics and Automation*, 2000.

[18] C. Ampatzis, E. Tuci, V. Trianni, A. L. Christensen and M. Dorigo, "Evolving Self-Assembly in Autonomous Homogeneous Robots: Experiments with Two Physical Robots," *Artificial Life*, vol. 15, no. 4, pp. 465-484, October 2009.

[19] H. Wei, H. Li and T. Wang, "An Evolutionary Swarm Self-assembly Robot: from Concept to Prototype," in *IEEE International Conference on Robotics and Biomimetics*, Tianjin, China, 2010.

[20] H. Wei, Y. Chen, J. Tan and T. Wang, "Sambot: A Self-Assembly Modular Robot System," *IEEE/ASME Transactions on Mechatronics*, vol. 16, no. 4, pp. 745-757, August 2011.

[21] R. Hui, N. Kircanski, A. A. Goldenberg, C. Zhou, P. Kuzan, J. Wiercienski, D. Gershon and P. Sinha, "Design of the IRIS facility – a modular, reconfigurable and expandable robot test bed," in *IEEE International Conference on Robotics and Automation*, 1993.

[22] R. M. Setchi and N. Lagos, "Reconfigurability and reconfigurable manufacturing systems: state-of-the-art review," in *IEEE International Conference on Industrial Informatics*, Berlin, Germany, 2004.

[23] G. Liu, S. Abdul and A. A. Goldenberg, "Distributed Modular and Reconfigurable Robot Control with Torque Sensing," in *IEEE International Conference on Mechatronics and Automation*, Luoyang, China, June 25 - 28, 2006.

[24] G. Liu and Y. Liu, "Spring Assisted Modular and Reconfigurable Robot: Design, Analysis and Control," in *IEEE International Conference on Robotics and Automation*, Anchorage, Alaska, USA, May 3-8, 2010.

[25] G. Liu, Y. Liu and A. A. Goldenberg, "Design, analysis, and control of a spring-assisted modular and reconfigurable robot," *IEEE/ASME Transactions on Mechatronics,* vol. 16, no. 4, pp. 695-706, August 2011.

[26] R. E. Joerres, "Springs" in Standard Handbook of Machine Design, 2nd ed., J. E. a. C. R. Mischke, Ed., New York: McGraw-Hill, 1996.

[27] T. G. Rogers and N. Y. Rochester, "Constant force compensation for power spring weight balance". U.S. Patent 5 054 162, 8 October 1991.

[28] G. Liu, "Decomposition-based friction compensation of mechanical systems," *Mechatronics* , vol. 12, no. 5, p. 755–769, 2002.

[29] "KR AGILUS sixx | KUKA AG," KUKA Robotics, [Online]. Available: https://www.kuka.com/en-us/products/robotics-systems/industrial-robots/kr-agilus-sixx. [Accessed 14 06 2016].

[30] "IRB 580 - Industrial Robots - Robotics | ABB," ABB, [Online]. Available: http://new.abb.com/products/robotics/industrial-robots/irb-580/irb-580-data. [Accessed 16 06 2016].

[31] B. Armstrong-Helouvry, P. Dupont and C. C. deWit, "A survery of models, analysis tools and compensation methods for the control of machines with friction," *Automatica,* vol. 30, no. 7, pp. 1083-1138, July 1994.

[32] G. Liu, A. A. Goldenberg and Y. Zhang, "Precise slow motion control of a direct-drive robot arm with velocity estimation and friction compensation," *Mechatronics,* vol. 14, no. 7, pp. 821-834, 2004.

[33] G. Liu and A. A. Goldenberg, "Uncertainty decomposition-based robust control of robot manipulators," *IEEE Transactions on Control Systems Technology,* vol. 4, no. 4, pp. 384-393, July 1996.

[34] G. Liu and A. A. Goldenberg, "Robust control of robot manipulators based on dynamics decomposition," *IEEE Transactions on Robotics and Automation,* vol. 13, no. 5, pp. 783-789, October 1997.

[35] G. Liu and A. A. Goldenberg, "Comparative study of robust saturation control of robot manipulators: analysis and experiments," *International Journal of Robotics Research,* vol. 15, no. 5, pp. 473-491, October 1996.

[36] G. Liu, S. Abdul and A. A. Goldenberg, "Distributed control of modular and reconfigurable robot with torque sensing," *Robotica,* vol. 26, no. 1, pp. 75-84, January 2008.