CUKUROVA UNIVERSITY

INSTITUTE OF NATURAL AND APPLIED SCIENCES

PhD THESIS

Implementation of Adaptive Interaction Torque Control on Modified Wrist Rehabilitation Robot (CWRR) Under Internal and External Disturbance Effects

Mustafa DAĞDELEN

Mechanical Engineering Department

March, 2024

CUKUROVA UNIVERSITY

INSTITUTE OF NATURAL AND APPLIED SCIENCES

PhD THESIS APPROVAL

Implementation of Adaptive Interaction Torque Control on Modified Wrist Rehabilitation Robot (CWRR) Under Internal and External Disturbance Effects

Mustafa DAĞDELEN

Mechanical Engineering Department

This Doctorate Thesis was evaluated by the following Jury Members on 08/03/2024 and was approved by unanimity of votes.

Jury : Assoc. Prof. Dr. Mehmet İlteriş SARIGEÇİLİ (Advisor) ……

: Assist. Prof. Dr. Durmuş Ali BİRCAN ……

: Assoc. Prof. Dr. Hüseyin MUTLU ……

: Assoc. Prof. Dr. Erinç ULUDAMAR ……

: Assoc. Prof. Dr. Melik KOYUNCU ……

This Thesis was written in the Department of Mechanical Engineering, Institute of Natural and Applied Sciences.
Thesis Number:

Prof. Dr. Sadık DİNÇER
Director
Institute of Natural and Applied Sciences

This work was supported by the Scientific Research Office of Çukurova University.
Project ID: FBA-2022-14180

CONTENTS

CUKUROVA UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES

PhD THESIS

Implementation of Adaptive Interaction Torque Control on Modified Wrist/Forearm Rehabilitation Robot (CWRR) Under Internal and External Disturbance Effects

Mustafa DAĞDELEN

Advisor: Assoc. Prof. Dr. Mehmet İlteriş SARIGEÇİLİ
2nd Advisor: Assoc. Prof. Dr. Necdet Sinan ÖZBEK

Department of Mechanical Engineering

ABSTRACT

In this thesis study, two-degrees-of-freedom, pneumatically actuated wrist/forearm rehabilitation robot called CWRR has ben modified to make it suitable for force/torque interaction control. For this aim, the mechanical and control equipment of the CWRR robot has been redesigned, manufactured and renewed. Extra sensors have been added to maximize the control performance of the robot. After the completition of the modernization process, flow modeling of the renewed precision pneumatic control valves has been carried out. It has been determined that proportional directional control valves did not comply with the existing flow models presented in the literature, and a new and effective compressible flow model has been proposed for proportional directional control valves. As the next stage, experimental studies have been carried out to identify the friction force/torque of the renewed pneumatic cylinders. At this point, novel automation algorithms for friction force/torque identification that will provide faster and most accurate friction parameter identification has been designed and implemented. In order to increase the safety of the interacting human user and to make the robot more compatible, an algorithm that detects human movement intentions has been developed and incorporated into the controller structure. Afterwards, a fuzzy logic adaptive PID controller structure has been designed that can track the target interaction torque successfully during the executed workout. For this purpose, fuzzy logic rules have been determined correctly, membership functions and the boundaries of these functions have been determined according to experimental data and experience of the expert. In order for the proposed robot to take correct control actions even under disturbing effects, algorithms have been developed that can reject both internal and external disturbance effects. To demonstrate the superior performance of the proposed controller, two different controllers have also been tested on the robot. One of these controllers is the Default controller that is built-in the valve by the manuafcturer and the other one is the conventional PID structure designed in series with this Default controller. These three different controllers have been tested in the real-time hardware-in-the loop environmet under four different scenarios wherein disturbance-free or under-disturbance conditions were evaluated. It has been confirmed by the tests that the proposed Cascade Fuzzy Adaptive PID structure have exhibited outstanding control performance in all scenarios compared to its counterparts.

Keywords: Adaptive control, Disturbance rejection algorithm, Friction identification, Pneumatics, Rehabilitation robot

ÇUKUROVA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DOKTORA TEZİ

Modifiye Edilen Bilek/Önkol Rehabilitasyon Robotu Üzerinde İç ve Dış Bozucu Etkiler Altında Uyarlamalı Etkileşim Tork Kontrol Uygulaması

Mustafa DAĞDELEN

Danışman: Doç. Dr. Mehmet İlteriş SARIGEÇİLİ
2. Danışman: Doç. Dr. Necdet Sinan ÖZBEK

Makine Mühendisliği Anabilim Dalı

ÖZ

Bu tez çalışmasında, CWRR adı verilen, iki serbestlik dereceli, pnömatik tahrikli bilek/önkol rehabilitasyon robotu kuvvet/tork etkileşimi kontrolüne uygun hale getirilecek şekilde modifiye edilmiştir. Bu amaçla CWRR robotunun mekanik ve kontrol ekipmanları yeniden tasarlanmış, üretimi gerçekleştirilmiş ve yenilenmiştir. Robotun kontrol performansını en üst düzeye çıkarmak için ekstra sensörler eklenmiştir. Modernizasyon sürecinin tamamlanmasının ardından yenilenen hassas pnömatik kontrol valflerinin akış modellemesi yapılmıştır. Oransal yön kontrol valflerinin literatürde sunulan mevcut akış modellerine uymadığı tespit edilmiş ve oransal yön kontrol valfleri için yeni ve etkili bir sıkıştırılabilir akış modeli önerilmiştir. Bir sonraki aşama olarak yenilenen pnömatik silindirlerin sürtünme kuvveti/torkunun belirlenmesine yönelik deneysel çalışmalar yapılmıştır. Bu noktada sürtünme parametresinin daha hızlı ve doğru tanımlanmasını sağlayacak sürtünme kuvveti/tork tanımlamasına yönelik yeni otomasyon algoritmaları tasarlanmış ve uygulanmıştır. Etkileşim halindeki insan kullanıcısının güvenliğini artırmak ve robotu daha uyumlu hale getirmek amacıyla, insanın hareket yönü isteğini tespit eden bir algoritma geliştirilmiş ve kontrolcü yapısına dahil edilmiştir. Daha sonra yürütülen çalışma sırasında hedef etkileşim torkunu başarılı bir şekilde takip edebilen bulanık mantık uyarlamalı PID kontrolcü yapısı tasarlanmıştır. Bu amaçla bulanık mantık kuralları doğru bir şekilde belirlenmiş, üyelik fonksiyonları ve bu fonksiyonların sınırları deneysel verilere ve uzmanın deneyimine göre belirlenmiştir. Önerilen robotun bozucu etkiler altında dahi doğru kontrol eylemleri gerçekleştirebilmesi için hem iç hem de dış bozucu etkileri reddedebilen algoritmalar geliştirilmiştir. Önerilen denetleyicinin üstün performansını göstermek için iki farklı denetleyici de robot üzerinde test edilmiştir. Bu kontrolörlerden biri üretici tarafından ve oransal yön kontrol valfine monte edilen Default kontrolcü, diğeri ise bu Default kontrolcüye seri bağlancak şekilde tasarlanan geleneksel PID yapısıdır. Bu üç farklı kontrolcü yapısı, gerçek-zamanlı döngü-içinde-donanım ortamında, bozucu etkilerden serbest veya bozucu etkiler altındaki koşulların değerlendirildiği dört farklı senaryo altında test edilmiştir. Önerilen seri bulanık mantık tabanlı PID yapısının tüm senaryolarda emsallerine göre üstün kontrol performansı sergilediği yapılan testler ile doğrulanmıştır.

Anahtar Kelimeler: Bozucu bastırma algoritması, Pnömatik, Rehabilitasyon Robotu, Sürtünme tanımlaması, Uyarlamalı kontrol

EXTENDED ABSTRACT

People may lose the mobility of their limbs in cases of paralysis, spinal cord injuries or aging. In other words, the range of motion of the limbs is reduced to a certain extent. After these unfortunate cases, they need intense, repetitive and long-term physical treatment and rehabilitation in order to regain their lost limb abilities and adapt to daily life again.

Today, physical therapy and rehabilitation can be applied physiotherapist-based or robotic-based. In physiotherapist-based physical therapy practices, the physiotherapist treats the patient by working in one-on-one interaction with the patient and applying the necessary contact and instructions. However, insufficient number of physiotherapists, difficulty in patient follow-up, and decreased efficiency of fatigue-based physical therapy practices are some of the problems. At this point, robot-based physical therapy applications have the potential to eliminate the problems in physiotherapist-based physical therapy applications. Because robots can demonstrate superior performance in repetitive and long-term applications. Therefore, researchers have focused on robot-based physical therapy applications and studied on the mechanical and controlling structures of the robot.

Mechanically, rehabilitation robots are designed as exoskeleton-based or end-effector-based. Exoskeleton-based robots can be worn by the user, and each human limb can be driven by the robot's different actuators. Therefore, this type of robots made it possible for the user's limbs to be driven independently and to track each axis independently, due to independent sensors placed on the axes. On the other hand, end-effector robots move the axes of the human limb by driving the point called the end-effector, with which the user interacts. These types of robots are generally fixed to the ground, and since a single point is driven and equipped with sensors, the remaining human limb axes cannot be followed.

Rehabilitation robots can also be classified as lower-limb or upper-limb robots. While lower extremity robots are used for the ankle, knee and hip regions of the human body, upper extremity robots are used for the wrist, forearm, elbow and shoulder regions. Since the main purpose of rehabilitation robots is to increase people's narrowed limb axis range of motion back to healthy human values, the range of motion offered by these robots for each axis becomes important. Therefore, for whatever region/limb the robot is used, it must have a range of motion at least close to that of a healthy person. For example, considering the upper-extremity limbs, a healthy human has a total range of motion of 115^{o}, 70^{o} and 150^{o} for wrist flexion/extension, wrist radial/ulnar deviation and forearm pronation/supination movements, respectively.

On the other hand, the actuators used on the robot also play an important role in obtaining correct control outputs. Electric motors have mostly been preferred due to their easy accessibility and easy controllability. However, electric motors have a rigid structure due to their high internal resistance, and since they are not back-drivable, they have the potential to present uncomfortable or

harmful environments to the patient. In order to somewhat soften the hard structure of electric motors and increase the safety level, some actuation equipment called series-elastic-actuator have been developed by incorporating elastic equipment into the motors and have been applied in rehabilitation robots. However, at this point, determining the stiffness levels of the elastic elements added to the actuators has made the use of this type of drive systems difficult. On the other hand, hydraulic cylinders have very high power/weight ratios and are suitable for precision applications. However, since these systems work with pressurized oil and are heavy, they are not suitable for some compact robotic applications. Therefore, the use of hydraulic cylinders in rehabilitation robots is almost non-existent. Another drive system called pneumatic drive systems, have gained an important place in rehabilitation robots. Pneumatic drive systems have found their place in rehabilitation robots as classical cylinders or artificial pneumatic muscles (pMA or McKibben muscles). While classical pneumatic cylinders have a piston structure that moves within the cylinder due to the applied air pressure, artificial pneumatic muscles work in a different way. Artificial pneumatic muscles have a structure that shortens in length when air is applied, due to their special design. When the air pressure recedes, their length becomes shorter again. While artificial pneumatic muscles perform well in linear movements, they require some add-ons for angular movements. An angular movement in a certain axis can be achieved with a pulley mechanism together with two artificial pneumatic muscles. However, they are not suitable for robots with particularly space constraints. In addition, pneumatic actuators are more suitable for human safety, as they have high power/weight ratios and low internal resistance. However, in order to control pneumatic systems effectively, nonlinear friction and flow models must be detected accurately.

The control method (i.e. controlled parameters) has a very important place in rehabilitation robots. In particular, robots that apply position control are only effective in the early physical therapy phase, while they do not provide a healing effect to the patient in the later stages. In addition, position control applications alone may cause harmful reactions to the patient due to their rigid structure. Based on these points, controlling the interaction force/torque in rehabilitation robots has provided the user with more effective healing ability and provided a safe and compliant environment to the patient. Therefore, isotonic, isometric or assistive exercise experiments are focused on robots. In isotonic exercises, when the patient interacts with the robot, she/he feels as if she/he is lifting a weight or moving a heavy object, that is, she/he performs work under constant force. These types of exercises require force/torque control and are well suited for human-robot interaction applications. On the other hand, in isometric exercises, a person applies force to a non-moving object and the object does not move no matter how much force he applies. In other words, whatever force it applies, it is met with an equivalent resistance force. In this exercise, the robot applies the same resistance force as the human does. Like isotonic exercises, these types of exercises require force/torque control. Finally, in assistive type exercise applications, the patient

moves the robot as much as she/he can for a certain amount of time. At this point, only force control is applied. From this point on, the human is supported by the robot through assistive forces to reach a certain goal. At this stage, positional force is applied. That is, hybrid position/force control is applied in assistive type exercise applications.

Finally, rehabilitation robots are expected to have algorithms that can reject disruptive effects so that they can take the correct control actions even under disruptive effects and apply the determined controller targets to the patient. In this way, it protects the patient from involuntary actions and provides a safe and compliant interaction environment.

In the light of the above information, in this thesis study, the existing pneumatic driven two-degrees-of-freedom wrist/forearm rehabilitation robot, which only has position control ability, was modernized by targeting the following criteria:

- Since the mechanical parts on the robot are both heavy and potentially dangerous for the patient, these parts were first redesigned based on human ergonomics, and were produced using polymer material that is both lightweight and more suitable for humans, with the help of 3D printing technology.
- The produced parts were then tried on the robot to determine whether there were any errors. Parts containing defects were re-designed and produced in a better way, and the final assembly process was carried out. After the final assembly process, it was confirmed that the range of motion of the robot was equal to the range of motion of a healthy person in some axes and greater in some axes. In addition, the pneumatic cylinders on the robot have been replaced with new technology cylinders that are both more compact and have less friction characteristics. In this way, the mechanical modernization process has been completed.
- After the mechanical modernization process was completed, the control equipment was modernized by choosing more sensitive equipment and incorporating some additional sensors. The solenoid-type pneumatic valves in the first version of the robot were replaced with proportional-type pneumatic control valves in the new version. The Arduino-based data acquisition and control card has been replaced with the more comprehensive and powerful QUANSER Q8 model data acquisition and control card. QUARC interface program is also provided for real-time and uninterrupted communication of the QUANSER Q8 card with MATLAB/Simulink. Standard potentiometers used for angle and speed measurement have been replaced with high precision encoders. In addition, pressure sensors were added, and force and torque sensors were added to increase the control performance of the robot. By using new and more powerful electrical power sources, control equipment is prevented from

being negatively affected by any power outage. In this way, the modernization of control equipment has been completed.

- After the modernization phase, a control scheme was created for the robot. In this study, isotonic type exercises were preferred and force/torque control was targeted. For this purpose, instead of directly controlling the interaction force or torque, the pressure within the pneumatic cylinders was controlled. For this purpose, the dynamics and relationships between pressure and output force or torque have been determined. At this point, it has been understood that in order to establish dynamic relationships correctly, the behavior of the valves must be determined and the friction characteristics of the cylinders must be determined. Therefore, mathematical models of the valves were determined, and as a result, it was observed that proportional pressure regulating valves have linear dynamics, while proportional directional control valves have very non-linear behavior. Proportional directional control valves have been successfully modeled and a new compressible flow model has been presented to the literature. However, in this study, proportional pressure regulating valves were chosen for pressure control and, directional control valves were used only to direct the air. Subsequently, it was determined that the methods available in the literature to determine the friction characteristics of pneumatic cylinders were insufficient, had the potential to produce incorrect results, and took a long time. Based on this, all experimental stages were automated, and a method that identifies friction with a single click in a short time, by the designed interfaces, was also introduced to the literature.
- Proportional pressure regulating valves actually have a control structure within themselves. However, while this controller exhibited good control performances under static conditions, it failed under dynamic conditions with poor performance such as very big steady-state error values and high-frequency chattering. Therefore, in this thesis study, a cascade controller scheme that is connected serially to the controllers of the valves has been considered. First of all, a conventional PID controller serially connected to the valve controller was designed and the results were reviewed. In conventional PID structure, gain values (i.e. K_p, K_i and K_d) must be optimized before being used in application. For this purpose, the optimization processes of these gain values were carried out using a closed-loop PID auto-tuning tool in a real-time simulation environment in MATLAB/Simulink. With optimized constant gain values, the conventional PID structure improved the poor performance of the valve's Default controller, but continued to exhibit a torque tracking with chattering. At this point, the fuzzy logic method was introduced and the gain values of the conventional PID structure were considered to be scheduled at different values under changing operating conditions. Because fuzzy logic can dynamically regulate rule-based input-output

relationships of systems operating in variable environments, and it can increase their control performance. For this reason, with the knowledge of experts working on the robot, input-output membership functions were determined. 49 fuzzy rules were carefully created and the information defuzzification mathematics was established correctly. The values obtained from fuzzy logic were finalized through an adaptation process in order to be suitable for the control valves. In this way, the gain values of the conventional PID structure have gained the adjustable/programmable feature.

- In order for the robot to provide a safe and compliant interaction environment for humans, a motion detection algorithm has been developed that automatically detects human movement intentions and allows controllers to react accordingly.
- In addition, algorithms have been developed that can detect internal and external disturbance effects and correct the disturbance effect by correcting the robot's controller output signals accordingly. Since a pneumatic system is used on the robot, internal disturbances have been developed to detect any pressure loss in the system. On the other hand, any deterioration in the interaction torque value during any exercise was considered as an external disturbance effect.

The efficiency and effects of all the improvements and the developed algorithms mentioned above were investigated through validation tests under different conditions/scenarios by studying the forearm pronation/supination movement on the robot, and the results listed below were obtained:

- Pneumatic drive systems have demonstrated superior performance by means of accurate estimation/identification of linear or non-linear (compressible flow pattern and friction in cylinders) characteristics. This thesis contains innovative and helpful information on how to obtain non-linear properties such as friction and compressible flow parameters.
- The motion detection algorithm worked very accurately and stably, providing a very safe and harmonious working environment for people.
- Fuzzy logic adaptive PID structure allowed superior interaction torque tracking by accurately programming PID gains under all specified operating conditions. With this thesis, it has been revealed that fixed-gain conventional controller structures cannot exhibit superior performance in rehabilitation robots.
- Algorithms that can detect disturbances have been successfully implemented and have improved controller performances to an excellent extent. While external disturbance effects have generally been studied in the literature (i.e. force/torque disturbance

effects coming from outside), in this study, an algorithm that can detect internal disturbance effects, especially for pneumatic systems, has been designed and its justifications have been discussed. The algorithm that detects pressure losses has worked very efficiently and protected the user. The information presented at this point has the potential to be a resource for other studies and is open to improvement.

GENİŞLETİLMİŞ ÖZET

İnsanlar felç, omurilik yaralanmaları veya yaşlanma durumlarında uzuvlarının sahip oldukları hareket kabiliyetlerini kaybedebilmektedirler. Yani uzuvlarının hareket açıklıkları belirli ölçüde azalmaktadır. Bu üzücü vakalar sonrasında kaybolmuş uzuv kabiliyetlerini geri kazanmak ve tekrardan günlük yaşama adapte olabilmek için yoğun, tekrarlı ve uzun süreli fiziksel tedavi ve rehabilitasyona ihtiyaç duyarlar.

Fiziksel tedavi ve rehabilitasyon günümüzde fizyoterapist tabanlı veya robotik tabanlı uygulanabilmektedir. Fizyoterapist tabanlı fizik tedavi uygulamalarında, fizyoterapist hasta ile birebir etkileşimli çalışarak ve gerekli temas ve yönergeler uygulayarak hastayı tedavi etmektedir. Fakat, fizyoterapist sayısındaki yetersizlik, hasta takibinin zorlaşması, yorulmaya dayalı fizik tedavi uygulamalarının veriminin azalması sorunların birkaçını oluşturmaktadır. Bu noktada robot tabanlı fizik tedavi uygulamaları fizyoterapist tabanlı fizik tedavi uygulamalarındaki sorunları giderme potansiyeline sahiptir. Çünkü robotlar tekrarlı ve uzun süreli uygulamalarda üstün performans sergileyebilmektedir. Bu yüzden, araştırmacılar robot tabanlı fizik tedavi uygulamalarına ve robotun mekanik ve kontrolcü yapıları üzerine yoğunlaşmışlardır.

Mekanik anlamda rehabilitasyon robotları dış-iskelet tabanlı veya uç-etkileyici tabanlı tasarlanmışlardır. Dış-iskelet tabanlı robotlar kullanıcı tarafından giyilebilmektedir ve her bir insan uzvu robotun farklı aktüatörleri tarafından tahrik edilebilmektedir. Bundan dolayı, bu tip robotlar kullanıcının uzuvlarının bağımsız şekilde tahrik edilebilmesini ve eksenlere yerleştirilen bağımsız sensörler sayesinde her bir eksenin bağımsız şekilde takip edilebilmesini mümkün kılmıştır. Diğer taraftan, uç-etkileyici robotlar ise uç-etkileyici olarak adlandırılan ve kullanıcının etkileşimde olduğu nokta tahrik edilerek insan uzvunun eksenleri hareket ettirilmektedir. Bu tip robotlar genelde yere sabitlenmiş bir yapıdadır ve tek bir nokta tahrik edilip sensörlerle donatıldığı için geride kalan insan uzvu eksenleri takip edilememektedir.

Rehabilitasyon robotları ayrıca alt-ekstremite veya üst-ekstremite robotları olarak da sınıflandırılabilmektedir. Alt ekstremite robotları insan vücudunda ayak bileği, diz ve kalça bölgeleri için kullanılıyorken, üst ekstremite robotları da el bileği, ön kol, dirsek ve omuz bölgeleri için kullanılmaktadır. Rehabilitasyon robotlarının esas amacı insanların daralan uzuv eksen hareket açıklıklarını tekrardan sağlıklı insan değerlerine çıkarmak olduğu için bu robotların her bir eksen için sunmuş oldukları hareket açıklıkları önem kazanmaktadır. Dolayısıyla, robot hangi bölge/uzuv için kullanılıyorsa o bölge için en azından sağlıklı insanın sahip olduğu hareket açıklıklarına yakın bir değere sahip olması gerekmektedir. Örnek olarak, üst-ekstremite uzuvları referans alındığında sağlıklı bir insan bilek fleksiyon/ekstansiyon, bilek radyal/ulnar deviasyon ve önkol pronasyon/supinasyon hareketleri için toplamda sırasıyla 115^o, 70^o ve 150^o'lik hareket açıklığına sahiptir.

Diğer taraftan robot üzerinde kullanılan tahrik organları da doğru kontrol çıktılarının alınabilmesi için önemli bir yer kapsamaktadır. Elektrik motorları, hem kolay erişilebilirliklerinden hem de kolay kontrol edilebilirliklerinden dolayı çoğunlukla tercih edilmişlerdir. Fakat elektrikli motorlar iç dirençlerinin yüksek olmasından dolayı hem sert bir yapıya sahiptirler hem de geri-sürülebilir olmadıklarından dolayı hastaya rahatsız edici veya zarar verici ortamlar sunabilme potansiyeline sahiptirler. Elektrik motorlarının sahip olduğu sert yapıyı bir nebze yumuşatmak ve güvenlik seviyesini artırmak için motorları elastik ekipmanlar dahil edilerek seri-elastik-aktüatör adında tahrik organları geliştirilmiş ve rehabilitasyon robotlarında uygulanmıştır. Fakat bu noktada motorlara ilave edilen elastik elemanların sertlik seviyelerini belirlemek bu tip tahrik sistemlerinin kullanımını zorlaştırmıştır. Diğer taraftan, hidrolik silindirler oldukça yüksek güç/ağırlık oranlarına sahiptirler ve hassas uygulamalar için uygundur. Fakat, bu sistemler basınçlı yağ ile çalıştıkları için ve ağır oldukları için özellikle bazı kompakt robotik uygulamaları için hiç uygun değildir. Bu yüzden hidrolik silindirlerin rehabilitasyon robotlarında kullanımı yok denecek kadar azdır. Bir başka tahrik sistemlerinden biri olan pnömatik tahrik sistemleri ise rehabilitasyon robotlarında önemli bir yer edinmiştir. Pnömatik tahrik sistemleri klasik silindirler veya yapay pnömatik kaslar olarak rehabilitasyon robotlarında yer edinmiştir. Klasik pnömatik silindirler uygulanan hava basıncı sayesinde silindir içinde hareket eden piston yapısına sahipken, yapay pnömatik kaslar daha farklı bir şekilde çalışmaktadırlar. Yapay pnömatik kaslar özel tasarımları sayesinde hava uygulandığında boyları kısalan yapıya sahiptirler. Hava basıncı geri çekildiğinde ise boyları tekrardan kısalmaktadır. Yapay pnömatik kaslar, doğrusal hareketlerde iyi bir performans sergilerken açısal hareketler için bazı eklentiler gerektirmektedir. İki yapay pnömatik kas ile birlikte bir makara mekanizması ile belirli bir eksende açısal hareket sağlanabilmektedir. Ancak özellikle alan kısıtlaması olan robotlar için uygun değildirler. Buna ilaveten, pnömatik tahrik organları, hem güç/ağırlık oranları yüksek hem de iç dirençleri düşük olduğu için insan güvenliği için daha uygun yapıdadır. Fakat pnömatik sistemlerin etkili bir biçimde kontrol edilebilmeleri için doğrusal olmayan sürtünme ve akış modellerinin doğru bir şekilde saptanması gerekmektedir.

Rehabilitasyon robotlarında kontrol metodu (yani kontrol edilen parametreler) çok önemli bir yer kaplamaktadır. Özellikle pozisyon kontrolü uygulayan robotlar sadece erken fizik tedavi aşamasında etkili olurken sonraki evrelerde hastaya iyileşme etkisi kazandırmamaktadır. Ayrıca sadece pozisyon kontrolü uygulamaları sert yapılarından dolayı hastaya zarar verici tepkiler de doğurabilmektedir. Bu noktalardan yola çıkarak, rehabilitasyon robotlarında etkileşim kuvveti/torkunu kontrol etmek hem kullanıcıya daha etkili iyileşme yetisi kazandırmış hem de hastaya güvenli ve uyumlu bir ortam kazandırmıştır. Bundan dolayı robotlar üzerinde, izotonik, izometrik veya yardımcı tarzda egzersiz denemeleri üzerine yoğunlaşılmıştır. İzotonik egzersizlerde hasta robot ile etkileşim halindeyken sanki bir ağırlığı kaldırıyormuş ya da sanki ağır bir objeyi hareket ettiriyormuş gibi hisseder, yani sabit kuvvet altında bir iş gerçekleştirir. Bu tip egzersizler kuvvet/tork kontrolünü gerektirir ve insan-robot etkileşimi uygulamaları için oldukça

uygundur. Diğer taraftan, izometrik egzersizlerde insan hareket etmeyen bir objeye kuvvet uygular ve ne kadar kuvvet uygulasa da obje hareket etmez. Yani ne kadar kuvvet uygularsa ona denk bir direnç kuvvetiyle. Bu egzersizde robot, insan ne kadar kuvvet uygularsa aynı şekilde direnç kuvveti uygular. İzotonik egzersizler gibi bu tip egzersizler de kuvvet/tork kontrolünü gerektirmektedir. Son olarak yardımcı tip egzersiz uygulamalarında hasta belirli bir miktarda gücünün yettiği kadar robotu hareket ettirir. Bu noktada sadece kuvvet kontrolü uygulanır. Bu noktadan sonra belirli bir hedefe ulaşması için insan robot tarafından yardımcı kuvvetler vasıtasıyla desteklenir. Bu aşamada ise pozisyon kuvveti uygulanır. Yani, yardımcı tip egzersiz uygulamalarında hibrit pozisyon/kuvvet kontrolü uygulanır.

Son olarak rehabilitasyon robotlarının bozucu etkiler altında bile doğru kontrol aksiyonlarını alarak hastaya belirlenen kontrolcü hedeflerini uygulayabilmesi için bozucu etkileri tolere edebilecek algoritmalara sahip olması beklenir. Bu şekilde hastayı istemsiz aksiyonlardan koruyarak ona güvenli ve uyumlu bir etkileşim ortamı sağlar.

Yukarıdaki bilgiler ışığında bu tez çalışmasında, hali hazırda var olan ve sadece pozisyon kontrolü yetisine sahip olan, pnömatik tahrikli iki serbestlik dereceli bilek/önkol rehabilitasyon robotu aşağıdaki kriterler hedef alınarak modernize edilmiştir:

- Robotun üzerindeki mekanik parçalar hem ağır hem de hasta için tehlike yaratacak potansiyelde olduğu için, öncelikle bu parçalar, insan ergonomisi temel alınarak yeniden tasarlanmış, 3D yazıcı teknolojisi yardımıyla hem hafif hem de insan için daha uygun olan polimer malzeme kullanılarak üretimi gerçekleştirilmiştir.
- Üretimi gerçekleştirilen parçalar, daha sonra robot üzerinde montaj provaları yapılarak herhangi bir hatanın var olup olmadığı belirlenmiştir. Kusur barındıran parçalar tekrardan daha iyi bir şekilde tasarlanıp, üretimi gerçekleştirilerek son montajlama işlemi gerçekleştirilmiştir. Son montajlama işlemi sonrasında, robotun sahip olduğu hareket açıklığının sağlıklı insanın sahip olduğu hareket açıklığına bazı eksenler de denk bazı eksenler de ise fazla olduğu doğrulanmıştır. Ek olarak, robot üzerindeki pnömatik silindirler hem daha kompakt hem de daha az sürtünme karakteristiğine sahip olan yeni teknoloji silindirler ile değişimi gerçekleşmiştir. Bu sayede mekanik anlamda modernizasyon işlemi tamamlanmıştır.
- Mekanik modernizasyon işlemi tamamlandıktan sonra, kontrol ekipmanları daha hassas ekipmanlar tercih edilerek ve bazı ek sensörler dâhil edilerek modernize edilmiştir. Robotun ilk versiyonunda selenoid tarzı pnömatik valfler, yeni versiyonda oransal pnömatik kontrol valfleriyle değiştirilmiştir. Arduino tabanlı veri toplama ve kontrol kartı, daha kapsamlı ve güçlü QUANSER Q8 model veri toplama ve kontrol kartı ile değiştirilmiştir. QUANSER Q8 kartının MATLAB/Simulink ile anlık ve kesintisiz haberleşmesi için QUARC ara yüz programı da temin edilmiştir. Açı ve hız

ölçümü için kullanılan standart potansiyometrik sensörler hassas enkoder açıölçerleri ile değiştirilmiştir. Buna ek olarak, basınç sensörleri ilave edilmiş, kuvvet ve tork sensörleri ilave edilerek robotun kontrol performansının artırılması amaçlanmıştır. Yeni ve daha güçlü elektrik güç kaynakları kullanılarak kontrol ekipmanlarının herhangi bir elektrik kesintisinden kaynaklı olumsuz etkilenmeleri engellenmiştir. Bu şekilde kontrol ekipmanlarının modernizasyonu tamamlanmıştır.

- Modernizasyon aşamasından sonra, robot için bir kontrol şeması oluşturulmuştur. Bu çalışmada izotonik tipte egzersizler tercih edilmiş ve kuvvet/tork kontrolü hedeflenmiştir. Bu amaçla, etkileşim kuvvetini veya torkunu direkt olarak kontrol etmek yerine, pnömatik silindirler içerisindeki basınç kontrol edilmiştir. Bu amaçla, basınç ve çıkış kuvveti veya torku arasındaki dinamikler ve ilişkiler belirlenmiştir. Bu noktada, dinamik ilişkilerin doğru bir şekilde kurulabilmesi için, valflerin davranışlarının belirlenmesi gerektiği ve ayrıca silindirlerin sahip olduğu sürtünme karakteristiklerinin saptanması gerektiği anlaşılmıştır. Bu yüzden, valflerin matematiksel modelleri saptanmış, sonucunda ise oransal basınç düzenleme valflerinin doğrusal bir dinamiğe sahipken, oransal yön kontrol valflerinin oldukça doğrusal olmayan bir davranışa sahip olduğu gözlemlenmiştir. Oransal yön kontrol valflerinin, başarılı bir şekilde modellemesi yapılmış, literatüre yeni sıkıştırılabilir akış modeli sunulmuştur. Fakat yine de bu çalışma da basınç kontrolü için oransal basınç düzenleme valfleri seçilmiş ve yön kontrol valfleri sadece havayı yönlendirmek amaçlı kullanılmıştır. Daha sonrasında, pnömatik silindirlerin sürtünme karakteristiklerini belirlemek için literatürde var olan yöntemlerin yetersiz olduğu, yanlış sonuçlar üretebilme potansiyeline sahip olduğu ve uzun zaman aldığı belirlenmiştir. Buradan yola çıkarak tüm deneysel aşamaların otomasyonu gerçekleştirilmiş, tasarlanan ara yüzler sayesinde tek tıkla ve kısa sürede sürtünme tanımlaması yapan bir yöntem ayrıca literatüre kazandırılmıştır.
- Oransal basınç düzenleme valfleri esasında kendi içlerinde bir kontrol yapısına sahiptir. Fakat bu kontrolcü, statik şartlar altında iyi kontrol performansları sergilerken dinamik şartlar altında oldukça büyük sabit hata değerleri ve yüksek frekansta titreşimli takip gibi kötü performansla sınıfta kalmıştır. Bu nedenle bu tez çalışmasında valflerin kontrolcülerine seri olarak eklenen bir kademeli kontrolcü şeması ele alınmıştır. Öncelikle konvansiyonel PID kontrolcü tasarlanmış, seri bir şekilde valfin kontrolcüsüne bağlanmış ve sonuçlar gözden geçirilmiştir. Konvansiyonel PID yapısında kazanç değerlerinin (yani K_p, K_i ve K_d) uygulamada kullanılmadan önce optimize edilmeleri gerekmektedir. Bu amaçla bu kazanç değerlerinin optimizasyon işlemleri MATLAB/Simulink ortamında gerçek zamanlı bir simülasyon ortamında "Closed-Loop PID Auto-tuning" aracı kullanılarak

gerçekleştirilmiştir. Optimize edilen sabit kazanç değerleriyle konvansiyonel PID yapısı valfin doğal kontrolcüsünün kötü performansını iyileştirmiş fakat titreşimli bir tork takibi sergilemeye devam etmiştir. Bu noktada bulanık mantık yöntemi devreye sokulmuş ve konvansiyonel PID yapısının kazanç değerlerinin farklı şartlarda farklı değerlerde programlanması düşünülmüştür. Çünkü bulanık mantık değişken ortamlarda çalışan sistemlerin kural tabanlı giriş-çıkış ilişkilerini dinamik olarak düzenleyebildiği için kontrol performanslarını artırabilmektedir. Bu sebeple, robot üzerinde çalışan uzman bilgileriyle, giriş-çıkış üyelik fonksiyonları belirlenmiş, 49 adet kural dikkatlice oluşturulmuş ve bilgi çıkarma matematiği doğru bir şekilde kurulmuştur. Bulanık mantıktan çıkan değerler kontrol valflerine uygun olabilmesi için, bir adaptasyon işleminden geçerek son halini almıştır. Bu sayede konvansiyonel PID yapısının kazanç değerleri ayarlanabilir/programlanabilir özelliğini kazanmıştır.

- Robotun insana güvenilir ve uyumlu bir etkileşim ortamı sunması için, insan hareket isteğini otomatik olarak algılayan ve kontrolcülerin buna göre tepki vermesini sağlayan bir hareket algılama algoritması da geliştirilmiştir.
- Ek olarak, iç ve dış bozucu etkilerini saptayabilen ve bu durumlara göre robotun kontrolcü çıkış sinyallerini düzelten bozucu etkiyi düzelten algoritmalar geliştirilmiştir. Robot üzerinde pnömatik sistem kullanıldığı için, iç bozucu etkiler sistemdeki herhangi bir basınç kaybını algılayabilecek nitelikte tasarlanmıştır. Diğer taraftan, herhangi bir egzersiz sırasında etkileşim tork değerinde meydana gelen bozulmalar ise dış bozucu etki olarak değerlendirilmiştir.

Yukarıda bahsi geçen tüm iyileştirmeler ve geliştirilen algoritmalar robot üzerinde önkol pronasyon/supinasyon hareketi çalışılarak farklı şartlar/senaryolar altında doğrulama testleri sayesinde verimlilikleri ve etkileri araştırılmış ve aşağıdaki listelenen sonuçlar elde edilmiştir:

- Pnömatik tahrik sistemleri, doğrusal veya doğrusal olmayan (sıkıştırılabilir akış modeli ve silindirlerdeki sürtünme) özelliklerinin doğru saptanması/tanımlanması sayesinde oldukça üstün bir performans sergilemiştir. Bu tez çalışmasında bu özelliklerin nasıl elde edileceğine dair yenilikçi ve yardımcı bilgiler mevcuttur.
- Hareket algılama algoritması oldukça doğru ve kararlı bir şekilde çalışarak insana oldukça güvenli ve uyumlu bir çalışma ortamı sağlamıştır.
- Bulanık mantık uyarlamalı PID yapısı, belirlenen tüm çalışma şartlarında PID kazançlarını doğru bir şekilde programlayarak, üstün etkileşim tork takibine izin vermiştir. Bu tez çalışmasıyla, sabit kazançlı konvansiyonel kontrolcü yapılarının rehabilitasyon robotlarında üstün performans sergileyemedikleri gözler önüne serilmiştir.

- Bozucu etkileri saptayabilen algoritmalar başarılı bir şekilde uygulanmış ve kontrolcü performanslarını mükemmel ölçüde geliştirmiştir. Literatürde genellikle dış bozucu etkiler çalışılmışken (yani dışardan gelen kuvvet/tork bozucu etkileri), bu çalışma ile özellikle pnömatik sistemler için iç bozucu etkileri saptayabilecek algoritma tasarlanmış ve gerekçeleri tartışılmıştır. Basınç kayıplarını algılayan algoritma oldukça verimli bir şekilde çalışmış ve kullanıcıyı korumuştur. Bu noktada sunulan bilgiler, diğer çalışmalara kaynak olma potansiyeline sahiptir ve geliştirilmeye açıktır.

ACKNOWLEDGEMENTS

I would like to express my gratitude to those who supported me during my PhD period.

I gratefully thank my advisor Assoc. Prof. Dr. Mehmet İlteriş SARIGEÇİLİ and my co-advisor Assoc. Prof. Dr. Necdet Sinan ÖZBEK for their great and seamless support during my PhD thesis study. I also appreciate Sefa MERAL for his technical support during the manufacturing stage of designed rehabilitation robot. Additionally, I gratefully thank to Engineering Faculty and Mechanical Engineering Department of Çukurova University for the financial support. The study has also been funded partially by the Scientific Office of Çukurova University under the project number of FBA-2022-14180.

I overcame all the difficulties on this path to offer my son Uraz DAĞDELEN a better life and future.

LIST OF TABLES

LIST OF FIGURES

SYMBOLS AND ABBREVIATIONS

ROM:	Range of motion
DOF:	Degree of freedom
SEA:	Series elastic actuator
pMA:	Pneumatic artificial muscle
DCV:	Directional control valve
DCV-a & DCV-b:	Proportional directional control valves for chambers a and b
PRV:	Pressure regulating valve
PRV-a & PRV-b:	Proportional pressure regulating valves for chambers a and b
HRI:	Human-robot interaction
PID:	Proportional-Integral-Derivative controller
PDRA:	Pressure disturbance rejection algorithm
TDRA:	Torque disturbance rejection algorithm
WF/WE:	Wrist flexion/Wrist extension
WRD/WUD:	Wrist radial deviation/Wrist ulnar deviation
FP/FS:	Forearm pronation/Forearm supination
EF/EE:	Elbow flexion/Elbow extension
CWRR:	Çukurova Wrist Rehabilitation Robot
CB:	Control block
GUI:	Graphical user interface
MF:	Membership function
NS:	Negative small MF
NM:	Negative medium MF
N:	Negative MF
Z:	Zero MF
P:	Positive MF
PVL:	Positive very low MF
PM:	Positive medium MF
PB:	Positive MF
MAE:	Mean absolute error
IAE:	Integral absolute error
ITAE:	Integral time-weighted absolute error
RMSE:	Root mean squared error
Maximum AE:	Maximum absolute error
S1, S2:	Control signals for solenoid DCV-a of old version CWRR

S3, S4: Control signals for solenoid DCV-b of old version CWRR

f_s: Sampling frequency

P_w: Working pressure

u: Control signal for proportional directional control valve

u_a: Final control signal for DCV-a

u_b: Final control signal for DCV-b

z: Control signal for proportional pressure regulating valve

z_a: Final control signal for PRV-a

z_b: Final control signal for PRV-b

$[z_a]^r$: Raw control signal for PRV-a

$[z_b]^r$: Raw control signal for PRV-b

$[z_j]^T$: Output signal generated by TDRA algorithm for chambers a & b

$\dot{m}$: Air mass flow rate

$\dot{m}_p$: Theoretical mass flow rate produced by new flow model

$\dot{m}_r$: Residual mass flow rate due to modelling error

$\dot{m}_2$: Mass flow rate through port 2 of DCV

$\dot{m}_4$: Mass flow rate through port 4 of DCV

Δu_{null}: Control signal interval producing zero air mass flow rate

$[\Delta u_{p2}]^e$: Unstable control signal range at port 2 in exhausting state

$[\Delta u_{p2}]^f$: Unstable control signal range at port 2 in filling state

PS-0: Pressure sensor measuring air supply pressure

PS-a: Pressure sensor measuring air pressure in chamber a

PS-b: Pressure sensor measuring air pressure in chamber b

P_{comp}: Compressor pressure

P_s: Supply pressure

P_r: Pressure ratio

P_u: Upstream pressure

P_d: Downstream pressure

P_{atm}: Atmospheric pressure

P_{diff}: Loss in pressure under disturbance conditions

P_a: Absolute air pressure in chamber a

P_b: Absolute air pressure in chamber b

$[P_a]^t$: Theoretical chamber pressure based on new flow model

$[P_a]^m$: Measured pressure in chamber a

$[P_b]^m$: Measured pressure in chamber b

$[P_s]^m$: Measured supply pressure

$e_{P,j}$: Error in chamber pressures for a & b

$\dot{e}_{P,j}$: Rate of change of error in pressure

e_{var}: Error variance

$\dot{P}$: Rate of pressure change

P^r: Reference pressure

$[P_r]^f$: Final reference pressure

$[P_a]^r$: Reference pressure for chamber a

$[P_b]^r$: Reference pressure for chamber b

$[P_a]^d$: Pressure disturbance in chamber a

$[P_b]^d$: Pressure disturbance in chamber b

$[P_j]^c$: Correcting signals produced by PDRA for chambers a & b

k_{CCW}: Triggering signal based on CCW motion intention

k_{CW}: Triggering signal based on CW motion intention

C_d: Discharge flow coefficient

$\beta_p(P_r)$: Proposed new flow function with respect to pressure ratio

β_{max}: Attained maximum flow function

CPR: Cracking pressure ratio

a: Cracking pressure ratio value

$a(u, P_s)$: Modified cracking pressure ratio

T_u: Upstream temperature

k: Specific heat ratio

ε: Subsonic flow index

$\varepsilon(u)$: Flow index of new flow model

$k(P_s)$: Pressure effect coefficient of proposed flow model

$[k(P_s)]^e$: Empirical pressure effect coefficient for discharge state

$[k(P_s)]^f$: Empirical pressure effect coefficient for filling state

b: Critical pressure ratio

R: Ideal gas constant for air

C_1, C_2: Flow constants function of k and R

σ_{lam}: Laminar flow coefficient

c_1, c_2, c_3, c_4: Model parameters of Bobrow & McDonell

$\alpha_1(u)$, $\alpha_2(u)$: Parameters of new flow model

T_{net}: Net torque acting on the end-effector

T_i: Interaction torque

$[T_i]^t$: Target interaction torque

$[T_i]^m$: Measured interaction torque

T_p: Pneumatic torque

T_f: Friction torque

T_c: Coulomb friction torque

T_s: Static friction torque

ω_s: Angular Stribeck speed

B: Viscous friction coefficient

$[T_s]^{CCW}$: Static friction torque for CCW direction

$[T_c]^{CCW}$: Coulomb friction torque for CCW direction

$[w_s]^{CCW}$: Angular Stribeck speed for CCW direction

$[B]^{CCW}$: Viscous friction coefficient for CCW direction

$[T_s]^{CW}$: Static friction torque for CW direction

$[T_c]^{CW}$: Coulomb friction torque for CW direction

$[w_s]^{CW}$: Angular Stribeck speed for CW direction

$[B]^{CW}$: Viscous friction coefficient for CW direction

j: Stribeck shape factor

θ: Angular position of vane or end-effector attained in FP/FS

ω: Angular speed of vane or end-effector attained in FP/FS

ω_{fric}: Corrected speed for friction identification

A_v: Vane surface area of semi-rotary cylinder

r: Torque radius

P: Position of the center point on vane surface area

$ØD_1$: Output shaft diameter of vane type semi-rotary cylinder

$ØD_2$: Vane shaft diameter of semi-rotary cylinder

W: Vane width

H: Vane height

I: Mass moment of inertia

α: Angular acceleration

N_s: Number of static friction experiments

N_d: Number of dynamic friction experiments

$[T_f]^s$: Gathered static friction data

$[T_f]^d$: Gathered dynamic friction data

k: Counter variable in friction automation process

θ_{min}: Minimum angular test position for semi-rotary friction tests

θ_{max}: Maximum angular test position for semi-rotary friction tests

θ_t: Generated test position for semi-rotary friction tests

F_i: Interaction force

F_p: Pneumatic force

F_{ext}: External force

$[F_s]^{ext}$: Static friction force data for extension direction

$[F_s]^{ret}$: Static friction force data for retraction direction

$[F_d]^{ext}$: Dynamic friction force data for extension direction

$[F_d]^{ret}$: Dynamic friction force data for retraction direction

$[S]^{ext}$: Dynamic speed data for extension

$[S]^{ret}$: Dynamic speed data for retraction

F_f: Friction force

F_s: Static friction force

F_c: Coulomb friction force

v_s: Linear Stribeck speed

$[F_s]^e$: Static friction force for extension direction

$[F_c]^e$: Coulomb friction force for extension direction

$[v_s]^e$: Linear Stribeck speed for extension direction

$[B]^e$: Viscous friction coefficient for extension direction

$[F_s]^r$: Static friction force for retraction direction

$[F_c]^r$: Coulomb friction force for retraction direction

$[v_s]^r$: Linear Stribeck speed for retraction direction

$[B]^r$: Viscous friction coefficient for retraction direction

x: Linear position of double-acting linear cylinder's piston rod

v: Linear speed of double-acting linear cylinder's piston rod

M_p: Total mass of moving objects in translational form

A_a: Piston surface area in chamber a

A_b: Piston surface area in chamber b

L: Stroke length of double-acting linear pneumatic cylinder

$ØRD$: Rod diameter of double-acting linear pneumatic cylinder

$ØPD$: Piston diameter of double-acting linear pneumatic cylinder

P^{out}: Output or regulated air pressure by PRVs

V_a: Volume of the chamber a when cylinder is fully extended

i_a, i_b: Disturbance sum block activating signals

d_a, d_b: Switching signals to determine value of $[P_j]^c$

e^T: Error in torque

h: Switching signal for controller type

$K_{p,j}$: Proportional gain of main PID controller schema

$K_{i,j}$: Integral gain of main PID controller schema

$K_{d,j}$: Derivative gain of main PID controller schema

$[K_{p,j}]^f$: Fuzzy scheduled proportional gain

$[K_{i,j}]^f$: Fuzzy scheduled integral gain

$[K_{d,j}]^f$: Fuzzy scheduled derivative gain

$[K_{p,j}]^r$: Regulated proportional gain

$[K_{i,j}]^r$: Regulated integral gain

$[K_{d,j}]^r$: Regulated derivative gain

$[K_{p,j}]_{min,max}$: Minimum and maximum regulation parameters for $[K_{p,j}]^f$

$[K_{i,j}]_{min,max}$: Minimum and maximum regulation parameter for $[K_{i,j}]^f$

$[K_{d,j}]_{min,max}$: Minimum and maximum regulation parameter for $[K_{d,j}]^f$

$K_{pp,j}$: Proportional gain of TDRA

$K_{ii,j}$: Integral gain of TDRA

$K_{dd,j}$: Derivative gain of TDRA

T: Tracking of control signal (in auto-tuning process)

N: Filtering coefficient to implement derivative action

μ: Degree of MF

x_k: Corresponding value of MF

t_d: Disturbance instant

1. INTRODUCTION

Stroke and spinal cord injuries in addition to the natural aging can result in the loss of functionality in both upper and lower limbs of the human being. Disabilities stemming from these events can be addressed through rigorous, prolonged, repetitive physical therapy sessions aimed at physical rehabilitation (Xu et al., 2014). The process of physical rehabilitation involves systematically repeating specific movements with the affected limbs of the body.

Nowadays, physical rehabilitation can be handled in two categories as: (i) physiotherapist-based rehabilitation (i.e. manual therapy) and (ii) robotic-based rehabilitation (i.e. robotic therapy) (Cao et al., 2014). In physiotherapist-based rehabilitation, the patient is rehabilitated with one-to-one interaction principle by interacting directly with physiotherapist. The patient follows the exercise patterns and models specified by the physiotherapist. In this type of rehabilitation, following shortcomings can reduce the quality of the applied rehabilitation procedure:

- Since the increase in number of affected patients is quite high, the number of educated expert physiotherapist is not sufficient (Martinez-Peon et al., 2021; Sarhan et al., 2023)
- In this type of therapy, the tracking and monitoring of patients' current status and its recovery level is challenging. Consequently, it becomes difficult to ascertain and implement more effective future therapy methods.
- The efficiency of the physiotherapist can change from patient to patient due to the exhaustion. Because of it, the quality of the repetitions of exercises is not stable and that issue can reduce the overall efficiency of the rehabilitation period and can reduce the recovery speed of the patient.

To eliminate the encountered problems in manual therapy, robotic-based rehabilitation studies increased tremendously to take advantage of the robots in repetitive, precise and long-term work. Many rehabilitation robots, both commercial and academic, have been designed so far with changing type of body segment, mechanical structures, actuator types, and controller methods (Miao et al., 2018; Q. Wu et al., 2017).

First of all, robots are subcategorized as lower limb, upper limb or full body robots according to the rehabilitated zone of the human body. The lower limb rehabilitation robots serve to rehabilitate the ankle, knee and hip joints, whereas upper limb robots are used for wrist, forearm, elbow and shoulder rehabilitation (Mekki et al., 2018). The full body robots are complex and compound structures and they can be used for either upper or lower limb joint impairments. The mechanical structures of the rehabilitation robots can be classified as end-effector type and exoskeleton type either with serial or parallel mechanic form. The end-effector typed robot interacts with the user at a single interaction point and transmits power to the user at the interaction

zone. This type of robots is generally grounded and could not be worn by the user. On the other hand, exoskeleton type robots are worn by the user and each joint of the user is moved by the independent actuators of the robot. In this study, just upper limb rehabilitation is focused on wherein the wrist and forearm movements are investigated. Wrist joint has two types of movements called wrist flexion/extension and wrist radial/ulnar deviation, yielding a 2 DOF at wrist joint. On the other hand, forearm joint has only 1 DOF wherein the movements are so-called forearm pronation/supination.

From the wrist movements points of view, the detailed movements are shown in Figure 1.1 wherein the right hand of a healthy user is considered. The movements at wrist joint take place at different rotation axes which are depicted in Figure 1.1 (b) and 1.1 (e). Bearing on this information, the movements around wrist joint can be described as following:

- Wrist flexion is described as the motion of the hand towards the body while in the handshake position (Figure 1.1 (a)), whereas wrist extension is the reverse of this action (Figure 1.1 (c)).
- Wrist radial deviation is the movement of the hand towards the body with the palm facing downward (Figure 1.1 (d)), whereas wrist ulnar deviation is the opposite motion (Figure 1.1 (f)).

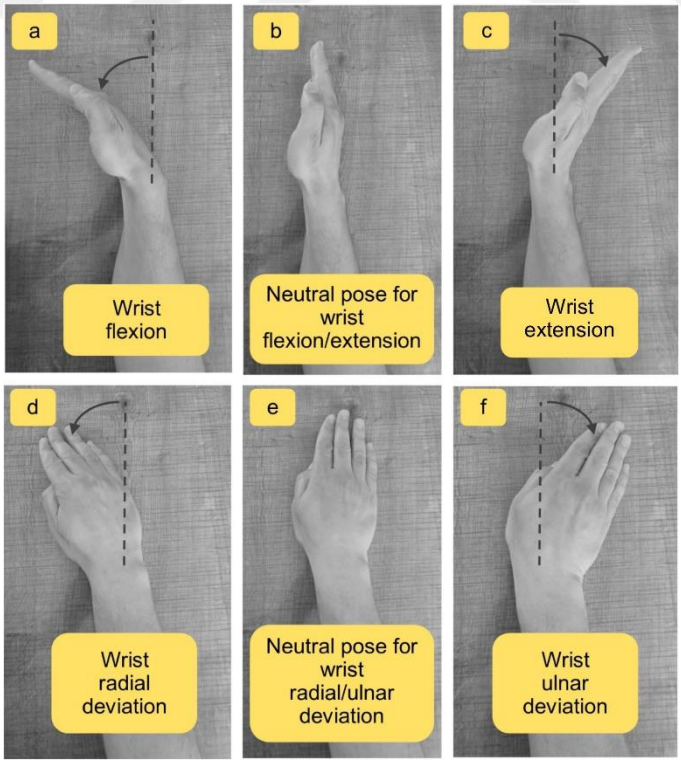

Figure 1.1. Detailed wrist movements

The forearm workouts for a healthy human are illustrated in Figure 1.2, for the right hand. Based on this figure, the neutral pose for the forearm is shown as in Figure 1.2. (b) and the definitions of the forearm movements can be made as following based on the illustrated neutral pose:

- Forearm pronation refers to rotating the forearm towards the body side while the hand is in handshake or punch form (Figure 1.2 (a)), whereas forearm supination is the opposite action (Figure 1.2 (c)).

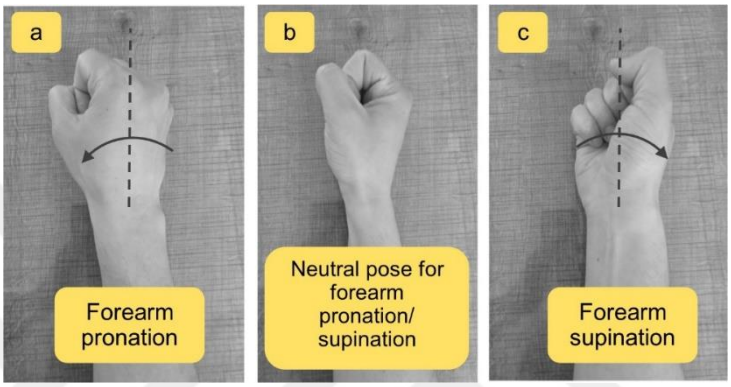

Figure 1.2. Detailed forearm movements

According to the literature, slightly different range of motions (ROM) recorded for the wrist and forearm movements. Indeed, the ROM values for the executed workouts for any healthy human can show changes as reported by (Rahman et al., 2014). The ROM values considered by Martinez et al. (2013) and Pehlivan et al. (2012) can be accepted and can be utilized as 115^{o} and 70^{o} in total for wrist flexion/extension and wrist radial/ulnar deviation workouts, respectively. In addition, a total of 150^{o} ROM for forearm pronation/supination can be taken as reference.

Actuator type in rehabilitation robots is one of the critical issues to get efficient rehabilitation together with safe and compliant interaction. Generally, classical electric motor actuators, modified versions of electric motor actuators called Series Elastic Actuators (SEA), classic pneumatic actuators and pneumatic artificial muscle actuators (pMA or McKibben actuators) are preferred in robotic based rehabilitation. Most of the related studies utilized electrical motor actuators for their ease of control and attainability. Nevertheless, due to the inherent high impedance and limited compatibility of these easily controllable actuators, researchers have explored modifications involving the addition of flexible elements to enhance compliance and safety in electric actuators. Toward this goal, modified versions of electric actuators have been designed in the name of series elastic actuators (SEA). Several studies have been conducted which utilize SEAs in rehabilitation robot applications (Yellewa et al., 2022; Yu et al., 2015). These modified actuators enable the robot to suit changing interaction forces/torques by protecting the user/environment with the help of elasticity of the utilized spring. The SEA may be a good solution

for compliant and precise trajectory performance compared to the counterparts, however designing a low weight SEA is quite hard and selection of spring stiffness is a complicated task (de Gaitani et al., 2022). Pneumatic actuators, either classical cylinders or pMAs work with compressed clean air, have a remarkable power-to-weight ratio and has an inherent compliancy due to the air compressibility. Classical pneumatic cylinders can be in linear or rotary form and they move/rotate with respect to the applied pressure to the respective air chambers (i.e. generally they have two air chambers to work in two directions). Pneumatic artificial muscles which is a different version of classic pneumatic cylinders work in antagonistic-like principle (Kalita and Dwivedy, 2020; Liu et al., 2020). According to the applied air pressure to their inner structure of muscles, they either contract or extend by means of utilized flexible material, and in result a certain translational work is produced. To obtain a specific rotational motion, generally, a kind of pulley and wire mechanism together with two muscles are needed. That situation may limit the usage of this type of actuators especially in space limited mechanisms and robots. Due to the issues above, in this study, classical pneumatic cylinders are opted to get efficient, safe and compliant rehabilitation. Obviously, the pneumatic cylinders have excessive non-linear behaviors such as friction due to the sealing between sliding elements to prevent any possible air leakages. Therefore, the study also aims to identify friction behaviors in utilized cylinders under variable operating conditions.

The control valves in pneumatics are other effective instruments which alter the controllers' performance. They receive an input signal from the controller framework and produce a specific air flow by means of a certain mass flow rate or pressure. The pneumatic valves can be classified as either solenoid or proportional. The solenoid valves operate in the principle of on/off, namely they can be fully opened or fully closed with respect to the input signal. Due to their control principle, they cannot enable the control of efficient operating speed. This kind of valves are cheap and easily attainable. Even though their controller performance is quite limited, yet many researchers used these valves by applying different kind of controller frameworks. However, obtaining an outstanding performance from this type of valve is quite challenging. On the other hand, proportional valves work on the principle of variable orifice opening with respect to the applied control signal. That means, the orifice opening can be proportionally adjusted. By the help of orifice opening adjustability, they can enable the efficient control of speed. Even though they are quite expensive compared to the solenoids, they provide excellent control performances. Beyond the control principle of valves, they are also categorized as their operation mission as Pressure Regulating Valves (PRV) and Directional Controller Valves (DCV). Pressure regulating valves produce desired pressure output, whereas directional control valves direct the pneumatic air and manipulate the air mass flow rate. In many of the pneumatic systems related studies, the proportional directional control valves were preferred by controlling the air mass flow rate via orifice control. It is to be noted that, the DCVs have quite non-linear flow behavior which necessitates the correct modelling of the flow through orifices. Compared to the DCVs, the PRVs

has rather linear behavior which makes it easy to control the pressure directly. Due to these reasons, in this study, the PRVs will be used for direct pressure control whereas the DCVs will be utilized for air directing purposes. The PRVs have built-in controller which is designed by the manufacturer. In static conditions, this built-in controller exhibits good controller performance, yet the performance is not efficient in dynamic conditions. To improve their controller performance in dynamic conditions, additional controller framework will be a good solution to make the system cascaded. With these motivations in mind, cascade controller design will be implemented to obtain the desired dynamic requirements.

The controller method applied to the robot, besides the mechanical structure and actuator type, is the most crucial subject and directly affects the efficacy of the robotic rehabilitation (Zhang et al., 2020). The choice of controller type is influenced by various parameters, including the type of therapeutic exercises to be provided by the robot, as well as the capabilities of the robot's design. The action of controllers is majorly specified based on type of therapeutic exercises such as passive, active assistive, isometric and isotonic (Akdoğan et al., 2018; Yellewa et al., 2022). In passive exercises, the user does not (or cannot) move the robot links actively. Instead, the robot moves the corresponding limb by applying only position control with specified range of motion, operating speed and number of repetitions. On the other hand, active assistive or assist as needed approach utilizes hybrid position or force/torque control. In this exercise, the user tries to move the robot links up to a certain position (force/torque control period) and the robot assists the user to complete predefined task or trajectory (position control period) if the user cannot move the link at any time. In isotonic exercises, the user applies a force to move the robot link with specified interaction characteristics such as lifting a constant weight and construct a human-robot interaction (HRI). During this exercise, force/torque control is required. Lastly, in isometric exercises, the user applies a force to robot link but the robot link does not move by fully resisting the user movement intent. The same with isotonic exercises, force/torque control methodology is requisite to conduct isometric exercises. Bearing on this information, as the robot and human user engage in physical interaction (i.e., HRI), the most appropriate controller is the interaction force/torque control, which ensures compliant and safe engagement. In these exercise methods, the passive exercise is not taken into consideration in this study since it does not satisfy the compliancy and safety criteria due to its stiff behavior which arises from position control objectives. To show the proposed controller’s performance of the robot, it is adequate to work on one of the remaining three exercises. Therefore, in this study, isotonic exercise is preferred. The benefit of utilizing isotonic exercises on the robot is that the level of the resistive interaction torque can easily be changed and reflected to the patient according to patient’s progress in the rehabilitation period. In addition to that, the generation of accurate control signals by the controllers is vitally dependent on the identification and modelling of the overall system for model-based control approaches. On the contrary, model-free based systems utilizes the power of machine-learning algorithms. In this type

of control approaches, correct control outputs are tried to be generated based on big data of the system, namely a relation between output and input data is derived.

According to the existing literature, low-level control strategies have shown limited success in interaction force/torque control approaches, as they primarily focus on trajectory tracking of either position or velocity. Thus, these controllers result in rigid mechanical structures that lack the necessary sensitivity to adapt to variations in human dynamics and environmental forces. Consequently, the utilization of these controllers is limited to the early stages of rehabilitation and is suitable only for seriously affected patients who have no limb movement ability (Zhang et al., 2020). Furthermore, as the patient makes progress in the range of motion and gains some sort of limb power, the low-level controllers do not add extra improvement to the recovery. Low-level control algorithms are applied generally by means of conventional controller frameworks such as Proportional-Integral-Derivative (PID) wherein the constant gains are utilized which struggles the controller performance to changing user and environment dynamics. On the other hand, high-level controllers, which are the most popular topic in robotic rehabilitation, aim to provide a compliant interaction by controlling the relation between force/torque and position (dos Santos and Siqueira, 2019; Sharifi et al., 2017; Sharifi et al., 2014). Impedance or admittance control approaches are examples of high-level control approaches. Furthermore, high-level control approaches can be used to provide the assistant forces/torques for the patients to complete the predetermined task trajectory in the name of "assist-as-needed" approach (Zhang et al., 2020; Y. Wang et al., 2022). The impedance or admittance control approaches satisfy the required interaction force/torque with respect to the virtual impedance parameters and human body dynamics. However, at this point, two problems can be encountered: (i) The specification of virtual impedance parameters can change from patient to patient, and modeling human body dynamics is not an easy task. (ii) Standard impedance/admittance control approaches, which rely on constant impedance parameter assumptions, may struggle to adapt to the evolving capabilities of a human user or any given environment (Miao et al., 2018). Some researchers have made notable contributions by implementing adaptive or learning-based impedance/admittance control to meet compliance requirements in the face of dynamic and unpredictable changes in environmental stiffness (Sharifi et al., 2022). However, the addition of adaptation mathematics to the conventional impedance approach has high possibility to increase complexity of the control approach which results difficulty in application.

Bearing on these observations, controlling the interaction force/torque in an enhanced manner is the main scope for the presented study. To provide interaction force/torque control, PID control framework powered with fuzzy logic is preferred for the purpose of PID gain scheduling, in the name of Cascade Fuzzy Adaptive PID. The adaptive scheduling of the controller gains enables the experts surpassing fully modelling of the complete system, yet modelling is done just for the generation of reference signals to be followed.

The ability of the robot to reject any kind of internal and external disturbances is another important issue to be handled for safe and compliant interaction. The rejection algorithms are embedded in the designed controller structures, and they manipulate the control signal under disturbance effects. The internal disturbances can be defined as the system malfunction, signal attenuations or sudden loss supplied energy. For pneumatically actuated systems, the sudden pressure losses occurring during the movement execution is a good example to the internal disturbances. On the other hand, external disturbances can be classified as the redundant force/torques applied by the unknown environments or interacting users.

Bearing on these observations, today's demands from rehabilitation robot studies can be summarized as following:

- Safe and compliant interaction of the patient with the robot
- User-friendly robot which can detect the motion intent of the patient and totally operates with respect to her/his intention
- An adaptive control performance of interaction force/torque rather than position and speed under changing user and environment profile
- Ability of the robot to reject any kind of disturbances arising from either user or environment

To reach these goals, this study aims to contribute to the literature by introducing following novelties:

- A modified pneumatically actuated robot to ensure compliant wrist and forearm rehabilitation by means of inherent compliancy due to the compressibility of air
- A user motion intention detection algorithm for safe interaction
- Design of adaptive control structure by utilizing fuzzy logic algorithm instead of classical control frameworks
- Identification and modelling of pneumatic tools including both actuating cylinders and controller valves, the design of new and efficient methodology to estimate the friction behavior under changing operating conditions and, a new compressible flow model for proportional pneumatic valves
- Design of internal and external disturbance rejection algorithms to present a robust controller structure

The remainder of the paper is organized as following:

In Section 2, the previous studies about the rehabilitation robot design, the related works on the actuator selection, the investigations about the friction force/torque identification, the flow model of the pneumatic valves and finally the control aspects will be presented.

In Section 3, the design and modification details of proposed wrist/forearm rehabilitation robot will be given first. Then, mathematical description of the robot dynamics as load dynamics of pneumatic actuator, modelling of the pressure regulating valve and directional control valve, identification of friction torque and the assumptions of unidentified dynamics will be presented. After presenting these two subsections, the main part of the material and method section which is control structure of the robot will be detailed.

In Section 4, the experimental results will be presented including valve modelling verification, identified friction force and torque, the validation results of identified friction under static and dynamic conditions, and finally the hardware-in-the loop controller performance results in real-time environment under disturbance free and disturbance conditions.

In Section 5, the study will be concluded by commenting on the gathered results.

2. PRELIMINARY WORK

2.1. Preliminary Work About General Design Approaches of Wrist/Forearm Rehabilitation Robots, Range of Motion Capabilities and Actuator Selection

General design approaches, range of motion (ROM) capabilities and actuator selection for wrist/forearm rehabilitation robot available in the literature has been tabulated in Table 2.1.

Table 2.1. Related preliminary studies about wrist/forearm rehabilitation robot

Name of the Rehabilitation Robot	**Degree of Freedom (DOF)**	**Provided Exercises**	**Actuator Type**	**Range of Motion (ROM)**
MIT-MANUS (*wrist module*) (Krebs et al., 2007)	3	WF/WE[1] WRD/WUD[2] FP/FS [3]	Electric motor	$60^o/60^o$ $30^o/45^o$ $70^o/70^o$
MAHI-EXO I (Gupta and O'Malley, 2006)	5	WF/WE WRD/WUD FP/FS EF/EE[4]	Electric motor	85^o (total) 85^o (total) 180^o (total) 90^o (total)
MAHI-EXO II (Pehlivan et al., 2011)	5	WF/WE WRD/WUD FP/FS EF/EE	Electric motor	72^o (total) 72^o (total) $>180^o$ (total) $>90^o$ (total)
RiceWrist (Gupta et al., 2008)	4	WF/WE WRD/WUD FP/FS	Electric motor	$42^o/42^o$ $33^o/19^o$ $90^o/90^o$
RiceWrist-S (Pehlivan et al., 2012)	3	WF/WE WRD/WUD FP/FS	Electric motor	120^o (total) 70^o (total) 180^o (total)
Wrist Gimbal (Martinez et al., 2013)	3	WF/WE WRD/WUD FP/FS	Electric motor	180^o (total) 60^o (total) 180^o (total)
Supinator Extender-SUE (Allington et al., 2011)	2	WF/WE FP/FS	Pneumatic	90^o (total) 90^o (total)
W-Exos (Gopura and Kiguchi, 2008)	3	WF/WE WRD/WUD FP/FS	Electrical motor	$60^o/50^o$ $30^o/20^o$ $80^o/75^o$
ITT-Wrist (Masia et al., 2009)	3	WF/WE WRD/WUD FP/FS	Electrical motor	$70^o/70^o$ $35^o/35^o$ $80^o/80^o$
Universal Haptic Device (Oblak et al., 2009)	2	WF/WE WRD/WUD FP/FS	SEA	No exact info
CRAMER (Allington et al., 2011)	3	WF/WE WRD/WUD FP/FS	Electrical motor	$90^o/70^o$ $10^o/15^o$ $90^o/85^o$

[1] WF/WE: Wrist flexion/ Wrist extension
[2] WRD/WUD: Wrist radial deviation/Wrist ulnar deviation
[3] FP/FS: Forearm pronation/Forearm supination
[4] EF/EE: Elbow flexion/Elbow extension

A pneumatically actuated forearm and wrist rehabilitation robot designed and prototyped in Çukurova University, Mechanical Engineering Department (Dağdelen, 2017). The name of the robot was called Çukurova Wrist Rehabilitation Robot (CWRR). The robot was composed of two actuated active degrees of freedom (DOF) and one passive DOF to align and comply with different sized users. With its special design, it could produce movements for wrist flexion/extension, wrist radial/ulnar deviation and forearm pronation/supination. The prototyped robot, CWRR, was end-effector type robot and it was grounded. This robot had only the capability of providing passive exercises with on-off position control approach. The summarized information will be presented in following subsections.

2.2. The Information About Previously Designed Old-Version CWRR

2.2.1. Overall Schematic Description of Old Version CWRR

Overall structure of the robot's old version is shown in Figure 2.1. The robot's structure is composed of two setups called as (i) Mechanical setup and (ii) Control setup. The control setup has two subsystems as main control setup and pneumatic control setup.

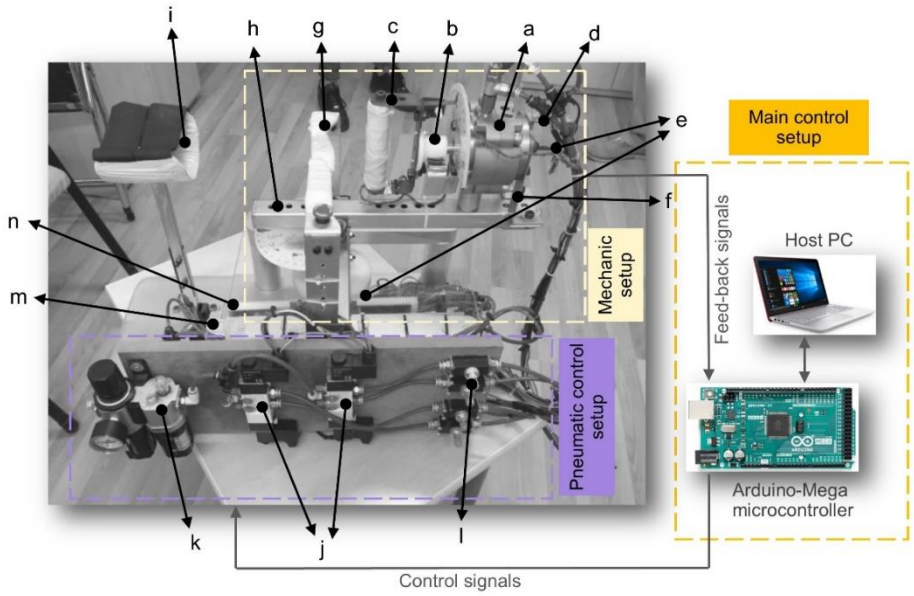

Figure 2.1. Old version of CWRR

To understand the operation of the old version of CWRR, the mechatronic configuration of the entire robotic setup is illustrated in Figure 2.2. Bearing on the schema given in Figure 2.2, a Host PC is utilized to embed the written control script to the microcontroller and monitor the angular position of wrist and forearm. In addition to that, Host PC supplies electric power for the microcontroller. During any specific workout, the angular position of forearm (θ_1) and wrist (θ_2) are measured with potentiometers called "potentiometer 1" and "potentiometer 2" and fed-back to the microcontroller. To execute the on-off control algorithm, the position information is evaluated,

and microcontroller sends switching (i.e. digital) signals labeled the four-way relay card for either enabling or disassembling 24 V DC power to the solenoids of the directional control valves (DCV) with respect to the specified range of motion (ROM). Four output signals of relay card are labeled as S1 and S2 for DCV-a and, S3 and S4 for DCV-b. Manual pressure regulating valve is utilized to adjust working pressure for the pneumatic system. In old version, working pressure (P_w) is preferred as 6 bar absolute.

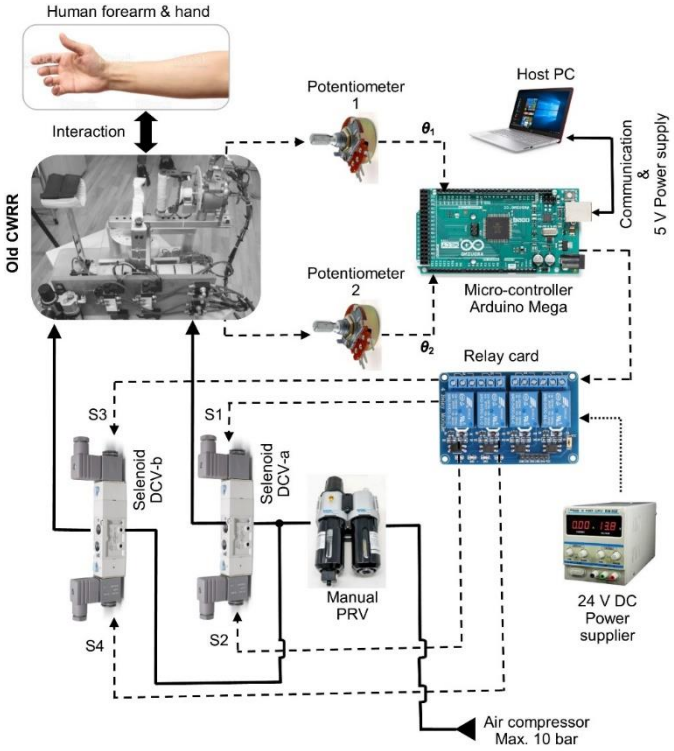

Figure 2.2. The mechatronic setup of old version CWRR

2.2.2. Mechanical Setup and Specifications of Old Version CWRR

In mechanical setup, the robot includes the necessary parts to provide suitable motions through the pneumatic actuators by taking compressed air. For clarity, the labels specified in Figure 2.1 to describe the robot's mechanical parts are specified as in Table 2.2.

Table 2.2. Mechanical part definitions of old version CWRR

Part Label	**Part Definition**	**Part Specification**
a	Vane type semi-rotary cylinder	CHELIC- RTM80/270 Operating pressure range: 0-10 bar Range of motion ability: 270^{o}
b	Flange between cylinder and end-effector	Produced from polymeric material
c	End-effector	Produced from steel material
d	Potentiometer support unit	Produced from steel material
f	Actuator support	Produced from steel material
g	Wrist support	Produced from steel material
h	Rotary platform	Produced from steel material
i	Forearm support	Produced from steel material
m	Rodless linear cylinder	MEDAN- PLF-0250M Stroke length: 250 mm Piston diameter: 25 mm Operating pressure: 8 bar
n	Rack and pinion gear mesh	Produced from polymeric material

The old version of the robot has the following mechanical issues as following with outfitted the parts as in Table 2.2:

- Most of the structural components are made of steel, which adds to the robot's total weight and may compromise user comfort and safety.
- The utilized vane type semi-rotary and linear pneumatic cylinders are heavy and have high friction torque/force characteristics reducing the trajectory performance.
- The most of the robot’s structural elements are not in adjustable form. Because of this, the old version robot lacks the sufficient passive DOFs to fit with varying user profiles (i.e. size, age, and sex).

Bearing on these mechanical issues, some of the parts are modified based on specified rationale in Table 2.3.

Table 2.3. Modifications to be carried out in mechanical setup

Modification Step	**Part to be Modified**	**Modification Rationale**	**Modification Method**
Step 1	(a) Vane type semi-rotary cylinder	The cylinder is too heavy, and the cylinder has high frictional characteristics.	Replacing of existing cylinder with a new one which has low weight and low friction characteristics
Step 2	(b) Flange between cylinder and end-effector	The part should be modified to enable mounting of torque and force sensors on it.	Designing of new flange structure and production of new part with 3D printing method
Step 3	(c) End-effector	The part is heavy, possibility to harm the user and not adaptable for mounting torque and force sensors on it.	Designing of end-effector with multiple parts with 3D printing method
Step 4	(d) Potentiometer support unit	The part is not suitable for mounting of utilized encoder and increasing weight of rotary platform.	Designing a new platform suitable for joining of encoder with the rotary cylinder
Step 5	(f) Actuator support	The support is made of steel raising the overall weight of the robot. Also, it has no passive DOF on it to fit with varying users.	Designing a new lightweight support which has passive DOF. Production with 3D printing method
Step 6	(g) Wrist support	It is a heavy, steel-built unit. This component makes the robot heavier overall and poses a risk to the user.	Designing of a new support unit and production with 3D printing method
Step 7	(i) Forearm support	It is a heavy, steel-built unit like wrist support. In addition, it lacks necessary passive DOFs to fit with varying users.	The design of a low weight support unit which has necessary passive DOFs and production with 3D printing method.
Step 8	(m) Rodless linear cylinder	The cylinder has excessively high friction force characteristics complicating control performance of the robot.	Replacing the current cylinder with a double acting, low friction characteristic cylinder.

2.2.3. Control Setup and Specifications

Control setup is responsible from generating accurate control signals for the pneumatic control valves with respect to the embedded script on Arduino interface with host PC. In control unit, the robot has following equipment as listed in Table 2.4 with respect to the labeling in Figure 2.1.

Table 2.4. Control setup part definitions and specifications

Part Label	**Part Definition**	**Part Specification**
	Host PC	64bit 2400 CPU 3,10 *Ghz* processor
	Microcontroller	Arduino Mega
e	Potentiometers	Standard 0-5 VDC analog potentiometers
j	Solenoid directional control valves	MINDMAN- MVSC-220-4E2R-BSP 5/3- exhaust center type Operating voltage : 24 VDC
k	Pressure regulating valve	MINDMAN- MAFR300-8A-G Manually controlled Output pressure ability: 0-10 bar
l	Speed control valves	MSC200-8A Manually controlled One way controlled (check valve type)

The old version of the robot has the following control issues as following:

- The robot has solenoid directional valves manipulated with either 0 V or 5 V DC control signals activating or deactivating the 24 V DC supply with relay cards. That means the orifice areas of the valves are either fully opened or closed wherein on-off and PWM controller approaches are generally applicable. This kind of valve utilization complicates the application of precise control approaches on the robot such as impedance/admittance control, interaction torque/force control etc.
- Data collection and control are achieved through the use of the Arduino microcontroller. For this reason, the processing of the data is slow and monitoring of controlled or manipulated variables is quite hard. In addition, the Arduino programming interface does not allow the implementation of sophisticated control algorithms.
- The old version of the robot has only position sensors which are analog output potentiometers and just enabling the position control approach on the robot. As it is known, the position information produced by potentiometers is noisy complicating to read correct and precise position information.
- The robot's previous control configuration lacked pressure sensors. This shortcoming makes it impossible to create a controller that adjusts the cylinder's chamber pressures.
- The robotic setup is lacking torque/force sensors. This shortcoming restricts the use of torque/force based controller techniques and makes it impossible for the expert to observe the torque/force interaction.

Based on this control issues, following modifications are considered to upgrade the robot’s controller performance as in Table 2.5.

Table 2.5. Modifications to be carried out in control setup

Modification Step	**Part to be Modified**	**Modification Rationale**	**Modification Method**
Step I	Microcontroller	Arduino microcontrollers does not allow to design and application of complex and efficient controller approaches.	Utilization of more efficient and fast data acquisition control card which enables complex programing and suits for real time hardware-in-the-loop simulations
Step II	(e) Rotary potentiometers	Potentiometers are not meant for use in precise control applications since they provide noisy signals.	The changing of potentiometers with excessively precise rotary encoders
Step III	(j) Solenoid directional control valves	Solenoid control valves is not suitable for precise and proportional control approaches. Mostly suitable for on-off control approach.	Replacing the solenoid directional control valves with proportional and precise ones.
Step IV	(k) Pressure regulating valve	Manual regulating valve does not permit precise working pressure regulation in the cylinder chambers.	Replacing manual regulating valve with proportionally controlled precise pressure regulating valve
Step V	(l) Speed control valves	The utilization of proportional directional control valves eliminates the usage of manual speed control valves.	Removal of manual speed control valves from the control setup
Step VI	Pressure sensors (does not exist)	The control setup of the robot does not contain any pressure sensors. This missing component does not permit the expert to track and control the chamber pressures.	Addition of pressure sensors close to the air inlet ports.
Step VII	Torque and force sensors (does not exist)	The shortage of the torque and force sensors does not permit the design of interaction torque based controllers. Also, the tracking of interaction torque/force is impossible.	Addition of sensitive torque and force sensors between the newly designed flange and end-effector
Step VIII	Sensor amplifier	The sensitive torque and force sensors need signal amplifier to produce readable signals for DAQ card.	Addition of sensor amplifier which tones up mV level output signals to V levels

2.3. Preliminary Work About Proportional Pneumatic Valve Modelling

In literature there is no qualitative search on the modelling of pressure regulating valves due to their linearity between input control signal and output pressure. The linear modelling of PRVs will be covered in Section 3 in detail.

From the directional control valves point of view, the flow in the valves was compared to the flow in a smooth opening placed in a simple pipe, during the early pneumatic control investigations. This assumption assumed one degree of freedom compressible flow and ignored air leakages. In the majority of pneumatic researches (Xiangrong and Goldfarb, 2007; Saleem et al.,

2017; Richer and Hurmuzlu, 2000; X. Brun et al., 1999; NOURI and SAUDI, 2014), this model, known as the standard model in the literature, was preferred. However, recent advancements in sensor technology and computational capabilities enabled us to manufacture more sophisticated and sensitive valves, for which the classic flow model cannot accurately anticipate the flow. As a result, the standard model was revised, and an improved flow model with more accurate outputs was proposed in later investigations (Arbab and Najafi, 2014; Harris et al., 2012; Meng et al., 2011; Van Der Merwe et al., 2013; Rad et al., 2014; Rad and Hancu, 2017).

As noted in preliminary work of proportional directional control valve, these valves have rather non-linear fashion compared to the PRVs wherein the flow behavior of the valve is represented in Figure 2.3 (a). These valves work in the input signal range of u= 0-10 V. To define the flow details of the respective valve, the spool mechanism of the valve is also depicted in Figure 2.3 (b). According to the command signal (u), the spool moves either in left (l) or right (r) direction. According to the manufacturer's technical data, the valve blocks completely both ports (2 and 4) and no flow is observed ($\dot{m}_2$=$\dot{m}_4$ =0 kg/s). Also, the valve has a certain zero flow region (Δu_{null}) such that $\dot{m}$=0 kg/s around 5 V. Between the signal range of 0≤u≤5 V, the spool moves in the right (r) direction and opens the flow passage from port 1 to port 2, and a certain mass flow rate $\dot{m}_2$ is sent to chamber "a". At this signal range, the passage from port 4 to port 5 is opened at the same time and the air in chamber "b" is exhausted. In the signal range of 5≤u≤10 V, the spool moves in the left (l) direction and air moves through the passage from port 1 to port 4 with a certain mass flow rate of $\dot{m}_4$, and the counter chamber which is "b" is filled. During this operating condition, the air passage from port 2 to port 3 is simultaneously opened and the air in chamber "a" is exhausted.

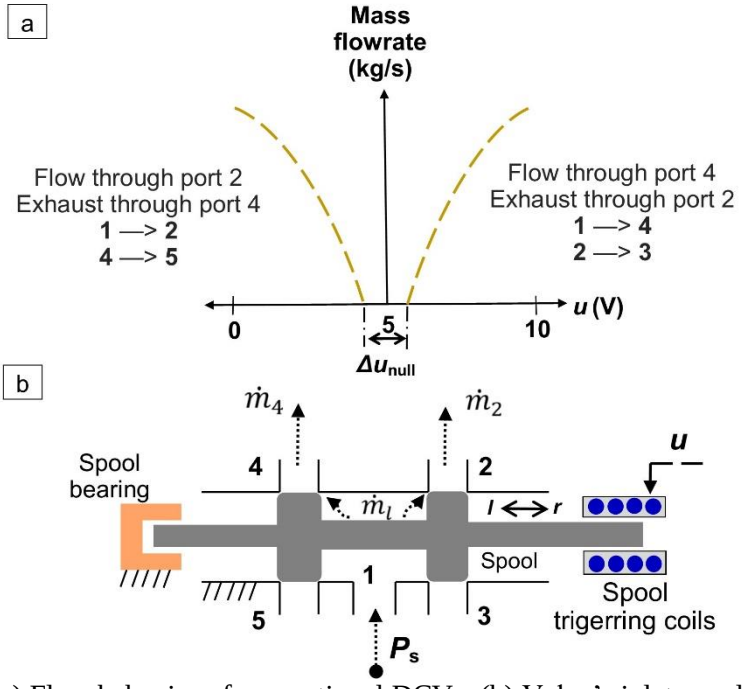

Figure 2.3. (a) Flow behavior of proportional DCVs; (b) Valve's inlet spool mechanism

In literature, there exist two commonly used flow models which relates the input control signal with resultant air mass flow rate. These models are represented based on the general formula presented in Equation 2.1 as following:

$$\dot{m} = C_d \cdot A(u) \cdot \beta(P_r) \tag{2.1}$$

In this equation, $\dot{m}$ is the mass flow rate, $A(u)$ is the orifice area of the valve that is dependent on the control signal u, C_d is the discharge coefficient of the valve, and $\beta(P_r)$ is the flow function that is dependent on the pressure ratio (P_r). The pressure ratio (P_r) is the ratio of the valve's downstream (outlet) pressure (P_d) to its upstream (inlet) pressure (P_u). Figure 2.4 depicts a general perspective of the flow function for the standard and improved models.

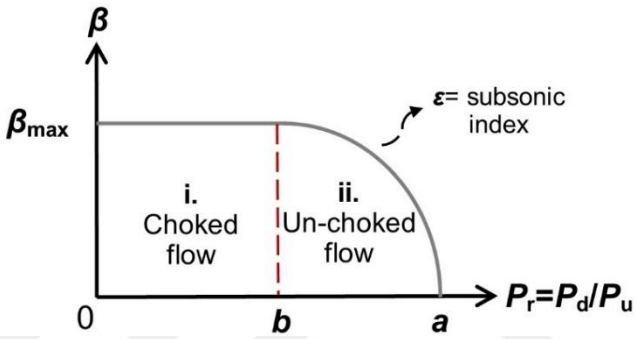

Figure 2.4. The curve of flow function

Figure 2.4 depicts two distinct behaviours: (i) choked flow and (ii) un-choked flow. The flow continues at a constant and maximum level (β_{max}) in region "i" until the pressure ratio (P_r) equals the critical pressure ratio (b). When the pressure ratio exceeds this crucial threshold (b), the flow behaves differently and follows the pattern in region "ii". Namely, flow amount decreases in a parabolic fashion. The flow continues until the pressure ratio (P_r) equals the cracking pressure ratio (a), at which time the air flow is terminated ($\dot{m} = 0$ kg/s). The cracking pressure ratio (CPR), which describes the ratio of the downstream pressure to the upstream pressure (i.e. P_d/P_u) at the valve for which there is no flow due to pressure difference (Rad et al., 2014; Rad and Hancu, 2017), is one of the common and crucial characteristics for these two models. In both models, the CPR value is typically assumed to be a=1. This means that the flow will continue until the pressure on the valve's input and exit sides is equal.

The pressure ratio (P_r) at the valve is computed differently depending on the flow direction. That is, the filling state occurs when the valve feeds air into a closed chamber, and the exhausting state occurs when the valve discharges compressed air from the chamber to the atmosphere. When filling, the downstream pressure (P_d) equals the pressure in the chamber, and the upstream pressure (P_u) equals the supply pressure (P_s) (i.e. P_d=P_a or P_b; P_u=P_s). When

exhausting, the downstream pressure is atmospheric air pressure, and the upstream pressure is chamber pressure (P_d=P_{atm}; P_u=P_a or P_b).

Equation 2.2 expresses the flow function for the standard flow model, where the constants C_1 and C_2 are defined by Equations 2.3 and 2.4, respectively. T_u, k, and ε are the upstream air temperature, specific heat ratio of air, and subsonic index, respectively. The CPR is considered to be a=1 in the standard model for all operating situations, and the critical pressure ratio is likewise preferred to be b=0.528. Furthermore, the subsonic index is most utilized at a value of 0.5.

$$\beta\,(P_r) = \begin{bmatrix} C_1 \cdot \dfrac{P_u}{\sqrt{T_u}} & if\ P_r \le b \\ C_2 \cdot \dfrac{P_u}{\sqrt{T_u}} \cdot P_r^{\frac{1}{k}} \cdot \left(1 - (P_r)^{(k-1)/k}\right)^{\varepsilon} & if\ P_r \ge b \end{bmatrix} \tag{2.2}$$

$$C_1 = \sqrt{\frac{k}{R} \cdot \left(\frac{2}{k+1}\right)^{(k+1)/(k-1)}} \tag{2.3}$$

$$C_2 = \sqrt{\frac{2 \cdot k}{R \cdot (k-1)}} \tag{2.4}$$

Equation 2.5 depicts the flow function utilized in the modified flow model, where the constant C_1 is given in Equation 2.3.

$$\beta\,(P_r) = \begin{bmatrix} C_1 \cdot \dfrac{P_u}{\sqrt{T_u}} & 0 \le P_r \le b \\ C_1 \cdot \dfrac{P_u}{\sqrt{T_u}} \cdot \left[1 - \left(\dfrac{P_r - b}{a - b}\right)^2\right]^{\varepsilon} & b \le P_r \le \sigma_{lam} \\ C_1 \cdot \dfrac{P_u}{\sqrt{T_u}} \cdot \left[\dfrac{1 - P_r}{1 - \sigma_{lam}}\right] \cdot \left[1 - \left(\dfrac{\sigma_{lam} - b}{a - b}\right)^2\right]^{\varepsilon} & \sigma_{lam} \le P_r \le 1 \end{bmatrix} \tag{2.5}$$

The modified model's critical pressure ratio (b) is commonly regarded to be greater than 0.528. Furthermore, there is no precise information on the critical pressure ratio value specification because the value of this parameter varies from valve to valve, even among valves of the same type. To put it another way, Harris et al. (2012) proposed b=0.23, whereas Van der Merwe et al. (2013), Rad et al. (2014), and Rad and Hancu (2017) all proposed b as a function of control signal (b=f(u)). In contrast to the standard model, the parabolic variation in the "ii. un-choked" flow behaviour occurs by an unusual subsonic index value (ε) rather than a value of 0.5. Although the CPR value is advised to be between 0 and 1, a=1 is assumed in the majority of research (Arbab and

Najafi, 2014; Harris et al., 2012; Meng et al., 2011). Furthermore, the laminar flow parameter, σ_{lam} exists; however, because the value of σ_{lam} is relatively close to one, this value has been ignored.

The preferred flow models in pneumatic system investigations, as well as the corresponding parameter assumptions in the literature, are summarized in Table 2.6. When Table 2.6 is examined in depth, the range of parameter values is readily apparent. For example, in some researches, the discharge coefficient is considered to be C_d=0.8 (Xiangrong and Goldfarb, 2007; Saleem et al., 2017), whereas in others, it is believed to be C_d=0.25 (Richer and Hurmuzlu, 2000). The reasons for these utilizations are not explicitly explained in the papers described above.

Table 2.6. Utilized flow models in literature and parameter assumptions

Previous Studies	**Flow Model**	**Control Valve**	**(*a*) Assumption**	**(*b*) Assumption**	**(C_d) Assumption**
Xiangrong and Goldfarb (2007)	Standard model	Proportional	a=1	b=0.5	C_d=0.8
Saleem et al. (2017)	Standard model	Proportional	a=1	b=0.529	C_d=0.8
Arbab and Najafi (2014)	Improved model	Solenoid	a=1	Catalogue or calculation	Catalogue or calculation
Richer and Hurmuzlu (2000)	Standard model	No info	a=1	b=0.528	C_d=0.25
Brun et al. (1999)	Standard model	No info	a=1	Catalogue or calculation	Catalogue or calculation
NOURI and SAUDI (2014)	Standard model	Proportional	a=*1*	b=0.528	No usage
Harris et al. (2012)	Improved model	Solenoid	a=1	around 0.23 (changing with their specified flow paths)	Changing with their specified flow paths
Meng et al. (2011)	Improved model	Proportional	a=1	Constant b, but no value specification	C_d=f(P_r)
Van der Merwe et al. (2013)	Improved model	Proportional	a=f(u)	b=f(u)	C_d=0.5750
Rad et al. (2014)	Improved model	Proportional	a=f(u)	b=f(u)	C_d=0.5504
Rad and Hancu (2017)	Improved model	Proportional	a=f(u)	b=f(u)	C_d=0.5513

Because of these inconsistencies and shaky assumptions mentioned in Table 2.6, Rao and Bone (2008), Ning and Bone (2005), and Bobrow and McDonell (1998) proposed that these two commonly used flow models (i.e. standard and improved) could not accurately predict the flow in valves, and they proposed quite distinct and efficient empirical models. To resolve the inappropriate modelling issue, Bobrow and McDonell (1998) proposed a new empirical model stated as in Equation 2.6 as following:

$$\dot{m} = \begin{bmatrix} \frac{1}{c_p \cdot T_u} \cdot (P_u - P_d)^{0.5} \cdot (c_1 \cdot u + c_2 \cdot u^2) & \text{filling} \\ \frac{1}{c_p \cdot T_u} \cdot (P_u - P_d) \cdot (c_3 \cdot u + c_4 \cdot u^2) & \text{exhauisting} \end{bmatrix} \quad (2.6)$$

In Equation 2.6, the model parameters c_1, c_2, c_3, and c_4 are determined experimentally. Further, u corresponds to the valve control signal. Also, the CPR is assumed to be a=1 for all operating conditions of the valve in the model of Bobrow and McDonell (1998).

The model developed by Bobrow and McDonell (1998) was more accurate than the standard and improved models. However, as with prior investigations, the CPR was assumed to be a=1 for all operating conditions of the valve. In this regard, the FESTO MPYE-5-1/8-HF-010-B (i.e. utilized valve in this thesis study) proportional DCV has been used in this investigation, where the actual flow has been carefully evaluated. When data from the tests was evaluated, it has been revealed that the CPR values fluctuate significantly depending on both the control signal (u) of the valve and the supply pressure (P_s). Furthermore, the flow properties of the valve have changed significantly depending on the flow direction (filling or exhausting). It was discovered that the tested valve had extremely distinct flow characteristics, particularly in small orifice sizes (the control signal around 5 V).

2.4. Preliminary Work About Friction in Pneumatic Cylinders

The friction identification in pneumatics have been a challenging issue. Different kinds of efforts have been spent to identify and understand its behavior. Some studies neglect the friction due to the usage of low-stiction cylinder (Z. Lin et al., 2018), but the efficient control performances could not be obtained. Furthermore, the effects of friction were tried to be reduced by the application of vibration implementation by Qian et al. (2022). However, these types of applications cannot completely cancel the friction and have technical limitations to apply in controller approaches. Due to the limitations of these types of alternative methods, identification of nonlinear friction characteristics is a primary requisite to greatly improve the control performances.

The friction models for sliding mechanical elements in the literature can be divided into two families, namely classical (or static) friction models and complex (or dynamic) friction models (Odabaş and Morgül, 2020). Coulomb (Guo et al., 2013), Coulomb-viscous (Yerlikaya and Balkan, 2018), and Stribeck (Beerens et al., 2022) models belong to the classical friction models whereas friction models such as LuGre (Mashayekhi et al., 2022), modified LuGre (Saha et al., 2016) and Dahl (Piatkowski, 2014) form the complex friction models. More dynamic friction models exist in the literature (Keck et al., 2017; Liang et al., 2012), but these complex models were not preferred for the friction force/torque definition of the pneumatic cylinders due to their complexity and difficulty in estimating their parameters. Unlike existing models, some researchers proposed

empirical friction models for the pneumatic cylinders (Belforte et al., 1989). However, the usage of their models seems to be limited in pneumatic applications due to the difficulty of nonlinear fitting operation to the model parameters.

In the literature, for linear pneumatic cylinders, mostly Coulomb-viscous, Stribeck and LuGre friction models were utilized in control applications. As an example, Nouri (2004), Kosari and Moosavian (2015) utilized Stribeck friction model in their work. W. Lin et al. (2018) and Wang et al. (2004) applied Coulomb-viscous model. A more enhanced model called modified LuGre was investigated by Tran and Yanada (2013).

However, there have been conducted limited number of studies which focused on the determination of the friction characteristics in the vane type semi-rotary pneumatic cylinders. The first valuable study was carried out by Belforte and Raparelli (1997). In that study, the friction characteristics were defined with respect to only a single parameter i.e. chamber pressure. Empirical models were proposed that were quite hard to apply in control algorithms due to the lack of relation between speed and friction. In another study, the parameters of the LuGre friction model were tried to be determined with limited amount of experimental data by Schlüter and Perondi (2018). In that study, the estimation of friction parameters was performed at only one specific working pressure. In other words, the effect of pressure variations was not evaluated in broad sense.

2.5. Preliminary Work About Pneumatic Control Approaches on Rehabilitation Robots

There exist numerous control studies for control approaches in rehabilitation robots. In this section, position based low-level control frameworks are not taken into consideration. Instead, the high-level control approaches wherein either conventional or adaptive interaction force/torque control is evaluated will be presented.

For instance, a 3 DOF wrist rehabilitation robot was proposed by Mayetin and Kucuk (2022) wherein both position-based and force-based impedance control approaches were applied to obtain a good quality HRI. Chen et al. (2019) proposed an elbow exoskeleton for upper limb rehabilitation including forearm motions. Stiffness simulation was successfully applied with the application of series elastic actuators and cable drives. In another research, a five DOF lightweight upper limb rehabilitation robot was controlled by the impedance control approach by Wu et al. (2019). Yet another adaptive impedance control for lower limb rehabilitation robots was investigated by Brahmi et al. (2020). Wendong et al. (2020) proposed an electrically actuated upper limb rehabilitation robot based on HRI for the prediction of human motion intent rather than control of interaction force/torque. Aktan and Akdoğan (2021) studied different types of controllers (i.e. force, impedance control etc.) on an upper limb rehabilitation robot called DIAGNOBOT. (Shi et al. (2021) focused on the human-robot-centered adaptive control for safe interaction purpose in

lower limb rehabilitation robot. Sanjeevi and Vashista (2021) addressed stiffness modulation of a cable-driven leg exoskeleton for effective HRI.

In the existing literature, there have been studies that applied fuzzy logic in rehabilitation robotics; however, the majority of these studies focused on evaluating the position-tracking ability and improvements in the range of motion of the affected limb. As an example, Amiri et al. (2022) proposed an adaptive fuzzy logic controller for a multi-joint lower limb assistive robot. However, human-in-the-loop effects were not evaluated in depth and only the positional trajectory tracking performances were addressed. A different fuzzy logic control example was implemented by Huang et al. (2021). The patient improvement was evaluated from the range of motion point of view, but the human force/torque interaction dynamics were not considered. Bouteraa et al. (2021) proposed an EMG-triggered fuzzy logic-based control for a wrist robot. The ability of fuzzy was discussed for only feature extraction (i.e. intended motion) and, the force/torque interaction properties were not included. Sharma et al. (2021) presented an optimal fuzzy logic control strategy for a 4 DOF lower limb rehabilitation robot. The position-tracking performance was discussed, and human robot interaction dynamics were ignored. The studies that discuss the adaptive high-level control (either impedance or admittance) are quite limited. As an example, AYAS and ALTAS (2017) proposed a fuzzy adaptive admittance control for a lower limb robotic platform. The main goal in their study was producing either resisting or assisting forces to the ankle joint to help the patient to complete the trajectory rather than interaction control. However, the ability of the fuzzy logic control was not evaluated in depth from the interaction force/torque point of view.

3. MATERIAL AND METHOD

3.1. Design Details of Robot's New Version

The robot's new version is modified with respected to the rationale mentioned in Section 2.2.2. The mechanical setup is improved by utilizing 3D printing methodology for production of newly designed parts to reduce the overall weight of the robot and prevent potential hazards from the users. New version is presented in two setups as (i) mechanical setup and (ii) control setup as in Figure 3.1.

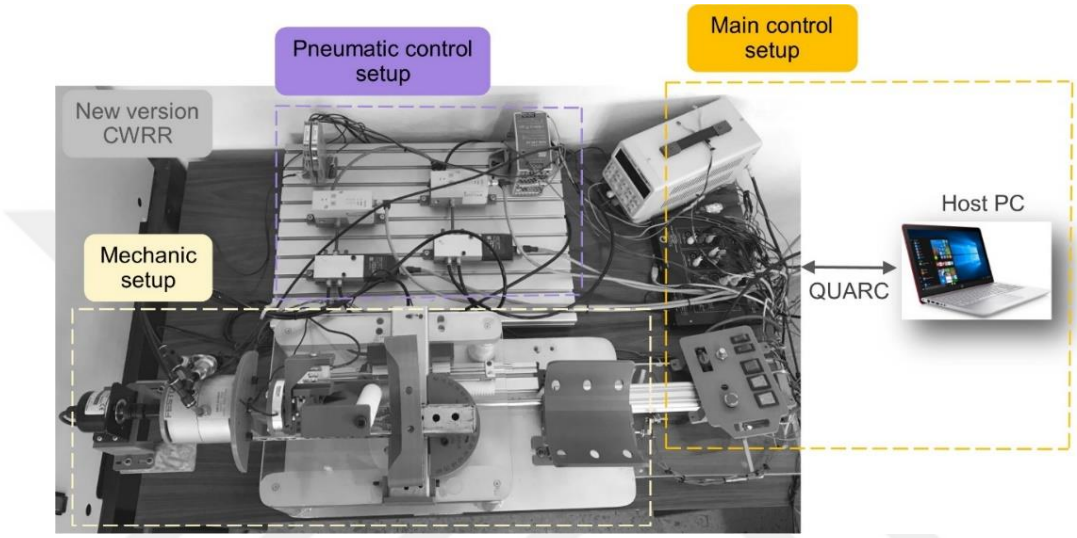

Figure 3.1. General representation of new version CWRR

Each setup of new version will be covered in the following subsections. In addition, the modification carried out on either mechanical or control setup will be mentioned in the respective sections.

3.1.1. Mechanical Setup Definitions and Modifications

The mechanical modifications mentioned in Table 2.3 are carried out and the robot is enhanced by utilizing lightweight and low-friction actuators and lightweight part designs. For this aim, 3D printing method is preferred. The same robotic ground is used without any modifications. In addition to that, the same rotary platform and rack and gear mesh are utilized. Many of the remaining mechanical parts are designed such that passive DOFs exist to accommodate to varying sized users. New version of the robot's mechanical setup is presented in Figure 3.2 in detailed form.

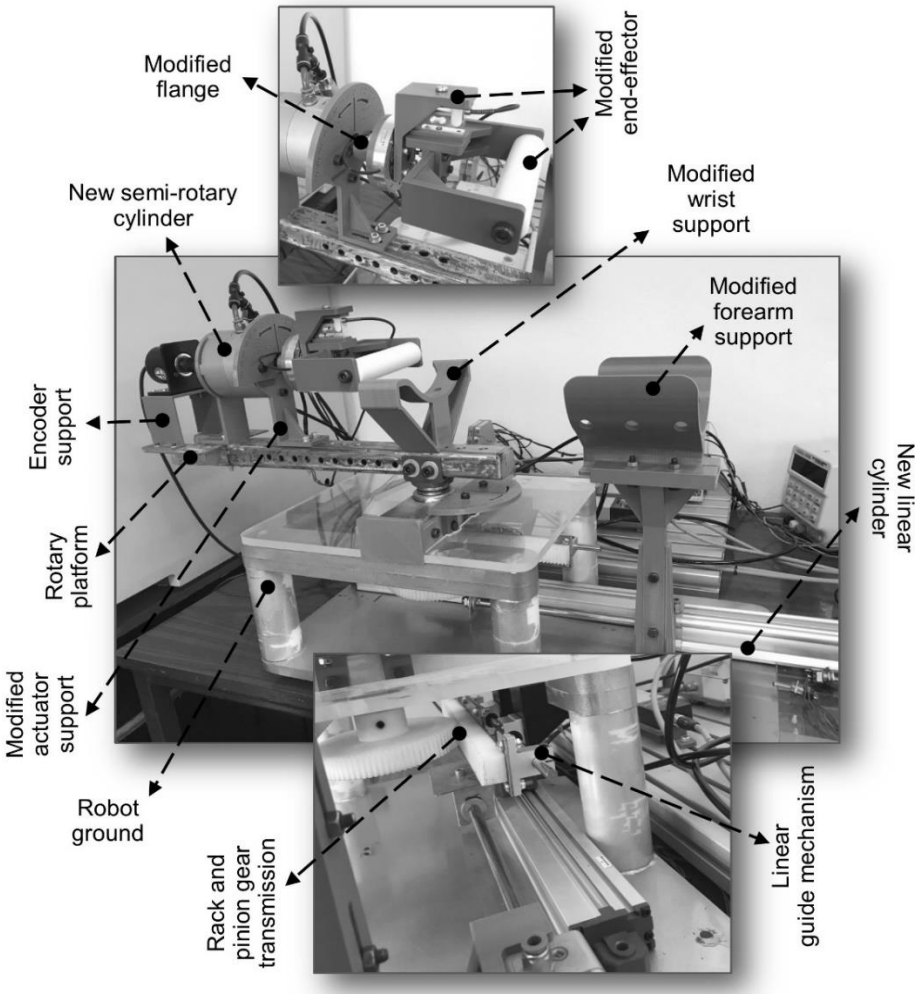

Figure 3.2. Mechanic setup of new version CWRR

The information of the upgraded mechanical setup with respect to the steps given in Table 2.3 is detailed in Table 3.1.

Table 3.1. Mechanical part definitions of new version CWRR

Part Definition	**Part Specification**
New semi-rotary cylinder	FESTO DRVS-32-270-P Vane type and Double shaft property Maximum working pressure: 8 bar (Step 1 applied)
Modified flange	Designed to adapt torque sensor Produced with 3D printing method (Step 2 applied)
Modified end-effector	Designed to adapt torque and force sensor Produced with 3D printing method (Step 3 applied)
Encoder support	Designed to have one passive DOF Produced with 3D printing method (Step 4 applied)
Modified actuator support	Designed to have one passive DOF Produced with 3D printing method (Step 5 applied)
Modified wrist support	Designed to provide more space for the movements Produced with 3D printing method (Step 6 applied)
Rotary platform	No change The same platform is used
Modified forearm support	Designed to have two passive DOFs Design based on multiple parts Production with 3D printing method (Step 7 applied)
New linear cylinder	FESTO DSBC-32-300-PPVA-N3 Double acting type Piston diameter:32 mm Stroke length: 300 mm Maximum working pressure: 12 bar (Step 8 applied)
Rack and pinion gear mesh	No change The same gears are utilized

3.1.2. Control Setup Definitions and Modifications

To modify the control setup, the steps stated in Table 2.5 are followed. Towards this goal, the control setup of old version CWRR wherein the pneumatic control setup and main control setup exist is improved by utilizing proportional pneumatic control valves and functional data acquisition and control card. Additionally, pressure sensors are added to control setup to efficiently track and control the chamber pressures, and torque and force sensors together with their amplifier are assembled to the system to control the interaction torque at the end-effector. Improved version of control setup of CWRR is depicted in Figure 3.3 in detail.

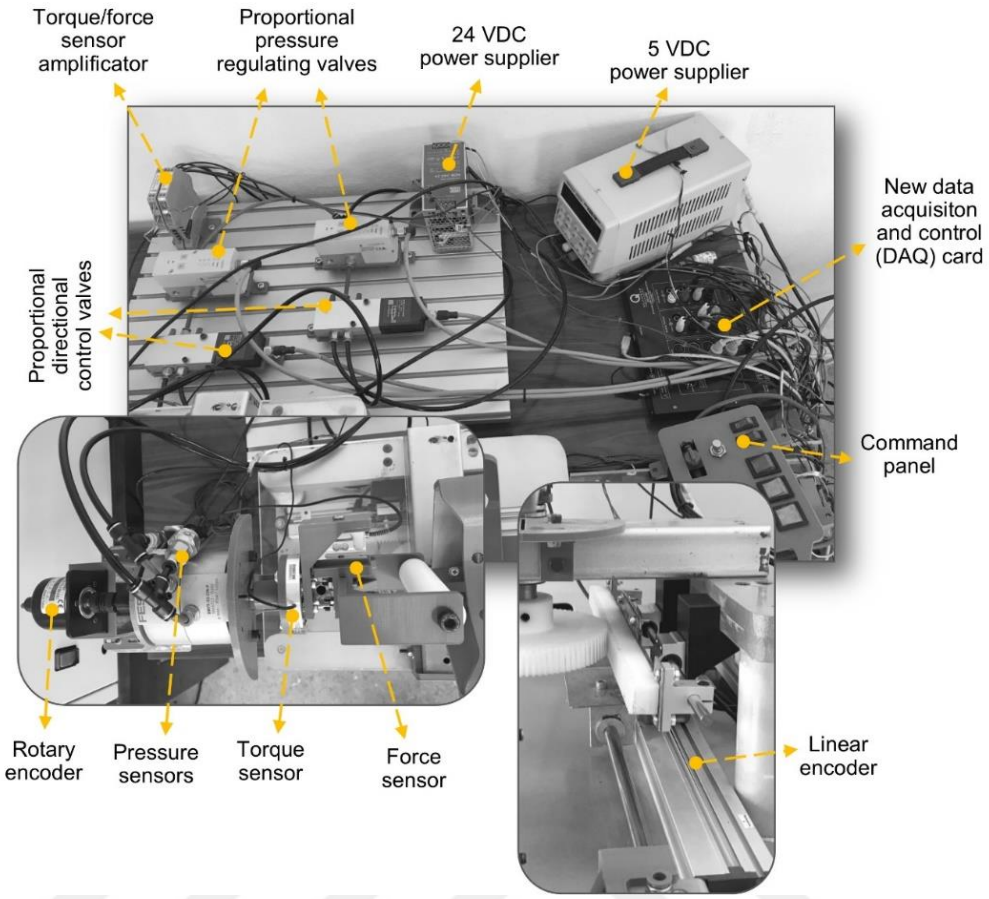

Figure 3.3. Control setup of new version CWRR

The control equipment described in Figure 3.3 are detailed in Table 3.2.

Table 3.2. Control equipment definitions of new version CWRR

Equipment Definition	**Equipment Specification**
Data acquisition and control (DAQ) card	QUANSER Q8 USB 16-bit resolution Output signal range: +/- 5 V or +/- 10 V # of analog input: 8 # of analog output: 8 # of digital input: 8 # of digital output: 8 (Step I applied)
Rotary encoder	ATEK ARC-S-50-1024-TTL-3-3M-A-8-C Incremental type (Step II applied)
Linear encoder	ESKON ES500-B2-010-TTL-3-3M-C-R Resolution: 0.01mm Measuring range: 300mm (Step II applied)
Proportional directional control valves	FESTO MPYE-5-1/8-HF-010-B 5/3- Closed center type Supply voltage: 24 VDC Control signal input: 0-10 VDC (Step III applied)
Proportional pressure regulating valves	FESTO VPPM-6L-L-1-G18-0L10H-V1P Supply voltage: 24 VDC Control signal input range: 0-10 VDC Output pressure range: 0-10 bar (Step IV applied)
Pressure sensor	Standard analog type Supply voltage: 5 V Output voltage: 5 V (Step VI applied)
Torque sensor	ME-Meßsysteme-TS70a-10Nm Torque measurement range: +/- 10 Nm (Step VII applied)
Force sensor	ME-Meßsysteme-KD40S-100N Force measurement range: +/- 100N (Step VII applied)
Torque and force sensor amplifier	ME-Meßsysteme-GSV-1H-010/2K5/2 # of analog output: 1 Input sensitivity: 0.2/0.5/1/2 mV/V Output voltage: +/- 10 V (Step VIII applied)

3.1.3. Mechatronic Representation of New Version CWRR

The mechatronic setup of new version CWRR is illustrated in Figure 3.4 to clarify the working principles and the connections between the whole equipment including main control setup and pneumatic control setup. The mechatronic setup consists of a host PC including MATLAB/Simulink. Bi-directional communication between host PC and DAQ card is carried out with an interface packet module, QUARC. The DAQ card leads all the setup by both receiving the

sensory data and generating control signals for the control valves with respect to the embedded controller framework.

On the mechatronic setup there exist many sensory information. Among them, P_a and P_b are the absolute pressures measured with pressure sensors PS-a and PS-b for the chambers "a" and "b", respectively. θ and x represent the measured angular position of the semi-rotary cylinder and linear position of the linear cylinder. T_i and F_i are the interaction torque and interaction force measured on the end-effector between the user and robot. On the other hand, u_a and u_b are the outputted control signals for proportional DCV-a and DCV-b respectively. Finally, z_a and z_b correspond to the control signals for proportional PRV-a and PRV-b, respectively.

5 VDC power supplier supply electrical energy to the pressure sensors whereas 24 VDC power supplier provides energy for proportional DCV, proportional PRV and torque/force sensor amplifier. The rotary and linear encoders are supplied from directly DAQ card by utilizing encoder connection ports. Pressurized air with a maximum P_{comp}=10 bar is supplied by an air compressor to the proportional pressure regulating valves (PRV-a and PRV-b). These valves adjust the air pressure by taking the control signals z_a and z_b. Then, PRVs send the regulated air to the proportional DCV-a and DCV-b. These directional control valves direct the regulated air to the cylinder chambers (i.e. chambers "a" and "b") with respect to the input control signals u_a and u_b, respectively.

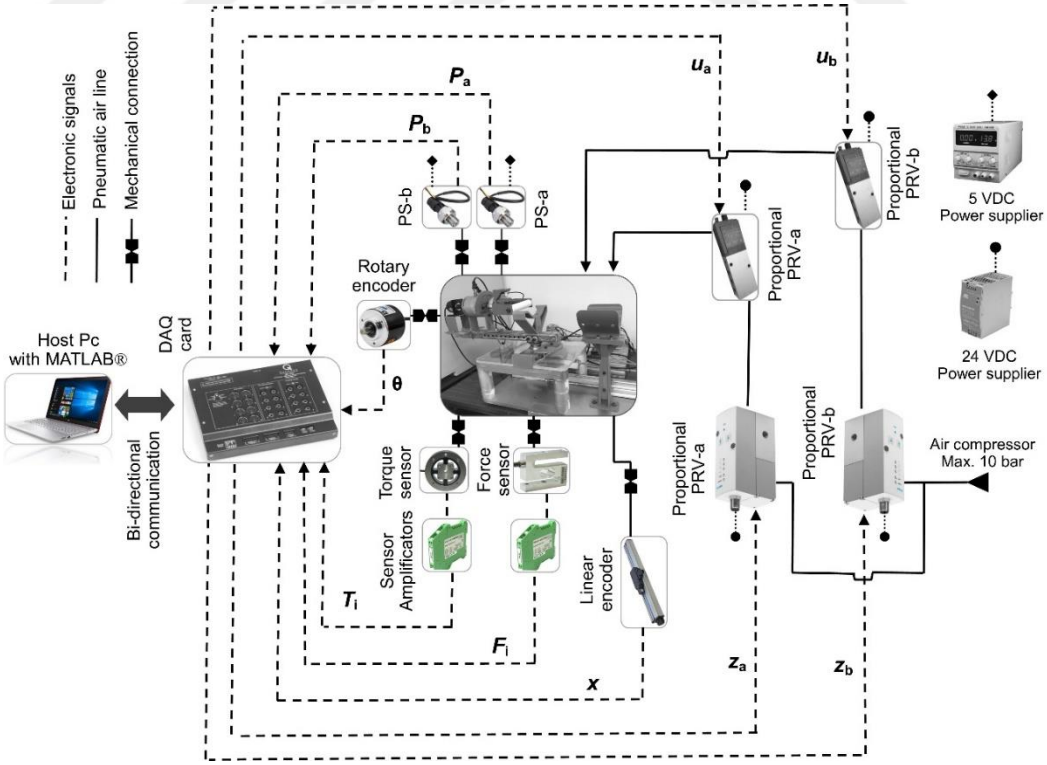

Figure 3.4. The mechatronic setup of new version CWRR

3.2. Mathematical Model of Robot Dynamics

In this section, the dynamics of the robot's moving parts will be presented including the internal and external dynamics of both semi-rotary cylinder and linear cylinder. In addition to that, the control behaviors of the pneumatic valves will be covered to derive a logic between input control signals and desired (i.e. controlled) interaction torque.

The main power is provided from the pneumatic cylinders which take the pressurized pneumatic air and generates a net torque/force with respect to its internal dynamics. Therefore, firstly the load dynamics of the pneumatic actuators will be covered.

3.2.1. Load Dynamics of Pneumatic Actuators

The robot poses two different type of pneumatic actuators as (i) vane type semi-rotary pneumatic cylinder and (ii) double acting pneumatic cylinder. Since their working principle and general structures are different, their dynamic investigations will be covered in seperate subsections.

Dynamics of the Vane Type Semi-Rotary Cylinder

To understand the working principles and generate a dynamic model of the actuator, it is requisite to analyze the inlet structure of the semi-rotary cylinder (i.e. FESTO DRVS-32-270-P). The inlet structure of the cylinder is represented as in Figure 3.5.

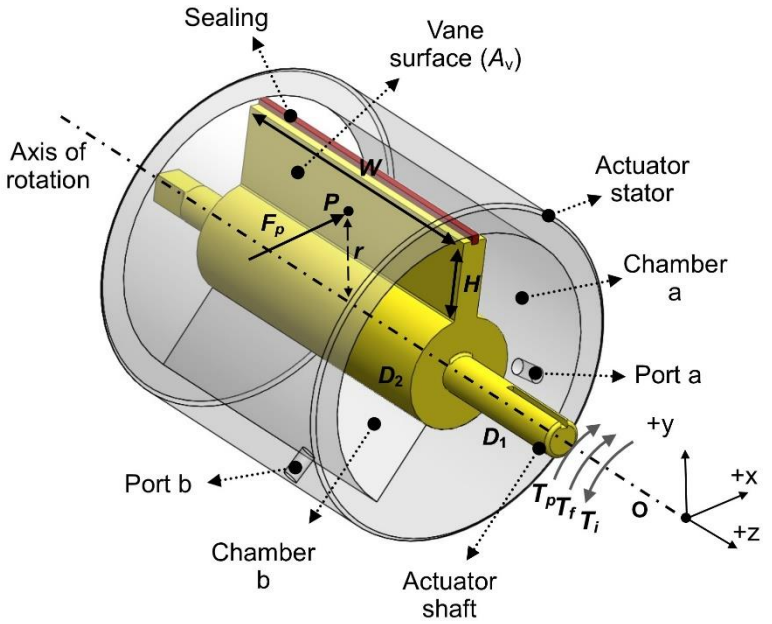

Figure 3.5. Inlet structure of utilized vane-type semi rotary cylinder (FESTO DRVS-32-270-P)

The structure of the cylinder is based on a rotating vane inside a cylindrical stator. It has two chambers called as chamber "a" and chamber "b". The cylinder takes the pressurized air

through the port a and port b for chambers “a” and “b”, respectively. According to the pressurized chamber, the cylinder vane is rotated either in Counter-Clock-Wise (CCW) or Clock-Wise (CW) direction as a result of generated pneumatic force, F_p. This generated force acts at the middle position of the vane surface area which is in a distance r from the axis of rotation of the vane. The cylinder shaft has two stage shaft. One of this stage is the output shaft where the power is transmitted either in forward or backward side and has D_1 diameter in both of the sides. The second stage shaft is combined with the vane and has D_2 diameter. W and H are the vane width and height, respectively. A_v is the vane surface area where the pneumatic force is acted on. The vane has peripheral sealing material to prevent potential pressurized air leakages. T_p, T_f and T_i represent the pneumatic torque, friction torque and interaction (i.e. external torque), respectively. I and α are the mass moment of inertia of rotating parts of the cylinder and the angular acceleration, respectively.

To derive the load dynamics of the cylinder, Newton’s second law is applied in rotary form as in Equation 3.1.

$$T_{net} = T_p - T_f + T_i = I \cdot \alpha \tag{3.1}$$

Where T_p is defined as a result of F_p defined as in Equation 3.2.

$$T_p = F_p \cdot r \tag{3.2}$$

F_p is related with the chamber pressures as in Equation 3.3.

$$F_p = (P_a - P_b) \cdot A_v \tag{3.3}$$

From Equations 3.2 and 3.3, the pneumatic torque takes the form of Equation 3.4:

$$T_p = (P_a - P_b) \cdot A_v \cdot r \tag{3.4}$$

Vane surface area is calculated as $A_v = H \cdot W$ and r is calculated with geometrical approach as $r = (D_2 + H)/2$. The correct calculation of A_v and r parameters is highly dependent on correct determination of D_2, H and W parameters. Except D_1, the other size measures are obtained with mechanical inspection by disassembling the cylinder and measuring sizes with a caliper in sensitive way as in Figure 3.6.

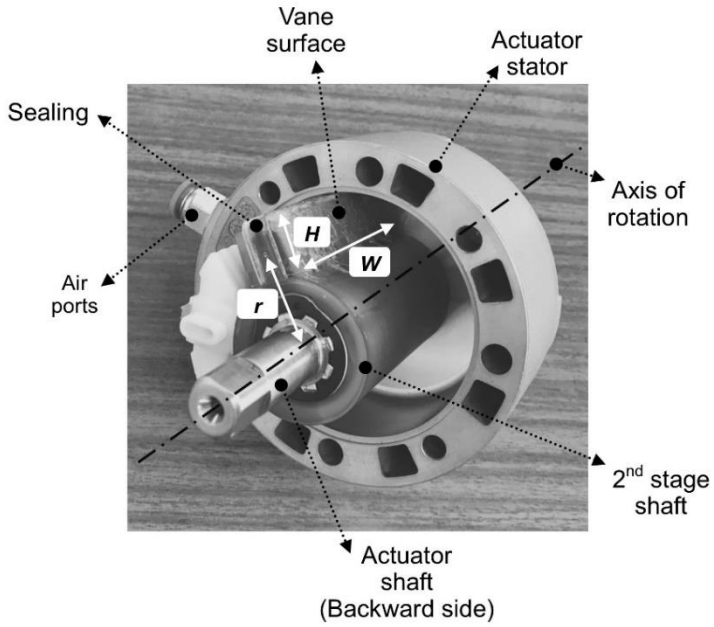

Figure 3.6. Disassembled FESTO DRVS-32-270-P

According to the inspection, D_2, H and W parameters are measured as 31 mm, 15 mm and 57 mm, respectively.

Dynamics of Double-Acting Linear Pneumatic Cylinder

The general structure of the preferred double acting pneumatic cylinder (FESTO DSBC-32-300-PPVA-N3) is shown in Figure 3.7. As indicated in Figure 3.7, sealing elements are placed around the piston and the piston rod to prevent pressurized air leakages. A_a, A_b, P_a and P_b represent the effective piston areas and the absolute air pressures in chambers "a" and "b", respectively. *Ø PD*, *Ø RD* and *L* show the piston diameter, piston rod diameter and the stroke length of the cylinder, respectively. The kinematic parameters of the moving piston i.e. linear position, speed and acceleration are symbolized by *x, v* and *a*, respectively. F_{ext} and F_f represent the external force applied to the piston rod and overall friction force resisting to the motion. Lastly, M_p is the total mass of the piston rod and any attached part (if exists).

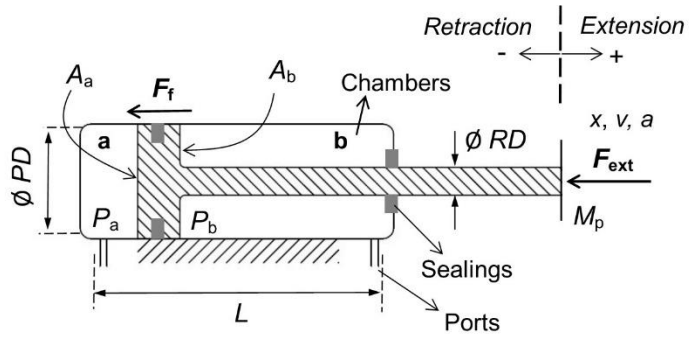

Figure 3.7. Inner structure and assigned parameters for linear cylinder (FESTO DSBC-32-300-PPVA-N3)

The equation of motion in pneumatic cylinders is expressed as in Equation 3.5 (Ohara et al., 2022), whereby the pneumatic force (F_p) is given in Equation 3.6:

$$F_{net} = F_p - F_f - F_{ext} = M_p \cdot a \tag{3.5}$$

$$F_p = (P_a \cdot A_a - P_b \cdot A_b - P_{atm} \cdot A_{rod}) \tag{3.6}$$

In the following sub-sections, the mathematical modelling of the control valves will be covered. Then, the friction model together with its identification methodology will be assessed.

3.2.2. Modelling of Proportional Pressure Regulating Valves

The proportional pressure regulating valves works in linear fashion as in Figure 3.8. That, means a direct and unique relation exists between the applied control signal (z) and output pressure (P_{out}). The valves have input control signal span of 0-10 V DC and output pressure range of 1-11 bar absolute. For example, application of z_a=1 V to PRV-a produces P_a= 2 bar absolute pressure for the chamber "a". Similarly, the application of z_b=1 V to PRV-b yields P_b=2 bar absolute for chamber "b". Based on this information, the mathematical relations between control signal, z and output pressure, P_{out} of each PRV are represented in Equations 3.7 and 3.8.

$$[P_a]^{out} = z_a + P_{atm} \text{ (bar)} \tag{3.7}$$

$$[P_b]^{out} = z_b + P_{atm} \text{ (bar)} \tag{3.8}$$

Equation 3.9 generalizes the output pressure of these controllers where j corresponds to the chambers "a" and "b", respectively.

$$[P_j]^{out} = z_j + P_{atm} \text{ (bar)} \quad j=a, b \tag{3.9}$$

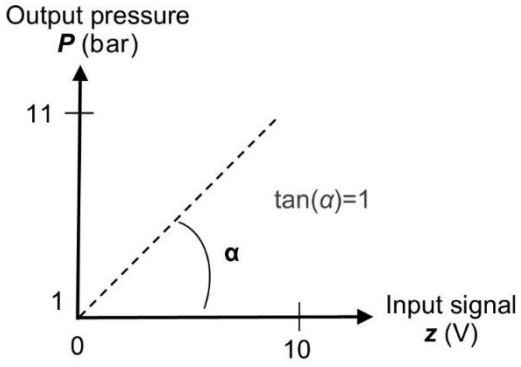

Figure 3.8. The relation between control signal and pressure output for FESTO VPPM-6L-L-1-G18-0L10H-V1P

3.2.3. Modelling of Proportional Directional Control Valves

Definition of New Model

The model of Bobrow and McDonell (1998) has been enhanced further by considering all of the aforementioned requirements in Section 2.3, and a novel efficient compressible flow model has been developed. The new flow model employs the correct mapping of cracking pressure ratio (CPR) at each operating condition, hence eliminating the mistakes existing in previous flow models. In addition, the new model includes a subsonic index (ε) that is depending on the control signal rather than a constant 0.5 value. It is discovered that the subsonic index (ε) value behaves non-linearly in relation to the valve control signal (i.e. $\varepsilon(u)$). Furthermore, the influence of supply pressure is taken into consideration using newly derived empirical coefficients. The model of Bobrow and McDonell (1998) underwent additional modifications to encompass all these influences. The updated flow model is introduced in Equation 3.10, showcasing the equation for the novel flow function as outlined in Equation 3.11 (Dağdelen et al., 2023a).

$$\dot{m}_p = \frac{\left[\frac{1}{c_p \cdot T_u}\right] \cdot \beta_p(P_r)}{k(P_s)} \tag{3.10}$$

$$\beta_p(P_r) = \left[\left|(P_u \cdot a(u, P_s) - P_d)\right|^{\varepsilon(u)} * (\alpha_1(u) \cdot u + \alpha_2(u) \cdot u^2)\right] \tag{3.11}$$

In the new flow model, a(u, P_s) corresponds to the CPR value dependent on the supply pressure and the valve control signal; $\varepsilon(u)$ defines the flow index which is a function of control signal; $\alpha_1(u)$ and $\alpha_2(u)$ are the model parameters identified at a specific supply pressure. $k(P_s)$ is the parameter of supply pressure effect. C_p corresponds to the specific heat coefficient at constant pressure and T_u corresponds to the temperature of the supplied pneumatic air. It should be noted that the identification of the correct CPR, model parameters and coefficients of supply pressure effect is critical to efficiently use the new model in control applications.

Figure 3.9 (a) depicts a comparison of the flow curves for the new and existing (i.e. standard and newly developed) models. The dashed line shows the new flow curve, which includes the correct CPR information, while the continuous line reflects the present flow curves (i.e. either standard or improved) with a subsonic index (ε) of 0.5. On the other hand, Figure 3.9 (b) depicts pressure change with respect to pressure ratio during air flow. When the existing models are applied to central orifice openings (i.e. small orifice openings), the flow is assumed to follow the pattern in the continuous line. As a result, despite the fact that the flow is stopped at point p_1 in reality by following the dashed line, existing models inaccurately imply that the flow follows the continuing line from point p_2 to the CPR value of 1 (a=1). In practice, however, if the flow is stopped at point p_1, the pressure ratio, P_r, will not change, and the CPR value will not approach to 1 (a=1). As a result, at the real flow termination point, p_1, these models yield a residual and constant

mass flow rate ($\dot{m}_r$). This problem leads to a significant discrepancy in pressure tracking performance, as depicted in Figure 3.9 (b), which has been confirmed through preliminary experiments. By using the new flow model, the pressure tracking performance is aimed to be enhanced as in dashed line in Figure 3.9 (b).

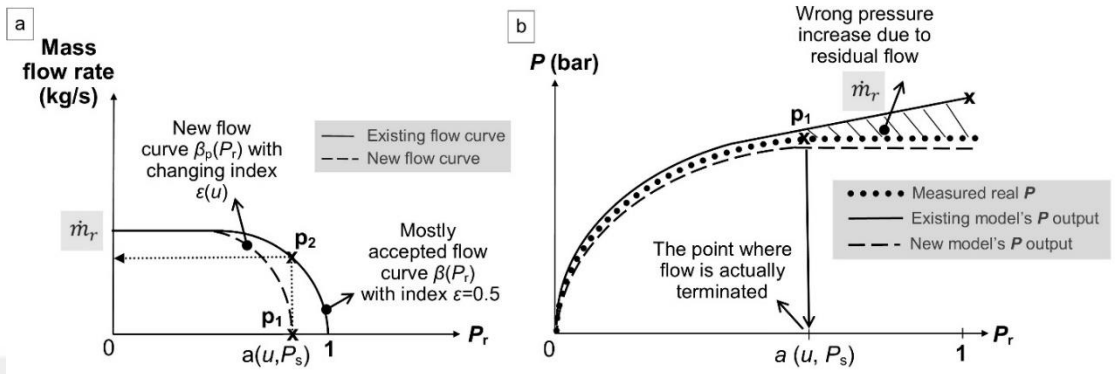

Figure 3.9. (a) Existing flow curves; (b) Pressure outputs of models

Determination and Mapping of CPR

The ISO 6358-1989 and ISO 6358-1:2013 standards are generally used to determine flow rate parameters in compressible flows. However, some researchers observed that the stages in these standards were time consuming and required expensive equipment such as a flow meter (Kawashima et al., 2004). As a result, the researchers tended to use a simpler and alternative method, in which the respective port/orifice of the valve to be modelled was connected to a fixed volume chamber (or the pneumatic cylinder itself could be used), and the pressure-time variation $P(t)$ of this fixed volume was investigated under various operating conditions (Valdiero et al., 2011; Kawashima et al., 2004; Kawashima et al., 2000). Taking these observations into account, an alternative methodology and setup will be employed to efficiently and easily investigate the effect of supply pressure (P_s) on flow rate characteristics. Figures 3.10 (a) and (b) illustrate the modified schematic and built setup for the identification of CPR.

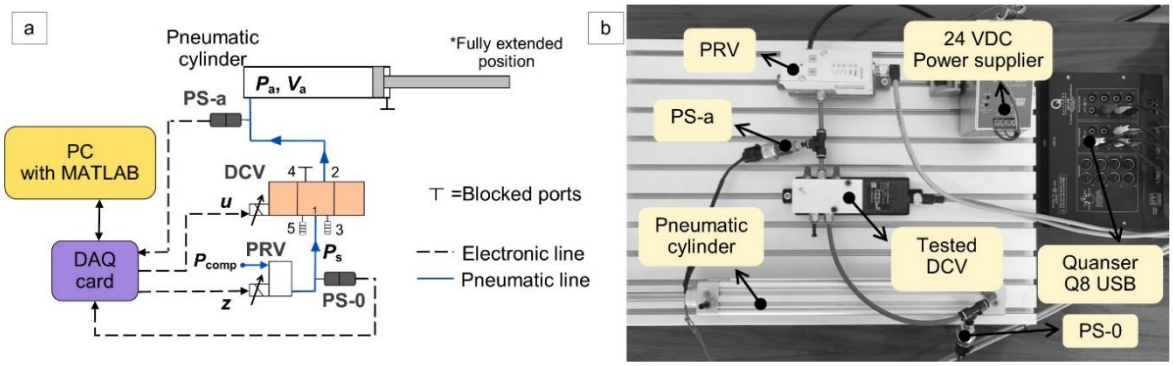

Figure 3.10. (a) Schematic illustration of setup; (b) Built setup for CPR identification

The built setup in Figure 3.10 consists of the QUANSER Q8 USB data acquisition card (DAQ), a host PC equipped with MATLAB/Simulink, a fully extended pneumatic cylinder to satisfy the constant volume air chamber (FESTO DSBC-32-300-PPVA-N3), the proportional directional control valve under examination (FESTO MPYE-5-1/8-HF-010-B), a pressure regulating valve (FESTO VPPM-6L-L-1-G18-0L10H-V1P), two pressure sensors (PS-0 and PS-a) to measure the supply pressure and chamber pressure, a power supplier (MeanWell NDR-240-24), and an air compressor serving as the pressure source (P_{comp}=8-10 bar). Furthermore, for real-time control and measurement procedures, the QUARC module is used as an interface between MATLAB/Simulink and the QUANSER Q8 USB. While the proportional directional control valve (DCV) and pressure regulating valve (PRV) are powered by a 24 V DC external power supplier, the pressure sensors (PS-0 and PS-a) are powered directly by utilized DAQ card. u signal controls the DCV to provide various orifice openings, while the z signal controls the PRV to ensure that various supply pressures (P_s) are given to the DCV's supply port.

Table 3.3 displays the catalogue information for the tested valve. The manufacturer's catalogue already includes the parameters b and C. The discharge coefficient (C_d) can also be calculated using the necessary formulae, as detailed by Rad et al. (2014). When using the standard or improved flow models, these values can be useful. However, because the effect of these parameters will be reflected by the model parameters (i.e. α_1, α_2 and ε), there will be no need to employ these catalogue values in the new model.

Table 3.3. Features and parameter values of FESTO MPYE-5-1/8-HF-010-B

Features and Parameters	**Value**
Nominal width	6 mm
Operating pressure	0-10 bar
Critical pressure ratio (b)	0.26
Sonic conductance (C)	3.1 l/s·bar
Nominal flow rate (Q)	700 l/min
Setpoint values (Control range)	0-10 VDC
DC supply voltage	17-30 VDC
Discharge flow coefficient (C_d)	0.5507

The flow rate characteristics changing according to the control signal (u) of the relevant valve ($\dot{m}$) was already depicted as in Figure 2.3. However, the real behavior of the valve is observed as in Figure 3.11.

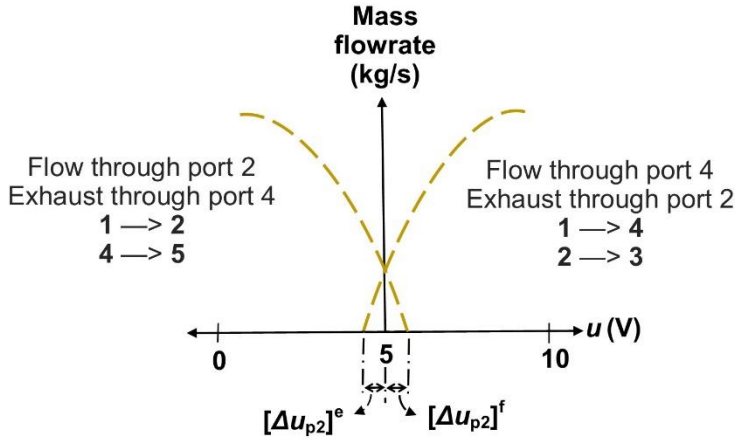

Figure 3.11. The relation between control signal and real flow behaviour for FESTO MPYE-5-1/8-HF-010-B

As a first observation, the valve has air leakages through port 2 and port 4 at a significant mass flow rate level ($\dot{m}_l$) due to the uncertainty characteristics ($[\Delta u_{p2}]^f$ and $[\Delta u_{p2}]^e$). When the measured data is evaluated, the valve continues to supply air to the chamber up to a certain signal value ($[\Delta u_{p2}]^f$) even when it should terminate the air flow through port 2 at $u \geq 5$ V. Similar to that problem, the air exhausting from the chamber continues until a certain control signal ($[\Delta u_{p2}]^e$) even though it should stop the air flow through port 2 at $u \leq 5$ V. From this critical point of view, the need for a precise model of valve around central position of it (i.e. small orifice openings) arises. The identification of real flow behavior requires intensive experimental work where the details will be discussed in the following sections.

The CPR (cracking pressure ratio) and the instability $[\Delta u_{p2}]^f$ during the filling state are assessed within the signal range of 0 to 6 volts (i.e. $0 \leq u < 6$ V), utilizing a specific signal increment (Δu). This range is established based on initial experimental trials conducted on the valve, as detailed in the preceding section. The parameters are examined through the following outlined procedures:

1. The designated supply pressure (P_s) for parameter examination is specified. To correspond with this specified supply pressure, a signal of $z = P_s$ V is transmitted to the PRV.
2. A signal of $u = u_n$ V is directed to the DCV, resulting in a specific orifice opening being established.
3. When the pressure in the cylinder chamber (P_a) achieves a constant value and the pressure change becomes almost zero ($\dot{P}_a = 0$), the maximum pressure ratio obtained (P_a/P_s) is calculated. The estimated P_a/P_s value is saved as the CPR value for the signal under investigation (u).

4. To investigate an alternate valve opening ($u_{n+1}=u_n+\Delta u$), the air within the chamber is entirely released into the atmosphere, setting the chamber pressure to atmospheric levels ($P_a= P_{atm}$=1 bar). This action prepares the cylinder for the subsequent experiment. Achieving this involves issuing a complete exhaust command to port 2 of the DCV by setting the signal to u=10 V.
5. After inspecting all valve openings at a specific supply pressure (P_s) in the range of u=0-6 V, a different supply pressure is chosen (($P_s)_{n+1}=P_s+\Delta P_s$)) and steps 2, 3, and 4 are repeated.

Utilizing the collected data, the $[\Delta u_{p2}]^f$ parameter can also be determined. As illustrated in Figure 3.11, the distinction between the control signal indicating actual zero airflow and the signal theoretically representing zero airflow (5 V) is identified as ($[\Delta u_{p2}]^f$) for port 2.

The CPR (a) and instability $[\Delta u_{p2}]^e$ in exhausting state is evaluated at $4\leq u<10$ V signal range with a certain signal step (Δu). The parameters are investigated by following the procedures:

1. The supply pressure (P_s) for which the parameters are to be estimated is determined. Air is supplied to the chamber until the air pressure in the chamber equals to the specified supply pressure ($P_a=P_s$). This is achieved with sending the control signal $z= P_s$ V to the PRV and fully opening the DCV to the chamber (u=0 V).
2. To set a certain valve opening, $u=u_n$ V signal is sent to the DCV. When the chamber pressure returns to a constant value and pressure change become nearly zero ($\dot{P}_a = 0$) during exhausting, pressure ratio (P_{atm}/P_a) is calculated. This value is recorded as the CPR for the tested signal (u).
3. To examine another valve opening, Step 1 is repeated. After the chamber pressure has been adjusted, the following control signal ($u_{n+1}=u_n+\Delta u$) is sent to the DCV. Step 2 is applied and the relevant data is retrieved again.
4. After the inspection of all valve openings in the range of $4\leq u<10$ V at specified particular supply pressure (P_s), another supply pressure is selected (($P_s)_{n+1}=P_s+\Delta P_s$)) and steps 1, 2 and 3 are repeated.

3.2.4. Identification of Friction

The friction in mechanical elements is an inherent property due to the irregularities existing on the surface on the materials. The friction force in resistive form acts opposite to the movement direction. The friction can be evaluated in pre-sliding and pure-sliding region in general sense. The identification of friction forces/torques requires intensive experimental analysis, since this kind of forces cannot be measured or estimated by utilizing any kind of sensor technology. Furthermore, identification procedures show slight differences with respect to the preferred model.

In literature there exist many friction models such as Coulomb-Viscous, combined Stribeck, LuGre and Modified LuGre, etc. Among these models, LuGre and modified LuGre models have more comprehensive structure since they can even account for pre-sliding friction behavior. However, the identification of LuGre model parameters necessitates excessive experimental effort and this is not an easy task. In addition to that, since the proposed robot is evaluated under motion (i.e. pure sliding conditions) the utilization of LuGre model is not intended in this study for both actuators (i.e. vane type semi-rotary and double-acting linear pneumatic cylinders). Another important reason supporting not preferring the LuGre model or its variations is that the Stribeck model can estimate precisely the pure sliding friction and, can be identified in more easy experimental procedure.

Stribeck friction is mostly a function of operating speed and can be represented as in Figures 3.12 (a) and (b) with the mathematical expressions as in Equations 3.12 and 3.13 for rotary and linear cylinders, respectively. In dead, only difference between these equations is notation of model parameters. In more broad sense, in vane type semi-rotary actuator, since rotational motion is observed, torque (T) and angular speed (ω) are utilized whereas in linear cylinder force (F) and linear speed (v) are opted for the definition of equations. Bearing on this information, Stribeck model includes five parameters as static friction torque/force (T_s/F_s), coulomb friction torque/force (T_c/F_c), Stribeck speed (w_s/v_s), Stribeck shape factor (j, common in both equations), and viscous friction coefficient (B, common in both equations). Operating angular and linear speed (w and v) are the independent parameters in models presented in Equations 3.12 and 3.13, respectively.

$$T_f = T_c + (T_s - T_c) \cdot e^{-\left(\frac{w}{w_s}\right)^j} + B \cdot w \tag{3.12}$$

$$F_f = F_c + (F_s - F_c) \cdot e^{-\left(\frac{v}{v_s}\right)^j} + B \cdot v \tag{3.13}$$

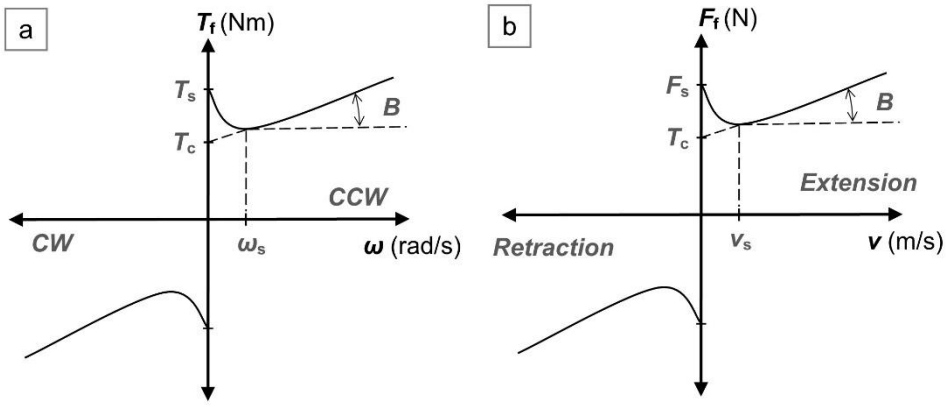

Figure 3.12. General representation of Stribeck friction model for (a) vane type semi-rotary cylinder; (b) double acting pneumatic cylinder

However, the friction behavior in pneumatic cylinders can change with respect to the operating (i.e. working) pressure due to the non-linear contact attitude of the sealing (Nouri, 2004). Therefore, in this thesis study, modified version of Stribeck function has been utilized for vane type semi-rotary cylinder as in Equation 3.14, since the forearm pronation/supinations will be studied. However, the friction force analysis of linear pneumatic cylinder has been conducted on a single specified working pressure, namely in depth evaluation under changing operating conditions has not been considered. Based on this modified model, all four parameters are functions of working pressure, P_w. Only the Stribeck factor (j), can be assumed to have a value between 0.5 and 2 (Yue et al., 2017) which is considered to be independent of P_w.

$$T_f\,(P_w, w) = T_c(P_w) + [T_s(P_w) - T_c(P_w)] \cdot e^{\left[-\frac{w}{w_s(P_w)}\right]^j} + B(P_w) \cdot w \tag{3.14}$$

Friction identification of Vane Type Semi-Rotary Cylinder

From the friction torque identification of semi-rotary cylinder point of view, it is necessary to estimate the parameters of T_c, T_s, w_s, and B by experimental techniques under changing operating speed (w) and working pressure (P_w). To identify the Stribeck model parameters, the following general experimental procedures are employed:

Step 1: In order to obtain static friction torque (T_s), the cylinder is kept (i.e. not moving) at rest at a counter working pressure applied to the one of the chambers based on the selected movement direction. Then, while cylinder is at rest (i.e. not moving) a ramping pressure is supplied to the cylinder chamber to obtain the desired movement direction until the first motion is noticed. At the instant of movement initialization, the resultant net friction torque (T_p) based on Equation 3.4 will be equal to the static friction torque.

Step 2: Different working pressure levels are supplied to the cylinder chambers to allow the cylinder to rotate at constant angular speeds in each experiment session. During the movement of the cylinder, net torque is computed using Equation 3.4 similar to Step 1. Simultaneously, the angular speed of the cylinder is monitored and recorded for the subsequent steps.

Step 3: The data gathered from previous two steps (i.e. Steps 1 and 2) are further processed to pick out the useful torque (T_f) and speed (w) data.

Step 4: A non-linear curve fitting method is applied to the data obtained from previous step (i.e. Step 3) to identify friction parameters.

However, in literature, there is no sufficient information about the friction characterization of vane type semi-rotary pneumatic cylinders. This shortage prevents the utilization of vane type

cylinders in precise control applications. In addition to this shortcoming, the aforementioned steps are utilized manually in most of the pneumatic friction identification studies. The following are other factors that highlight how serious and challenging the issue is:

- A variety of experimental procedures, including cylinder motion control, data collection, signal processing, data classification, and non-linear parameter identification, are involved in the identification of friction parameters. As a result, the probability to be faced with user or systematic errors is quite high in manual experimental applications. These errors may have negative impacts on generating the necessary and correct controller outputs.
- Because of the processing time and workload associated with collecting and analysing experimental data, the studies that are already in place in literature have only processed a small amount of data. It is important to remember that the amount of processed experimental data has a significant impact on how accurate the predicted friction parameters are.

Based on these observations in manual experimental methodology, a mechatronic setup has been constructed. All tests have been automated using MATLAB/Stateflow® tools (Dağdelen and Sarıgeçili, 2023a). In this manner following contributions are provided as following:

- The estimation time (i.e., experiment duration) and possible user errors have been greatly minimized. The operator's workload has been effectively reduced.
- The vane type cylinders can be characterized from the perspective of friction torque characteristics across a wide range of operating pressures and speeds with the aid of automation flowcharts.
- Successful automation has allowed for the accurate processing of a large amount of data. As a result, more accurate results have been obtained.
- To facilitate the usage of automation algorithm, an easy to use graphical user interface (GUI) has been designed which takes the necessary information about the cylinder and test conditions and then outputs the identified friction parameters with just one click of a button.

Figure 3.13 illustrates the mechatronic setup of the friction torque automation. Four primary control blocks (CBs)- CB-I, CB-II, CB-III, and CB-IV are employed in this schema to manage each of the required experimental stages. The remaining blocks (i.e. net torque calculation block, data classification block) in the schema perform switching and classification activities in addition to the essential computations.

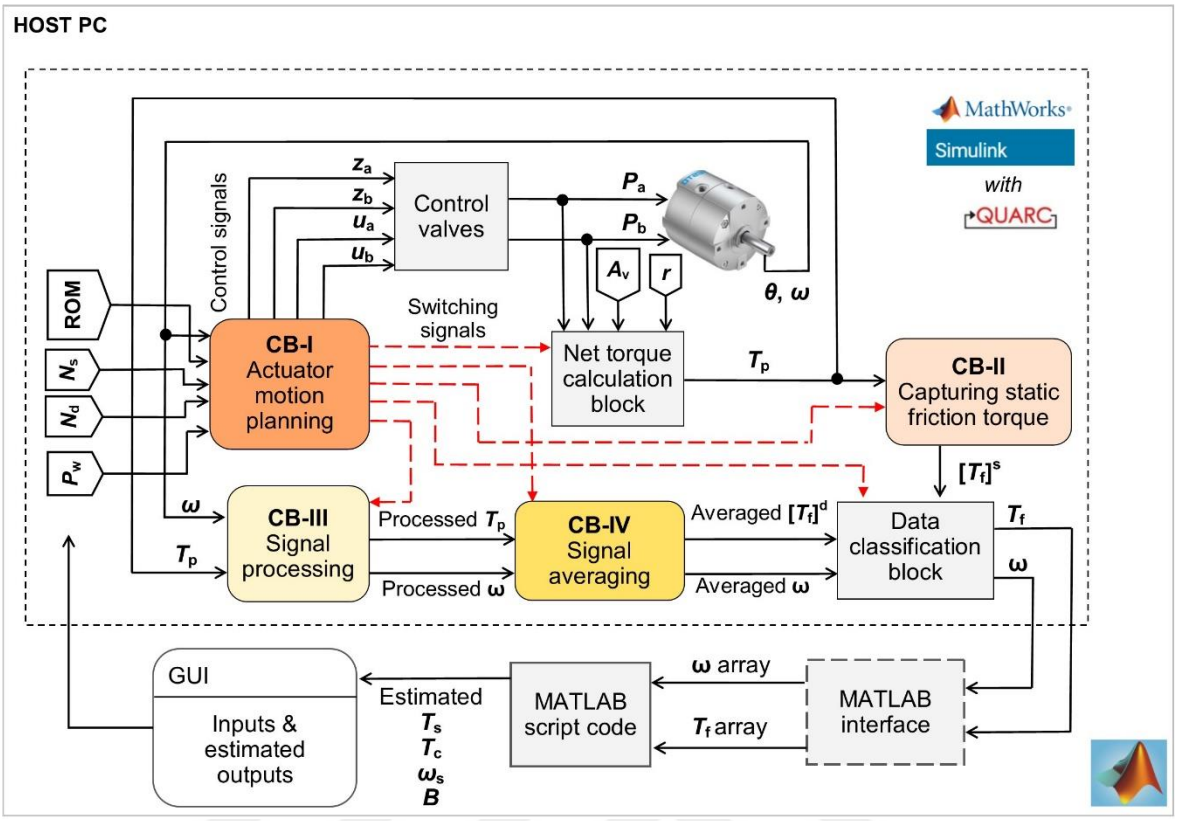

Figure 3.13. Control and measurement schema for friction identification vane type semi-rotary cylinder

The primary block, known as control block-I (CB-I), is responsible for managing the cylinder's motion using the control signals (u_a, u_b, z_a, z_b) in accordance with the input parameters (P_w, N_s, N_d, and ROM). P_w defines the working pressure to be applied to the cylinder's chambers. N_s and N_d are the number of static and dynamic experiments to be conducted. ROM is the range of motion of the rotary cylinder in angular form. The values of P_w, N_s and N_d are user defined, whereas ROM value is specified in the manufacturer's catalogue. Generated control signals by CB-I are sent to the control valves (i.e. both PRVs and DCVs), and in result suitable pressure levels are adjusted and the pneumatic air is correctly directed to the cylinder's chambers. In addition to that, switching signals whose value either 0 or 1 (i.e. shown in red dashed line in Figure 3.13), are output from CB-I and send to the other blocks (i.e. net torque calculation block, CB-II, CB-III, CB-IV and data classification block) for enabling or disassembling the operators embedded in the respective blocks.

The raw pneumatic torque (T_p) is evaluated by the control block-II (CB-II) in order to obtain the static friction torque (T_s). Equation 3.4 is applied in the "Net torque calculation block" to determine T_p in relation to the input data (A_v and r) and the chamber pressures (P_a and P_b). P_a and P_b are the absolute chamber pressures fed back from pressure sensor PS-a and PS-b. A_v and r define the vane surface area and torque radius whose specifications were already given in Section 3.2.1.

The mission of the control block-III (CB-III) is to process pneumatic torque (T_p) and angular speed (w) data. The processed T_p and w data is then fed to the control block IV (CB-IV) to generate mean and single values of each T_p and w data for data classification block. Classified data in data classification block are then saved in array format in MATLAB interface for the further processing. The graphical user interface (GUI) receives the predicted friction torque parameters from the script code wherein a non-linear curve fitting operation is executed on the w and T_f arrays.

According to the static friction torque identification methodology, a ramping pressure is applied to one of the chambers during the static experiments while the actuator is at rest. Namely, the ramping pressure applied to chamber "a" for the CCW direction inspections, whereas it is applied to chamber "b" for CW direction. During the test application, at the moment of movement initialization, the maximum pneumatic torque is recorded, and this torque information (T_p) is designated as static friction torque (T_s). For this manner, flow charts for the static experiment protocols in the CCW and CW directions are presented in Figures 3.14 (a) and (b), respectively.

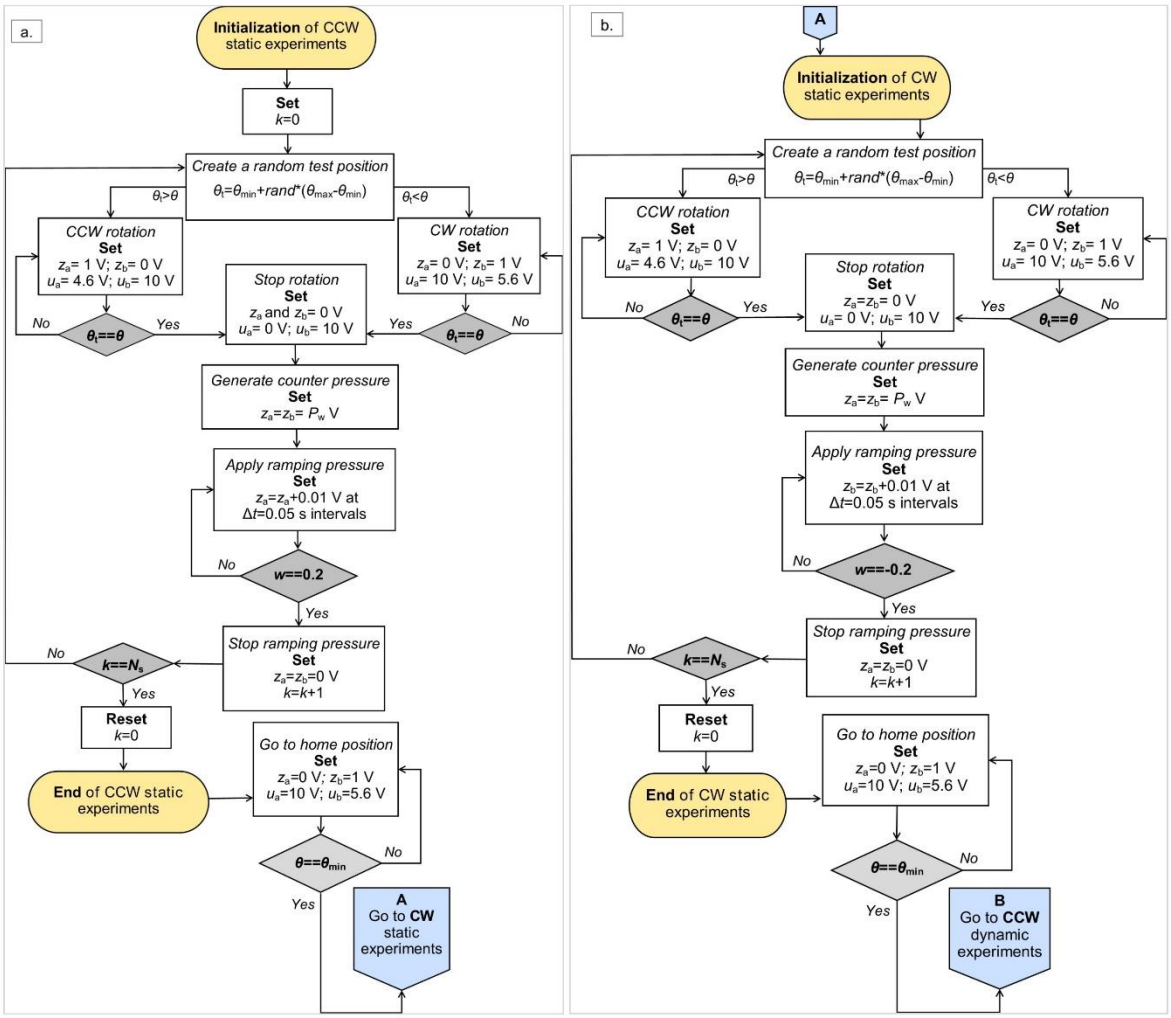

Figure 3.14. Static friction identification algorithm for (a) CCW; (b) CW directions

To understand the logic and relation between the control signals and valve's behaviour in the automation flowcharts, the following short explanations are presented:

To avoid unexpected torque levels at the end points due to the lack of speed control, the actuator is tested between 20% and 80% of the maximum ROM, or θ_{min} to θ_{max}. The definition of a counter variable, k and its assignment of zero value is the first step in the static friction torque identification experiments (Figure 3.14 (a)). The following formula creates a random angular test point (θ_t) for each test, between the positional limits θ_{min} and θ_{max} as in Equation 3.15:

$$\theta_t = \theta_{min} + rand()(\theta_{max} - \theta_{min}) \tag{3.15}$$

With the help of *rand*() function, a fractional number is generated in each test application bearing on the logic in Figures 3.14 (a) and (b). Based on this generated random number, a test position, θ_t is generated for the comparison purposes in the flowchart. The cylinder is rotated either in CCW or CW direction by comparing the test position, θ_t with its current position, θ by sending the suitable control signals (u_a, u_b, z_a and z_b) to the PRVs and DCVs. Once the cylinder reaches the specified test position, the PRV-a and PRV-b are arranged to generate atmospheric pressure which is 1 bar absolute (i.e., z_a=z_b=0 V) to terminate its motion. Next to this, by setting z_a=z_b=P_w V, the same designated working pressure of P_w is applied simultaneously to both chambers. After then, the pressure in counter chamber "b" is maintained at the designated constant working pressure, P_w (z_b=P_w V), while the z_a signal value is increased by 0.01 V at 0.05 s time intervals (z_a=z_a+0.01 V). During the application of ramping pressure, the angular speed of the cylinder vane (ω) is measured and when it just passes the threshold value of 0.2 rad/s (i.e. the condition of $\omega \geq 0.2$ rad/s is satisfied), it is accepted that the first motion is captured by the algorithm. At this point, the ramping signal application is terminated by shutting off the both PRVs by sending the signals of z_a=z_b=0 V.

After increasing the value of k by 1, it is compared to the user-defined number of static experiments (N_s) value. The same processes are continued until the parameter k equals the value N_s. After the equivalence is confirmed, the CCW static experiments are concluded and k is reset to zero value. Next, the experiments for CW direction shown in Figure 3.14 (b) are carried out. The CW and the CCW static experiments are similar, yet there are stages in which the valve arrangement is altered. The process of creating test points has similarities in that sense. On the contrary, the ramping pressure is supplied to chamber "b" by raising 0.01 V at 0.05 s time intervals of the z_b signal (z_b=z_b+0.01 V) until ω just passes -0.2 rad/s threshold speed while the counter chamber, or chamber "a", is maintained at the prescribed constant P_w value (z_a=P_w V). At that point, ramping pressure application is halted and collected pneumatic torque (T_p) is allocated to static friction torque (T_s) for the CW direction. In a similar manner, N_s is compared to the value of counter parameter (k), which has been increased by 1. Until k equals N_s, the same procedures are followed.

The system switches from static to dynamic experiments when they are equal (i.e. the condition of $k=N_s$ is satisfied).

In dynamic studies, the cylinder is moved at different constant speeds throughout the experiment session to calculate the corresponding pneumatic torque. The dynamic experiments' process steps are displayed in Figures 3.15 (a) and (b). To avoid unexpected collisions at high angular speeds, the cylinder is tested between the positional limitations of θ_{min} and θ_{max}, much like in static testing.

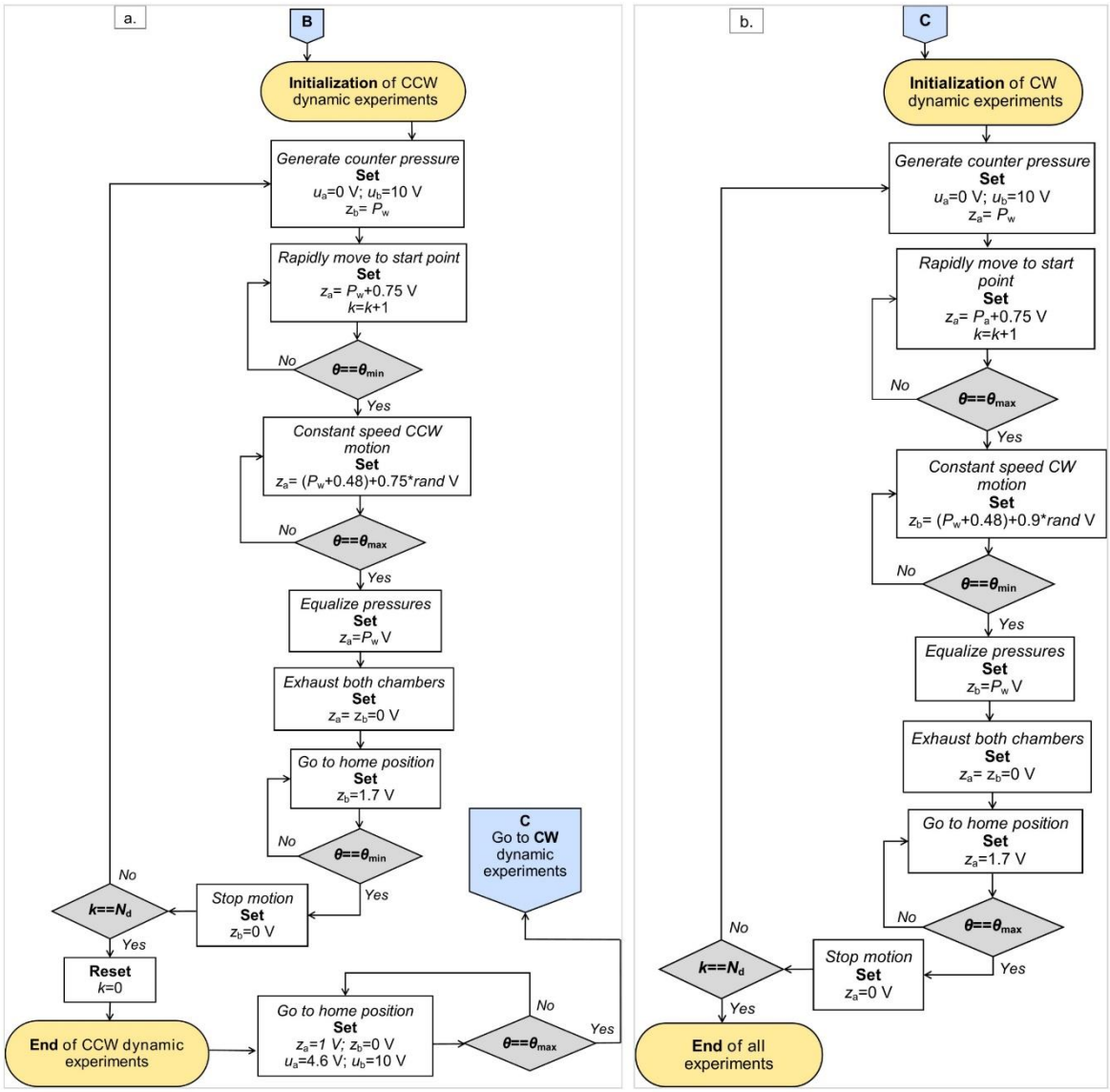

Figure 3.15. Dynamic friction identification algorithm for (a) CCW; (b) CW directions

The first step in the CCW dynamic tests (i.e. the point B in the flowchart) is to charge chamber "b" with a counter pressure (P_w). The starting position for the CCW dynamic tests is designated as θ_{min}, which is the cylinder position. When the vane position, θ reaches θ_{min}, the

control signal, z_a is set as per Equation 3.16, causing the pressure level in chamber "a" to be randomly produced pressure for each test to move the cylinder:

$$z_a = (P_w + 0.48) + 0.75rand() \text{ V} \quad (3.16)$$

The cylinder is enabled to conduct constant speed CCW rotation up to specified θ_{max} as per different signal configurations assigned by Equation 3.16. When the cylinder's vane reaches this position, chamber "a" pressure is set to P_w initially (i.e. z_a=P_w V). After a certain amount of time, z_a=z_b=0 V is set to equalize both chambers' pressures to atmospheric pressure. Subsequently, the cylinder returns to its initial position, θ_{min}, to carry out an additional dynamic test by rotating at a specific speed in the CW direction. The counter variable, *k*, is compared with the user-defined dynamic test number (N_d) at its home position. Until they are equal, the same procedures are repeated. *k* value is reset to zero and the CCW direction dynamic experiments come to an end when equality is reached. The cylinder is then moved to the θ_{max} position in order to carry out the CW dynamic experiments with respect to the signal arrangements specified in the flowchart as in Figure 3.15 (a) (i.e. the point C in the flowchart). The CW direction dynamic experiments are carried out as the last phase of the friction identification experiments as per the algorithm presented in Figure 3.15 (b). By setting z_a=P_w V, a counter pressure is established in chamber "a", contrarily to CCW direction experiment's procedures. The cylinder is rapidly rotated to θ_{max}. The control signal of pressure level in chamber "b" is then set as in Equation 3.17 to create different constant speed angular motions from θ_{max} to θ_{min} at different tests:

$$z_b = (P_w + 0.48) + 0.9rand() \text{ V} \quad (3.17)$$

The pressure in both chambers is first adjusted to the working pressure (P_w) and subsequently to the ambient pressure when the cylinder reaches the θ_{min} position. To perform a second dynamic test, the cylinder is sent back to the θ_{max} position. The experiment stages are then repeated after comparing the counter variable's value, *k* to N_d. All of the experiments are completed when this equality is satisfied.

As an important note, the constant parameters of Equations 3.16 and 3.17 (i.e. 0.48 & 0.75 in Equation 3.16, and 0.48 & 0.9 in Equation 3.17) have been determined through experimental work, or the "trial and error" method. The vane is demanded to be rotated in the CCW and CW directions at a minimum of +10 rad/s and a maximum of -10 rad/s during identification, respectively. Since the friction torque acts linearly after the Stribeck speed, it is not rational and necessary to exceed these speed restrictions.

Different processing processes are needed for the signals obtained from sensors during both static and dynamic experiments. It is necessary to capture the greatest pneumatic torque at the

beginning of rotation motion in a static test. Hence, Figure 3.16 (a) illustrates a ramping pneumatic torque (T_p) example based on sensory data computed using Equation 3.4 as a result of ramp pressure as seen in a static test in the CCW direction. As can be seen in Figure 3.16 (b), the ramping torque is applied until the control system detects an angular speed (ω) of 0.2 rad/s. At that point, the maximum torque value (max T_p) is identified and saved as the static friction torque ($[T_s]_n$, $[T_s]_{n+1}$,...) for each individual test. The static friction torque (T_s) parameter is determined by averaging the static friction torque values that have been gathered ($[T_s]_n$, $[T_s]_{n+1}$,...) at the conclusion of the static testing.

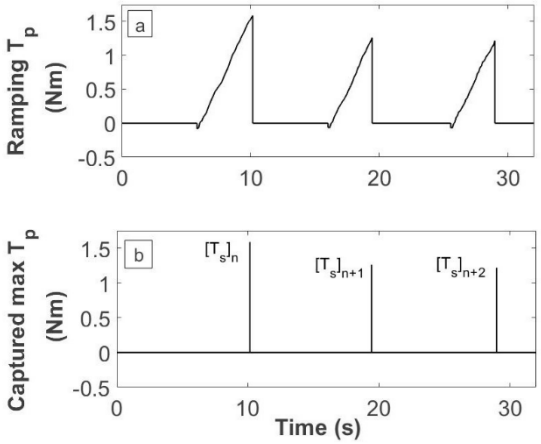

Figure 3.16. (a) Ramping signal application; (b) Static friction torque (T_s) measurement

In dynamic tests, the pneumatic torque together with constant angular speed data are needed to be detected. Thus, Figures 3.17 (a) and (c), respectively, illustrate sample pneumatic torque (T_p) and angular speed (ω) signals obtained with respect to time for a single test. According to Figure 3.17 (a), a dynamic test consists of three parts: (*i*) applying pressure to achieve motion at a constant speed; (*ii*) returning to the starting position for the next dynamic test; and (*iii*) setting the counterworking pressure (P_w). The first section's results provide the only pertinent information for estimating friction torque at constant speed (*i*). However, during a dynamic test, the speed and torque signals are both erratic, therefore signal processing is essential to obtaining meaningful data. Examining the variation in the angular speed (ω) and pneumatic torque (T_p) in section (*i*) as presented in Figure 3.18 reveals that they are not constant across the whole range of motion. In order to leverage nearly constant speed and constant torque portions for additional processing, the signals fetched during the entire ROM must be separated into two sections: unsteady and steady sections. When taking into account the CCW rotation from θ_{min} to θ_{max} in a dynamic test, the signals fetched for more than 60% of the ROM value are classified as steady sections. Conversely, in a dynamic test, the signals fetched for less than 40% of the ROM value are defined as the steady section for CW rotation from θ_{max} to θ_{min}. Next, for each rotation direction of a specific test, the signals in the unsteady regions are trimmed. The uncropped signals seen in Figures 3.17 (b) and (d)

are utilized by averaging the recorded values throughout their respective time domains to compute a single constant angular speed (ω) and a single friction torque (T_f).

Other three parameters, T_c, ω_s, and B, can be estimated using a non-linear curve fitting operation applied to the friction torque ($[T_f]_n$, $[T_f]_{n+1}$, $[T_f]_{n+2}$…) and constant speed data ($[\omega]_n$, $[\omega]_{n+1}$, $[\omega]_{n+2}$…) pairs. Since the quality of the estimation depends on the quantity of processed data and the quality of the curve fitting, the non-linear curve fitting is the most important step in the parameter estimation process. For this purpose, the least squares non-linear curve fitting (*lsqcurvefit*) is applied and integrated into the graphical user interface (GUI) with script code (see Appendix 1). More accurate results are obtained in these kinds of fitting operations with correct initial value assumptions of parameters and their constraints. Determining the accurate initial assumptions is not a straightforward process. The iterative experimental technique is one method to determine accurate initial guess values. That method involves testing the actual system with different initial predictions. When the estimated results remain constant, it indicates that the initial guess values were accurate. The same methodology is used in the present study, and the initial values for the T_c, ω_s, and B parameters are specified as [1, 1, 2] and [-1, -1, 2] for the CCW and CW direction, respectively. In addition, Table 3.4 lists the upper and lower parameter limitations of three parameters that are set to prevent erroneous findings.

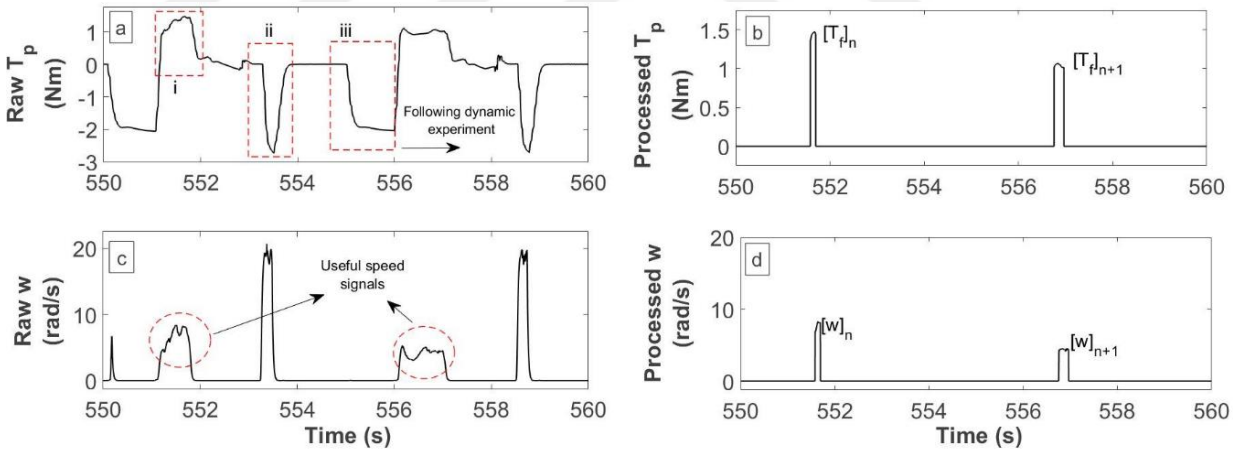

Figure 3.17. Torque and speed signal measurements and processing (a) Raw pneumatic torque (Raw T_p); (b) Processed Pneumatic torque (Processed T_p); (c) Raw operating speed (Raw ω); (d) Processed operating speed (Processed ω)

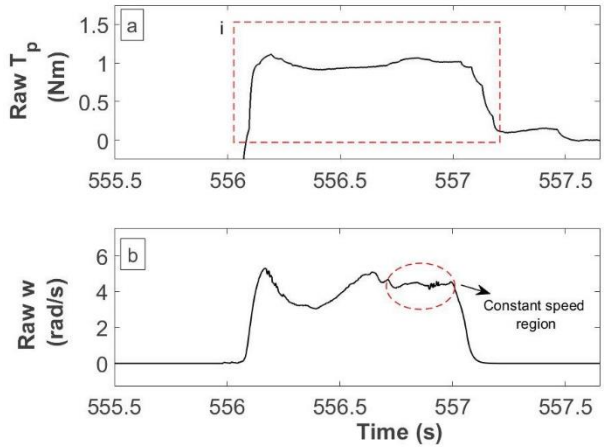

Figure 3.18. Detailed signals in a dynamic experiment (a) T_f; (b) ω

Table 3.4. Defined parameter bounds

	CCW		**CW**	
Parameters	**Upper Bounds**	**Lower Bounds**	**Upper Bounds**	**Lower Bounds**
$\boldsymbol{T_c}$	T_s	0	0	T_s
$\boldsymbol{w_s}$	1	0	0	-1
$\boldsymbol{B}$	2	0	2	0

To fully automatically acquire the friction torque characteristics of a rotary pneumatic actuator, an easy-to-use and effective graphical user interface (GUI) is required. Because of this, a GUI in MATLAB/AppDesigner® environment has been built and is shown in Figure 3.19. The sections of "input parameters", "experiment control panel", "obtained friction torque-speed curve", and "estimated parameter results" make up the four primary elements of the developed GUI. Actuator size information and experiment parameters are entered by the user in the "input parameters" section (*top left section*). The actuator size characteristics are entered as torque radius (r), vane surface area (A_v), and range of motion (ROM). The experiment parameters are the absolute working pressure (P_w), the number of dynamic experiments (N_d), and the number of static experiments (N_s). The "experiment control panel" (*bottom left section*) allows the user to control the experimental operations once the cylinder to be tested has been installed on the setup. The estimated parameters can be obtained by clicking the "click to ESTIMATE" button once the experiments are completed. The final results are then shown separately for the CCW and CW directions (*upper right section*) in "estimated parameter results" section. In "obtained friction torque-speed curve" section (*bottom right section*), the friction torque-speed curve is finally automatically drawn in relation to the identified friction parameters.

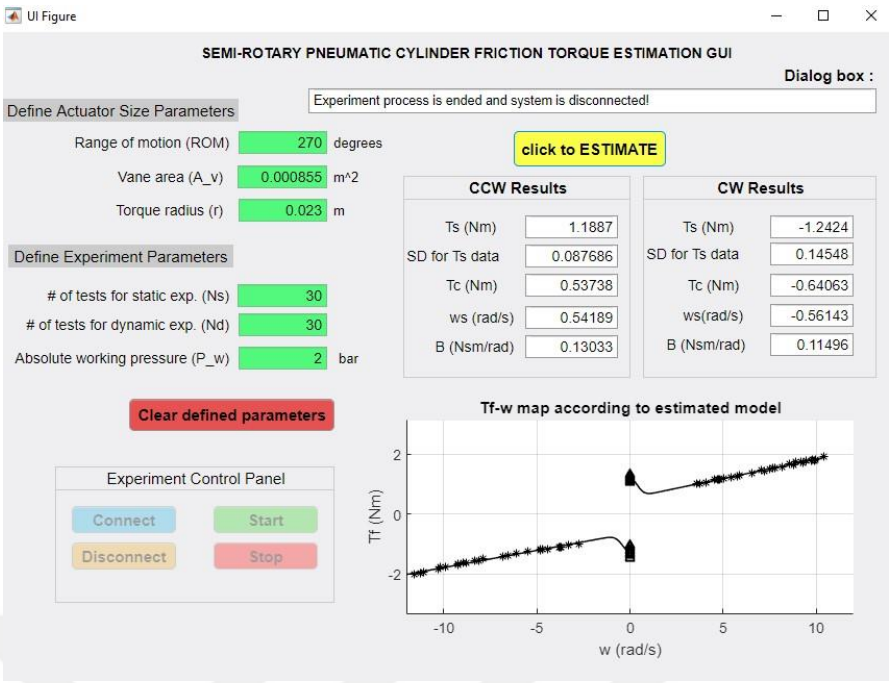

Figure 3.19. Designed graphical user interface

For the tests to be conducted correctly, the GUI has established some guidelines and restrictions to degrade any possible bugs. These are listed as follows:

• The input boxes do not allow the user to enter zero or negative values.

• The value of the range of motion (ROM) ought to be between 0 and 360 degrees.

• The input boxes need to be completed in the right sequence. The following input box cannot be entered if one input box has not received the right value format. When a valid value is submitted, the input box's colour goes back to green.

• For a more thorough statistical analysis of the parameters, there must be a minimum of 30 tests for both the static and dynamic experiments (N_s and N_d). Therefore, the values less than 30 are not allowed to be entered to the respective input boxes of N_s and N_d.

• The "Experiment Control Panel" is not activated until all required parameters are entered.

• Upon the successful completion of the tests, the "click to ESTIMATE" button becomes active.

• The user clears all input boxes by clicking the "Clear defined parameters" button to begin a new experiment.

Friction identification of Double Acting Linear Pneumatic Cylinder

The identification procedure of friction force in linear pneumatic cylinder is similar to the friction torque identification algorithm mentioned in the previous section, yet there are some differences at following issues:

- The utilized experimental setup
- The valve configurations (i.e. the control signal and automation steps)
- The measured and manipulated variables
- The designed graphical user interface (GUI) structure
- Embedded script code (see Appendix 2)

The experimental setup built for the automated system is shown schematically in Figure 3.20. It consists of same instruments (i.e. presented in Tables 3.1 and 3.2) existing in the robot's modified version with a different mechatronic configuration. Namely, it consists of the data acquisition and control card, the linear incremental encoder, pneumatic cylinder to be tested, two pressure sensors, one proportional pressure regulator valve, two 5/3 proportional directional control valves, and 5 VDC & 24 VDC power suppliers. In Figure 3.20, the dashed and continuous lines represent electronic signals and the pneumatic air, respectively.

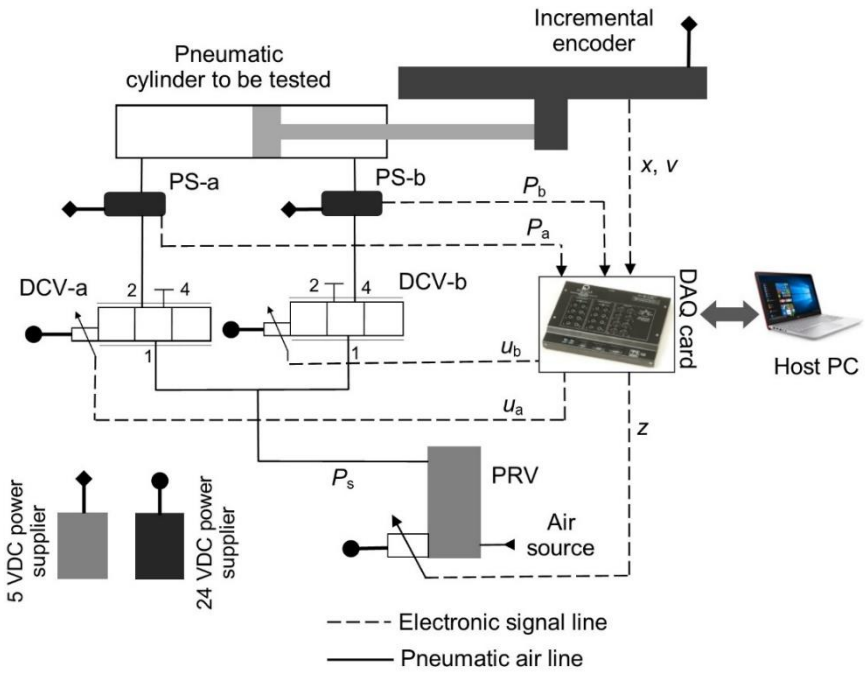

Figure 3.20. Experimental setup for friction force identification of linear pneumatic cylinder

The general control scheme of the system is given in Figure 3.21 (Dağdelen and Sarıgeçili, 2023b). "Encoder reading block" produces the linear position obtained from the linear encoder. Linear speed of the cylinder is derived by taking the first derivative of the linear position, x. The speed signal is further filtered by transfer function method to eliminate any noise and in result, the "Raw v" data is produced. "Pressure reading blocks" generates the measured pressure readings (P_a, P_b) from the sensors PS-a and PS-b. "Friction force calculation block" utilizes the user defined inputs piston diameter (PD) and rod diameter (RD) of the cylinder to calculate the raw friction force (raw F_f) with respect to the Equations 3.5 and 3.6.

Three Stateflow charts (i.e. Stateflow chart-I, -II and -III) and a data classifier block have been constructed in the control scheme (Figure 3.21). The Stateflow chart-I (named as Cylinder motion planning) is responsible for automating the steps of the static and dynamic experiments. It consists of blocks and functions which control the movement of the cylinder that will carry out the experiments sequentially in the order of (1) static extension, (2) static retraction, (3) dynamic extension and (4) dynamic retraction. Linear position (x) and raw linear speed (Raw v), cylinder stroke length (L), number of static friction experiments (N_s) and number of dynamic friction experiments (N_d) are fed as an input to the Stateflow chart-I. This chart generates the switching signals by the constructed events and transition algorithms inside it. The raw linear speed (Raw v), the raw friction force (Raw F_f), as well as the switching signals from the Stateflow chart-I are sent simultaneously as input to the Stateflow chart-II (i.e. Signal processing) for performing the signal processing tasks to draw the meaningful data from the complete set of raw data. As a result, the processed friction force (Processed F_f) and linear speed (Processed v) signals are generated as an output from the Stateflow chart-II and fed as an input into the Stateflow chart-III (i.e. Signal averaging and value extraction). The averaged signals are fed to the last block that is called "Data classifier". In this chart, the average values (Averaged F_f and Averaged v) from the processed force and speed signals are calculated and recorded at the end of each experiment based on the extension or retraction experiments. In this way, the obtained data has been classified with respect to extension–retraction direction as well as static–dynamic experiment data in "Data classifier" block. As a result, the gathered results are saved in MATLAB workspace in the form of vector arrays for further calculations, respectively as: static friction force results for extension direction ($[F_s]^{ext}$); static friction force results for retraction direction ($[F_s]^{ret}$); dynamic force results for extension ($[F_d]^{ext}$); dynamic force results for retraction ($[F_d]^{ret}$); dynamic speed results for extension ($[S]^{ext}$); dynamic speed results for retraction ($[S]^{ret}$).

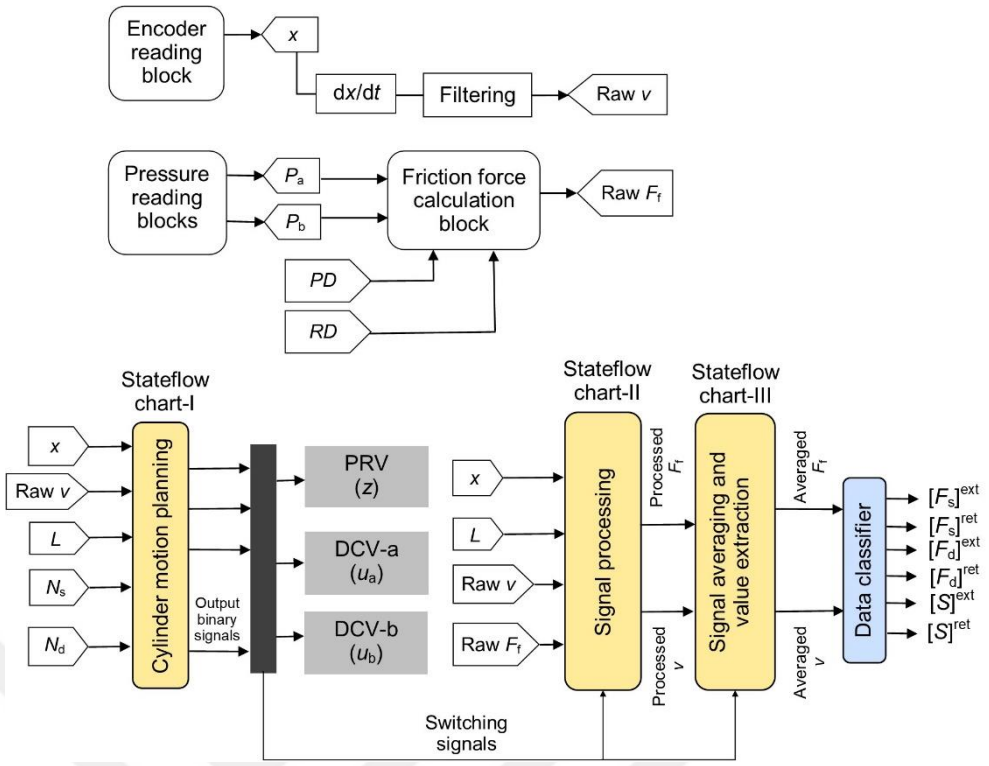

Figure 3.21. MATLAB/Simulink block diagram for friction force automation of double acting linear pneumatic cylinder

To estimate the maximum static friction force value, the system is designed to catch the friction force at the onset of the piston motion. The algorithm of the static friction experiment for extension direction is given as a flowchart in Figure 3.22. A counter variable (k) is defined as the first step. This variable increases by one at the beginning of each static experiment. The static experiments are carried out between 10% and 90% of the cylinder stroke length (L) to avoid spurious results due to the adhesion and cushioning effects at the end points in pneumatic cylinders. Therefore, the test position (p) for each static friction experiment is assigned randomly in the interval of $0.1L \leq x \leq 0.9L$ by the random number generator function (*rand*) in MATLAB. The current linear position (x) of the piston is compared with the generated test position (p). Then, the piston is moved automatically to the next test position accordingly. Simple on-off position control is applied at this step, because the main purpose here is not to achieve precise position control. When the piston comes to the test position with an acceptable position error, the pressure inside chamber "a" is increased by sending ramping pressure signal z to the PRV (i.e. with 0 V initial value and slope of 0.02 V/s). This signal yields the pressure increase rate of 0.02 bar/s with quite well linear relationship. For an extension experiment, the pressurized air is sent to chamber "a" while the air in opposite chamber "b" is fully exhausted to the atmosphere. The pressurized air is fed until the first initiation of motion is captured by the system.

The correct detection of motion initiation to catch the maximum static friction force value cannot be achieved reliably from observing the first displacement in position measurements acquired from encoder. Due to the noise and error inherent in the electronic system, the false static friction force values can be read due to misleading position change information. It was clearly stated in a previous study (de Wit et al., 1995) that instead of change in position information, speed data with a sudden jump should be used to accept the motion initiation.

To utilize the same approach in this study, which speed value should be used as the threshold speed was unclear. Therefore, the automation algorithm of the static experiments was tested at different threshold speed values to obtain correct static friction forces. The results were compared and they were quite consistent between 0.001 and 0.05 m/s threshold speeds. Very small and irrational friction force values were obtained at values lower than 0.001 m/s due to the noise and error. On the other hand, at threshold values greater than 0.05 m/s, the algorithm did not behave as desired due to the sudden motion of the piston and insufficient time to fetch the force information. Therefore, in this study, to catch up the initiation of movement in the algorithm, the linear speed (v) of the piston is measured and when it just first reaches 0.02 m/s value, the z signal is reset to zero and the PRV is shut down (Figure 3.22).

When all these experiment stages are completed, the counter value (k) is compared with the static experiment number (N_s) defined by the user. The same procedures are repeated until the counter value (k) is equal to the number of static experiments (N_s). When the experiments in the extension direction are completed, the counter variable (k) is reset to zero and the test procedures in the retraction direction start.

The algorithm of the static friction experiment for the retraction direction is similar to the algorithm of the extension as in Figure 3.22. The only difference is that when applying the ramping signal, the pressure is directed to chamber “b” instead of chamber “a” (Meanwhile, the DCV-a is adjusted such that the chamber “a” is fully exhausted to the atmosphere). Therefore, the steps in “Application of ramping signal” block in Figure 3.22 are reconfigured as: “Set DCV-b to feed Chamber b” and “Set DCV-b to exhaust Chamber a”.

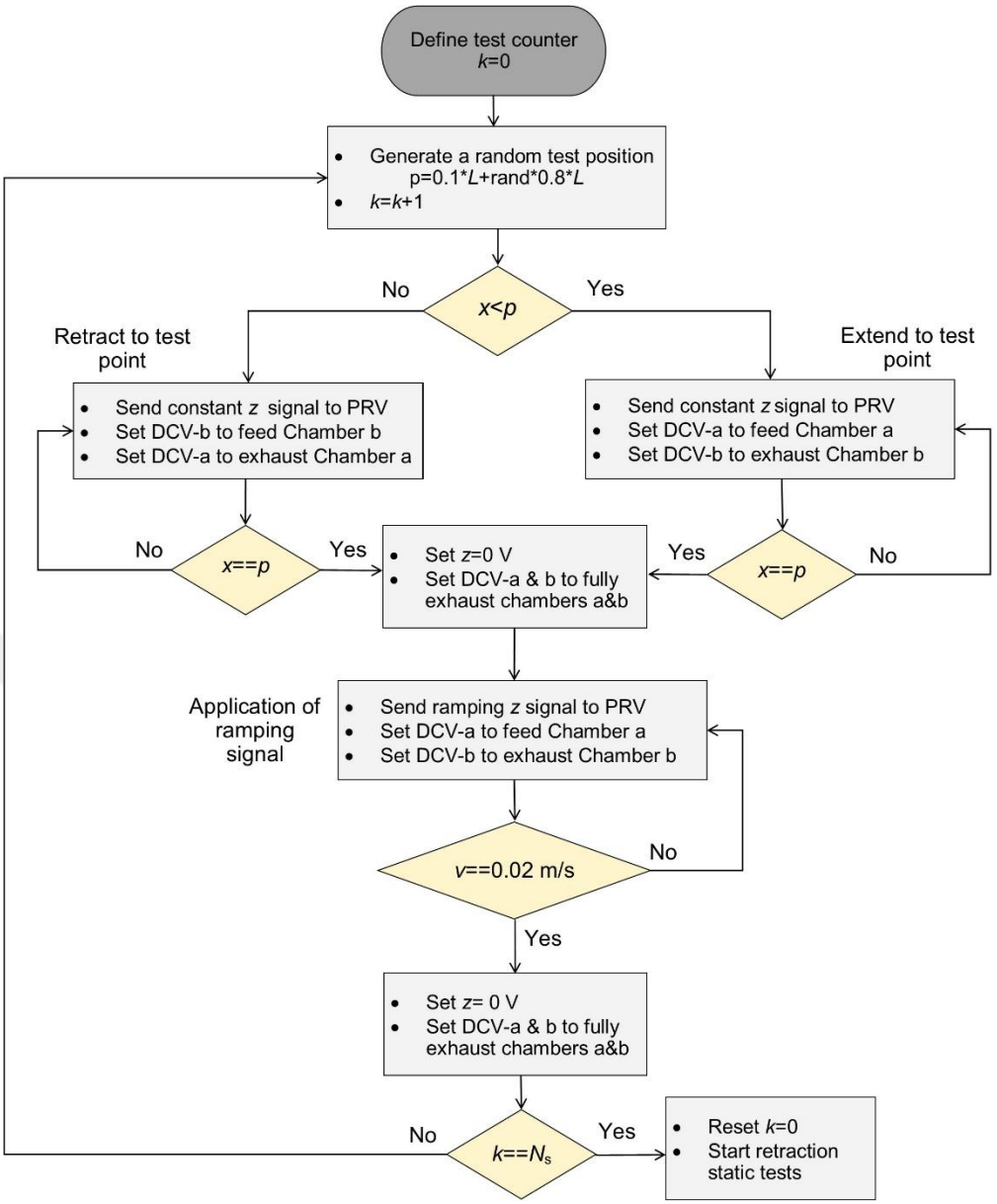

Figure 3.22. Flow chart of static experiments (for extension direction)

After the static experiments are performed successfully, the system automatically starts the extension dynamic experiments and then continues with the retraction dynamic experiments. The dynamic test procedures involve moving the piston of pneumatic cylinder at a particular constant speed under net forces developed by pressure differences according to Equation 3.5 and 3.6. Therefore, the net force vs. constant speed value pairs should be acquired for each particular dynamic experiment from the measurement of piston speed (v) and chamber pressures (P_a and P_b). In order to obtain constant speed, separate proportional directional control valves for each chamber of the cylinder are utilized. During the extension movement, while the valve on the extension side supplies pressurized air from the pressure regulating valve to the chamber "a" (DCV-a-through port 2), the other valve discharges the air in the chamber "b" to the atmosphere (DCV-b through port 4). On the other hand, for the retraction movement, the DCV-a is set to exhaust the air in the chamber

"a" whereas the DCV-b is set to supply pressurized air. These valve configurations are controlled by Stateflow chart-I. Sample measurements (P_a, P_b and v) of a dynamic experiment is shown in Figure 3.23. The constant speed region can be easily observed especially in the time interval of t=726 and t=726.6 s.

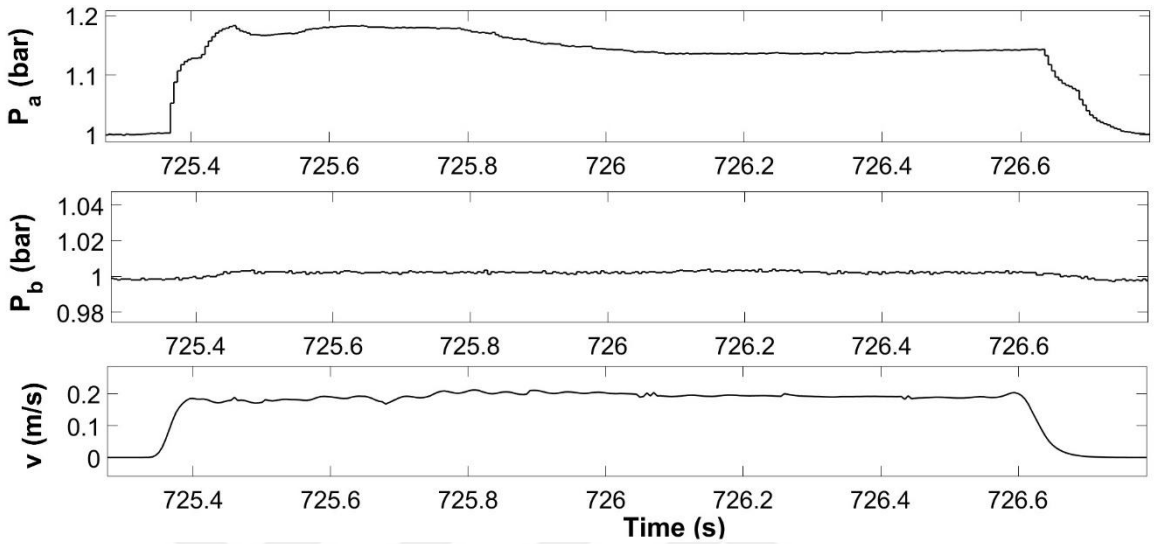

Figure 3.23. P_a, P_b and v signals for a constant speed extension movement

In the dynamic experiments, the piston's motion is restricted between the positions of 0.2L and 0.8L to prevent impact at high linear speeds. Also the experiments should be carried out at various piston speed and friction force value pairs in order to collect satisfactory amount of data for efficient curve fitting. Hence, a constant pressure is supplied from the PRV by setting control signal z for both the extension and retraction dynamic experiments. The constant pressure from the PRV is directed to the corresponding chamber for the extension (or retraction) motion through either the DCV-a or DCV-b, while the other valve is set to exhaust air fully. However, the different speeds of extension or retraction are obtained by controlling the valve openings for the corresponding DCV. The valve openings are controlled by control signals, namely u_a for the DCV-a and u_b for the DCV-b. Both of the DCVs are same type and these valves are normally fully closed at 5 V. Based on the catalog information of the valves, the maximum air flow at port 2 is obtained when the control signal is 0 V. On the other side, the maximum airflow at port 4 is obtained when the control signal is 10 V. As a result, a systematic way can be defined for changing the control signal based on the number of dynamic experiments. At this point one approach would be to shift the control signal from fully closed to fully open valve positions at equal intervals. However, fully open valve positions would result in very high operating speeds, such as greater than 2 m/s, which are impractical in applications and cause of difficulty in signal processing due to the permissible time interval for processing. Hence, a maximum speed around 1 m/s for both of the operating directions is selected in this study. Also, the experimental trials showed that there might be slight

differences in the behavior of the directional control valves from the catalog information. After some successive trials, it has been realized that the DCV-a can generate speeds of 0 to 1 m/s through port 2 for the extension with the control signals from 4.2 V to 5 V, while the DCV-b can produce speeds between 0 and -1 m/s through port 4 for the retraction with the control signals from 5 V to 6 V (for pneumatic cylinders whose *PD* value is around 25 mm). Therefore, in order to perform the dynamic experiments at different linear speeds in the range of 0 to 1 m/s, a random u_a signal between 4.2 V and 5 V for the extension direction is generated for the DCV-a (Equation 3.18) whereas a random u_b signal between 5 V and 6 V for the retraction direction is generated for the DCV-b (Equation 3.19).

$$u_a = (4.2 + 0.8 \cdot rand) \text{ V} \tag{3.18}$$

$$u_b = (5 + rand) \text{ V} \tag{3.19}$$

Flow charts for the extension and retraction dynamic experiments are shown in Figures 3.24 and 3.25, respectively. The dynamic experiments begin in the direction of extension by setting z=1.5 V and u_b=0 V. P_s becomes absolute 2.5 bar and the DCV-b is fully opened to the atmosphere as a result of these control signal arrangements. As in the static experiments, the counter value (k) is increased by one and the u_a signal is generated by utilizing the "*rand"* function in MATLAB based on Equation 3.18. This control signal is sent to the DCV-a. In this way, the piston extends at a certain constant speed until the piston position reaches 80% of the cylinder stroke length (x = 0.8L i.e. upper limit for the extension). The chamber pressures of the cylinder (P_a and P_b) and the linear speed of the piston (v) are measured and recorded throughout the motion. At the end of each experiment, the piston returns to the position corresponding to 20% of its stroke length (x = 0.2L i.e. lower limit for the extension) in order to execute the subsequent dynamic experiment. Then, the counter value (k) is compared with the number of dynamic experiments (N_d) entered by the user. The dynamic experiments are repeated at different speeds until these two parameters (k and N_d) become equal.

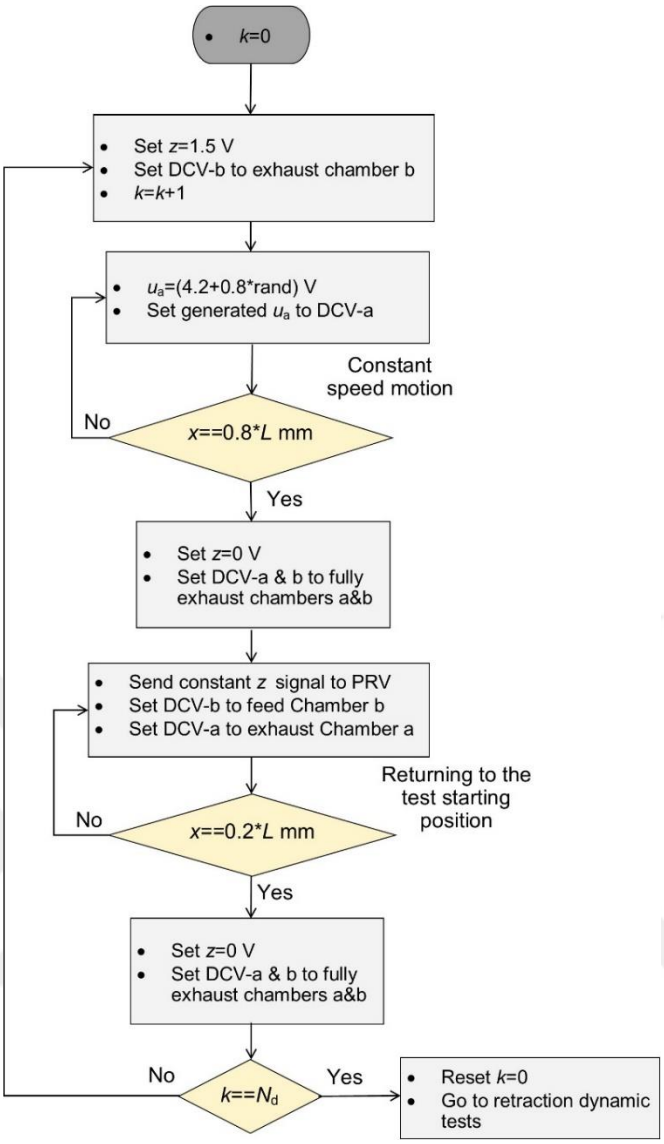

Figure 3.24. Flowchart for dynamic friction identification (for extension direction)

When the dynamic experiments in the extension direction are completed, the counter value is reset to zero value again and the piston automatically moves itself to the starting position of x=0.8L for the dynamic tests in the retraction direction. The dynamic identification of friction of retraction direction is represented as in Figure 3.25. Here, similar to the dynamic experiments in the direction of extension, the control signals are set as z=1.5 V to set P_s=2.5 bar absolute pressure and u_a=10 V in order to exhaust the pneumatic air in chamber "a" to the atmosphere directly. The randomly generated u_b signal from Equation 3.19 is sent to the DCV-b and the piston is allowed to move in the retraction direction with a constant speed until its position reaches 0.2L. Once the piston reaches this position, the movement of the piston is stopped by setting z=0 V and after a while the piston is returned back to 0.8L for the next experiment. At this point the counter value (k) is compared with the dynamic test number (N_d) and the same steps are applied until the equality of these parameters (k and N_d) is satisfied. When the experiments in the direction of retraction are

completed, the piston is pulled back to its home position (x = 0.1L) indicating that all the experiments are completed.

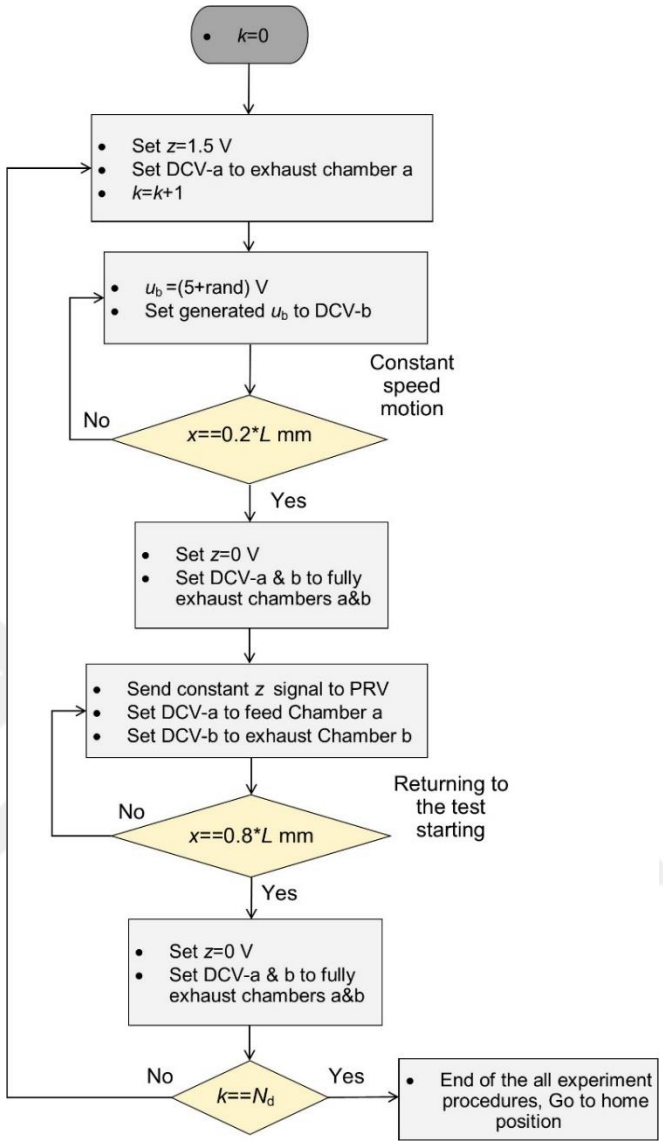

Figure 3.25. Flowchart for dynamic friction identification (for retraction direction)

Once the dynamic experiments are finished, the raw friction force (Raw F_f) values are generated by "Friction force calculation block" and the raw linear speed (Raw v) values are produced by "Encoder reading block". However, these signals cannot be directly used in the estimation of the friction force parameters even after the proper signal filtering from noise. Because, several phenomena arise during each experiment until steady state constant friction force and constant speed values are observed once the pressurized air is applied to the respective chambers. An example "Raw F_f" and "Raw v" vs. time graphs for the extension dynamic test values are shown in Figure 3.26 where a dynamic experiment period is also represented with a dashed rectangular box. Also, each consecutive experiment is symbolized by sub-index n, $n+1$ and $n+2$. In Figure 3.26, the piston is extended from the point "i" to the point "ii". Then, between the points "ii" and "iii", there is no motion since the system waits for the following command and no

friction force is generated. The cylinder is retracted from point “iii” to point “iv” and is ready to proceed to the next experiment period after a certain stalling time.

As a result, the useful data for one dynamic experiment period can be obtained only from the point “i” to the point “ii” where the other regions should be cropped. Even in this region, the irregularities of speed and the corresponding friction force values should be cropped for identifying the constant speed and constant friction force values (circled with the dashed lines in Figure 3.26). This cropping process is performed in the Stateflow chart-II where the retraction data as well as the unnecessary data are updated to zero for the extension dynamic experiments. For the retraction dynamic experiments, the extension data as well as the unnecessary data are equated to zero value. Even this cropping process is not sufficient for the calculation of constant speed and constant friction force values. Therefore, the average values of the cropped signals for friction force (Processed F_f) and linear speed (Processed v) are generated by the Stateflow chart-III.

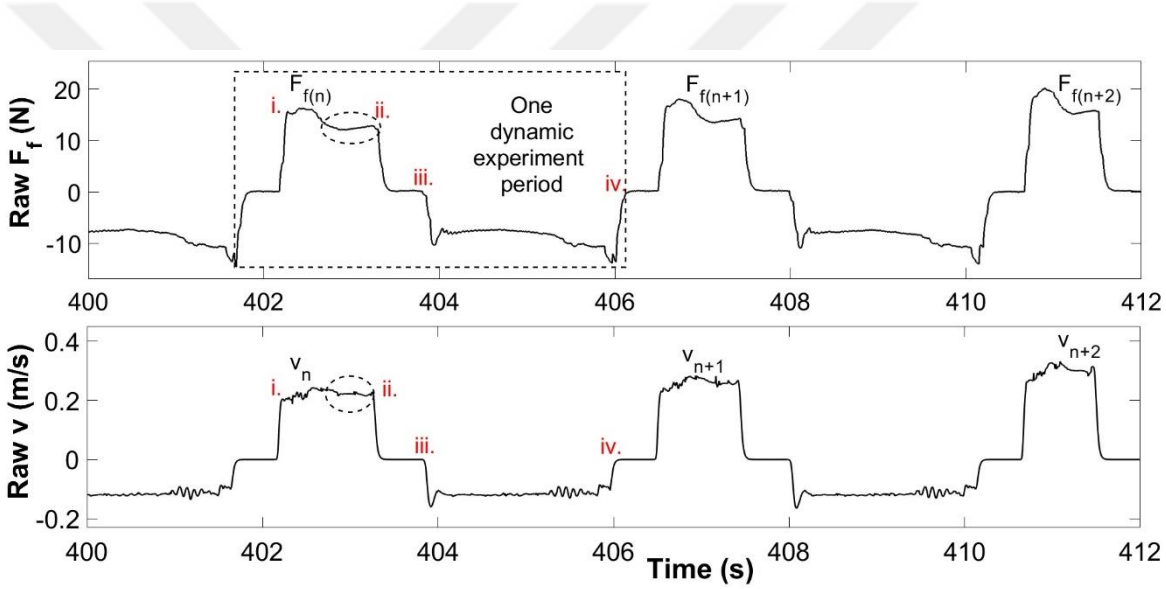

Figure 3.26. Raw F_f and v signals during the dynamic experiments (for extension direction)

The data obtained from the static and dynamic experiments for both the extension and retraction directions are used in nonlinear curve fitting operation for the estimation of friction parameters. The least square curve fitting method has been applied. The curve fitting algorithm has been selected optionally as “*Levenberg-Marquardt*”. The same experimental approach as in friction torque identification procedures in previous section has been applied for definition of initial guesses for fitting operation. As a result, good initial guesses for the tested cylinders have been obtained. The preferred initial values are tabulated in Table 3.5. In addition to that, the upper and lower bounds for the parameters, F_c, v_s and B, are defined as in Table 3.6.

Table 3.5. Initial guess values for curve fitting process

Parameters	**Extension**	**Retraction**
F_c	5	-5
v_s	0.1	-0.1
B	100	100

Table 3.6. Defined parameter bounds

Extension	**Retraction**
$0<F_c\leq F_s$	$F_s\leq F_c<0$
$0<v_s\leq 0.1$	$-0.1\leq v_s<0$
$0<B\leq 100$	$0<B\leq 100$

A graphical user interface (GUI) has been designed in MATLAB/AppDesigner to facilitate the usage of complete automated system and adapted to whole setup for entering the required parameters of the experiments, connecting to the experimental setup, performing or stopping the experiments and presenting results. The designed GUI primarily receives the input parameters (*L, PD, RD*, N_s and N_d*)* from the user and then present the results of the estimated friction model together with the estimated parameters (F_s, F_c, v_s and B) to the user with a single calculation button.

The GUI consists of three main sections which are the “Inputs”, “Control Buttons” “Estimated Outputs” as shown in Figure 3.27. The “Inputs” section is located on the top left and the “Control Buttons” section is placed on the top right of the GUI. There is a dialog message box section under the section II for the warnings and instructions. The user can easily perform the experiments by following the directions presented in the dialog box. Finally, the “Estimated Outputs” section is placed on the bottom right of the GUI.

On the “Inputs” section, there are three subsections. Under the “Define cylinder size” subsection, the parameters that specify the dimensions of the pneumatic cylinder such as the stroke length (*L*), the piston diameter (*PD*) and the cylinder rod diameter (*RD*) of the pneumatic cylinder are entered by the user. Under the “Define static experiment condition” subsection, the number of static experiments (N_s) is submitted. Under the “Define dynamic experiment condition” subsection, user defines the number of dynamic experiments (N_d). The “Control Buttons” section includes the buttons that have functions such as connecting to the experimental setup (Connect), disconnecting from the setup (Disconnect), starting experiments (Start) and stopping experiments (Stop). The user determines the course of the experiment using the control buttons. In addition, there are two auxiliary buttons named as “Clear All” and “Calculate”. The user clicks “Clear All” to clear existing parameter values to input new parameters for a new experiment. When all the experiment steps are completed, the user presses the “Calculate” button. Then, the estimated results are provided to the user in the “Estimated Outputs” section separately for the extension and retraction directions. The friction force-linear speed maps are demonstrated on the left bottom with the title of “Estimated friction force-linear speed map”.

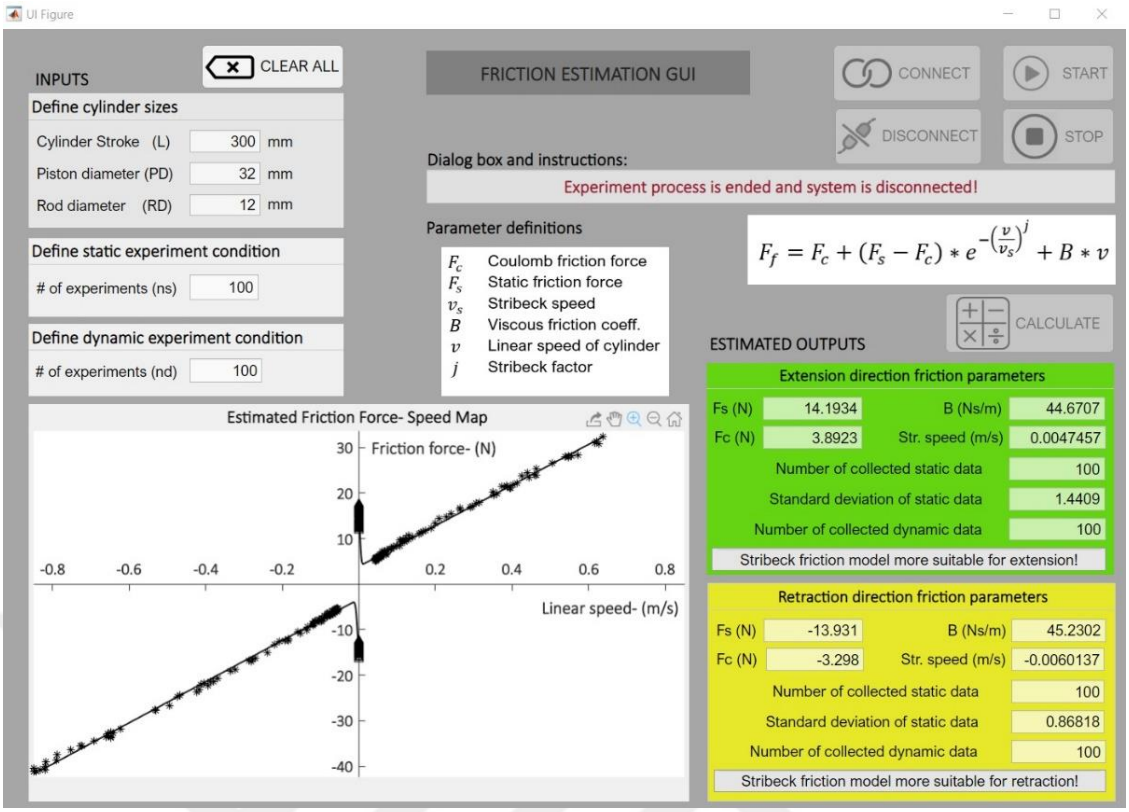

Figure 3.27. Designed graphical user interface

3.2.5. The Assumptions of Unidentified Dynamics of the Robot

During the real-time hardware in the loop simulations (i.e. executed workouts on the robot), the following assumptions are considered:

- The moment of inertia of the robot's rotating links are ignored. The underlying reason behind that is the workouts are carried out at low and constant speeds. Therefore, since the acceleration term will be close to zero level, there will be no need to calculate the moment of inertia.
- Pneumatic air has been accepted as an ideal gas due to the fact that all mathematical representations of pneumatic systems stemmed from the fundamental ideal gas equation, $P \cdot V = m \cdot R \cdot T$.
- It has been accepted that the pressure and temperature in the cylinder chamber were distributed uniformly. These assumptions have been tested by positioning the pressure sensors to detect chamber pressures (i.e. P_a and P_b) extremely close to the cylinder's air ports.
- During the derivation of equations utilized in any parameter identification and validation processes, the kinetic and potential energy aspects of pneumatic air have been ignored.

- Air leakages along the pneumatic pipelines and in the cylinder chambers have been ignored because pneumatic airflow has been monitored in a completely enclosed environment and no leakages or pressure losses have been found in the real setup. In long-piped systems, the turbulent scenario should be studied to analyze pressure attenuations and time delays, as explained by Richer and Hurmuzlu (2000).
- Throughout the tests, the temperature of the pneumatic air has been supposed to be identical to the temperature of atmospheric air (T_a=T_u=T_{atm}). Nevertheless, when the chamber volume of the cylinder is large, temperature measurements gain significance due to heat transfer considerations from the cylinder walls. This specific phenomenon was investigated by Carneiro and De Almeida (2007).

3.3. General Control Approach and Controller Tools

The proposed robot works in the principle of controlling the chamber pressures based on target interaction torque $[T_i]^t$. During the design stage of the controller the following issues have been considered to satisfy the compliancy, safety and efficiency during the workouts:

- The robot should identify the motion intent of the user for compliant human-robot interaction. To satisfy this demand, a user motion intention detection algorithm is needed.
- The robot's controller framework should be capable of estimating friction torque under varying operating conditions. To achieve this, an algorithm must be employed to detect and identify concurrent friction behavior. This will enable the generation of appropriate controller outputs aligned with the identified friction.
- The robot needs to respond appropriately to systematic (i.e. internal) disturbances, like pressure losses during workouts. Hence, it's crucial to integrate a pressure disturbance rejection algorithm into the controller framework. This algorithm will manage pressure disturbances and prevent sudden, unintended movements, ensuring the user's safety.
- The robot may face with external disturbances since it is interacting with the user and unidentified environments. In case of any external disturbance in the form of either speed or torque, the robot should act safely as desired. Therefore, a torque disturbance rejection algorithm should be incorporated in the considered controller framework.

Based on the aforementioned issues, an efficient controller framework is designed as in Figure 3.28. In this block, black continuous and black hidden lines represent the feedforward and feedback signals, respectively. On the other hand, black dotted lines represent the pneumatic air flow line.

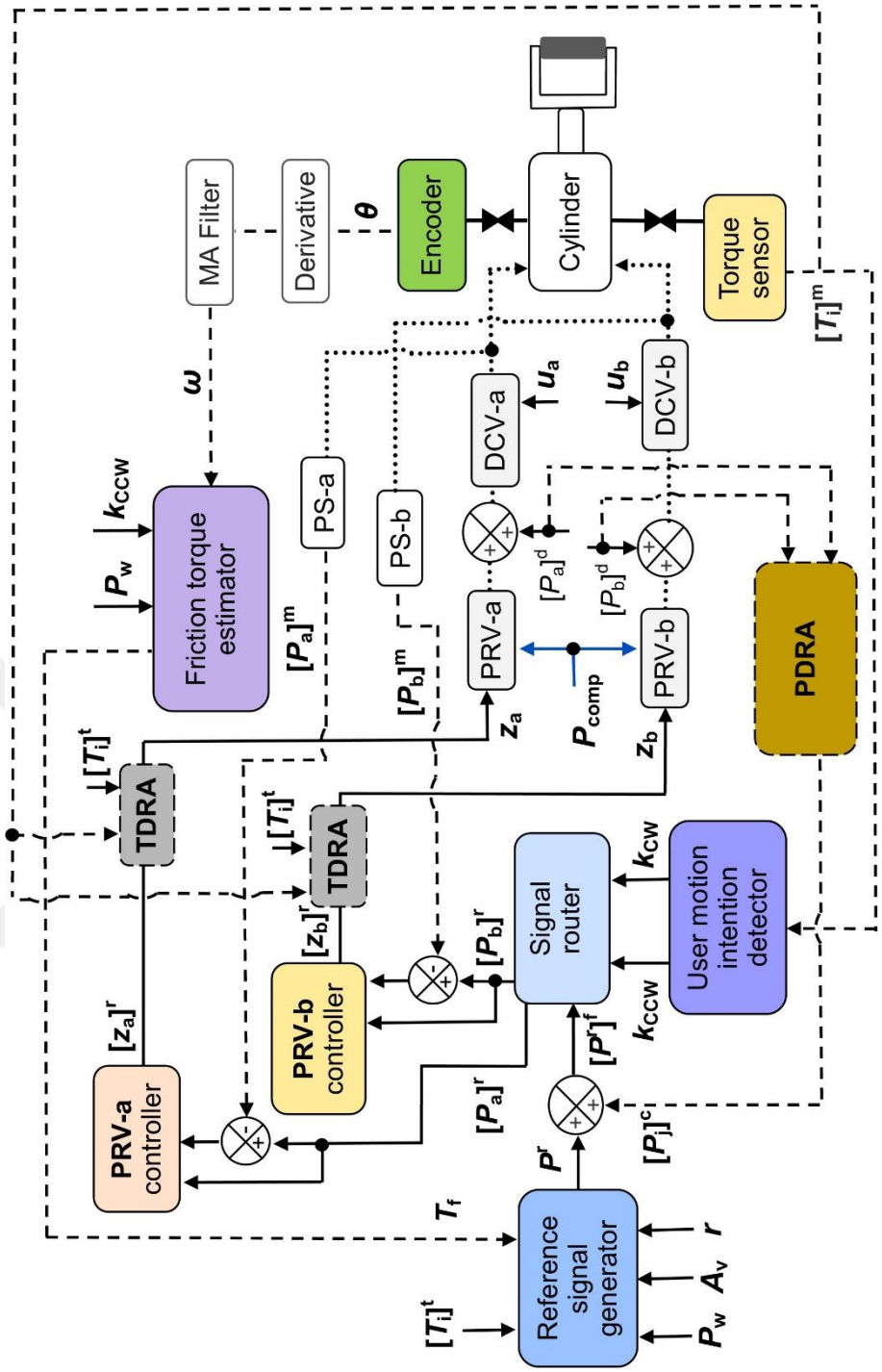

Figure 3.28. General controller block schema

For the control of independent chamber pressures, separate but identical controller blocks are designed for main control valves as depicted in Figure 3.28 (i.e. PRV-a and PRV-b). The directional control valves (DCV-a and DCV-b) are just utilized for directing the flow of pneumatic air. With respect to this information, the manipulated signals are defined as z_a and z_b which controls the pressure output from PRV-a and PRV-b. The control signals of DCV-a and DCV-b (i.e. u_a and u_b) are not manipulated, they are set to a specific value throughout all control period.

In this schema, “Friction torque estimator” block generates friction torque, T_f according to working pressure (P_w) and operating speed (w) and feeds the friction information to the main block which is “Reference signal generator”. In this block a single reference pressure, P^r is produced based on input data of working pressure, P_w, vane surface area, A_v, torque radius of the cylinder, r, specified target interaction torque, $[T_i]^t$ and friction torque, T_f defined by operating conditions. P^r is then sent to the sum block for the generation of final reference pressure, $[P^r]^f$ based on any corrective pressure data $[P_j]^c$ (j= a or b with respect to the chamber) generated by “Pressure disturbance rejection algorithm (PDRA)” in any case of pressure disturbances as $[P_a]^d$ and $[P_b]^d$. “User motion intention detector” block takes the measured interaction torque, $[T_i]^m$ and generates triggering signals (i.e. k_{CCW} and k_{CW}) for the routing of reference pressure to the chambers. Based on the algorithm embedded in “Signal router” block, reference pressures for the chambers “a” and “b” are specified as $[P_a]^r$ and $[P_b]^r$, respectively.

Produced reference pressures, $[P_a]^r$ and $[P_b]^r$ are sent to the sum blocks. Sum blocks generate error in chamber pressures (P_a and P_b) as $e_{P,a}$ and $e_{P,b}$ where these error signals are sent as an input to the “PRV-a controller” and “PRV-b controller” blocks, respectively. $e_{P,a}$ and $e_{P,b}$ are calculated as $[P_a]^r$-$[P_a]^m$ and $[P_b]^r$-$[P_b]^m$, respectively. “PRV-a controller” and “PRV-b controller” blocks generate raw control signals ($[z_a]^r$ and $[z_b]^r$) and are fed to the “Torque disturbance rejection algorithm (TDRA)”. TDRA evaluates the specified and measured interaction torques in case of any disturbance cases and generates final control signals for PRV-a and PRV-b, as z_a and z_b, respectively.

In the following sections, the blocks inserted in the controller framework will be covered in depth.

3.3.1. User Motion Intention Detector and Signal Routing

User intention detection block is designed to sense the natural movement intention of the user which aims to provide compliant interaction. The details of working principles of this block are demonstrated as in Figure 3.29. It operates on the basis of creating a triggering signal (k) for the controller by sensing interaction torque $[T_i]^m$ between the robot and user at HRI zone by using MATLAB/Stateflow® charts. This block activates/deactivates three states with respect to $[T_i]^m$ in comparative manner (i.e. measured torque is compared with a specific threshold value). The threshold values to settle any movement intention is specified experimentally as 0.25 Nm for CCW and -0.25 Nm for CW directions. Due to the noise in the torque signal, the selection of lower threshold values leads to incorrect user intention recognition, which causes the robot to move in an undesirable manner. On the other hand, the choice of high threshold values makes it difficult to identify user intention even when the user wishes to do a specific motion at low contact torque levels.

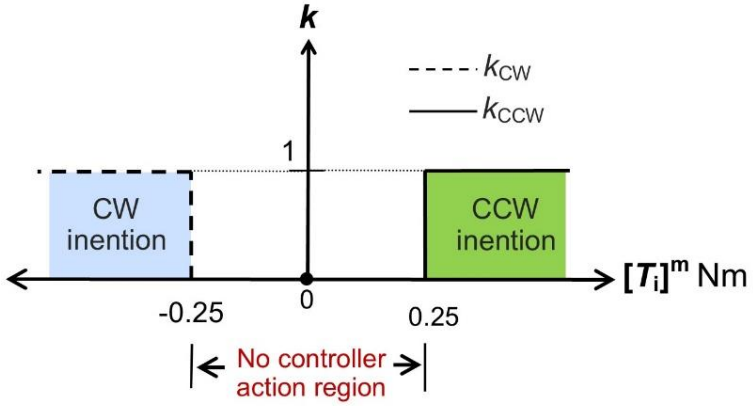

Figure 3.29. Motion direction detection chart

In Figure 3.30, the MATLAB/Stateflow chart based on the algorithm in Figure 3.29 is presented. In this chart, States "0", "1" and "2" represent the "no intention", "CCW intention" and "CW intention", respectively. The value of *k* is switched from zero to one with respect to Figure 3.30. The signal, *k*, serves multiple purposes such as enabling/disabling the controller actions and playing a role in determining the movement direction, thereby influencing the operational status of other auxiliary blocks, for example "Signal routing" and "Friction torque estimation".

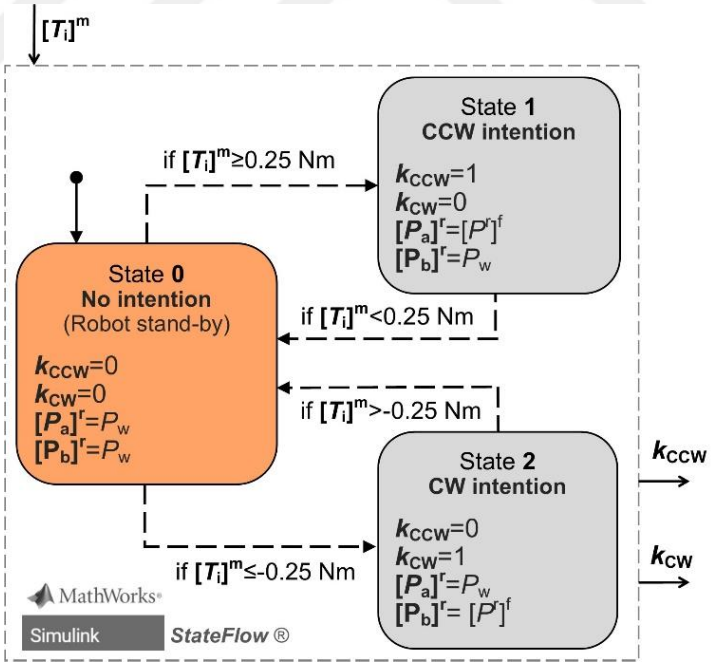

Figure 3.30. User motion intention detection algorithm and signal switching algorithm

Final reference pressure, $[P^r]^f$ directed to the "Signal switching" is then directed to the "PRV-a controller" and "PRV-b controller" blocks based on the algorithm in "Signal switching" block as in the following rules:

- "Controller no-action" mode: $[P_a]^r=P_w$ and $[P_b]^r= P_w$
    - *The action of the robot*: No movement is observed; the end-effector does not move to protect the user from any unintended movements. For the purpose of giving more detail, during any specific movement, if the user wants to stop or feels any pain, the controller automatically stops the movement of the end-effector.
- "CCW intention" mode: $[P_a]^r=[P^r]^f$ and $[P_b]^r=P_w$
    - *The action of the robot*: The robot rotates in CCW direction at desired interaction torque, $[T_i]^t$.
- "CW intention" mode: $[P_a]^r=P_w$ and $[P_b]^r=[P^r]^f$
    - *The action of the robot*: The robot rotates in CW direction at desired interaction torque, $[T_i]^t$.

The procedures to get $[P_r]^f$ signal from P^r and $[P_j]^c$ will be covered in "Reference signal generator" subsection.

3.3.2. Friction Torque Estimator

Friction torque estimator block generates simultaneously the friction torque information inherent in the system based on the input parameters of working pressure (P_w), operating speed (ω) and detected movement direction (k_{CCW}=1 to define CCW direction and k_{CCW}=0 to define CW direction). In this block, the identified friction parameters are embedded by utilizing one dimensional (1D) lookup tables for each Stribeck friction parameters (i.e. T_s, T_c, w_s and B) based on specified working pressure in MATLAB/Simulink as in Figure 3.31. Real P_a value is fed as input for CCW direction movements since the controlled chamber is "a" whereas real P_b value is utilized since chamber "b" is controlled for CW direction movements.

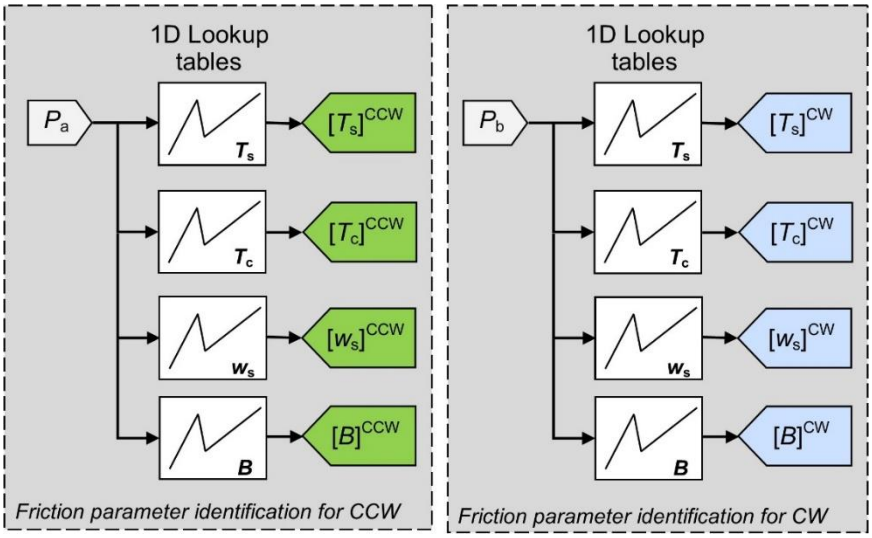

Figure 3.31. Lookup table schema for parameter identification based on chamber pressures

Identified friction parameters from lookup tables are then sent to the switching algorithm as in Figure 3.32. With the help of this algorithm, a single parameter value is executed based on the movement direction (i.e. k_{CCW}=1 enables the usage of CCW parameters whereas k_{CCW}=0 enables the usage of CW parameters) with the help of switching operation. To unify the parameter values and cancel out the negative sign, the values of CW direction ($[T_s]^{CW}$, $[T_c]^{CW}$, $[w_s]^{CW}$) is multiplied with "-1" gain except viscous friction coefficient, $[B]^{CW}$ since it has positive value for both CCW and CW directions.

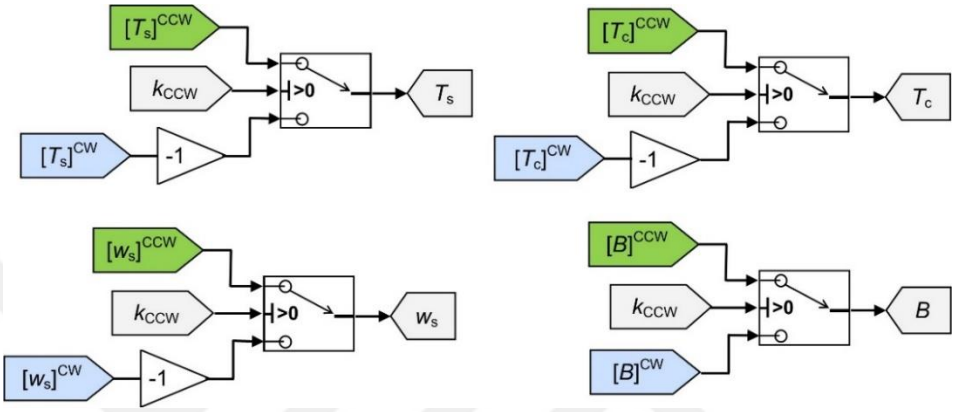

Figure 3.32. Execution of friction parameters based on movement direction

The unified parameters are then fed to the block schema given in Figure 3.33 to generate friction torque. In this schema, instead of original rotational speed, ω a regulated speed parameter is constructed to make the speed value positive in both CCW and CW directions. The regulated speed parameter, ω_{fric} is obtained based on the movement direction (k_{CCW}) and original speed (ω) signals as well as the equation presented in Equation 3.20. Other than identified parameters, Stribeck shape factor is assumed to be j=2.

$$\omega_{fric} = \begin{bmatrix} \omega \; if \; k_{CCW} = 1 \\ -\omega \; if \; k_{CCW} = 0 \end{bmatrix} \tag{3.20}$$

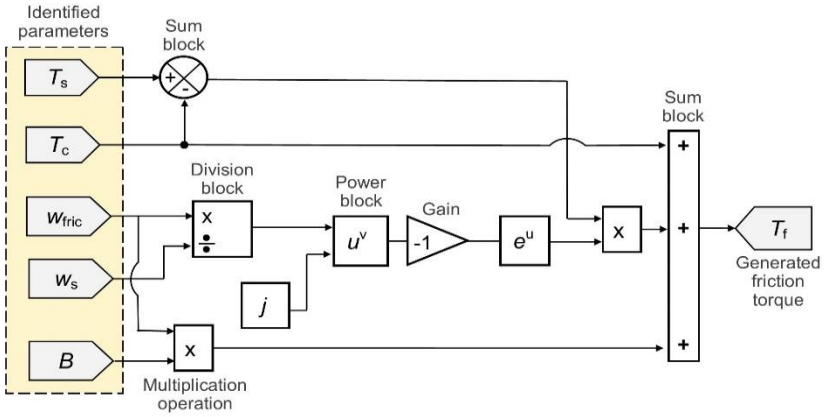

Figure 3.33. T_f calculation block based on executed parameters

Generated overall, T_f information is then sent to the "Reference signal generator" algorithm.

3.3.3. Pressure Disturbance Rejection Algorithm (PDRA)

From the considerations in controller design, an algorithm wherein the pressure losses in air flow line is automatically detected is designed as in Figure 3.34. In this algorithm, during the executed motions, the counter chamber pressure losses are considered. Namely, during any specific CCW movements, since the controlled chamber is "a", the losses in chamber "b" is considered. In a similar fashion, for CW movements, the losses in counter chamber "a" is evaluated.

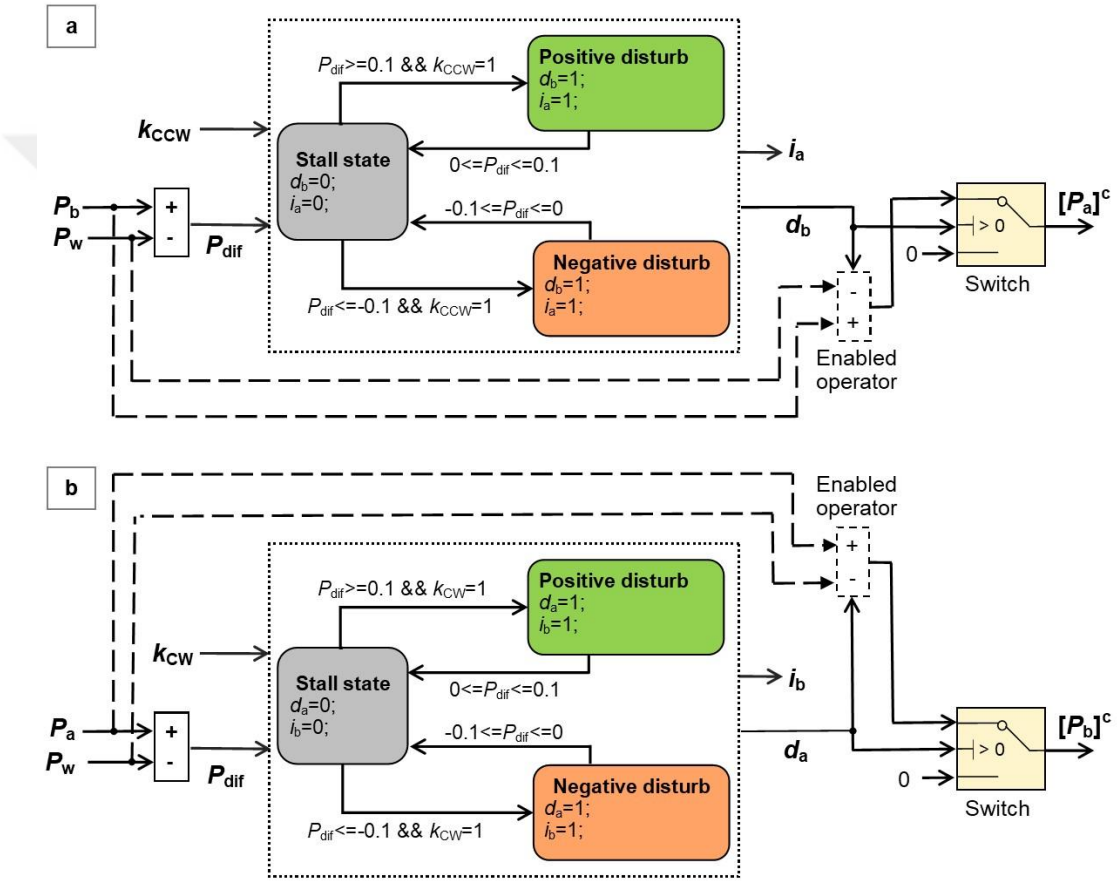

Figure 3.34. Pressure disturbance detection algorithm (PDRA) for (a) CCW; (b) CW motions

In this detection algorithm, the signals representing the movement direction (k_{CCW} and k_{CW}), the specified working pressure (P_w), and chamber pressures (P_a and P_b) are taken as an input and resultantly, the output data for defining final corrective signals ($[P_a]^c$ and $[P_b]^c$) are generated as i_a, i_b, d_a and d_b

The algorithms in Figures 3.34 (a) and (b) calculate the pressure difference, P_{dif} to detect any level of pressure losses (i.e. or disturbances) of $[P_a]^d$ and $[P_b]^d$. Throughout the controller execution, the algorithm firstly takes the direction information from "User motion intention detection" algorithm. According to the values of k_{CCW} and k_{CW}, one of the algorithms "a" or "b" in

Figure 3.34 is activated. For example, if k_{CW}=1, the algorithm "b" is activated and the difference in pressures P_w and P_a is calculated. If P_{dif} is bigger than 0.1 or smaller than -0.1, the loop is triggered with "Positive disturb" and "Negative disturb" blocks, respectively. With the successful execution of the loops, suitable i_a and d_b signals are generated and finally, corrective signal $[P_a]^c$ is sent to the disturbance sum block to generate accurate reference signal $[P_r]^f$. i_a and i_b signals are utilized for either activating or deactivating the disturbance sum block in Figure 3.28. Furthermore, d_a and d_b signals are used for the switching in the final part of the algorithms to determine the value of $[P^r]^f$ to become either zero or P_{dif}. If $-0.1 \leq P_{dif} \leq 0.1$ condition is satisfied, the "Stall state" block is activated; no actions are permitted from the disturbance observer algorithm; and $[P_a]^c$ becomes zero. As a summary, the algorithm is designed by taking into the considerations as in Table 3.7.

Table 3.7. Pressure disturbance compensator conditions and rules

k_{CCW}	$-0.1 \leq P_{dif} \leq 0.1$ (Stall state active)	$P_{dif} \geq 0.1$ or $P_{dif} \leq -0.1$ (Negative or Positive disturb in active state)
0	$i_a=d_b=0$; $[P_a]^c=0$	$i_a=d_b=0$; $[P_a]^c=0$
1	$i_a=d_b=0$; $[P_a]^c=0$	$i_a=d_b=1$; $[P_a]^c=P_{dif}$
k_{CW}	$-0.1 \leq P_{dif} \leq 0.1$ (Stall state active)	$P_{dif} \geq 0.1$ or $P_{dif} \leq -0.1$ (Negative or Positive disturb in active state)
0	$i_b=d_a=0$; $[P_b]^c=0$	$i_b=d_a=0$; $[P_b]^c=0$
1	$i_b=d_a=0$; $[P_b]^c=0$	$i_b=d_a=1$; $[P_b]^c=P_{dif}$

3.3.4. Torque Disturbance Rejection Algorithm (TDRA)

Torque disturbance rejection algorithm, depicted in Figure 3.35, is connected serially to the "PRV-a controller" and "PRV-b controller" blocks to make the system more robust. The reason behind that is, even the reference pressures, $[P_a]^r$ and $[P_b]^r$ is tracked during a specific exercise, the torque perturbations cannot be taken into account by the controller blocks without any torque disturbance algorithm. And therefore, the system can deviate from the specified interaction torque.

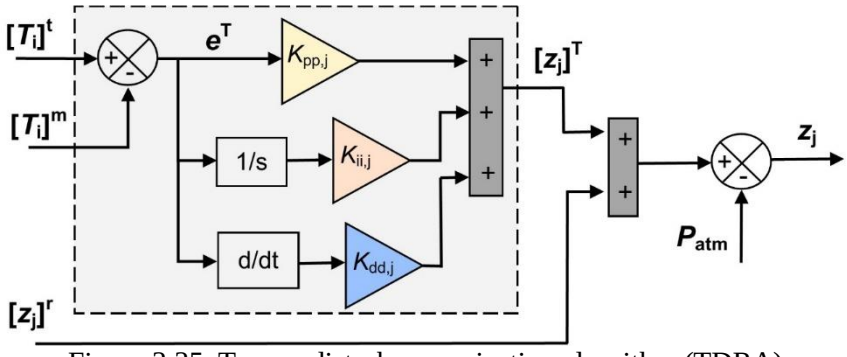

Figure 3.35. Torque disturbance rejection algorithm (TDRA)

The TDRA framework takes the error in torque, e^T and generates a corrective control signal $[z_j]^T$ for each chamber (i.e. j=a,b representing chamber "a" and chamber "b") to prevent the torque disruptions. $K_{pp,j}$, $K_{ii,j}$ and $K_{dd,j}$ are the proportional, integral, and derivative gains of PID structure

in TDRA, respectively. Based on this information, TDRA is mathematically represented as in Equation 3.21.

$$\left[z_j\right]^T = K_{pp,j} \cdot e^T(t) + K_{ii,j} \cdot \int e^T(t) \cdot dt + K_{dd,j} \cdot \frac{de^T(t)}{dt}; j = \text{a, b} \tag{3.21}$$

These gains are experimentally identified under changing interaction torque levels and torque disruptions separately for "PRV-a controller" and "PRV-b controller". According to the experimental optimization, K_{pp}=0.165, K_{ii}=4 and K_{dd}=0.0002 are obtained for both "PRV-a controller" and "PRV-b controller" with experimental trial and error. With respect to the block diagram presented in Figure 3.35, the final output signal z_j can be expressed mathematically as in Equation 3.22. P_{atm} is subtracted from the resultant signal to obtain absolute valve pressure output.

$$z_j = \left[z_j\right]^T + [z_j]^r - P_{atm};\ j = \text{a, b} \tag{3.22}$$

The generation of the raw controller signal, z_j will be detailed in the following controller design section.

3.3.5. Reference Signal Generator

In this block, in order to prevent rapid motion and offer more antagonistic-like movement of human body structure, the operation of the cylinder is manipulated so that while controlling the pressure inside one chamber (i.e. either in "*a*" or "*b*"), the counter chamber is set to working pressure, P_w. Therefore, during a particular motion, this block derives only one reference pressure, P^r output. This reference signal is obtained by utilizing Equation 3.23 with respect to input parameters ($[T_i]^t$, P_w, A_v and r) and friction torque information fed from "Friction torque estimator" block. The Equation 3.23 is constructed by modifying the Equations 3.1, 3.2, 3.3 and 3.4 under the constant speed conditions (i.e. ω=constant and α=0 rad/s^2) whereby eliminating the estimation of mass moment of inertia, I.

$$P^r = P_w + \frac{\left[T_f\right] - [T_i]^t}{A_v \cdot r} \tag{3.23}$$

The generated reference pressure (P^r) is transmitted to the disturbance sum block to be compensated by the signal, $[P_j]^c$ coming from PDRA block (Section 3.3.3) in case of any disturbance in pneumatic line (i.e. loss in pressure due to the malfunction or inadequate compressor pressure). Reference signal then takes its final form as in Equation 3.24 as following:

$$[P^r]^f = P^r + [P_j]^c \tag{3.24}$$

Final corrected reference signal is then transmitted to the "User motion intention detection and Signal Routing" block to set the correct reference signals for chambers "a" and "b", respectively as in mentioned previously in Section 3.3.1.

3.4. Internal Structure of "PRV-a Controller" and "PRV-b Controller"

The inlet structures of "PRV-a controller" and "PRV-b controller" are identical. Inside these controller blocks, there are three different controller frameworks existing in the name of "Default", "Cascade PID" and proposed "Cascade Fuzzy Adaptive PID". In "Default" control approach, the input signal is directly applied to the controller valves (PRV-a and PRV-b) with respect to the final reference pressure $[P^r]^f$. On the other side, in "Cascade PID" framework, a conventional PID structure with constant optimized gains is input before the valve inlets and the system works with respect to the error in reference pressure, e_p. Finally, in "Cascade Fuzzy Adaptive PID" framework, PID gains are scheduled with the efficient fuzzy logic design wherein the error (e_p) and error change rate ($\dot{e}_p$) is evaluated. In the following sections, since the inlet structure of the "PRV-a controller" and "PRV-b controller" from the structure point of view is identical, instead of representing their controller schema separately, a single generalized block schema is represented by utilizing the *j* subscript to denote either controllers or chambers "a" and "b".

These controllers are activated or deactivated based on the switching algorithm by utilizing "*if-action*" algorithm as in Figure 3.36 in MATLAB/Simulink. In this algorithm, a local variable called "*h*" is constructed in switching interface. This local variable can have the values of "1", "2" or "3" for "Default", "Cascade PID" or "Cascade Fuzzy Adaptive PID". These values are sent to the *if* block for activating one of the designed controllers. Generated triggering signal after *if* operation activates one of the controller blocks in "action" block by passing thorough "Merge" block. The robot is manipulated under activated controller with generated control signal output.

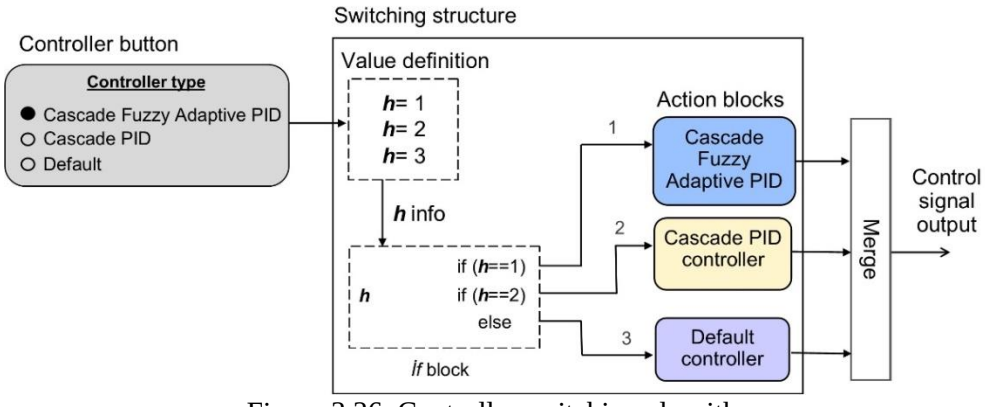

Figure 3.36. Controller switching algorithm

The details of the controller designs will be covered for “Default”, “Cascade PID” and “Cascade Fuzzy Adaptive PID”, respectively in the following sections.

3.5. Design and Application of Default Controller

The “Default” controller is the simplest and basic controller among the controllers. The “Default” controller is simply the built-in controller in controller valves (i.e. PRV-a and PRV-b) specifically designed by the manufacturer (i.e. FESTO). The working principle and details of these valves have been already given in Section 3.2.2. According to the manufacturer’s catalog, no detail of the control algorithm is given, however there are three different controller modes available in the form of “a. Fast Control Response”, “b. Universal Control Response (factory setting)” and “c. Precise control response”[5].

The block schema of “Default” controller is depicted in Figure 3.37. According to this schema, the reference pressure ($[P_j]^r$) is fed as input and the suitable raw control signal ($[z_j]^r$) is produced with respect Equation 3.25 based on the unity gain criteria.

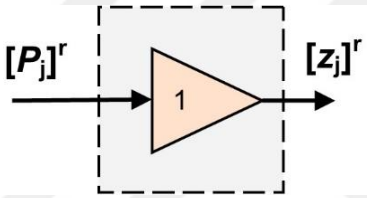

Figure 3.37. Block schema of Default controller

$$[z_j]^r = [P_j]^r \tag{3.25}$$

3.6. Design and Application of Cascade PID Controller

From the application point of view, as the performance expectations in terms of robustness and precision continue to grow, classical PID control strategies become limited, making it difficult to achieve desired results for uncertain systems. Using traditional controllers within cascade frameworks provides a solution that tackles the issues of traditional configurations while retaining their functions and characteristics.

3.6.1. Block Schema of Cascade PID Controller

To improve the performance of static and dynamic responses of the robotic system, cascade PID controllers have been designed for each control valve as in Figure 3.38, where the error in reference chamber pressure values ($e_{P,\,j}$) are fed as input and resultantly, final raw control signal $[z_j]^r$ is generated for the control valves (PRV-a and PRV-b). According to the conventional PID framework as in Figure 3.38, the outputs of the PIDs’ are mathematically represented as in Equation 3.26.

[5] https://www.festo.com/net/en-gb_gb/SupportPortal/Files/12877/PSI_241_1_VPPM_en.pdf

$$[z_j]^r = K_{p,j} \cdot e_{P,j} + K_{i,j} \cdot \int e_{P,j} \cdot dt + K_{d,j} \cdot \frac{de_{P,j}}{dt}; j = \text{a, b} \quad (3.26)$$

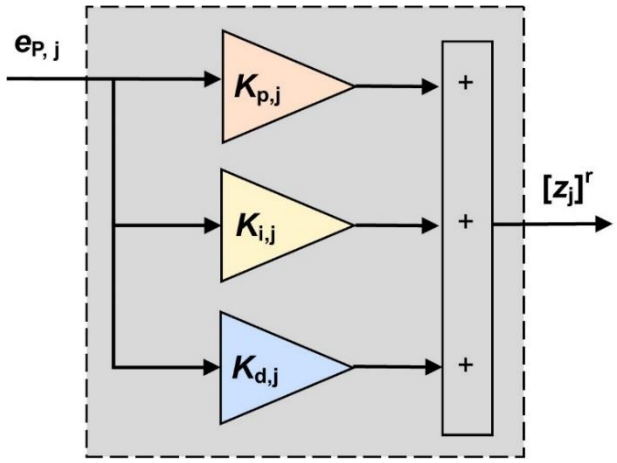

Figure 3.38. Block schema of Cascade PID controller

3.6.2. Optimization of PID Gains

Even the valves are the same type, their responses have shown differences which have been proven by quick experiments on the valves. To enhance the valve's own performance, each valve has been separately investigated for tuning of PID gains ($K_{p,j}$, $K_{d,j}$ and $K_{i,j}$). The tuning of these gains is a critical task to get the best controller outputs. Therefore, an optimization algorithm is needed to specify ideal $K_{p,j}$, $K_{i,j}$ and $K_{d,j}$. In the literature, both on-line and off-line optimization algorithms have been employed. The off-line methodology involves constructing a transfer function model of the system and evaluating the system's response to a step input. This approach is efficient and straightforward to implement. However, accurately defining the transfer function for certain applications, such as highly nonlinear systems like pneumatics, can be extremely challenging. Therefore, in this study, an on-line optimization algorithm has been chosen as the preferred approach.

The online methodology employs a real-time closed-loop algorithm called the "PID Autotuner" in MATLAB/Simulink. This algorithm operates by performing a frequency response estimation, where control signals are applied to the plant (i.e., pneumatic cylinder) to tune the PID gains based on the estimated frequency response. For the proposed pneumatic systems, the target bandwidth and target phase margin are specified as 2 rad/s and 60 degrees. These values are determined based on the systems operating conditions (i.e. slow and stable speed movements). By default, PID auto-tuner test the control signals at the corner frequencies (ω_c) of [1/10, 1/3, 1, 3, 10]. Once the tuning experiments are concluded, the block utilizes the estimated frequency response to calculate the optimal PID gains. The objective is to strike a balance between performance and robustness while meeting the specified control bandwidth and phase margin.

The PID auto-tuner algorithm is presented in the form of block schema as in Figure 3.39. In this schema, *e* is the error in pressure; *P*, *I* and *D* are the proportional, integral and derivative gains, respectively. *T* is the parameter which enables and disables the tracking of output signal, *u*. And finally, *N* is the filtering coefficient to implement derivative action. The algorithm includes three stages as i. evaluation, ii. optimization and iii. testing. In evaluation process, the tracking performance of the system is observed with initial parameters to be optimized. Soon after, the optimization process tunes the respective gains ($K_{p,j}$, $K_{i,j}$ and $K_{d,j}$) to improve the coarse performance of the system. After optimization, in testing process, the system is controlled with newly produced and optimized gains by the system.

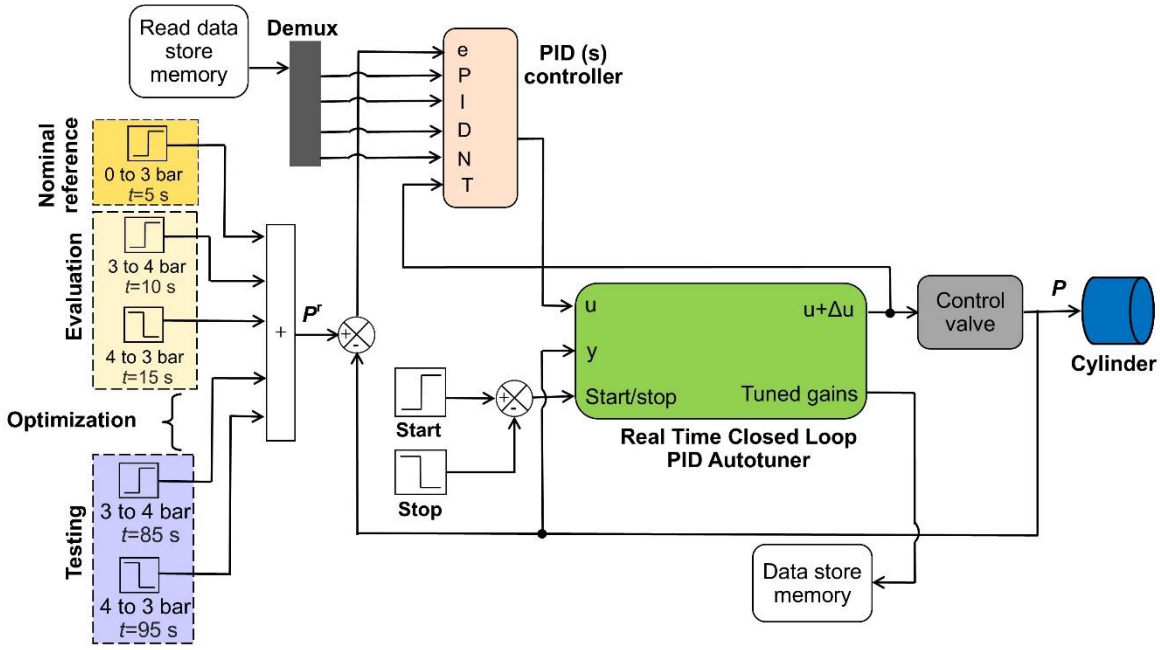

Figure 3.39. Real time closed loop PID auto-tuning schema

$K_{p,j}$, $K_{i,j}$ and $K_{d,j}$ gains are initialized with the values of [0.25, 1.5, 0.0005], respectively. For both of the valves (PRV-a and PRV-b), the derivative coefficient (*N*) and experiment duration (*t*) is set to 100 and 95 seconds, respectively. After the auto-tuning process, the optimum values of [0, 2.034, 0.0098] and [0, 2.042, 0.0107] are obtained for PRV-a and PRV-b, respectively for K_p, K_i and K_d. The sample graphics showing the details of whole process is given in Figure 3.40 (a). The performance of the system with initial optimized gains are presented in Figures 3.40 (b) and (c), respectively. The improvement in tracking performance can easily be detected from the settling time and rise time point of view.

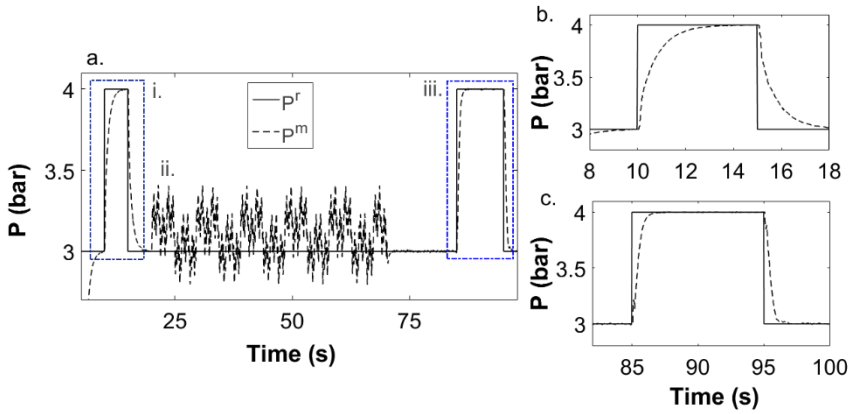

Figure 3.40. Real-time PID auto-tuning results of PRV-a in Simulink/MATLAB (a) Whole tuning process; (b) Performance before tuning; (c) Performance after tuning

The algorithm has been executed many times to get non-zero $K_{p,j}$ but every time zero $K_{p,j}$ is produced. At this point, experimental based manual trial and error methodology has been applied to estimate the best $K_{p,j}$. It has been obtained that that $K_{p,a}$=0.45 and $K_{p,b}$=0.55 were the best values for PRV-a and PRV-b, respectively.

3.7. Design and Application of Cascade Fuzzy Adaptive PID Controller

A fuzzy module essentially consists of three stages: fuzzification, a rule base that establishes the connection between input and output variables, and defuzzification (Özbek and Eker, 2019). The fuzzy rules, based on desired closed-loop control features, aim to optimize the controller's gains across various operating conditions. The proposed fuzzy scheme specifically tailored for interaction torque control. The linguistic variables and fuzzy rules are formulated to capture the complex relationship between input variables, such as force and torque, and output control actions. Fuzzy control is particularly effective in managing uncertainties present in the interaction torque control of pneumatic forearm rehabilitation robots. Uncertainties can arise from variations in patient characteristics, environmental conditions, or even the inherent variability in pneumatic actuation systems. Fuzzy logic provides a framework for representing and handling uncertain information, allowing the control system to adapt and respond appropriately to varying conditions. A fuzzy logic control system is utilized to mimic patient discomfort during rehabilitation treatment for the developed rehabilitation robot.

3.7.1. Block Schema of Cascade Fuzzy Adaptive PID Controller

For the presented study, a new Cascade Fuzzy Adaptive PID technique, wherein human-robot interaction considerations are integrated into the fuzzy control design to ensure safe and intuitive interaction, is addressed by Dağdelen et al., 2023b. Implementing the adaptive intelligent control algorithm into the created rehabilitation system allows the system to respond to changes in the rehabilitation conditions in an efficient and reliable manner.

The designed fuzzy adaptive scheme is represented in Figure 3.41, in which the controller has two inputs (error between reference and measured pressures, $e_{P,j}$ and rate of change of this error, $\dot{e}_{P,j}$) and a single output for each gain (i.e. scheduled PID parameters of $[K_{p,j}]^f$, $[K_{i,j}]^f$ and $[K_{d,j}]^f$, j=a,b).

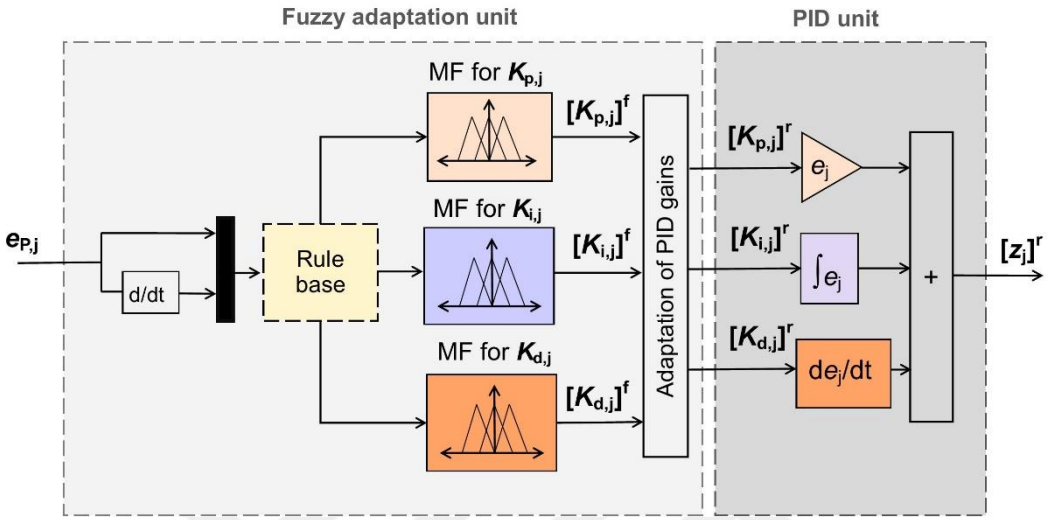

Figure 3.41. Block schema of Cascade Fuzzy Adaptive PID controller

3.7.2. Design of Fuzzy Logic Framework

Design of Fuzzy logic framework includes the specification of fuzzy rules (i.e. rule base), design and construction of input and output membership functions (MF) and finally determination of defuzzification method to derive the fuzzy adapted gains as $[K_{p,j}]^f$, $[K_{i,j}]^f$ and $[K_{d,j}]^f$ to be fed to the regulation unit.

Fuzzy membership functions (MFs) are chosen in accordance with Mamdani inference criteria. With a specific degree of membership, the error ($e_{P,j}$) and the rate of change in error ($\dot{e}_{P,j}$) in reference pressure ($[P^r]^j$) are fuzzified from crisp values to fuzzy sets with linguistic variables. Both inputs and outputs have been defined using seven membership functions (MFs). The MFs of inputs (i.e. $e_{P,j}$ and $\dot{e}_{P,j}$) are all triangular type and labeled as Negative Small (NS), Negative Medium (NM), Negative (N), Zero (Z), Positive (P), Positive Medium (PM) and Positive Big (PB). It should be noted that triangular and trapezoidal membership functions enable faster information processing compared to Gaussian, sigmoidal, or generalized bell membership functions. The MFs for input variables are shown in Figure 3.42 for the error ($e_{P,j}$) and rate of change in error ($\dot{e}_{P,j}$), respectively. The output MFs for each PID gain are shown separately in Figure 3.43. Five MFs have been utilized for each fuzzy adapted gain labeled as Zero (Z), Positive Very Low (PVL), Positive (P), Positive Medium (PM), Positive Big (PB).

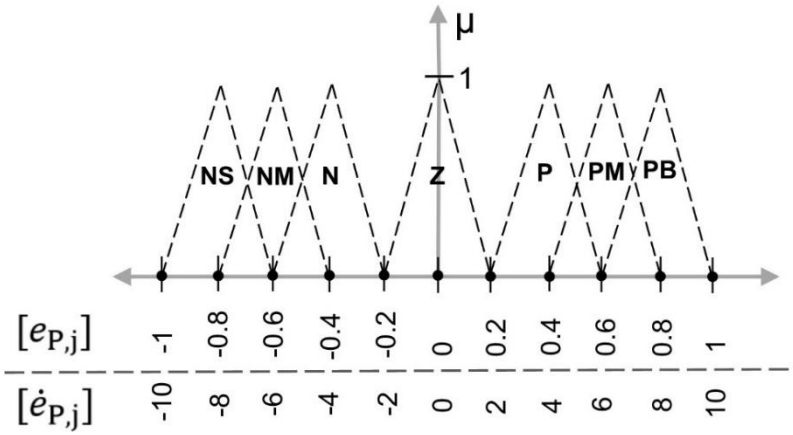

Figure 3.42. Input MFs for (a) e_p; (b) $\dot{e}_p$

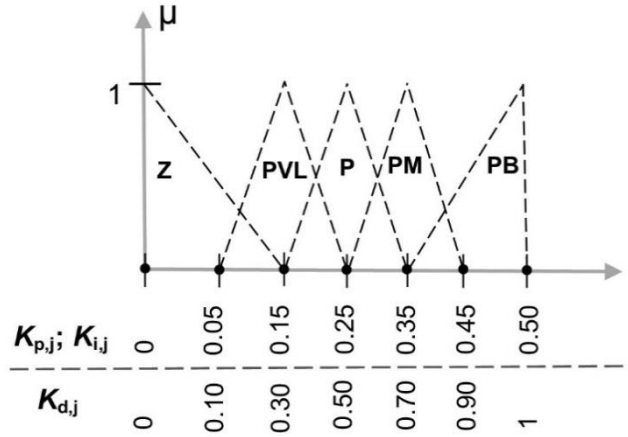

Figure 3.43. Output MFs for (a) $K_{p,j}$; (b) $K_{i,j}$ and (c) $K_{d,j}$

49 fuzzy rules have been obtained based on the expert information in Tables 3.8, 3.9 and 3.10 for $K_{p,j}$, $K_{i,j}$ and $K_{d,j}$ gains, respectively by considering the following criteria:

- While $e_{P,j}$ and $\dot{e}_{P,j}$ are in positively or negatively increasing manner, the values of PID gains should be selected at high levels to reach the reference as quick as possible.
- If the $e_{P,j}$ and $\dot{e}_{P,j}$ are around zero, namely the controller is close to the reference, the PID gains are stabilized, and no-sharp actions are permitted by the controller. To do it, generally Z or PVL membership functions are chosen at these states.
- At the condition of high $K_{p,j}$, the level of $K_{i,j}$ is selected at low values for fast convergence. In addition to that, $K_{d,j}$ gain is increased for only high $K_{i,j}$ levels. For low or moderate $K_{i,j}$ gains, $K_{d,j}$ are selected from Z or PVL MFs.

Based on the specified criteria, the fuzzy control surfaces of outputs are shown in Figures 3.44, 3.45 and 3.46 for $[K_{p,j}]^f$, $[K_{i,j}]^f$ and $[K_{i,j}]^f$, respectively.

Table 3.8. Fuzzy rules and fuzzy control surface for $[K_{p,j}]^f$

		$e_{P,j}$						
		NS	**NM**	**N**	**Z**	**P**	**PM**	**PB**
$\dot{e}_{P,j}$	**NS**	PB	PB	PM	P	PVL	PVL	Z
	NM	PB	P	P	PVL	PVL	Z	Z
	N	P	P	PVL	PVL	PVL	Z	Z
	Z	PVL	PVL	PVL	PVL	PVL	P	PM
	P	PVL	PVL	Z	PVL	P	PM	PB
	PM	Z	Z	Z	PVL	PM	PB	PB
	PB	Z	Z	Z	PVL	PB	PB	PB

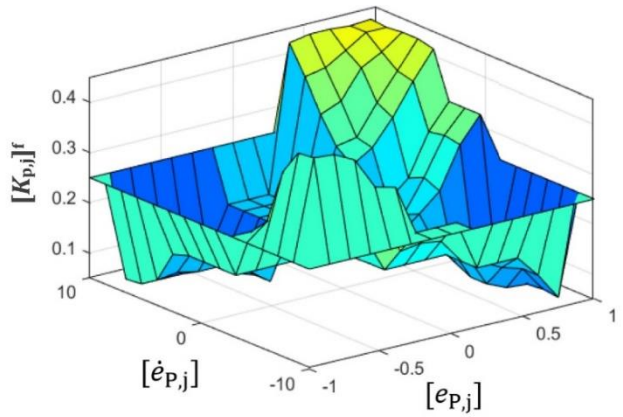

Figure 3.44. Control surface for $[K_{p,j}]^f$ based on defined rules on Table 3.6

Table 3.9. Fuzzy rules and fuzzy control surface for $[K_{i,j}]^f$

		$e_{P,j}$						
		NS	**NM**	**N**	**Z**	**P**	**PM**	**PB**
$\dot{e}_{P,j}$	**NS**	P	P	PM	P	PVL	PVL	Z
	NM	PVL	PVL	P	PVL	PVL	Z	Z
	N	P	P	Z	Z	Z	Z	Z
	Z	PVL	PVL	Z	Z	Z	P	PM
	P	PVL	PVL	Z	Z	Z	PM	PB
	PM	Z	Z	Z	PVL	PM	PVL	PVL
	PB	Z	Z	Z	PVL	PB	PVL	PVL

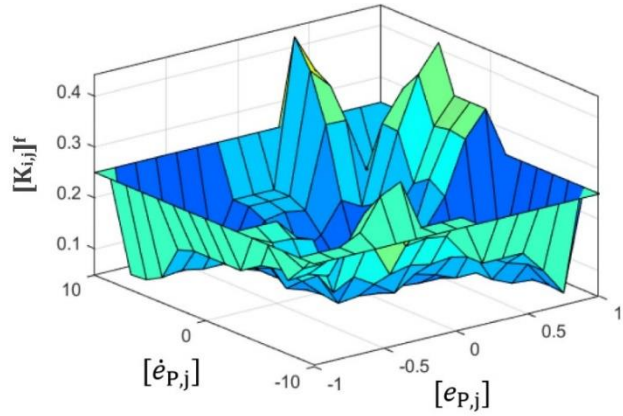

Figure 3.45. Control surface for $[K_{i,j}]^f$ based on defined rules on Table 3.7

Table 3.10. Fuzzy rules and fuzzy control surface for $[K_{d,j}]^f$

		$e_{P,j}$						
		NS	**NM**	**N**	**Z**	**P**	**PM**	**PB**
$\dot{e}_{P,j}$	**NS**	PVL	PVL	PB	PVL	Z	Z	Z
	NM	Z	Z	PVL	Z	Z	Z	Z
	N	PVL	PVL	Z	Z	Z	Z	Z
	Z	Z	Z	Z	Z	Z	PVL	PB
	P	Z	Z	Z	Z	Z	PB	PB
	PM	Z	Z	Z	Z	PB	Z	Z
	PB	Z	Z	Z	Z	PB	Z	Z

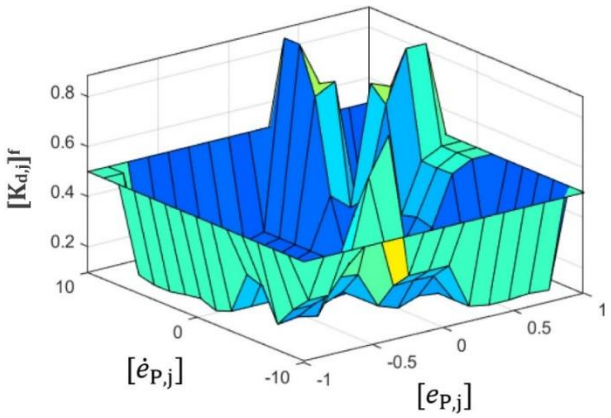

Figure 3.46. Control surface for $[K_{d,j}]^f$ based on defined rules on Table 3.8

The center of gravity approach is employed to translate the fuzzy output variables to the non-fuzzy parameters as defuzzification method by applying Equation 3.27 for each scheduled PID gain:

$$[K_{p,j}]^f, [K_{i,j}]^f, [K_{d,j}]^f = \frac{\sum_{k=1}^{5} x_k \cdot \mu(x_k)}{\sum_{k=1}^{5} \mu(x_k)}, j = \text{a, b} \tag{3.27}$$

In Equation 3.27, k=1,2,…5 defines the index for membership functions. Further, x_k and $\mu(x_k)$ are the corresponding value and the degree of the respective MF. Each scheduled gain ($[K_{p,j}]^f$, $[K_{i,j}]^f$ and $[K_{d,j}]^f$, j=a,b) is then sent to the regulation block and Equations 3.28-30 are applied.

$$[K_{p,j}]^r = \left([K_{p,j}]_{max} - [K_{p,j}]_{min}\right) \cdot [K_{p,j}]^f + [K_{p,j}]_{min}, j = \text{a, b} \tag{3.28}$$

$$[K_{i,j}]^r = \left([K_{i,j}]_{max} - [K_{i,j}]_{min}\right) \cdot [K_{i,j}]^f + [K_{i,j}]_{min}, j = \text{a, b} \tag{3.29}$$

$$[K_{d,j}]^r = \left([K_{d,j}]_{max} - [K_{d,j}]_{min}\right) \cdot [K_{d,j}]^f + [K_{d,j}]_{min}, j = \text{a, b} \tag{3.30}$$

3.7.3. Identification of Regulation Parameters

The values of regulation parameters have been determined with excessive experimental methods during the construction of the fuzzy rules. To reach good controller performances, different values for regulation parameters are used for the controller valves, PRV-a and PRV-b to more accurate controller outputs. As a result of the experimental work, the following values have been obtained for regulation parameters as in Table 3.11.

Table 3.11. Adaptation parameters for PRV-a and PRV-b

Maximum Adaptation Parameters	**Values**	**Minimum Adaptation Parameters**	**Values**
$[K_{p,a}]_{max}$	0.760	$[K_{p,a}]_{min}$	0.005
$[K_{p,b}]_{max}$	0.600	$[K_{p,b}]_{min}$	0.005
$[K_{i,a}]_{max}$	2.030	$[K_{i,a}]_{min}$	1.850
$[K_{i,b}]_{max}$	2.040	$[K_{i,b}]_{min}$	1.750
$[K_{d,a}]_{max}$	0.01	$[K_{d,a}]_{min}$	0
$[K_{d,b}]_{max}$	0.0108	$[K_{d,b}]_{min}$	0

After the necessary regulation operation is conducted, the final gains $[K_{p,j}]^r$, $[K_{i,j}]^r$, and $[K_{d,j}]^r$ are sent to PID operation and the raw control signal, $[z_j]^r$ (i.e. $[z_a]^r$ and $[z_b]^r$ for the PRV-a and PRV-b) is produced.

4. RESULTS AND DISCUSSIONS

4.1. Experimental Results of Proportional Directional Control Valve (DCV)

In this section, the modelling results for proportional directional control valve will be presented. To provide a clear understanding of the results, in the first subsection, cracking pressure ratio (CPR) identification results of DCV will be covered. Then, identification of proposed new compressible flow model's parameters will be given including curve fitting operations based on real (i.e. experimental) data. Following this subsection, the verification results that gathered from real-time hardware-in-the-loop environment will be presented.

4.1.1. The Results for Cracking Pressure Ratio (CPR) Identification

The experimental methods indicated in Section 3.2.3 have been successfully automated using MATLAB/Stateflow tools, and the CPR identification have been achieved under different operating conditions (i.e. changing u and P_s) (Dağdelen and Sarıgeçili, 2021). Only valve port 2 has been tested in both the filling and exhausting phases. The valve control signal has been investigated using Δu=0.1 V signal steps. Furthermore, supply pressure range $2 \leq P_s \leq 7$ bar with ΔP_s=1 bar pressure steps has been examined.

The graphs illustrating the variations in P_a and $\dot{P}_a$ during testing at a supply pressure of P_s=5 bar absolute are displayed in Figure 4.1. The filling state is depicted in Figures 4.1 (a) and (b), while the exhausting state is shown in Figures 4.1 (c) and (d). From the presented graphs, it is quite apparent that when the cylinder pressure (P_a) remains constant and the pressure change approaches zero ($\dot{P}_a = 0$), the compressed air is entirely released to the atmosphere by activating the appropriate control signals in the valves (P_a=P_{atm}). Subsequently, by implementing a specified signal increment (Δu), a different control signal (u_{n+1}) is examined to map the CPR. Likewise, during the exhausting phase, once the air pressure decreases to a specific level and stabilizes ($\dot{P}_a = 0$), the air pressure is elevated to the tested supply pressure (P_s) to test the subsequent signal (u_{n+1}). The testing steps are conducted in a predefined sequence.

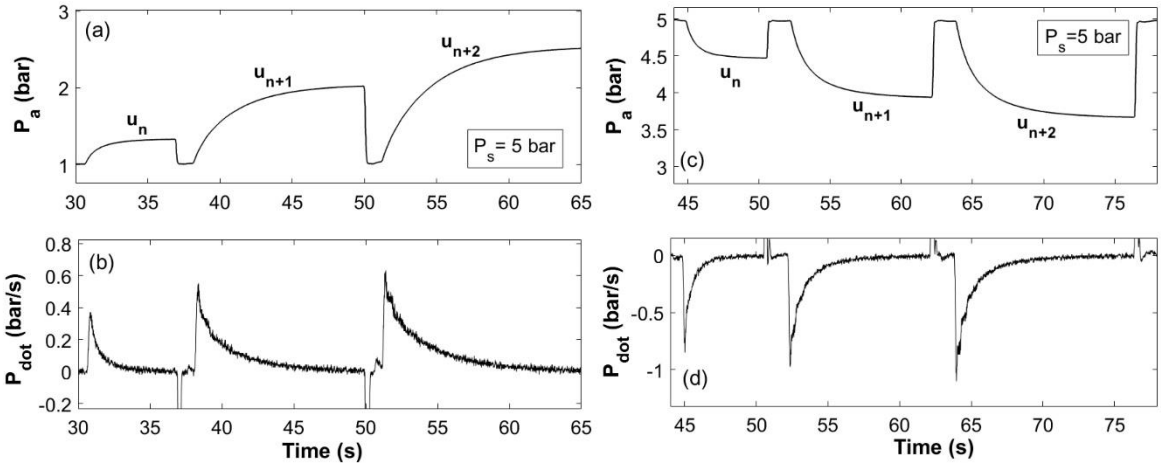

Figure 4.1. P_a and $\dot{P}_a$ curves for port 2 for (a) & (b) filling state; (c) & (d) exhausting state

When all obtained experimental data (P_a and $\dot{P}_a$) are evaluated, the CPR (a) curves for both filling and exhausting states can be constructed for all tested operating conditions (u and P_s) as shown in Figures 4.2. (a) and (b), respectively. It is verified that the CPR of port 2 remains constant around 1 between $0 \leq u \leq 4.5$ V in filling state. However, it exhibits excessively non-linear behavior over a wide and significant signal range of $4.5 \leq u \leq 5.7$ V and takes values ranging from $0.145 \leq a \leq 1$ depending on the operating conditions (u and P_s). When the exhausting state of the tested port is examined, the CPR takes values around 1 in $5.9 \leq u \leq 10$ V signal range. Similarly, in the signal range of $4.7 \leq u \leq 5.9$ V, the CPR takes values in the range of $0.145 \leq a \leq 1$. For these reasons, the regions where the CPR exhibits non-linearity are shown only in detail in Figures 4.2. (a) and (b). Based on these results, the $[\Delta u_{p2}]^f$ and $[\Delta u_{p2}]^e$ values of the relevant valve are determined as 0.7 V and 0.3 V, respectively.

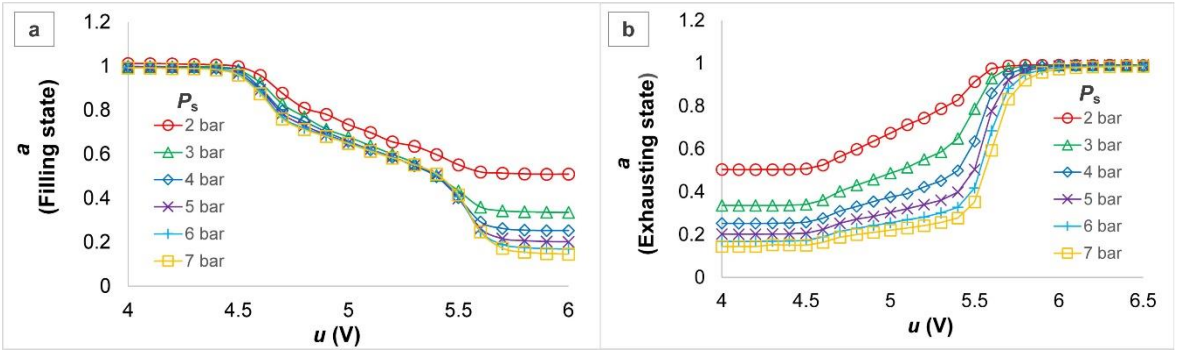

Figure 4.2. The CPR curves at different operating conditions (u and P_s) for (a) filling state; (b) exhausting state

In the following subsection, the identification results of new compressible flow model's parameters will be covered.

4.1.2. The Identification Results of New Model's Parameters (α_1, α_2 and ε)

The identification of model parameters is carried out using the equation specifying the chamber pressure (P_a) and its variation with time ($\dot{P}_a$). In the case of a constant volume chamber ($\dot{V}_a = 0$), the relationship between mass flow rate and pressure change can be expressed as Equation 4.1.

$$\dot{m}_m = \frac{\dot{P}_a}{k \cdot R \cdot T_a} \cdot V_a \tag{4.1}$$

In Equation 4.1, k, R and T_a denote the specific heat constant, the ideal gas constant and the temperature of air, respectively. V_a corresponds to the chamber volume of the cylinder. According to the specified operating conditions (u and P_s), the pressure P_a in the chamber "a" is measured instantly from the pressure sensor, PS-a. After the differentiation of the pressure signal, the pressure change rate $\dot{P}_a$ signal is generated. However, since the existing signal noise will be increased after the differentiation process, the pressure change rate is calculated after the appropriate filtering process. At this step, Moving Average Filter (MAF) is preferred. Obtained $\dot{P}_a$ signal is then used in conjunction with Equation 4.1 to calculate the measured mass flow rate ($\dot{m}_m$). The data obtained at this stage (P_a, $\dot{m}_m$) are recorded for using in non-linear curve fitting operations of Equation 3.10 and 3.11. For the implementation of these processes, the experimental setup in Figure 3.10 (b) is used.

The non-linear least squares approach are utilized for curve fitting tasks. The parameters α_1, α_2 and ε are solved with initial guess values of [0.00025, 0.00025, 0.75] and [-0.00025, -0.00025, 0.75] for filling and exhausting cases, respectively.

Model parameters are identified especially for unstable ranges of the valve ($4.5 \leq u \leq 5.7$ V for filling and $4.5 \leq u \leq 5.9$ V for exhausting) and the results are presented in Table 4.1. In addition, non-linear curve fitting operations are shown as samples in Figures 4.3 and 4.4. The variation in the behavior of the tested valve at each control signal is clearly evident in these sample graphs.

Table 4.1. Identified model parameters after curve fitting

		P_s=5 bar					P_s=5 bar			
Port 2	**u (V)**	**α_1**	**α_2**	**ε**	**Port 2**	**u (V)**	**α_1**	**α_2**	**ε**
Filling state	**4.5**	6.920e-9	3.020e-8	0.500	**Exhausting state**	**4.7**	9.43e-07	-4.05e-07	1.735
	4.6	2.595e-9	1.089e-8	0.700		**4.8**	4.57e-07	-1.68e-07	1.344
	4.7	1.528e-9	6.022e-9	0.750		**4.9**	7.340e-7	-2.270e-7	1.550
	4.8	1.501e-11	5.634e-9	0.592		**5**	7.870e-7	-6.710e-8	1.484
	4.9	1.316e-11	4.798e-9	0.654		**5.1**	6.060e-8	-5.760e-8	1.648
	5.0	9.062e-10	4.266e-9	0.816		**5.2**	5.050e-8	-4.390e-8	1.646
	5.1	1.045e-9	3.968e-9	0.950		**5.3**	8.330e-8	-4.190e-8	1.462
	5.2	1.056e-9	4.140e-9	1.04		**5.4**	7.440e-8	-3.390e-8	1.649
	5.3	1.016e-9	4.007e-9	0.900		**5.5**	5.440e-8	-2.060e-8	1.596
	5.4	1.127e-9	4.784e-9	1.288		**5.6**	5.010e-8	-2.020e-8	0.932
	5.5	1.076e-9	4.59e-9	1.020		**5.7**	2.820e-8	-2.010e-8	1.010
	5.6	-				**5.8**	9.910e-8	-4.170e-8	1.126
	5.7	-				**5.9**	8.830e-8	-5.600e-8	1.016

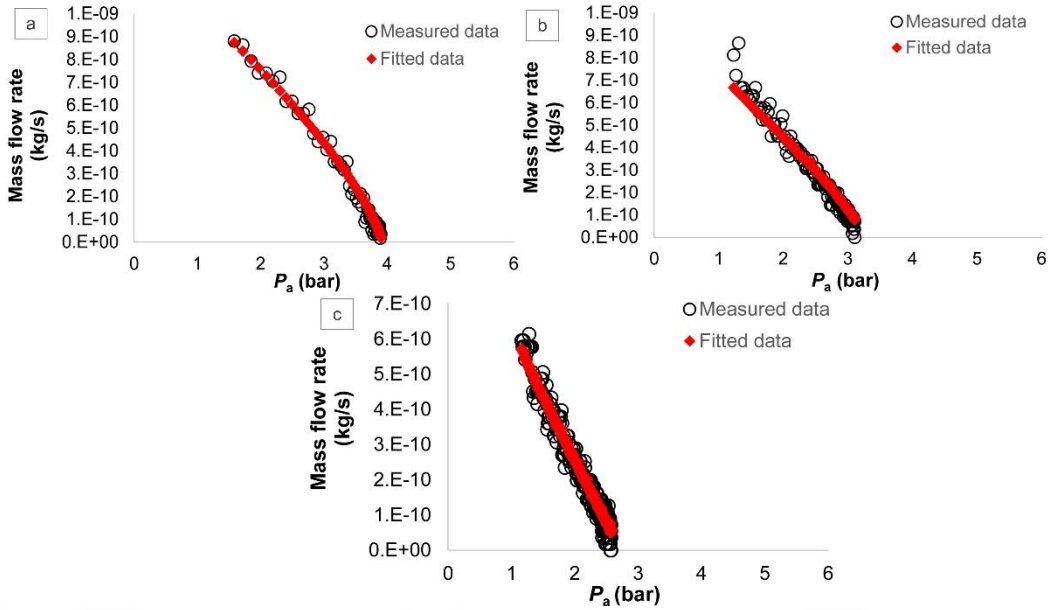

Figure 4.3. Curve fitting plots on different signals (Filling state); (a) u=4.7 V; (b) u=5.0 V; (c) u=5.3 V; For all cases P_s=5 bar absolute

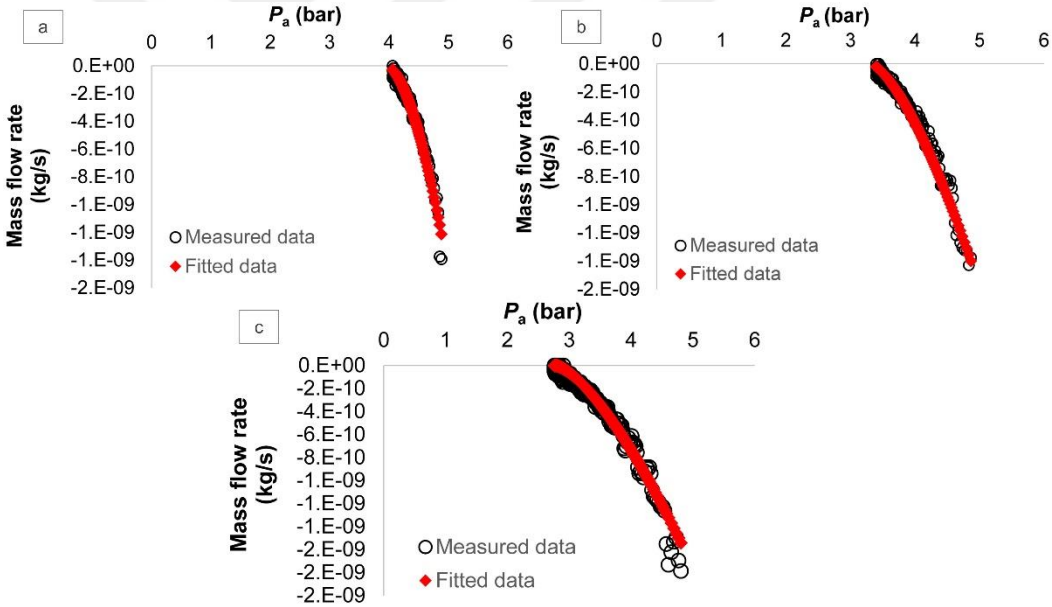

Figure 4.4. Curve fitting plots on different signals (Exhausting state); (a) u=4.7 V; (b) u=5.0 V; (c) u=5.3 V; For all cases P_s=5 bar absolute

Model parameters identified at P_s=5 bar have also been evaluated under different P_s (in the range of 2-7 bar absolute) conditions to check the mass flow rate characteristics. It was observed that supply pressure drastically affected the flow characteristics. Therefore, $[k(P_s)]^f$ and $[k(P_s)]^e$ empirical coefficients have been developed to compensate the effect of the supply pressure (P_s) to the produced theoretical mass flow rate ($\dot{m}_m$). The coefficients are shown in Equations 4.2 and 4.3. As it is clear, $[k(P_s)]^f$ is a third order polynomial and $[k(P_s)]^e$ is the fourth order polynomial functions which have R^2=1 fit quality.

$$[k(P_s)]^f = -0.0106 \cdot {P_s}^3 + 0.153 \cdot {P_s}^2 - 0.749 \cdot P_s + 2.231 \quad (4.2)$$

$$[k(P_s)]^e = -0.0437 \cdot {P_s}^4 + 0.879 \cdot {P_s}^3 - 6.381 \cdot {P_s}^2 + 19.496 \cdot P_s - 19.5 \quad (4.3)$$

4.1.3. Validation Tests and Results of New Flow Model

Figure 4.5 depicts the model validation block schema built in MATLAB/Simulink. It should be mentioned that validation testing is carried out in the valve's unstable region under randomly chosen u and P_s operating conditions. For both the filling and exhausting states, the CPR (a) values have been integrated in the simulation model as a 2-D Lookup tables. Furthermore, 1-D Lookup tables have been employed for each model parameter (α_1, α_2 and ε) separately by utilizing the results obtained as in Table 4.1. For filling and exhaustion scenarios, the solver type has been set to fixed-step, and validation data has been gathered with a sample time of 0.01 s. The steady-state conditions are reached at t=10 s, hence only 15 s has been monitored in the following resultant figures.

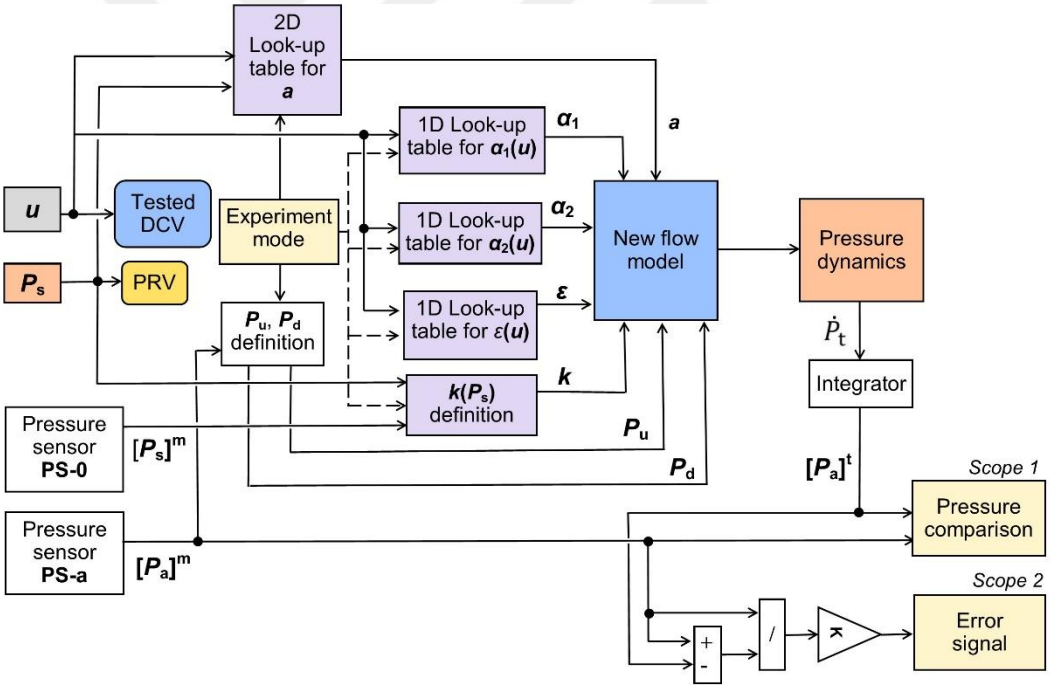

Figure 4.5. MATLAB/Simulink validation model

To carry out validation tests, the experiment mode (filling or exhausting) is initially specified, followed by the definition of the valve control signal (u) and supply pressure (P_s) values to be tested in the model. The CPR (a) value, derived from the 2-D lookup table, and the model parameters (α_1, α_2, and ε) read independently from the 1-D lookup tables, are fed into the "New flow model" block. In addition to these parameters, depending on the experiment mode, the $k(P_s)$ is input into the "New flow model" block using Equation 4.2 or Equation 4.3. The experimentally

assigned upstream and downstream pressure values (P_u, P_d) are likewise quickly supplied into the "New flow model" block, and the theoretical mass flow rate ($\dot{m}_p$) is generated using Equations 3.10 and 3.11.

The signal $\dot{m}_p$ generated from the "New flow model" block is directed to the "Pressure dynamics" block, where the theoretical pressure change rate ($\dot{P}_t$) is calculated using Equation 4.1. Subsequently, the integral of the pressure change rate signal is computed to determine the absolute theoretical pressure value ($[P_a]^t$). This derived value is then compared against the instantaneous chamber pressure ($[P_a]^m$) measured directly using the pressure sensor (PS-a). This comparison serves to evaluate the efficacy of the new model, assessing both its pressure tracking performance and the percentage error between $[P_a]^m$ and $[P_a]^t$.

The control signals of u=4.7 V, u=5.0 V, and u=5.3 V are randomly selected for filling state, since the performance of the new flow model is tested at unstable region of the valve. To evaluate the model's performance at low, moderate, and high pressure levels, P_s=3, 5 and 6 bar absolute supply pressure levels have been evaluated for all specified control signals. The results of u=4.7 V, u=5.0 V, and u=5.3 V are shown in Figures 4.6, 4.7, and 4.8, respectively. In this validation scenario, it is guaranteed that the cylinder chamber is empty at the beginning of the experiments, i.e. the pressure inside the chamber is the atmospheric pressure. (P_a=P_{atm}=1 bar absolute at t=0 s). Then, a certain test signal is sent to the DCV and the measured pressure and theoretical pressure resulted by the model are traced at the same scope. The validation tests are allowed to run until steady state condition in the chamber pressure, P_a is achieved. As it is clearly evident from figures, the new model produced very successful tracking performance nearly for all tested operating conditions in filling state. Among them, the superior performance of the model has been obtained at u=5.3 V (P_s=6 bar absolute) with 0.0039% and the worst performance has been observed in u=5.3 V (3 bar) with 8.755% absolute error.

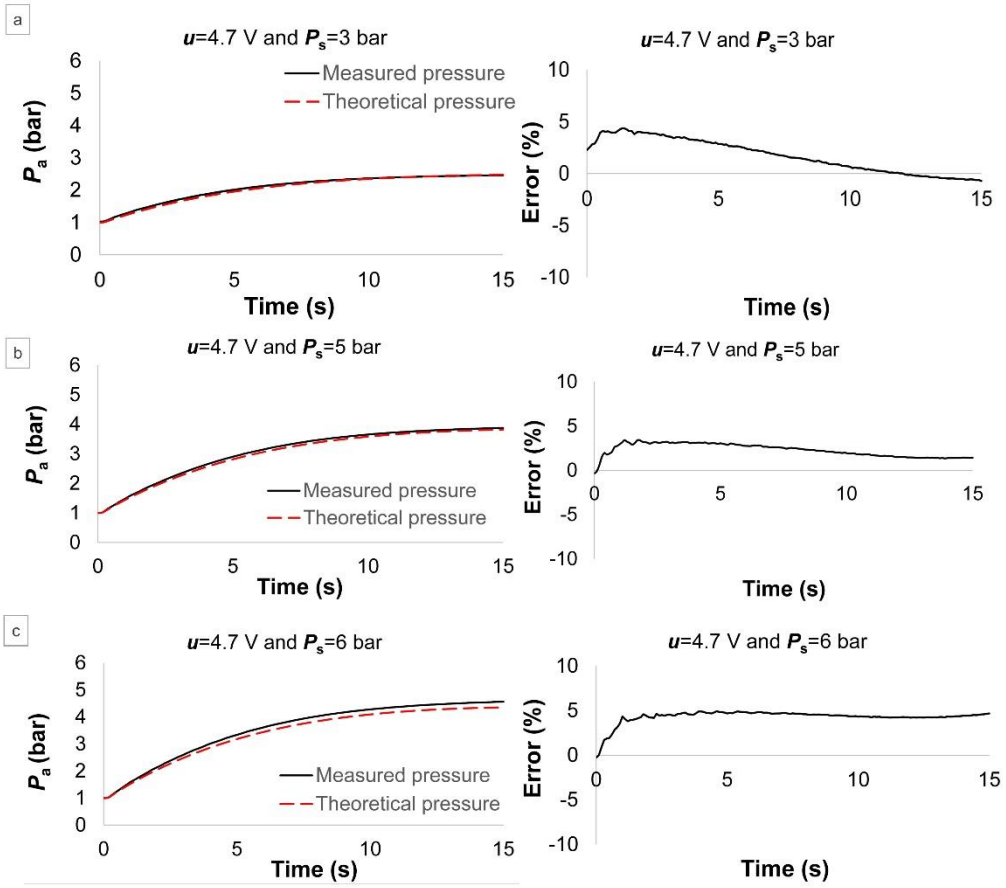

Figure 4.6. Model validation at u=4.7 V signal at different P_s conditions (Filling state); (a) P_s=3 bar; (b) P_s=5 bar; (c) P_s=6 bar

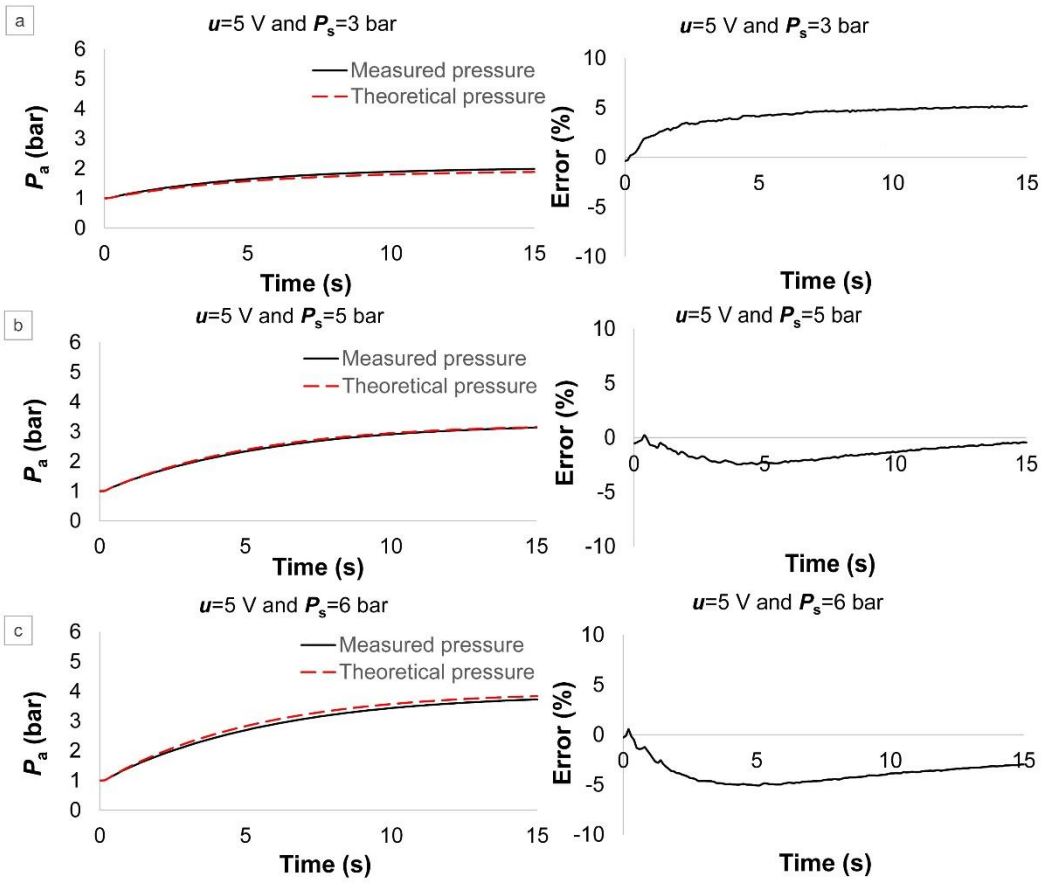

Figure 4.7. Model validation at u=5.0 V signal at different P_s conditions (Filling state); (a) P_s=3 bar; (b) P_s=5 bar; (c) P_s=6 bar

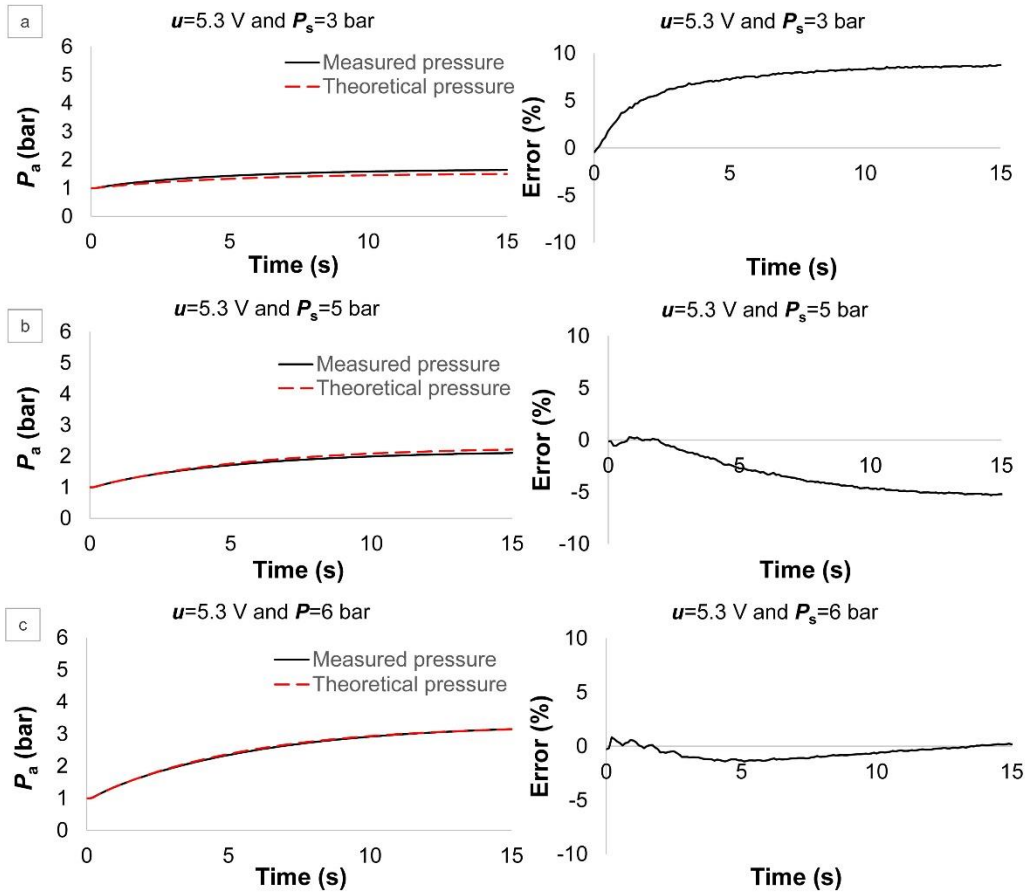

Figure 4.8. Model validation at u=5.3 V signal at different P_s conditions (Filling state) (a) P_s=3 bar; (b) P_s=5 bar; (c) P_s=6 bar

Similar to the filling state, to evaluate the performance of exhausting state of the valve at small orifice openings, the control signals of u=4.7 V, u=5.0 V, and u=5.3 V have been tested. All these signals have been observed in P_s=3, 5 and 6 bar absolute supply pressure levels. In this context, a desired pressure, P_s is fed to the chamber and then the pressurized air is exhausted with the tested control signal (u), thus the figures differ slightly from the filling state. This is achieved by fully opening the DCV to the chamber (u=0 V) and equalizing the pressure level to the desired test supply pressure level (P_a=P_s) by sending z= P_s V signal to PRV. Then, demanded test signal (u) is sent to DCV for exhausting operation. In this case, firstly a rapid-increase and then a slow-decrease at pressure, P_a is observed. Based on this observation, the performance at exhausting state is evaluated in only slow-decrease pressure by comparing measured pressure and theoretical pressure resulted by new model. The results of tested signals are presented in Figures 4.9, 4.10 and 4.11. As it is clear from figures, there is a great agreement between the pressure data. The percentage error values are around 5% nearly for all validations. Among them, superior performance has been obtained at u=4.7 V (5 bar absolute) operating condition with 0.017% absolute error.

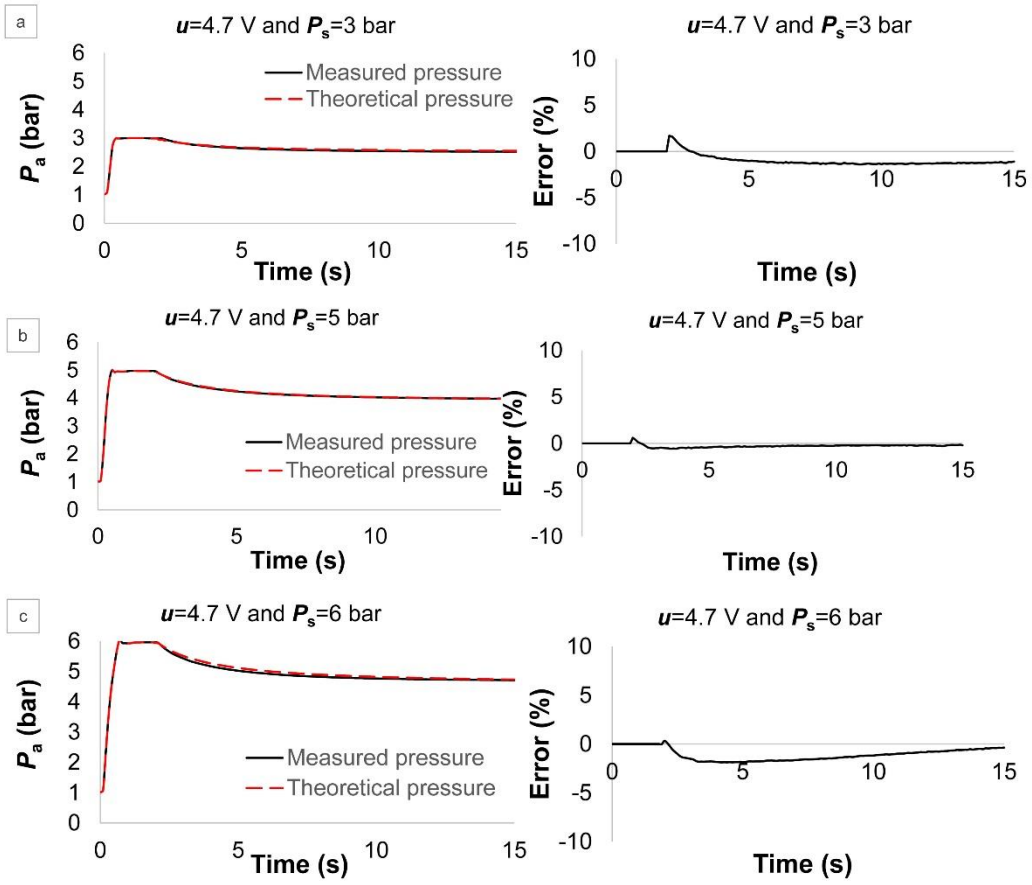

Figure 4.9. Model validation at u=4.7 V signal at different P_s conditions (Exhausting state); (a) P_s=3 bar; (b) P_s=5 bar; (c) P_s=6 bar

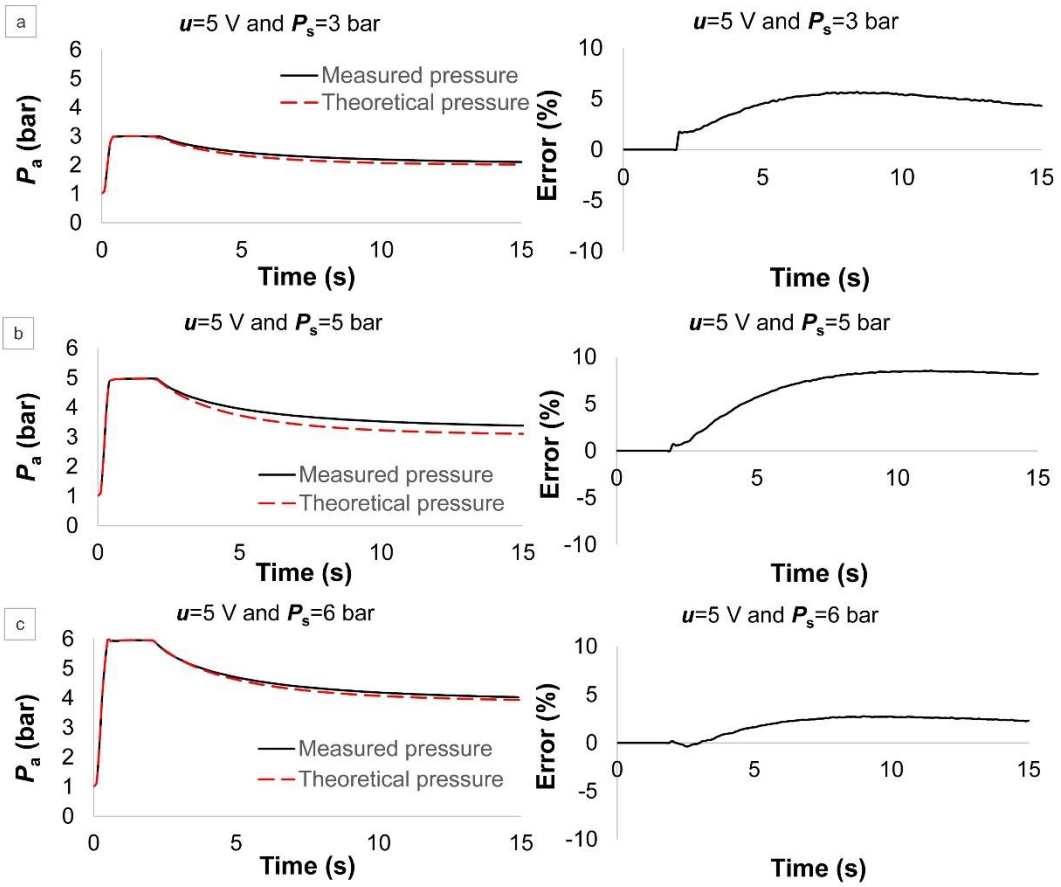

Figure 4.10. Model validation at u=5.0 V signal at different P_s conditions (Exhausting state); (a) P_s=3 bar; (b) P_s=5 bar; (c) P_s=6 bar

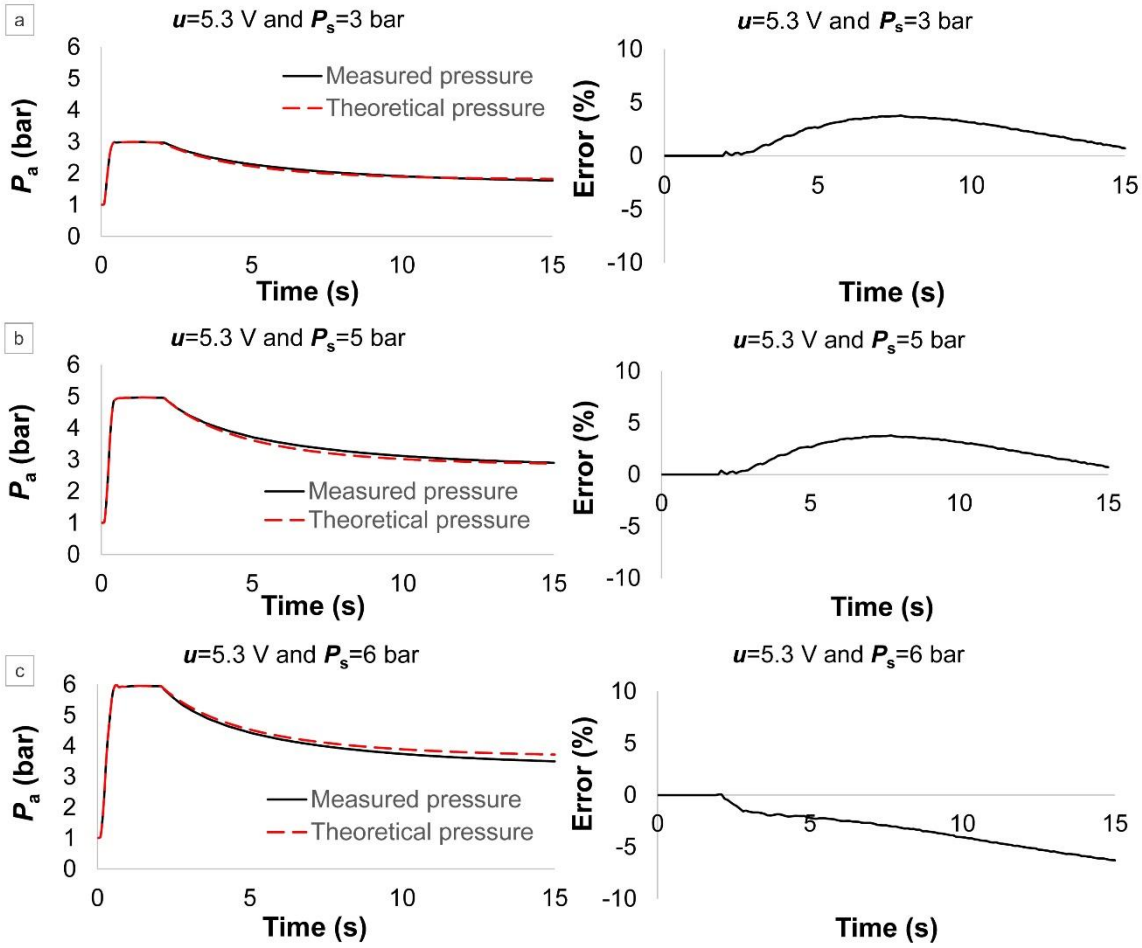

Figure 4.11. Model validation at u=5.3 V signal at different P_s conditions (Exhausting state); (a) P_s=3 bar; (b) P_s=5 bar; (c) P_s=6 bar

4.2. Experimental Results for Friction Torque and Force Identification for Vane Type Semi-Rotary Cylinder and Double Acting Linear Pneumatic Cylinder

In this section, the experimental results of identification algorithms of friction presented in Section 3.2.4 will be presented for vane type semi-rotary and double acting linear pneumatic cylinders. The results cover the parameter identifications, constructed friction curves based on identified parameters and finally, the real time hardware-in-the-loop validation results at specified operating conditions. At following first subsection, the results for vane type semi-rotary cylinder will be presented. And then, the results of double-acting linear pneumatic cylinder will be given.

4.2.1. The Results of Friction Identification Test of Semi-Rotary Pneumatic Cylinder

The proposed automated approach was implemented to test utilized semi-rotary pneumatic cylinder in robotic setup in real-time environment at varied working pressures ranging from P_w=1 to 6 bar by Dağdelen and Sarıgeçili (2023a). The number of static and dynamic experiments (N_s and N_d) was set to 100. During the identification step, the sampling frequency (f_s) in the MATLAB/Simulink® model was set to 500 Hz. Table 4.2 summarizes the identified friction torque parameters with respect to the proposed identification algorithm presented in Section 3.2.4. The

predicted friction torque surfaces with respect to angular speed and working pressure were also drawn, as shown in Figures 4.12 (a) and (b) for CCW and CW directions, respectively. Friction torque (T_f) characteristics of the semi-rotary cylinder at any working pressure (P_w) and operating speed (ω) can be simply driven from the friction torque surfaces.

Table 4.2. Estimated friction parameters at various working pressure levels

		Working Pressure, P_w (bar absolute)					
Direction	**Parameters**	**1**	**2**	**3**	**4**	**5**	**6**
CCW	$[T_s]^{CCW}$ (Nm)	1.196	1.223	1.296	1.334	1.362	1.320
	$[T_c]^{CCW}$ (Nm)	0.528	0.619	0.627	0.610	0.592	0.571
	$[\omega_s]^{CCW}$ (rad/s)	0.537	0.546	0.924	0.774	0.869	0.974
	$[B]^{CCW}$ (Nsm/rad)	0.097	0.099	0.150	0.163	0.167	0.186
CW	$[T_s]^{CW}$ (Nm)	-1.199	-1.221	-1.316	-1.400	-1.410	-1.370
	$[T_c]^{CW}$ (Nm)	-0.654	-0.702	-0.703	-0.634	-0.660	-0.686
	$[\omega_s]^{CW}$ (rad/s)	-0.550	-0.351	-1.000	-0.366	-1.000	-1.000
	$[B]^{CW}$ (Nsm/rad)	0.085	0.089	0.123	0.135	0.153	0.165

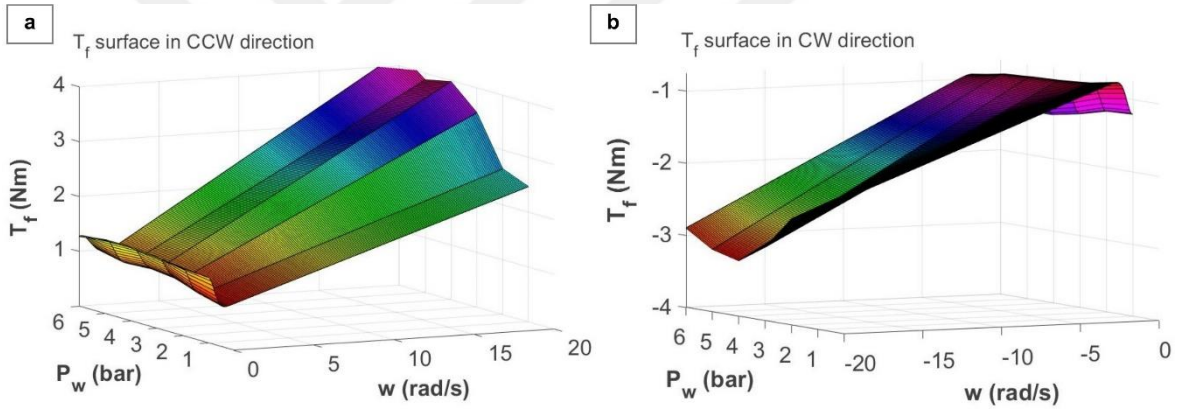

Figure 4.12. Friction torque (T_f) surfaces with respect to w and P_w; for (a) CCW; (b) CW direction

The accuracy of identified friction parameters in the proposed study has been confirmed through real-time validation tests employing both open-loop and closed-loop control approaches. Similar to the identification experiments, a sampling frequency (f_s) of 500 Hz has been consistently maintained across all validation experiments. These tests have been carried out on the robotic setup. Further elaboration and specifics regarding the validation tests can be found in the subsequent subsections.

Open-Loop Loading-Free Tests

During open-loop tests, the cylinder undergoes examination without any load or interaction torque (i.e., T_i = 0 Nm), and there's no assessment of tracking performance for any parameter. Rather, the focus lies on observing whether the simulation model, integrated with identified friction parameters, aligns the relevant chamber pressure with the actual pressure measured directly via

sensors from the chambers. For open-loop validation tests, a direct control methodology is employed by applying control signals directly to the input of the control valve (i.e., utilizing the default controller built into the control valve, as designed by the manufacturer).

To conduct the open-loop tests, both chambers of the cylinder are pressurized at a designated working pressure. Subsequently, the pressure corresponding to a particular chamber is incrementally raised to a randomly specified value, determined by the rotation direction. The recorded data includes measurements of pressure and angular speed. This data is then compared to the calculated chamber pressure value using Equation 3.12 to derive the specific constant angular speed associated with the estimated friction torque parameters.

To assess the performance of the estimated friction parameters across various working pressures, the cylinder underwent testing at P_w levels of 1, 5.4, and 6 bar. Each specified pressure level was tested at different operating speeds (ω) in both counterclockwise (CCW) and clockwise (CW) directions. The results of the rotation at ω =5 rad/s in the CCW direction were considered for the three specified working pressures. The comparison results between the measured pressure ($[P_a]^{real}$) and theoretical pressure ($[P_a]^t$) together with respective percentage error are presented in Figures 4.13, 4.14, and 4.15.

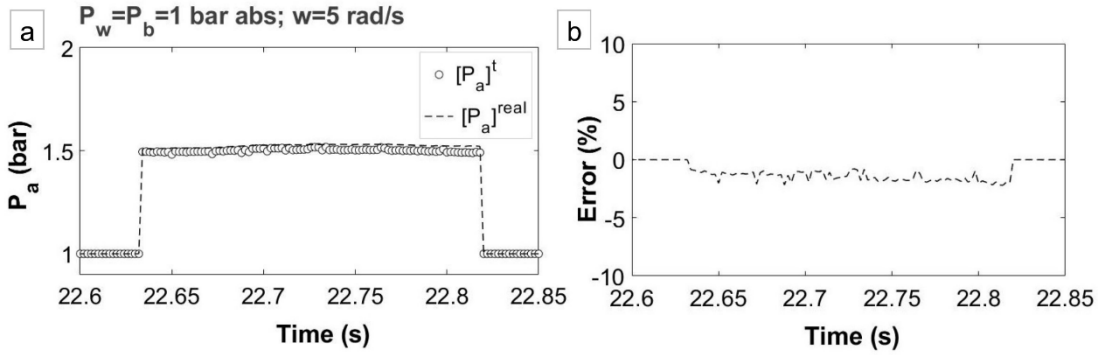

Figure 4.13. Open-loop performance for P_w=1 bar and w=5 rad/s; (a) Comparison of real and theoretical P_a; (b) Error (%)

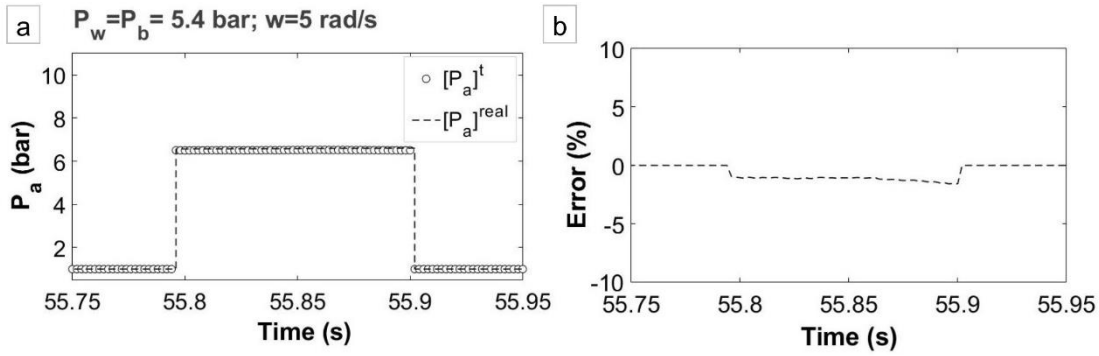

Figure 4.14. Open-loop performance for P_w=5.4 bar and w=5 rad/s; (a) Comparison of real and theoretical P_a; (b) Error (%)

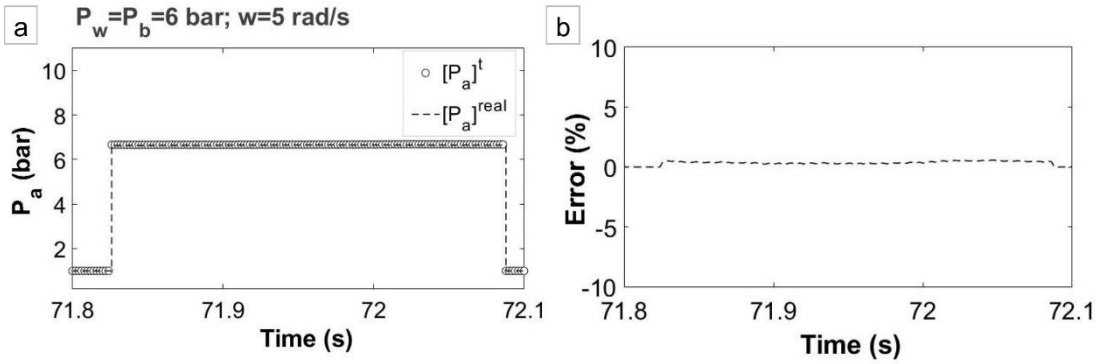

Figure 4.15. Open-loop performance for P_w=6 bar and w=5 rad/s; (a) Comparison of real and theoretical P_a; (b) Percent error

Closed-Loop Under-Loading Tests

The closed-loop experiments have been performed under loaded conditions to validate the effectiveness of the identified friction parameters in generating appropriate control responses. In this control methodology, achieving a stable external (or interaction) torque ($[T_i]^t$) at the end-effector involves regulating the chamber pressures (P_a and P_b). To simulate a load on the end-effector, a healthy human operator attaches their arm to the end-effector and manipulates it to exert a desired external torque, either CCW or CW direction, following these four steps:

- The user has contacted her/his arm to the end-effector along the z-axis, ensuring that both the end-effector and the user's hand are in their neutral positions, as depicted in Figure 4.16 (a) (i.e., the end-effector is perpendicular to the ground). The global axis is defined as x_0, y_0, z_0, while the axis of human operator is labeled as x_1, y_1, z_1 in Figure 4.16. A seamless interaction has been established between the end-effector and the user's hand, prohibiting any eccentricity among the axes (i.e. z_1 is aligned perfectly with z_0).
- Subsequently, a precise external (virtual) torque, denoted as $[T_i]^t$, has been defined within the MATLAB/Simulink® model to be experienced by the human operator while operating the cylinder.
- The human user has been instructed to rotate the end-effector in the counterclockwise (CCW) yielding an angular movement measured as θ, as shown in Figure 4.16 (b) (i.e. z_1 is rotated around z_0 in CCW sense).
- Following a specific stall time duration, the user has been prompted to rotate the end-effector in the clockwise (CW) direction resulting an angular movement measured as - θ, illustrated in Figure 4.16 (c) (i.e. z_1 is rotated around z_0 in CW sense).

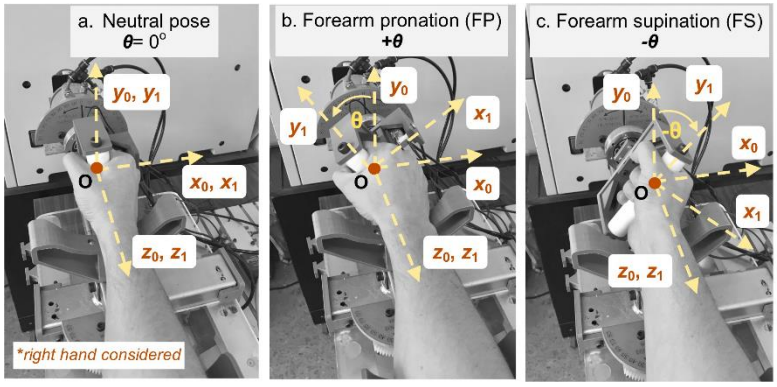

Figure 4.16. Loading-case scenarios; (a) Neutral pose; (b) Forearm pronation workout; (c) Forearm supination workout

The tests are conducted at consistently low angular velocities, hence disregarding the I and α terms (where I=0 kg·m^2; α=0 rad/s^2). Utilizing Equations 3.1-3.4, the necessary chamber pressure values are calculated based on the estimated friction torque (T_f) value and the desired external torque $[T_i]^t$ for a specific working pressure (P_w). Consequently, for validation objectives, the target chamber pressures are determined using Equations 4.4 and 4.5 as following:

$$[P_a]^r = P_w + \frac{[T_f] - [T_i]^t}{A_v \cdot r} \tag{4.4}$$

$$[P_b]^r = P_w + \frac{[T_f] - [T_i]^t}{A_v \cdot r} \tag{4.5}$$

In closed-loop under-loading validation experiments, a cascade control structure has been established by incorporating PID controllers to pressure regulating valves (PRV-a and PRV-b) instead of employing a direct control method utilized in open-loop testing. The underlying reason for this is the poor performance exhibited by the pressure regulating valves under dynamic conditions. The modified block diagram, depicted in Figure 4.17, illustrates the integration of PID controllers labeled as PID-a and PID-b for PRV-a and PRV-b valves, respectively. Within this diagram, the "Target pressure calculation block" determines the target chamber pressures ($[P_a]^r$, $[P_b]^r$) using input parameters such as the specified working pressure (P_w), desired external torque $[T_i]^t$, cylinder vane area (A_v), and torque radius of the cylinder (r). For CCW directions $[P_a]^r$ is regulated as per Equation 4.4 while $[P_b]^r$ is equated to the working pressure, P_w. On the contrary, for CW directions, $[P_b]^r$ is regulated as per Equation 4.5 while $[P_a]^r$ becomes P_w. The friction torque (T_f) is computed based on the defined T_f surfaces (Figures 4.12 (a) and (b)) as a function of working pressure (P_w) and angular speed (ω). Subsequently, the resulting $[P_a]^r$ and $[P_b]^r$ target values are forwarded to the PID controller setup as reference inputs.

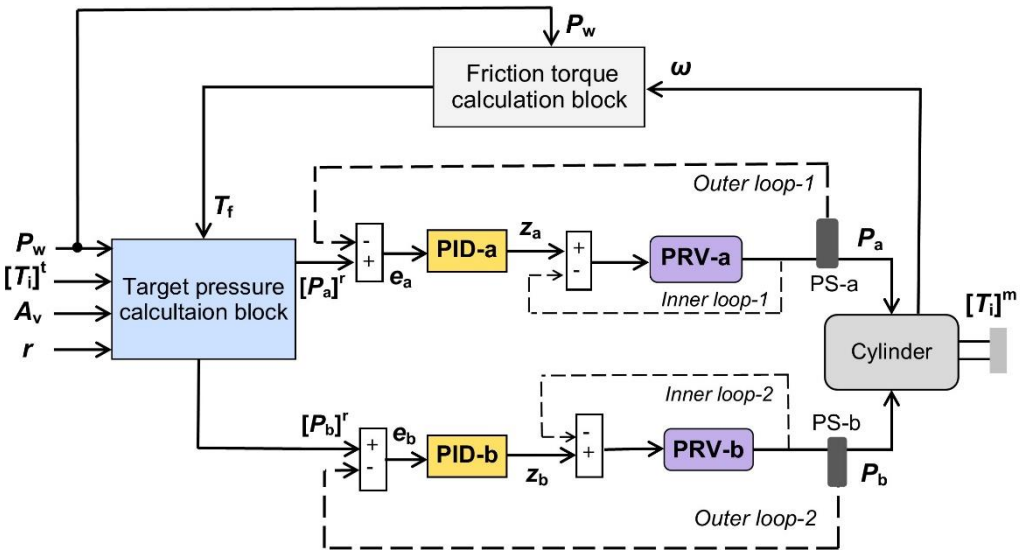

Figure 4.17. Control schema of closed-loop validation experiments

The tuning process for the PID parameters of both control valves (PRV-a and PRV-b) was previously carried out separately in a prior study. This involved comparing control responses such as rise time, settling time, delay time, and overshoot (Dağdelen and Sarıgeçili, 2022). Two scenarios were tested for the valves: (i) when PID was active (with the system cascaded) and (ii) when PID was passive (only the built-in controller of the valve was active). Throughout these tests, the valves' responses to a step input (i.e., reference) were monitored. The primary objective, consistent with all control applications, was to achieve a system with rapid response, minimal overshoot, and devoid of chattering. Using a trial and error experimental approach for the 4-bar step input, the PID parameters for the control valves were determined, as presented in Table 4.3.

Table 4.3. Tuned PID gains as a result of trial-and-error optimization

Parameters	**PID-a**	**PID-b**
K_p	0.715	0.715
K_i	7.250	7.000
K_d	0.016	0.012

Closed-loop validation tests assess whether the intended external torque that the user should experience can be achieved while the end effector is in motion, considering the estimated friction torque characteristics. Consequently, assessments were conducted using a random working pressure (P_w=5.4 bar) and evaluating two distinct interaction torques ($[T_i]^t$=0.75 Nm and $[T_i]^t$=1.5 Nm). The performance in torque tracking and the percentage of error compared to the desired torque are presented in Figures 4.18 and 4.19.

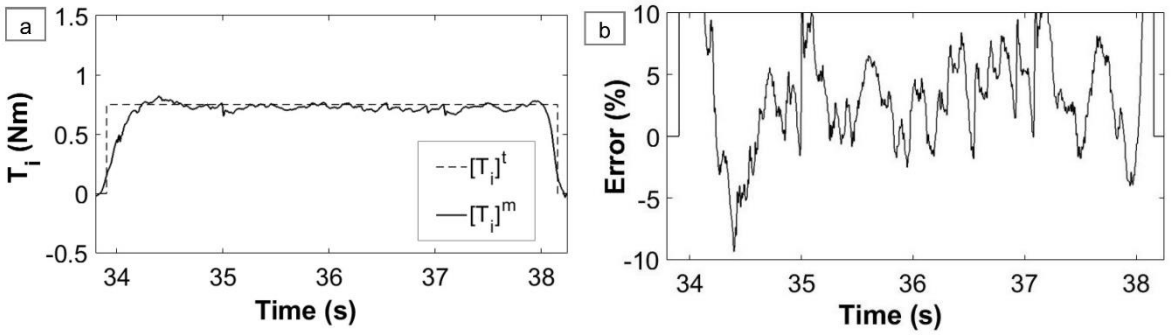

Figure 4.18. Closed-loop performance for P_w=5.4 bar and $[T_i]^t$=0.75 Nm (CCW direction); (a) Torque tracking performance; (b) Error (%)

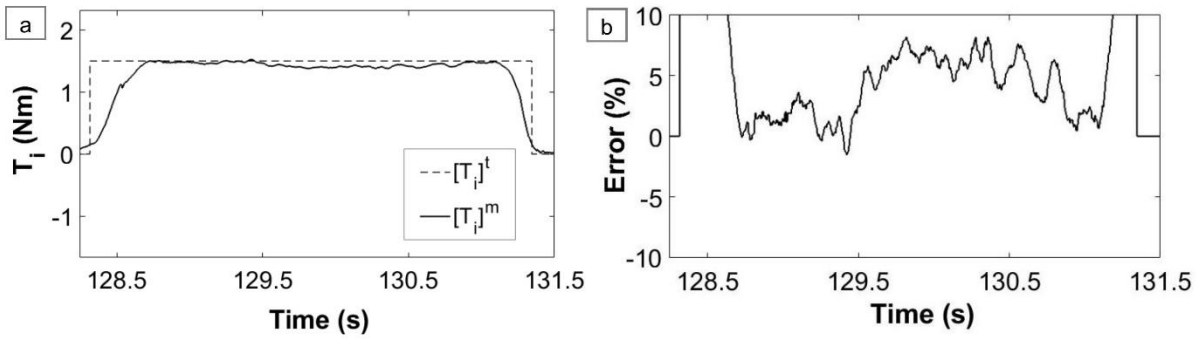

Figure 4.19. Closed-loop performance for P_w=5.4 bar and $[T_i]^t$=1.5 Nm (CCW direction); (a) Torque tracking performance; (b) Error (%)

When examining the validation outcomes from both open-loop and closed-loop experiments, the error levels appear to be within acceptable ranges. Specifically, the open-loop results demonstrate a nearly perfect alignment, displaying a maximum error level of 2.5%. As for the closed-loop analysis, the validation results underwent further scrutiny, utilizing Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) to evaluate the tracking performance, summarized in Table 4.4. These observations indicate that the estimated friction torque parameters effectively facilitated precise control actions to track the desired torque.

Table 4.4. Performance metrics for closed-loop validation experiments

Metrics	$[T_i]^t$**=0.75** (Nm)	$[T_i]^t$**=1.5** (Nm)
MAE (Nm)	0.043	0.143
RMSE (Nm)	0.006	0.081

4.2.2. The Results of Friction Identification Tests of Double-Acting Linear Pneumatic Cylinder

To show the effectiveness and accuracy of the friction identification automation algorithm presented in the subsection "*Friction Identification of Double Acting Linear Pneumatic Cylinder*" of Section 3.2.4, three pneumatic cylinders, two identical and one different, have been tested with the proposed automated system at the sampling frequency (f_s) of 200 Hz. Identical cylinders (FESTO DSNUP-25-225-P-A) have been preferred to check the consistency of friction results. The

third cylinder (FESTO DSBC-32-300-PPVA-N3) which was larger in size and different in structural form has been used to demonstrate the adaptability of the automated system to different pneumatic cylinders. Both static and dynamic test numbers have been defined as 100 (N_s=100 and N_d=100). Friction force estimation via manual method as presented in (Dağdelen and Sarıgeçili, 2020) has also been performed for all three cylinders to make a comparison between the manual and automated method and, to reveal the accuracy and consistency performance of the proposed automated system (Dağdelen and Sarıgeçili, 2023b).

The results of the friction force parameters i.e. estimated static friction force (F_s), Coulomb friction force (F_c), Stribeck speed (v_s) and viscous friction coefficient (B) are tabulated in Table 4.5 for two identical cylinders, symbolized as C1 and C2. The parameter values are provided both for automated and manual methods and both in the extension and retraction directions denoted by superscripts "e" and "r", respectively. The results yield the Stribeck friction model for both cylinders. The consistency between parameter values of two similar cylinders for the automated method can be clearly seen when compared to the manual method results. The friction force-speed maps of C1 and C2 cylinders with respect to the obtained results are given in Figure 4.20 to present the deviation in the manual method results and the consistency in the automated method results. Another interesting conclusion can be found out when the parameters in the extension and retraction directions are compared in the automated method and the manual method. The automated method yielded similar results for both directions whereas the manual method produced dissimilar results. This also presents that the values of parameters are sensitive to the direction of extension or retraction in the manual method since the force application by the technician might be affected for two different senses.

Table 4.5. Estimated friction parameters of identical cylinders (C1, C2)

	Parameters	**Automated Method**	**Manual Method**		**Automated Method**	**Manual Method**
Results for C1	$[F_s]^e$ **(N)**	4.714	4.351	**Results for C2**	4.656	4.255
	$[F_c]^e$ **(N)**	1.958	0.560		1.626	1.439
	$[v_s]^e$ **(m/s)**	0.062	0.115		0.0511	0.052
	$[B]^e$ **(N·s/m)**	19.99	23.630		20.302	26.350
	$[F_s]^r$ **(N)**	-4.918	-4.285		-4.944	-4.337
	$[F_c]^r$ **(N)**	-1.954	-1.748		-2.032	-0.317
	$[v_s]^r$ **(m/s)**	-0.0708	-0.0095		-0.0544	-0.14
	$[B]^r$ **(N·s/m)**	20.635	17.064		20.923	28.003

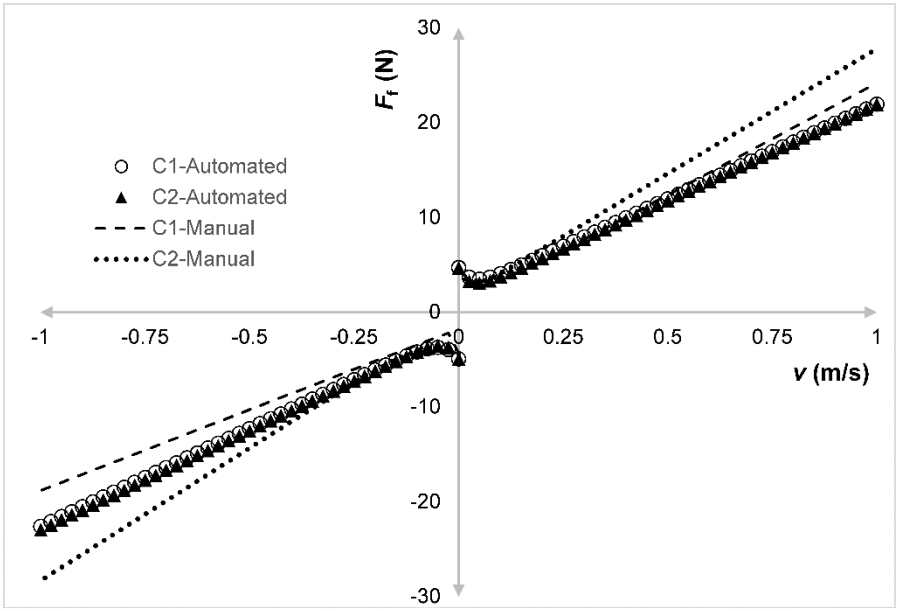

Figure 4.20. Friction force-speed curves for identical cylinders (C1 and C2)

The results of the friction force parameters for the dissimilar cylinder, C3, are shown in Table 4.6. The friction force-speed maps of C3 cylinder with respect to the obtained results are given in Figure 4.21. The performance of the automated method over the manual method can be obviously seen in Figure 4.21. As expected, the friction parameters for the cylinder 3 yielded different values compared to the other cylinders.

Table 4.6. Estimated friction parameters of cylinder C3

	Parameters	**Automated Method**	**Manual Method**
Results for C3	$[F_s]^e$ **(N)**	14.437	9.271
	$[F_c]^e$ **(N)**	4.479	4.316
	$[v_s]^e$ **(m/s)**	0.004	0.010
	$[B]^e$ **(N·s/m)**	45.796	39.189
	$[F_s]^r$ **(N)**	-14.536	-10.089
	$[F_c]^r$ **(N)**	-4.017	-3.922
	$[v_s]^r$ **(m/s)**	-0.005	-0.007
	$[B]^r$ **(N·s/m)**	47.283	35.239

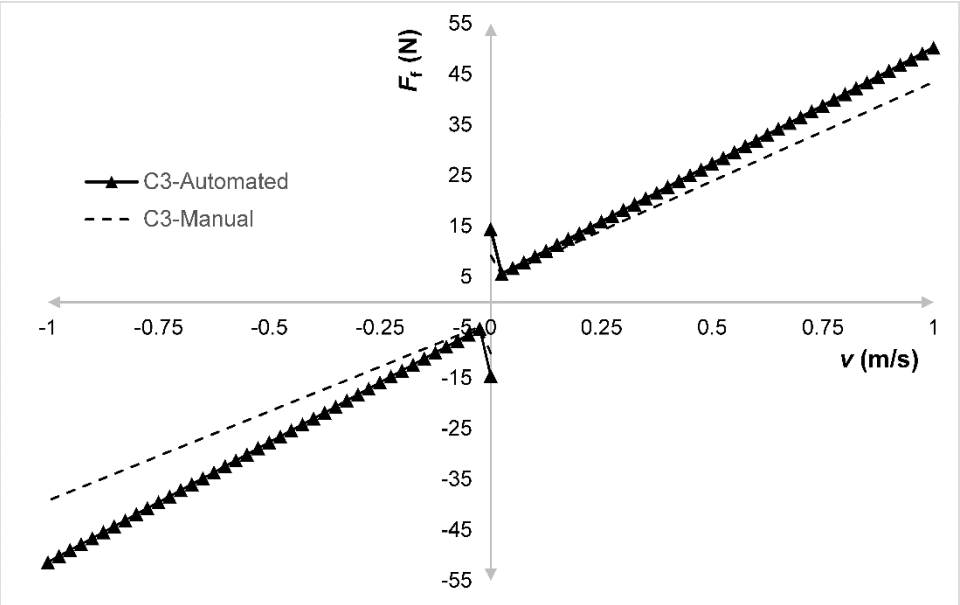

Figure 4.21. Friction force-speed curves for C3

Open-Loop Validation Results

In order to validate the estimated friction parameters from the automated and manual methods, pressure tracking tests have been carried out. Pressure tracking test mainly reverses the friction force estimation dynamic tests. In that regard, a random control signal for the corresponding DCV has been generated. The average constant speed value has been obtained from the motion of the piston. Then, the estimated friction force parameters together with the speed value have been entered into Equations 3.5, 3.6 and 3.13 to solve for the unknown chamber pressure (i.e. either P_a or P_b) driving the piston. Hence, the measured pressure from the sensor has been compared to the calculated theoretical pressure. During the validation experiments, the same experimental bench as in Figure 3.20 has been utilized.

For verification purposes, only one of the identical cylinders (C1) has been simulated at randomly specified u_a signal to obtain low and high level extension speeds. The supply pressure has been set up as 2.5 bar absolute (z=1.5 V) since the friction parameters have been estimated at this pressure level. P_b=P_{atm}=1.01 bar has been selected for extension motion calculations whereas P_a=P_{atm}=1.01 bar for retraction motion calculations. The simulation results of low and high speeds are shown in Figures 4.22 and 4.23, respectively. In these figures, both pressure tracking performances and percentage error deviations of both the automated and manual methods are presented. The automated method gives very close results to the actual measured pressure values meaning that the estimated friction parameters correctly represent the friction force at a particular speed value. Also, the error level of the automated method is less than 1% whereas the manual method can be erroneous up to 2.5%.

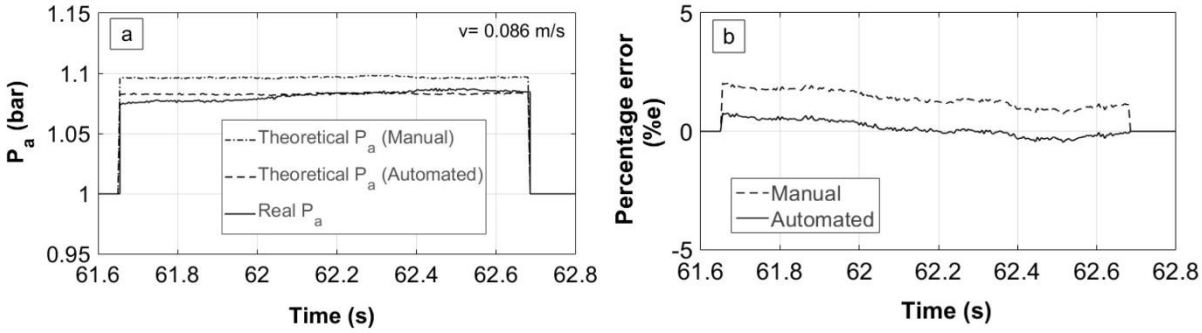

Figure 4.22. Pressure tracking performance comparison of C1 cylinder at low speed- extension

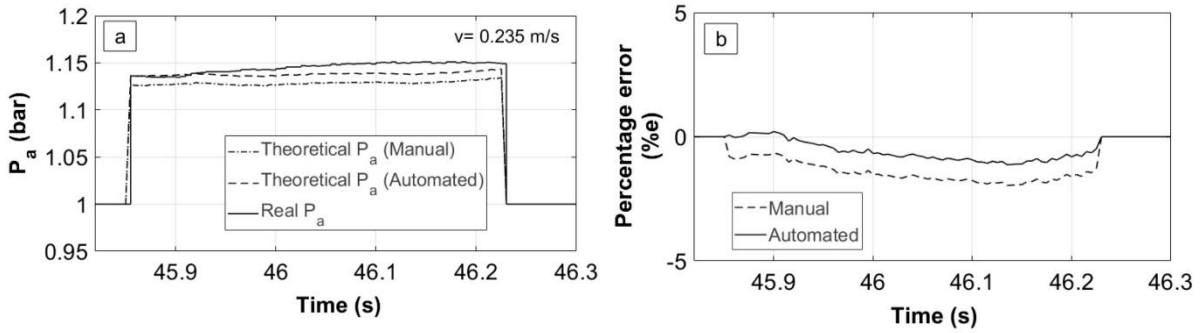

Figure 4.23. Pressure tracking performance comparison of C1 cylinder at high speed- extension

4.3. The Layout and Specifications of Real Time Hardware-in-the-Loop (HIL) Validation Experiments to Test the Controller's Performance with Changing Disturbance Scenarios

To prove the efficiency of proposed Cascade Fuzzy Adaptive PID controller and newly designed disturbance rejection algorithms (i.e. PDRA and TDRA), the robot will be tested at specified soft and hard interaction torque levels with a healthy user interacted as in the following experimental steps with the controllers in the query of I. Default, II. Cascade PID and III. Cascade Fuzzy Adaptive PID.

(1) Disturbance-free tests with PDRA and TDRA in passive state

First, the robot will be tested at disturbance free conditions by deactivating the PDRA and TDRA algorithms.

(2) Disturbance-free tests with PDRA and TDRA in active state

At this state, the robot will be evaluated under disturbance free conditions, but in this case PDRA and TDRA will be activated. The purpose of these tests is to obtain whether the designed rejection algorithms are correcting the control output of the system under unidentified environmental or user disturbances.

(3) Under torque-disturbance tests with PDRA and TDRA in active state

The robot will be tested at torque disturbance conditions by activating PDRA and TDRA algorithms.

(4) Under pressure-disturbance tests with PDRA and TDRA in active state

Finally, to verify the performance of PDRA algorithm, the robot will be tested under specified pressure conditions by activating both PDRA and TDRA in active state.

During the real-time hardware-in-the loop experiments for all scenarios, the working principles specified in the subsection of "*Closed-Loop Under-Loading Tests*" presented in Section 4.2.1 have been applied with the following simulation specifications:

- The running time of the HIL simulation has been set to infinity (∞) to test the system in more enhanced manner.
- The solver type has been selected as "Fixed step".
- All the evaluation tests have been carried out with f_s=200 Hz.
- The right hand of the user is preferred.
- The robot's controller acts when a specific motion intent is captured based on algorithm embedded in "User Motion Intention Detector and Signal Routing" block (Figure 3.30). Therefore, in all performance evaluation experiments, a step input in the form of $[T_i]^t$ is recognized. $[T_i]^t$=0.6 Nm and $[T_i]^t$=1.2 Nm target interaction torque levels have been set for the soft and hard interaction motions, respectively for both FP and FS motions.
- Working pressure of the pneumatic air has ben set as P_w=3 bar absolute.
- In disturbance free condition no kind of disturbances are given or applied. The robot is tested with stable operating speeds and any kind of pressure losses are disabled (i.e. it is ensured that pneumatic line operates in fully closed chamber). On the contrary, in under-disturbance tests, either pressure disturbances are intentionally created, or torque disturbances are implemented by applying abrupt operating speeds during normal operating speed conditions.

4.4. Experimental Results for Scenario 1

First of all, hard interaction levels have been tested for Scenario 1 (SC-1) for both of the movement directions. The performance results of the controllers at hard interaction, $[T_i]^t$=1.2 Nm-FP are presented in Figure 4.24. The major problem encountered in the Default and Cascade PID controllers has been obtained as high frequency chattering around target interaction torque which reduced the comfort and safety of the user during interaction (Figures 4.24 (a) and (b)). This problem is also harmful for control instruments which affects their operating life in negative manner. In addition to that, even if Default controller enabled smaller rise time with minimum overshot, the steady state error was the biggest among the tested controllers. Cascade PID algorithm, improved the steady-state error problem and slightly reduced the frequency and amplitude of the chattering compared to the Default controller results. However, the rise time of

Cascade PID is worse than the Default's one. As aspected, the proposed Cascade Fuzzy Adaptive PID algorithm offered the best performance by providing small rise time and oveshoot. In addition to that, the chattering proplem has been greatly minimized by providing safe and comfortable interaction (Figure 4.24(c)).

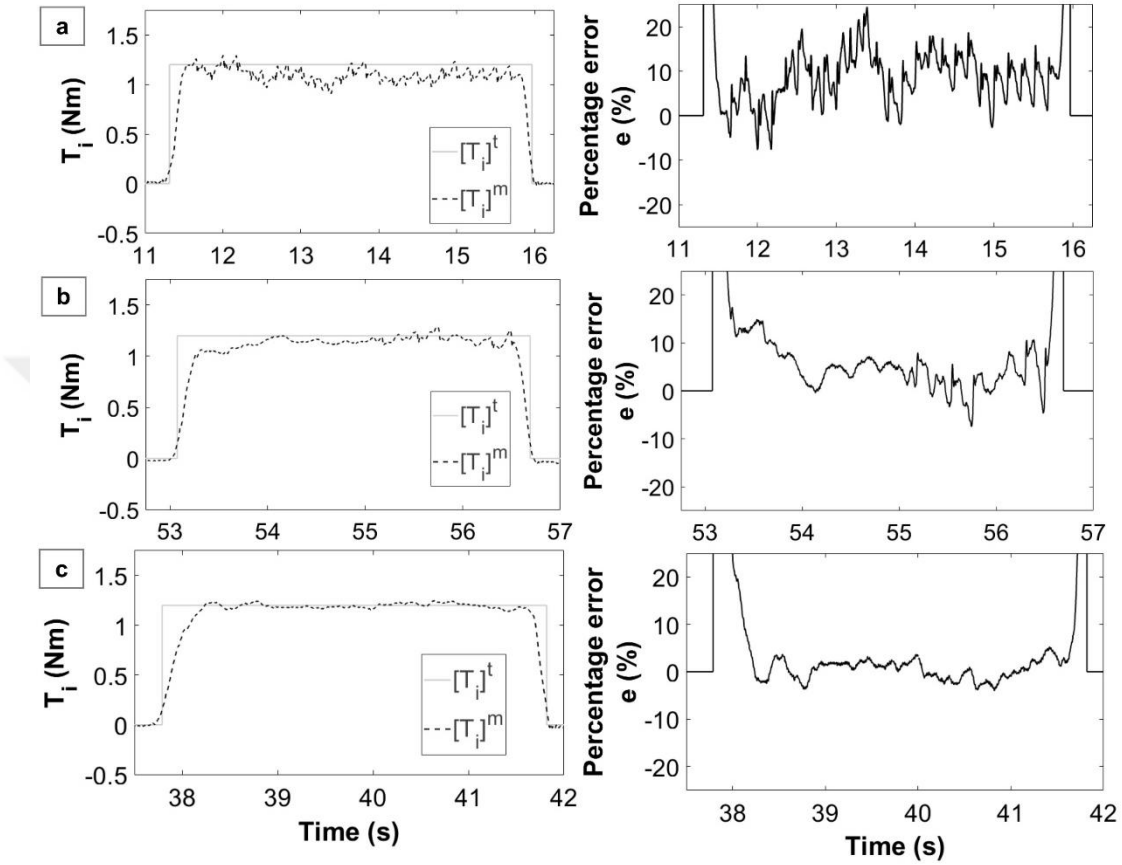

Figure 4.24. Performance comparison of controllers at $[T_i]^t$=1.2 Nm (FP; SC-1); (a) Default; (b) Cascade PID; (c) Cascade Fuzzy Adaptive PID controllers

The performance results for FS motion at $[T_i]^t$ = 1.2 Nm for FS workouts are given in Figure 4.25. Similar performances have been obtained as in 1.2 Nm-FP tests. In these tests, Default controller could not set to the target interaction torque and yielded huge steady-state error which is evident in percentage error (e%) graph (Figure 4.25 (a)). Furthermore, high frequency chattering has been obtained throughout the executed workout. On the other hand, Cascade PID algorithm provided much better performance from the chattering point of view, yet it failed on the rise and settling time points of view. At the begging of specified workout, it enabled moderate following performance with nonviolent chattering, however the amplitude of chattering increased and took a dangerous form near to the end of workout (Figure 4.25 (b)). Compared to the counterparts, Cascade Fuzzy Adaptive PID algorithm has shown good following performance by reaching the target in short time with almost zero overshoot. In addition to that, the existing chattering problem

in Default controller has been greatly reduced as it is clear in both interaction torque and percentage error graphs (Figure 4.25 (c)).

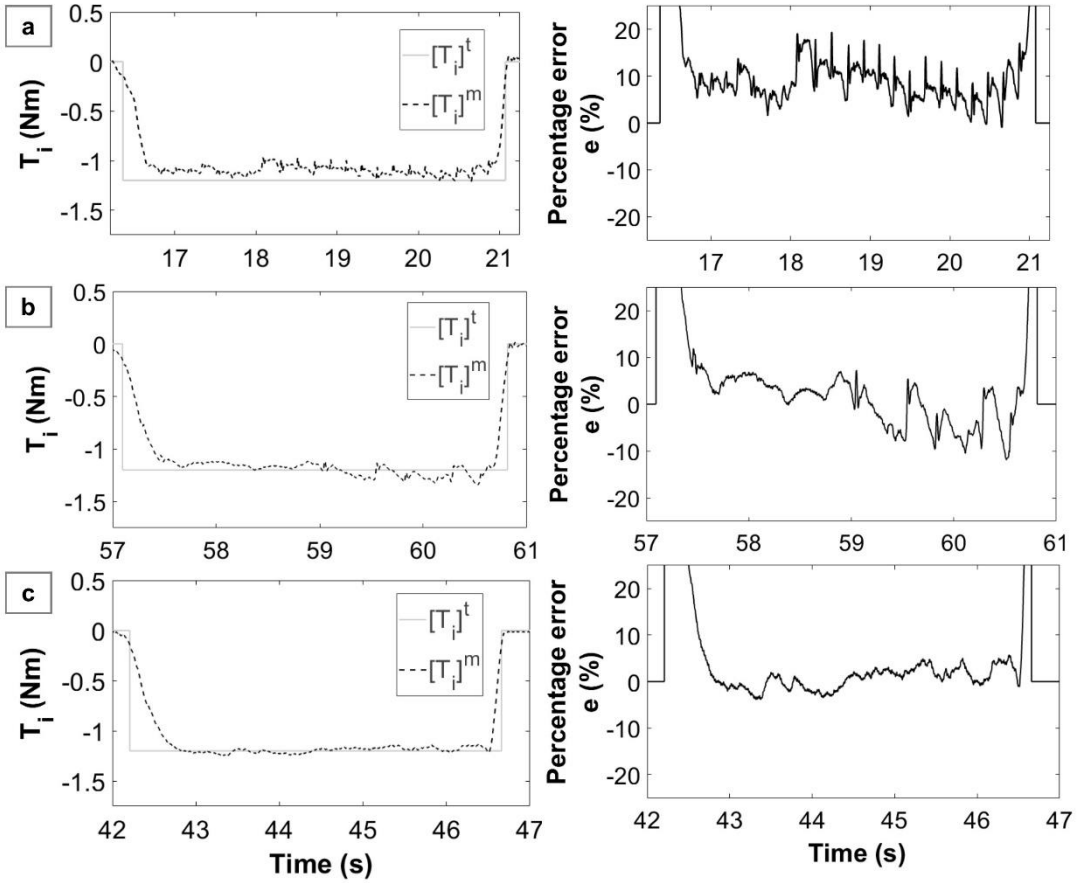

Figure 4.25. Performance comparison of controllers at $[T_i]^t$=-1.2 Nm (FS; SC-1); (a) Default; (b) Cascade PID; (c) Cascade Fuzzy Adaptive PID

To provide the efficiency of the proposed Cascade Fuzzy Adaptive PID framework, the performance metrics which are Mean Absolute Error (MAE), Integral Absolute Error (IAE), Integral Time-Weighted Absolute Error (ITAE), Error Variance (e_{var}), Root Mean Squared Error (RMSE) and Maximum Absolute Error (Maximum AE) have been calculated and presented in the form of radar graphs. In these graphs, each axis denotes one of the performance metrics whose respective values can be read from the common axis scale. The red dashed line, blue dotted line and green continuous line represent the performances of Default, Cascade PID and Cascade Fuzzy Adaptive PID controllers' performances, respectively. The worse performance of the Default and the best performance of the Cascade Fuzzy Adaptive PID can be detected by looking at the constructed radar performance graphs given in Figures 4.26 (a) and (b), for the cases of $[T_i]^t$=1.2 Nm and $[T_i]^t$=-1.2 Nm, respectively.

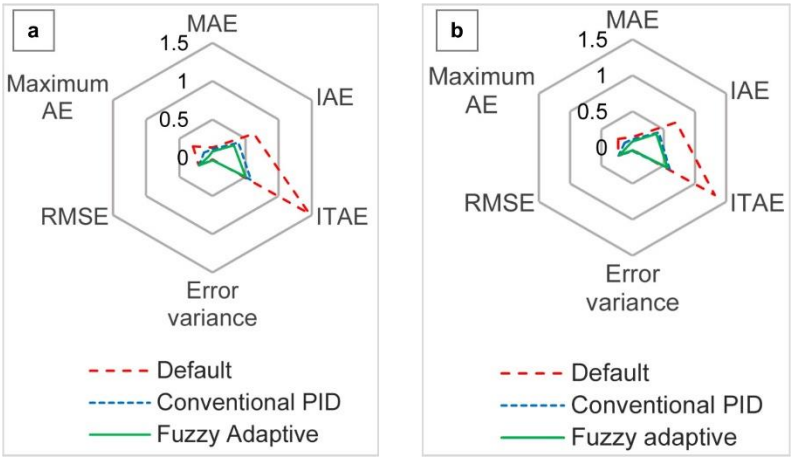

Figure 4.26. Radar graphs of performance metrics in hard interaction level for SC-1; (a) $[T_i]^t$=1.2 Nm-FP; (b) $[T_i]^t$=-1.2 Nm-FS

After conducting hard interaction validations tests, soft interaction level tests for Scenario 1 have been followed. For this aim, $[T_i]^t$ has been set to 0.6 Nm and -0.6 Nm for FP and FS workouts, respectively. The performances of FP workout are given in Figures 4.27 (a), (b), (c) and whereas the performances of FS workout are given in Figures 4.28 (a), (b), (c). Similar to the hard interaction test results, Default controller yielded the worse following performance with high frequency chattering and high steady state error for both of the workouts (Figure 4.27 (a) and Figure 4.28 (a)). Cascade PID framework slightly improved the following performance by reducing the steady state error level, however it provided bigger overshoot compared to the Default controller. In addition to that, it could not eliminate fully the chattering problem especially in FS workout (Figure 4.28(b)). On the other hand, proposed Fuzzy framework yielded better torque following performances compared to the counterparts. The performance of Cascade Fuzzy Adaptive PID was better in FS workout compared to the FP workout. In FP workout, unexpected jumps have been obtained which resulted in uncomfortable interaction, on the other hand, the performance was excellent, free of chattering and stable in FS workout (Figure 4.28 (c)).

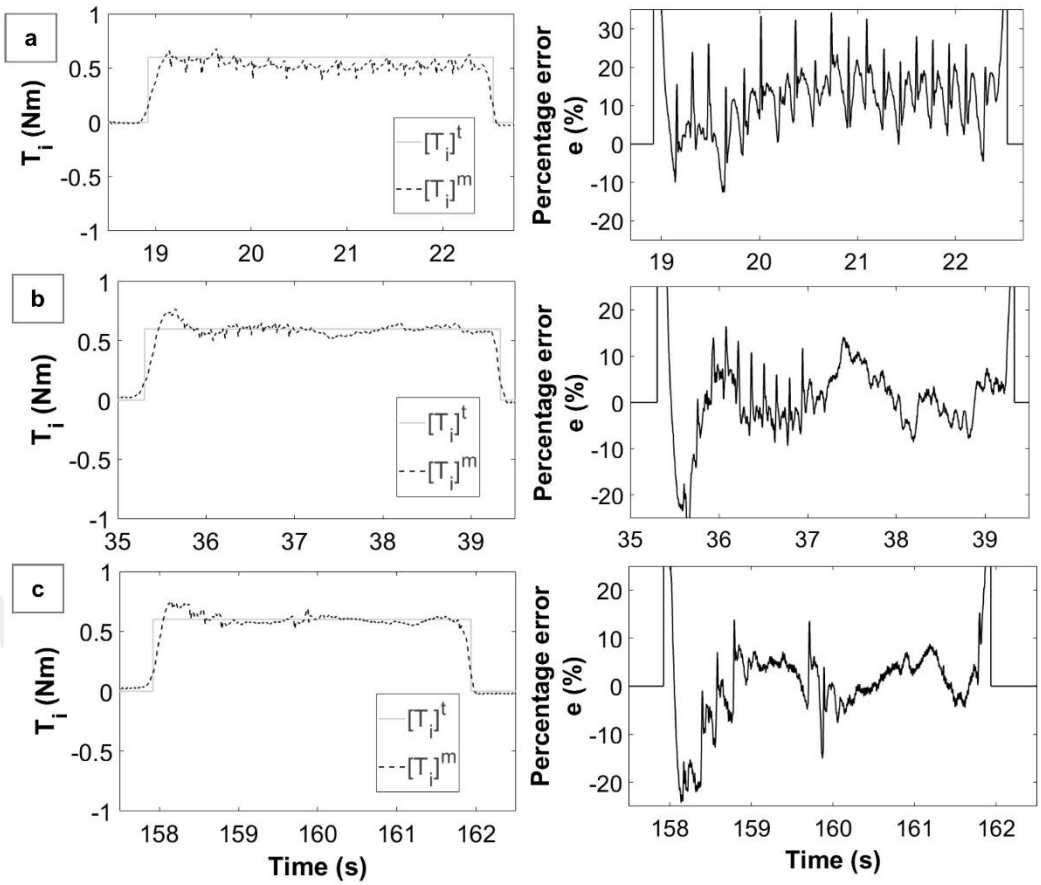

Figure 4.27. Performance comparison of controllers at $[T_i]^t$=0.6 Nm (FP; SC-1); (a) Default; (b) Cascade PID; (c) Cascade Fuzzy Adaptive PID

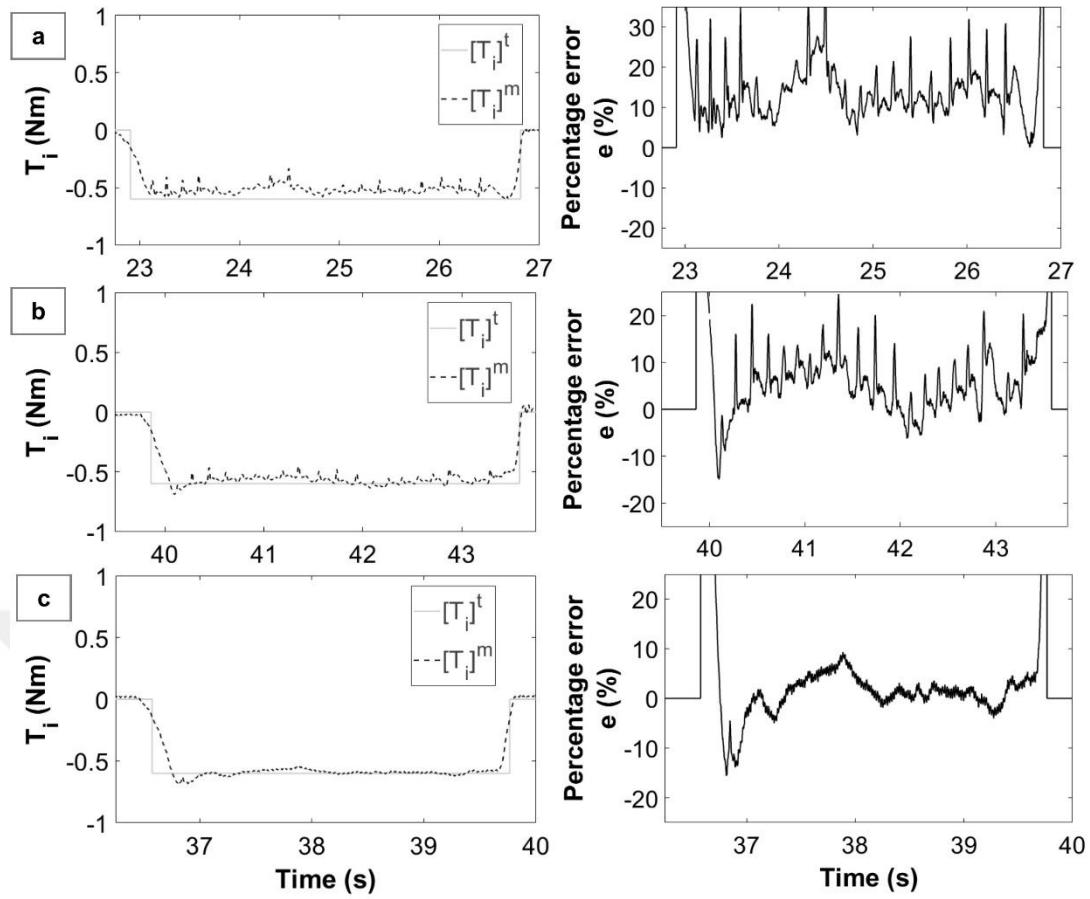

Figure 4.28. Performance comparison of controllers at $[T_i]^t$=-0.6 Nm (FS; SC-1); (a) Default; (b) Cascade PID; (c) Cascade Fuzzy Adaptive PID

The radar graphs for performance metrics are depicted in Figures 4.29 (a) and (b) for FP and FS workouts, respectively. The graphs verify the commentary made on the results in Figures 4.27 and 4.28.

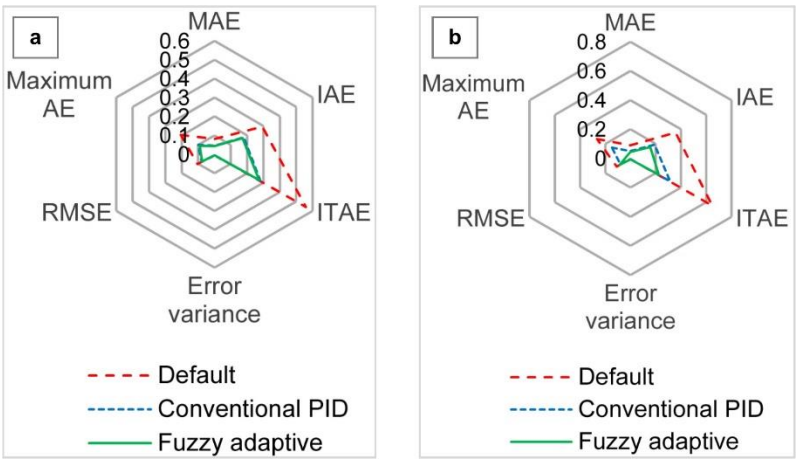

Figure 4.29. Radar graphs of performance metrics in hard interaction level for SC-1; (a) $[T_i]^t$=0.6 Nm-FP; (b) $[T_i]^t$=-0.6 Nm-FS

4.5. Experimental Results for Scenario 2

After conducting Scenario-1 tests, Scenario-2 (SC-2) tests have been conducted. The torque trajectory performance results for $[T_i]^t$=1.2 Nm for FP and $[T_i]^t$=-1.2 Nm for FS workouts are depicted in Figures 4.30 and 4.31, respectively. The radar graphs for performance metrics for FP and FS workouts are depicted in Figures 4.32 (a) and (b), respectively.

Since the PDRA and TDRA algorithms are activated in SC-2, the overall performances of all controllers have been greatly improved, yet the best one among the controllers was the proposed Cascade Fuzzy Adaptive PID framework. For the case of FP workout, by looking at the trajectory and error graphs, one can easily state that the steady state error problem of the Default controller has been eliminated except chattering (Figure 4.30 (a)) when compared to the SC-1 performance of the Default controller (Figures 4.24 (a)). Cascade PID framework quickly converged to the target torque and slightly reduced both amplitude and frequency of the chattering (Figure 4.30 (b)) when compared to the SC-1 results for the same condition (Figure 4.24 (b)). Proposed fuzzy framework performed outstanding performance with quick convergence, free of chattering and almost zero steady state error level. With the power of gain scheduling of the proposed fuzzy framework and designed rejection algorithms, the effects of unidentified disturbances and dynamics have been diminished and perfect torque trajectory performances have been obtained (Figure 4.30 (c)).

For the case of FS workout, similar performance levels have been obtained as in FP workout. As expected, the Default controller provided the worst performance with full of chattering and Cascade Fuzzy Adaptive PID framework yielded the best performance with free of chattering interaction together with an acceptable overshoot as clearly seen in Figure 4.31 (a), (b) and (c). From hard interaction point of view, the radar graphs of performance metrics for SC-2 are depicted in Figures 4.32 (a) and (b) for FP and FS workouts, respectively.

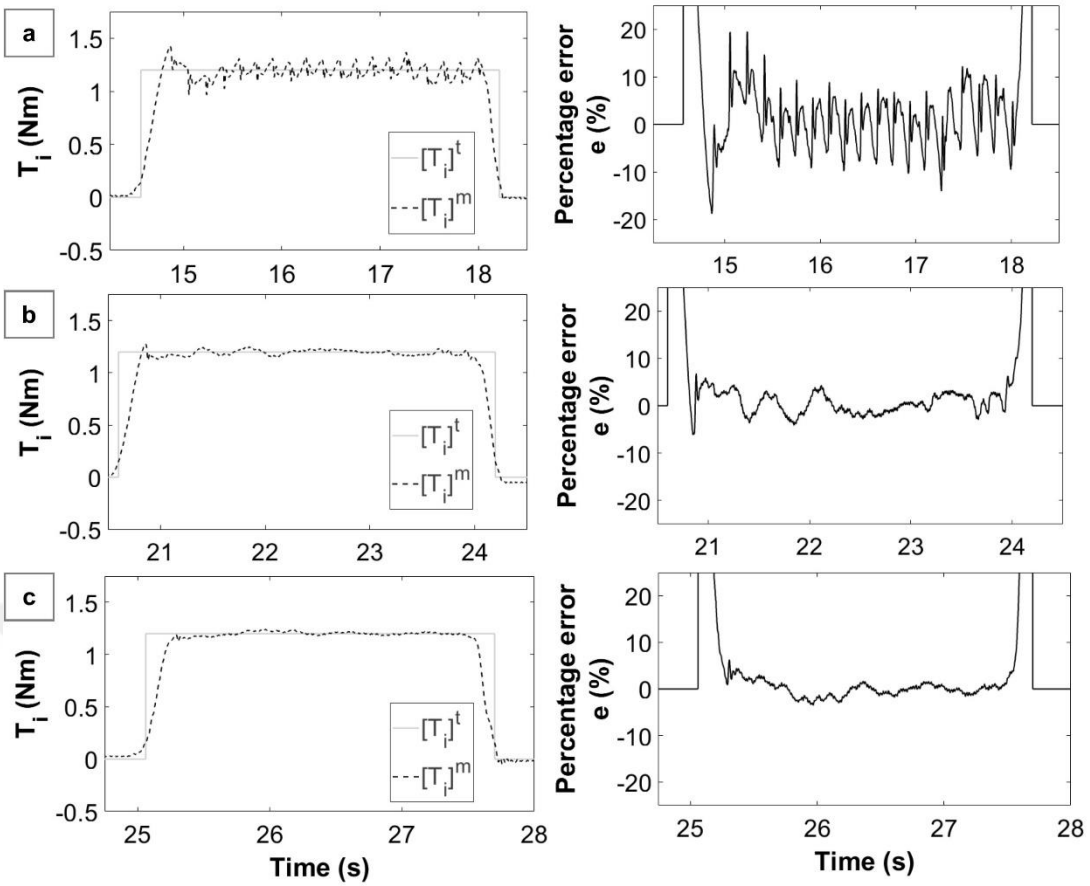

Figure 4.30. Performance comparison of controllers at $[T_i]^t$=1.2 Nm (FP; SC-2); (a) Default; (b) Cascade PID, (c) Cascade Fuzzy Adaptive PID

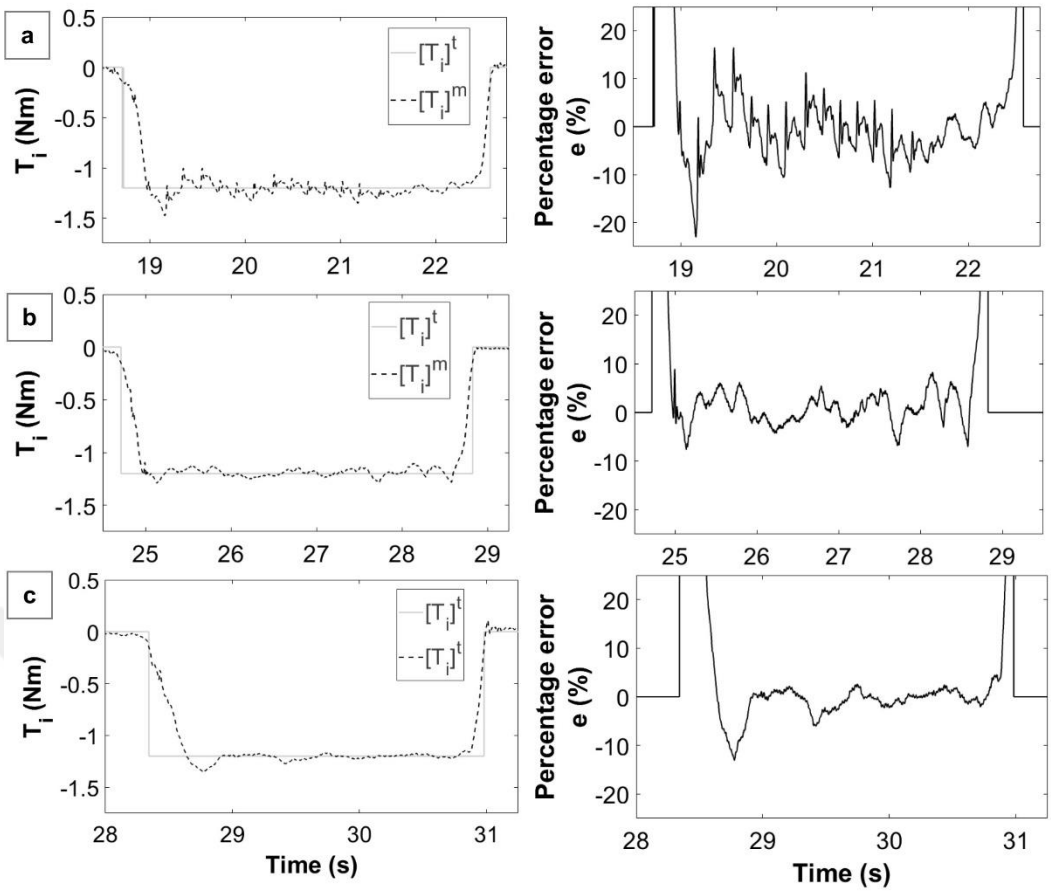

Figure 4.31. Performance comparison of controllers at $[T_i]^t$=-1.2 Nm (FS; SC-2); (a) Default; (b) Cascade PID; (c) Cascade Fuzzy Adaptive PID

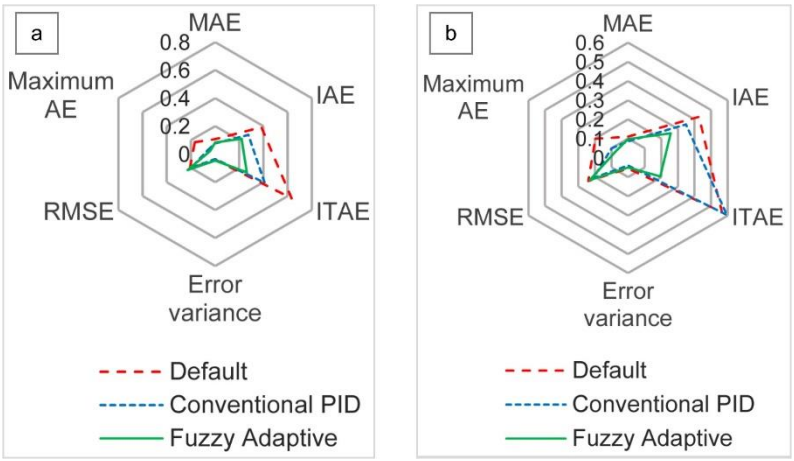

Figure 4.32. Radar graphs of performance metrics in hard interaction level for SC-2; (a) $[T_i]^t$=1.2 Nm-FP; (b) $[T_i]^t$=-1.2 Nm-FS

The validation results of soft interaction level are depicted in Figures 4.33 and 4.34, for FP (i.e. $[T_i]^t$=0.6 Nm) and FS ($[T_i]^t$=-0.6 Nm) workouts, respectively. For both of the workouts, proposed Cascade Fuzzy Adaptive PID has provided excellent torque trajectory following performance when the overall performance has been taken into account (Figures 4.33 (c) and 4.34 (c)). Even though, the performances of the Default and Cascade PID controllers have been

improved with the activation of disturbance rejection algorithms, the Default controller has provided the worst performance among the controllers (Figures 4.33 (a) and 4.34 (a)). The performance of the controllers at soft interaction level has been proven with radar graphs shown in Figures 4.35 (a) and (b), respectively.

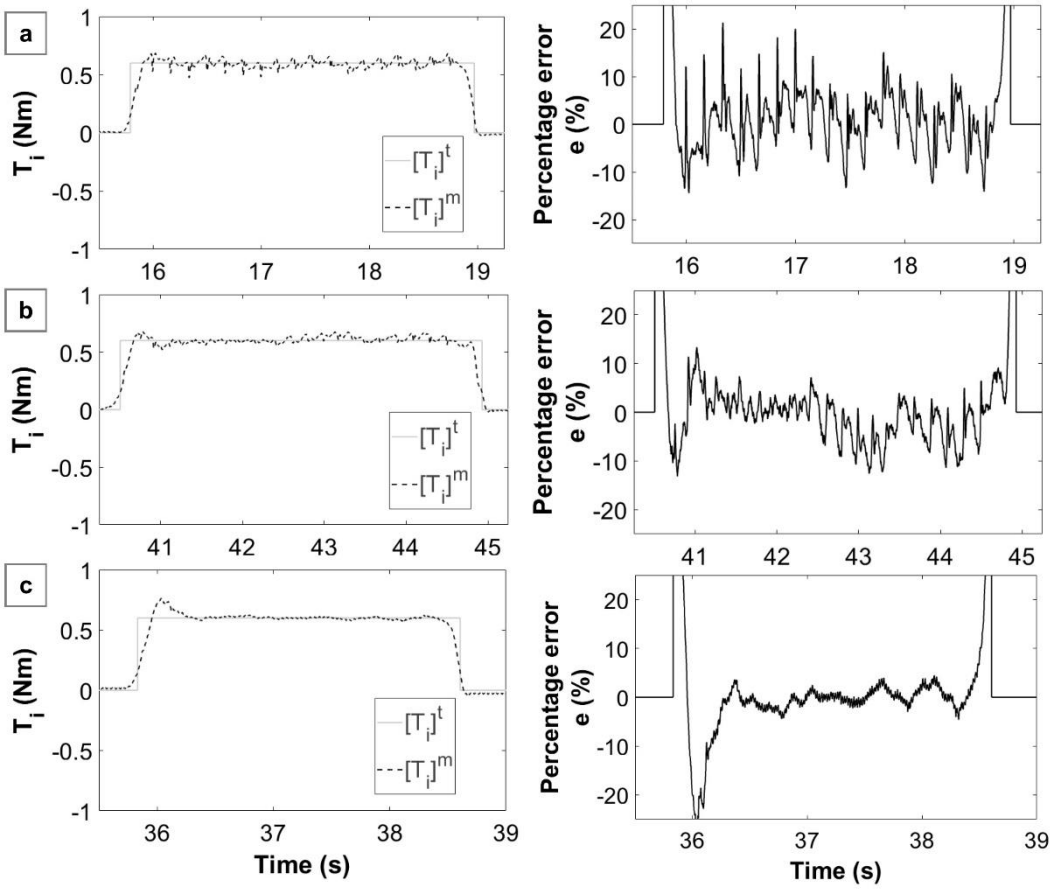

Figure 4.33. Performance comparison of controllers at $[T_i]^t$=0.6 Nm (FP; SC-2); (a) Default; (b) Cascade PID, (c) Cascade Fuzzy Adaptive PID

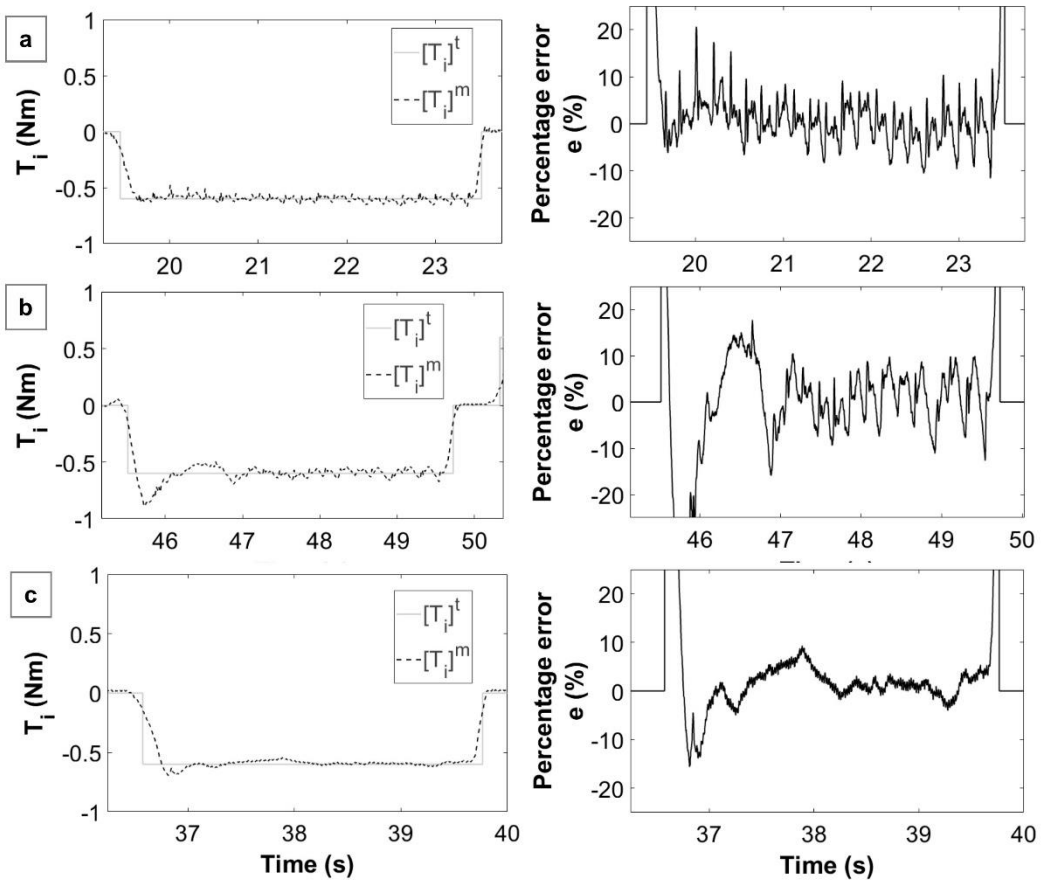

Figure 4.34. Performance comparison of controllers at $[T_i]^t$=-0.6 Nm (FS; SC-2); (a) Default, (b) Cascade PID, and (c) Cascade Fuzzy Adaptive PID

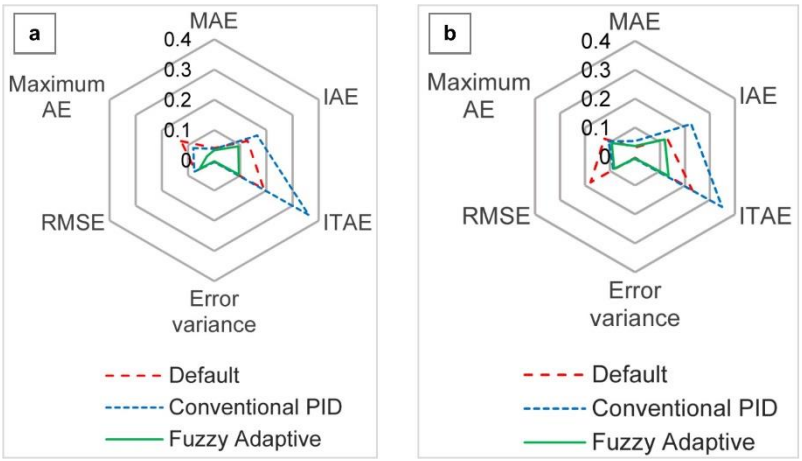

Figure 4.35. Radar graphs of performance metrics in hard interaction level for SC-2 (a) $[T_i]^t$=0.6 Nm-FP, (b) $[T_i]^t$=-0.6 Nm-FS

4.6. Experimental Results for Scenario 3

For Scenario-3, the performance of Torque Disturbance Rejection Algorithm (TDRA) has been evaluated for only FP workouts for the cases of hard and soft interaction levels. The performances of the Default, Cascade PID and Cascade Fuzzy Adaptive PID controllers are shown in Figures 4.36, 4.37 and 4.38, respectively. In all torque following performance graphs, the points

"a", "b" and "c" are specified on the graphs. The point "a" represents the instant at which the target interaction torque is converged. The point "b" is the disturbance instant at which, an intentional abrupt torque is applied by trying to rotate the end-effector in more fast manner. The point "c" states the point at which the workout period is ended. From the performance evaluation point of view, the following performances of the controllers between points "b" and "c" have been evaluated by looking at the torque tracking performance, the percentage error and operating speed variation.

The results of Default controller are shown in Figure 4.36. At the instant t_d=20.81 s, a redundant torque has been applied which is observable in Figure 4.36 (c). The Default controller could not converge to the target interaction torque after applied torque disturbance and performed chattering like profile (Figure 4.36 (a)). This kind of behavior drastically reduced the comfort of the user and the error levels reached to ±%20 which was not acceptable (Figure 4.36 (b)).

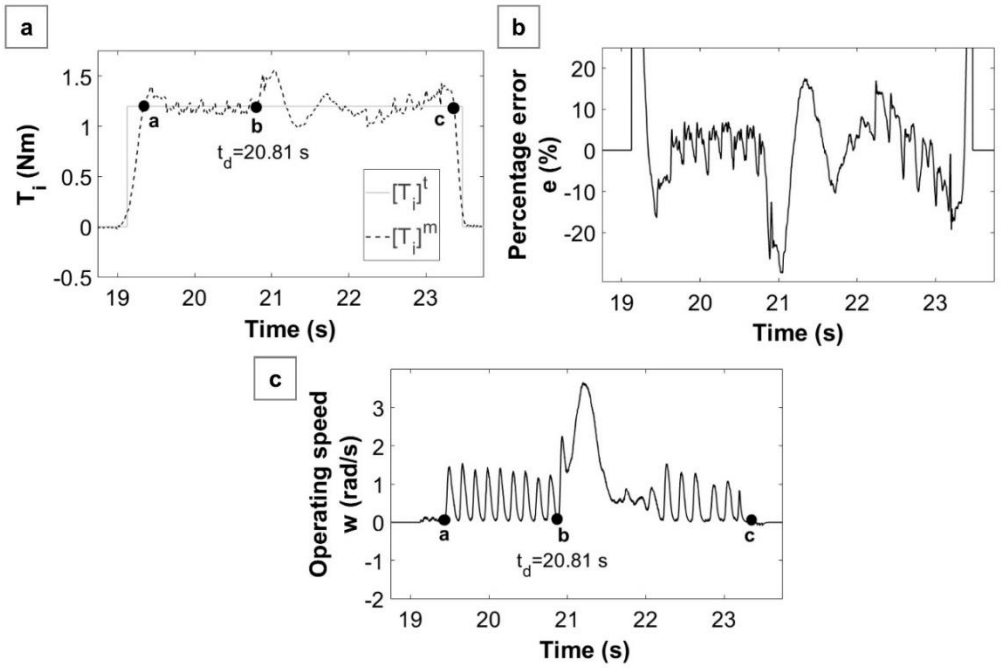

Figure 4.36. Performance of the Default controller at $[T_i]^t$=1.2 Nm (FP; SC-3)

The results of Cascade PID performance is depicted in Figure 4.37. At the instant, t_d=31.22 s, a specific torque disturbance has been applied which is clear in both torque tracking and operating speed graphs (Figure 4.37 (a) and (c)). Compared to the Default controller performance for the same condition, Cascade PID performed better with reduced chattering and percentage error during the workout. Even if, a stable interaction has been obtained after the disturbance input, it has behaved in low amplitude chattering profile and slightly reduced the comfort of the user. The maximum percentage error was around ±10% after the disturbance as observed in Figure 4.37 (b).

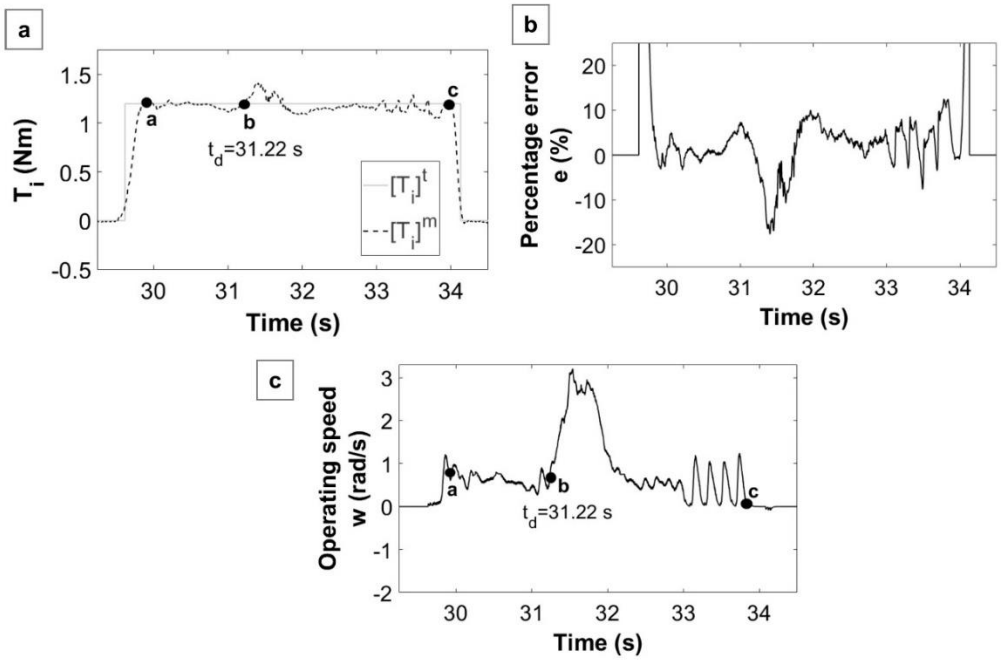

Figure 4.37. Performance of the Cascade PID controller at $[T_i]^t$=1.2 Nm (FP; SC-3)

For the evaluation of Cascade Fuzzy Adaptive PID framework, a torque disturbance has been given at t_d=56.93 s (Figure 4.38 (a)). Compared to its counterparts, proposed fuzzy framework has exhibited excellent tracking performance even after disturbance inclusion. After a short interval of time, the torque converged to the target torque and no chattering has been observed by fully protecting the user. The error levels after the disturbance instant was almost around zero as clearly seen in Figure 4.38 (b).

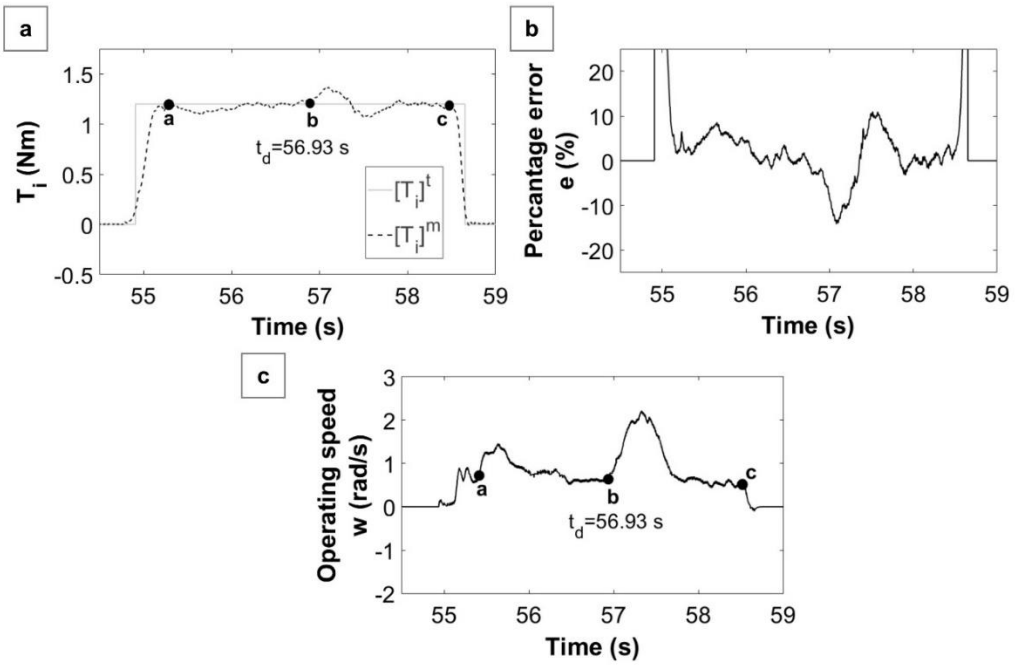

Figure 4.38. Performance of the Cascade Fuzzy Adaptive PID at $[T_i]^t$=1.2 Nm (FP; SC-3)

The radar graphs for performance metrics are shown in Figure 4.39. It is also clear that, Default controller exhibits the worst performance whereas the proposed Cascade Fuzzy Adaptive PID provided outstanding performance results.

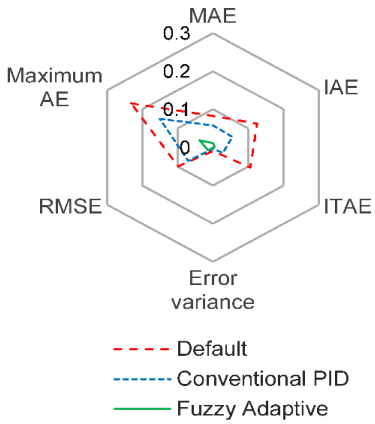

Figure 4.39. Radar graphs of performance metrics for $[T_i]^t$=1.2 Nm (FP; SC-3)

The results of SC-3 at soft interaction for FP workout ($[T_i]^t$=0.6 Nm) are presented in Figures 4.40, 4.41 and 4.42. At the instants of t_d=28.20 s, 73.68 s and 53.88 s, random torque disturbances have been applied for Default, Cascade PID and Cascade Fuzzy Adaptive PID controllers, respectively. When the results have been evaluated, similar performances with hard interaction level for SC-3 have been obtained for all controllers. Namely, the Default controller has exhibited the worst performance whereas proposed controller has shown the superior performance. The radar graphs of performance metrics are also depicted in Figure 4.43 wherein the outstanding performance of the proposed Cascade Fuzzy Adaptive PID has been proven over its counterparts.

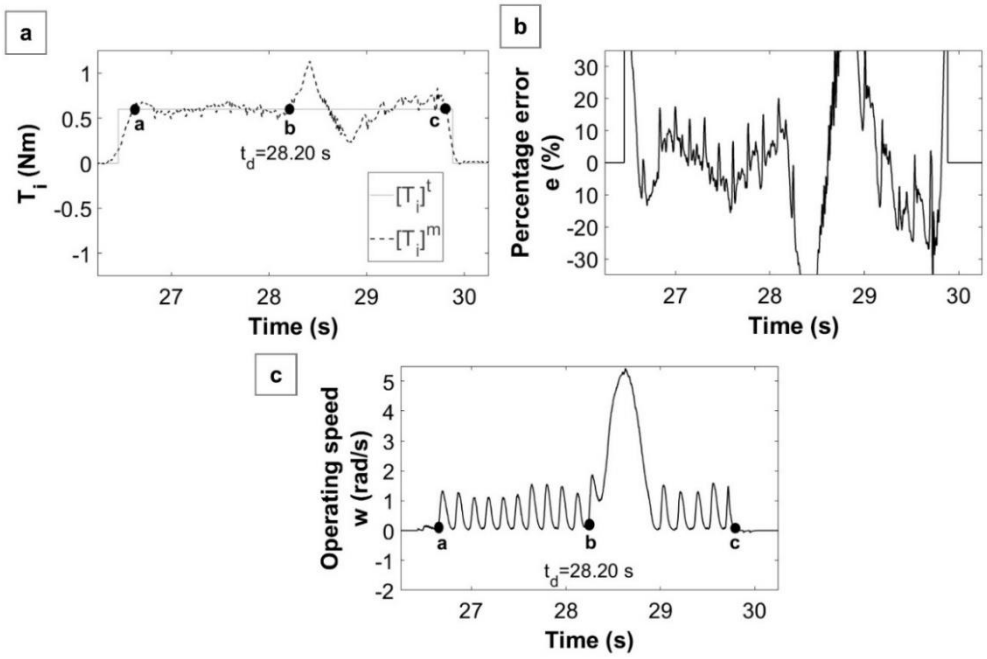

Figure 4.40. Performance of the Default at $[T_i]^t$=0.6 Nm (FP; SC-3)

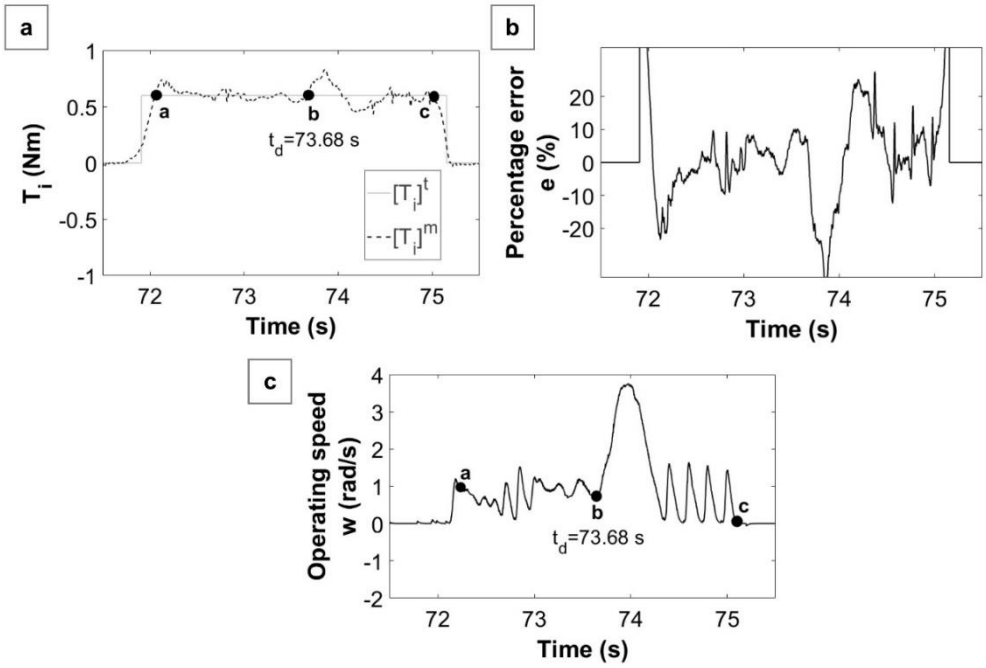

Figure 4.41. Performance of the Cascade PID at $[T_i]^t$=0.6 Nm (FP; SC-3)

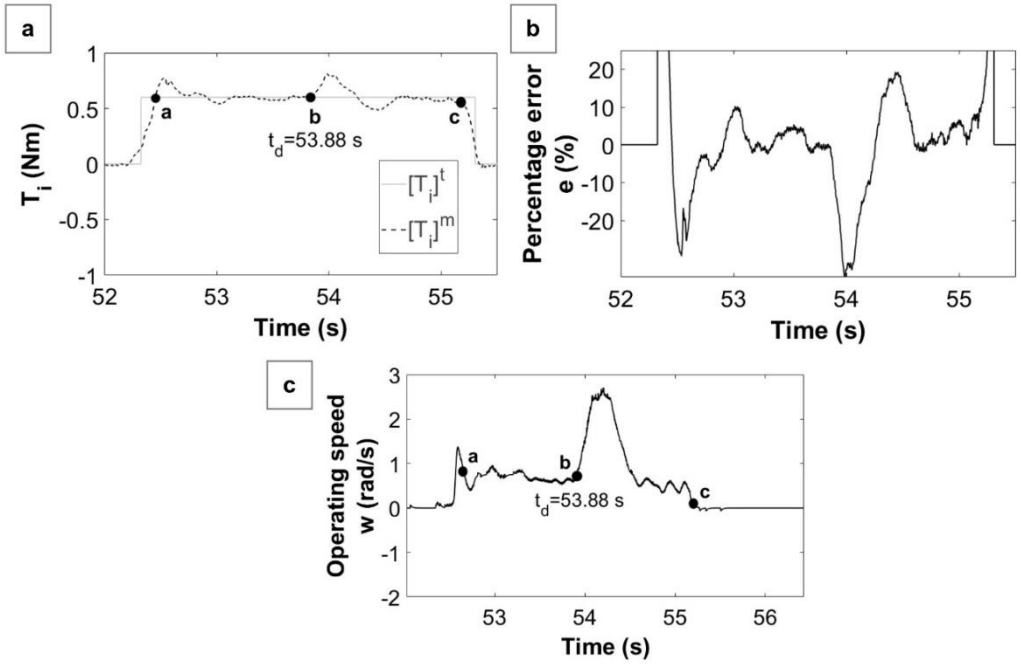

Figure 4.42. Performance of the Cascade Fuzzy Adaptive PID at $[T_i]^t$=0.6 Nm (FP; SC-3)

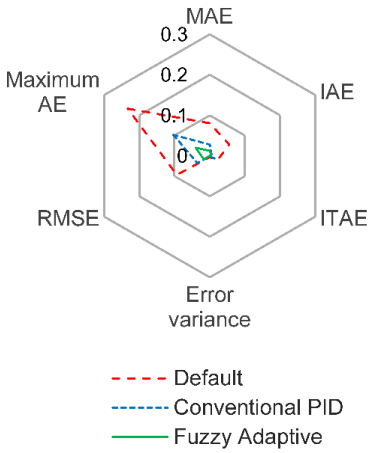

Figure 4.43. Radar graphs of performance metrics for $[T_i]^t$=0.6 Nm (FP; SC-3)

4.7. Experimental Results for Scenario 4

To test the controllers for Scenario-4, during any specific workout an intentional pressure loss at the exit of the pressure regulating valve has been created by sending a disturbance signal to the respective valve. For this study a pressure loss in counter chamber "b" has been specified as $[P_b]^d$= -0.4 bar while no disturbance has been applied in chamber "a" ($[P_a]^d$= 0 bar) for the cases of hard and soft FP workouts. Namely $[T_i]^t$=1.2 Nm and $[T_i]^t$=0.6 Nm for FP motion have been analyzed. To measure the performances of the controllers at specified conditions, torque tracking (T_i), and percentage error (e %), pressures in chambers "a" and "b" (P_a and P_b) have been monitored. The time interval between the disturbance instant and the time corresponding to the end of workout analyzed for the performance analysis and measurement.

The results of hard interaction disturbance analysis are shown in Figures 4.44, 4.45 and 4.46 for Default, Cascade PID and Cascade Fuzzy Adaptive PID controllers, respectively. The radar graphs for performance metrics of hard interaction case is depicted in Figure 4.47.

At t_d=46.04, 19.11, and 56.11 s, pressure disturbances in the form of $[P_b]^d$=-0.4 bar have been applied. According to the experimental results and analysis of the performance metrics, Cascade Fuzzy Adaptive PID has shown superior performance with chattering free behavior and low steady state errors for hard interaction during FP motion (Figure 4.46). Among the controller, Default controller produced high amplitude and frequency chattering around target interaction torque after disturbance instant (Figure 4.44). Cascade PID algorithm provided better performance than the Default controller by reducing the chattering effect, yet it could not totally eliminate both the chattering and steady state error (Figure 4.45).

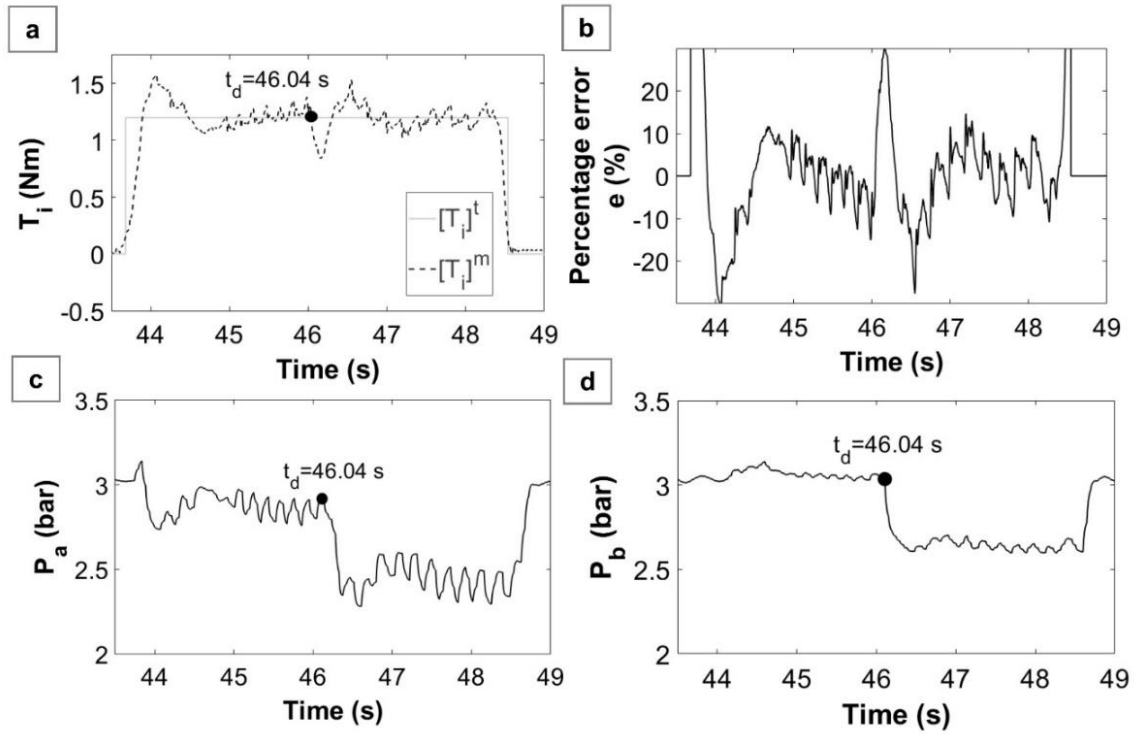

Figure 4.44. Performance of the Default controller at $[T_i]^t$=1.2 Nm (FP; SC-4)

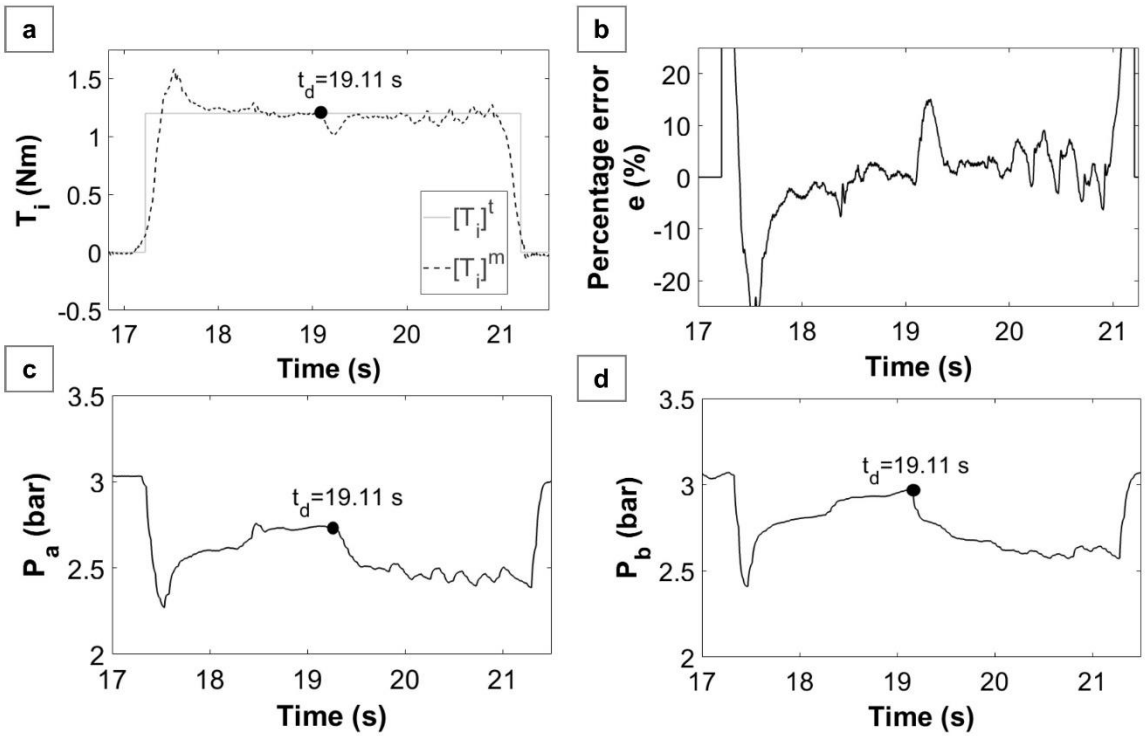

Figure 4.45. Performance of the Cascade PID controller at $[T_i]^t$=1.2 Nm (FP; SC-4)

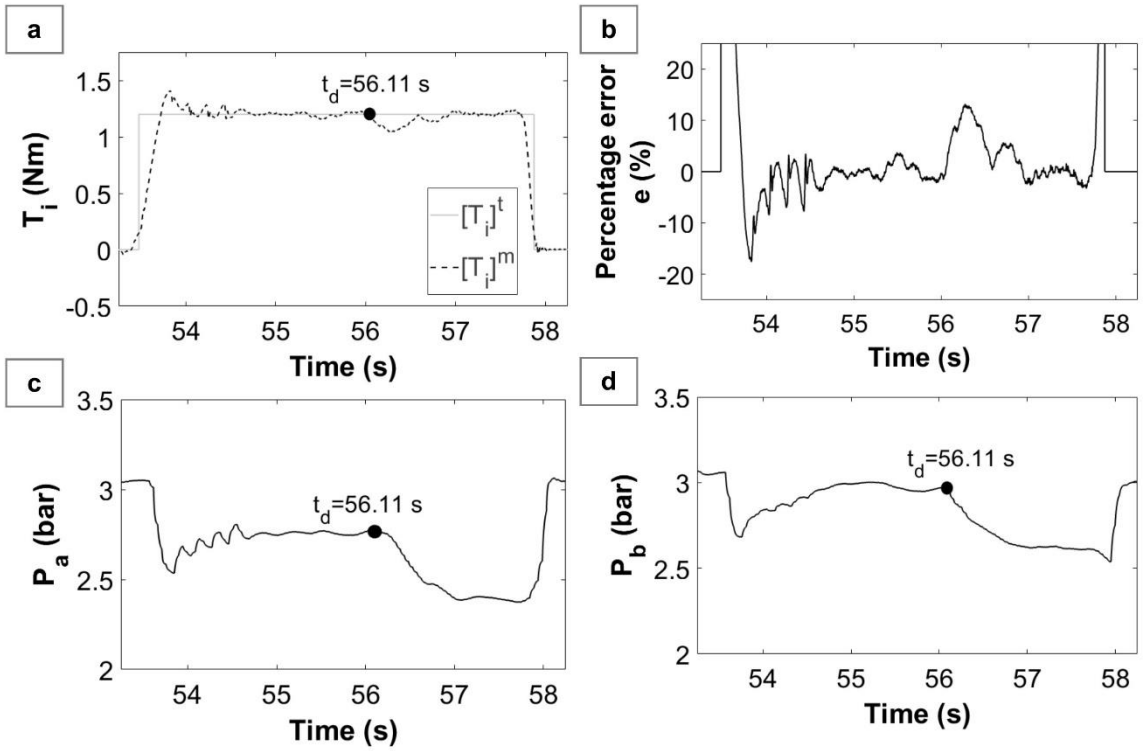

Figure 4.46. Performance of the Cascade Fuzzy Adaptive PID controller at $[T_i]^t$=1.2 Nm (FP; SC-4)

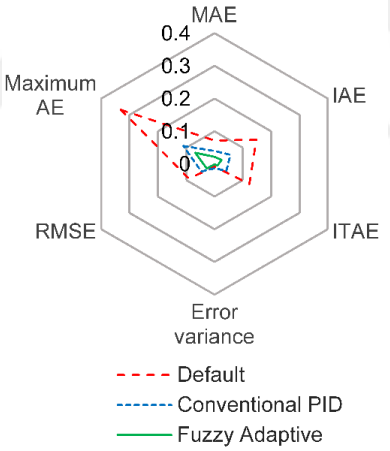

Figure 4.47. Radar graphs of performance metrics for $[T_i]^t$=1.2 Nm (SC-4)

The experimental outputs of soft interaction are shown in Figures 4.48, 4.49 and 4.50, and also the radar graphs of performance metrics for this condition are demonstrated in Figure 4.51. Bearing on the torque tracking performance presented in Figure 4.48 (a), Default controller yielded high frequency chattering and bad following performance after the disturbance instant t_d=101.47 s. By looking at the percentage error profile in Figure 4.48 (b), the worst performance has been obtained when Default controller was applied. From the Cascade PID controller point of view in Figure 4.49 (i.e. after the disturbance instant t_d=59.68 s), an improved tracking performance has been acquired when compared to results of Default controller (Figure 4.48). The frequency of chattering has been reduced and the comfort level has been increased. However, Cascade Fuzzy

Adaptive PID controller yielded perfect torque trajectory performance even after pressure disturbance at the instant of t_d=56.11 s (Figure 4.50). After the pressure disturbance, PDRA has worked well with the proposed Cascade Fuzzy Adaptive PID algorithm yielding fast convergence, almost free of chattering tracking performance (Figure 4.50 (a)).

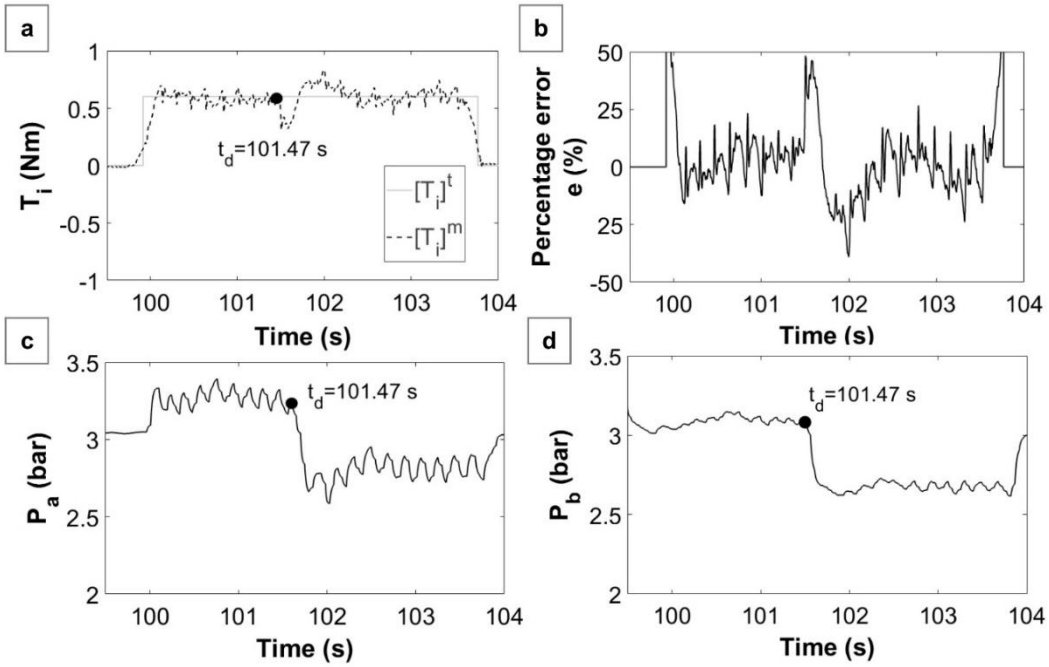

Figure 4.48. Performance of the Default controller at $[T_i]^t$=0.6 Nm (FP; SC-4)

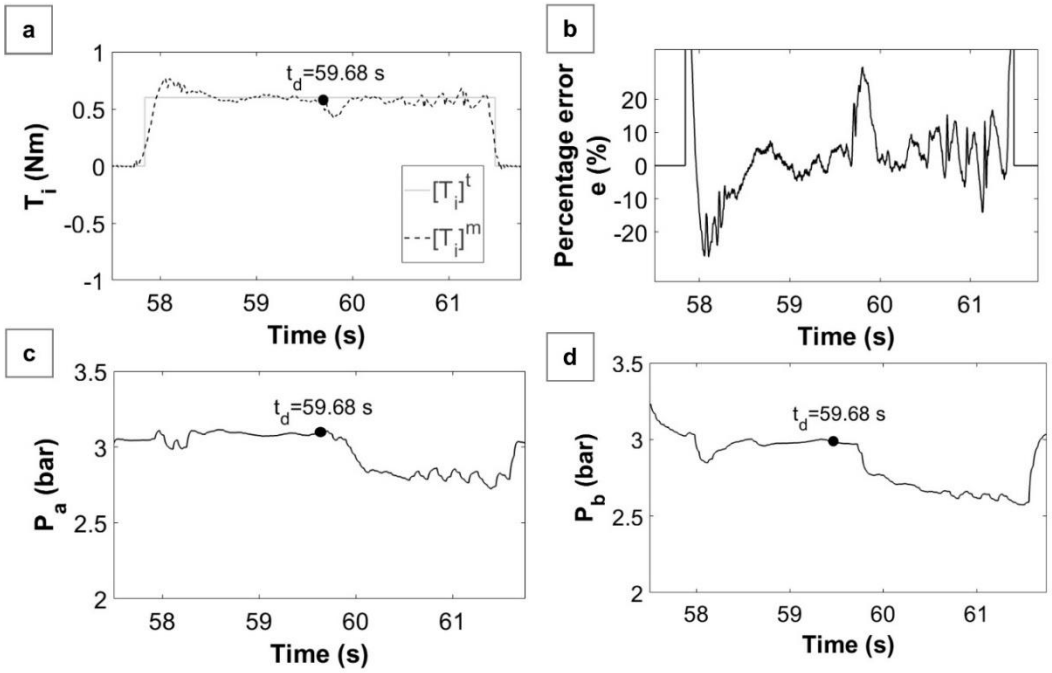

Figure 4.49. Performance of the Cascade PID controller at $[T_i]^t$=0.6 Nm (FP; SC-4)

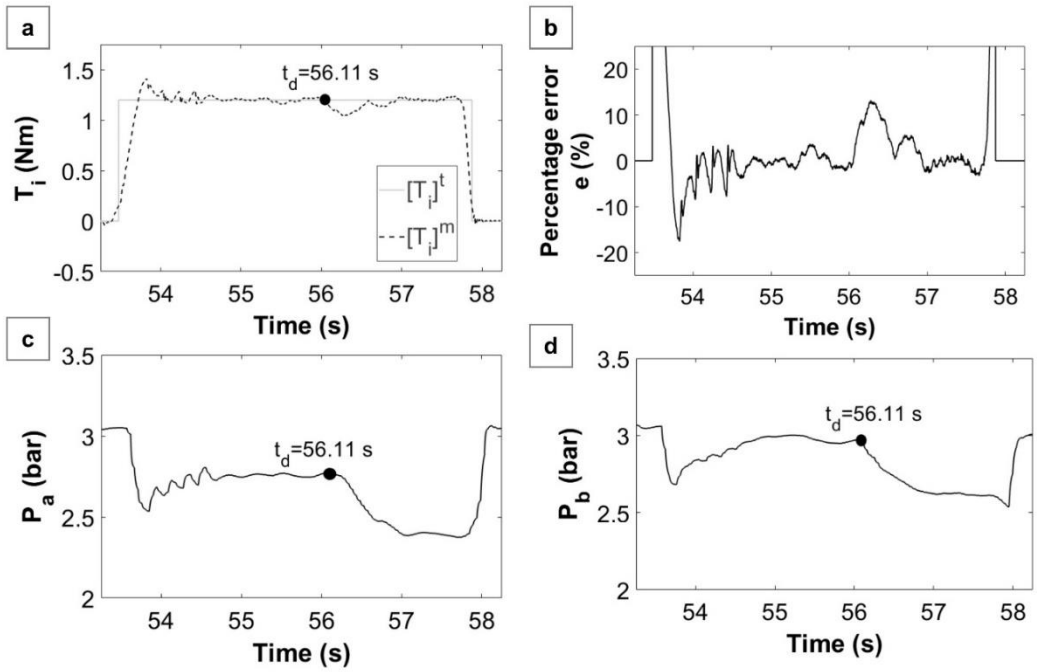

Figure 4.50. Performance of the Cascade Fuzzy Adaptive PID controller at $[T_i]^t$=0.6 Nm (FP; SC-4)

The radar graphs of performance metrics wherein the experimental data has been evaluated from Figures 4.48, 4.49 and 4.50 is illustrated in Figure 4.51.

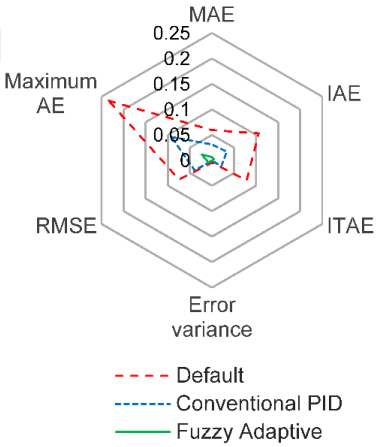

Figure 4.51. Radar graphs of performance metrics for $[T_i]^t$=0.6 Nm (SC-4)

.

5. CONCLUSIONS

In this thesis study, all the mentioned issues and proposed novelties have been successfully accomplished. To reach the main goal mentioned in Introduction section, firstly, the modification of the previously designed wrist and forearm rehabilitation robot (CWRR) has been conducted in the form of mechanic and control structure. From mechanical improvements point view, the parts have been designed based on increasing user comfort together with safety and decreasing overall weight of the robotic structure. And then, 3D printing technology has been utilized for the manufacturing of the modified mechanic parts. To enhance the control setup of the robot, new and sensitive equipment has been preferred. Especially, the pneumatic control valves have been changed with the proportional control valves to provide a wider range of control ability for the robot. In addition to that, pressure, force, and torque sensors have been included to the new version of the robot to increase the robot's control performance under changing and disturbed conditions.

After the modification procedure has been completed, the identification or modelling of the critical parts of pneumatics has been done, since the pneumatic systems have excessive non-linear behaviors such as friction in cylinders and non-linear compressible flow existing in the control valves. First, the control valves which are proportional pressure regulating valves (PRV) and directional control valves (DCV) have been analyzed in terms of input signal and either air pressure or mass flowrate output relation. It has been realized that the relation between input and output in PRVs was quite linear, yet it was excessively non-linear in DCVs. Therefore, most of the effort has been given for modelling of DCVs. For this purpose, a new and efficient compressible flow model for proportional DCVs has been proposed. The procedure to identify its parameters has been introduced in depth and successful identification study has been conducted. Bearing on the results, the real-time hardware-in-the-loop experiments have been carried out and identification of the valves have been verified.

A comprehensive friction identification for vane type semi-rotary cylinder and double-acting linear pneumatic cylinder has been introduced. Due to the difficulties in identification procedures which are majorly based on manual methods in literature, friction force/torque identification processes have been successfully automated and easy-to-use graphical user interfaces have been designed. With the implementation of automation algorithms, the friction identification time has been greatly fastened, more accurate parameter results have been obtained with the help of big data processing. The identified friction parameters yielded very good verification results under changing operating conditions.

After the identification of non-linear dynamics, an adaptive controller framework based on fuzzy logic has been designed. The adaptive algorithm has been implemented to schedule the Proportional-Integral-Derivative (PID) gains. As an important note, since the controlled valves (i.e. PRV) have their built-in controllers supplied by the manufacturer (FESTO), additional controllers

have converted the system to cascade control structure. Therefore, the proposed controller has been called as Cascade Fuzzy Adaptive PID. For the successful implementation of proposed controller, first, the tuning of PID gains has been conducted to set correct fuzzy variables by utilizing online closed loop PID auto-tuner algorithm in MATLAB/Simulink. Based on the optimized PID gains, fuzzy rules have been constructed based on expert knowledge. Correct input and output fuzzy membership functions have been built, the limits for membership functions have been set and accurate defuzzification method has been implemented. To provide the efficiency of proposed Fuzzy Adaptive PID framework, both the valves' Default controller and conventional Cascade PID algorithms have also implemented on the robot.

To satisfy the safety and compliancy, a user motion intention detection algorithm has also been designed and implemented. With the help of the intention detection algorithm, the controller is acted very properly with respect to the user's motion. In addition to that, the unexpected sudden motions have been prevented by stand-by the robot's actions.

Beyond this, efficient disturbance rejection algorithms have been constructed in the name of Pressure Disturbance Rejection Algorithm (PDRA) and Torque Disturbance Rejection Algorithm (TDRA). These rejection algorithms have embedded to all controller frameworks to investigate their performances in depth. The PDRA has worked properly and detected the any pressure losses and corrected to control outputs to protect the user from any unintended and fast movements. Furthermore, TDRA has manipulated the control output in the case of excessive torque disturbances and enabled a good convergence to the target interaction torque even after torque disturbances.

The robot has been tested with three different controllers under specified target interaction torque. Four different scenarios have been specified and based on these scenarios real-time hardware-in-the-loop experiments have been conducted. For all scenarios, proposed Cascade Fuzzy Adaptive PID algorithm has exhibited outstanding torque tracking performance with the power of gain scheduling and adaptability. Even, the counterpart controller performed worse than the proposed controller, the PDRA and TDRA algorithms improved their inherent control performances.

With this study, just the forearm pronation/supination movements have been investigated with the following critical issues:

- During the valve modelling stage, just control signals around 5 V (i.e. small orifice areas) could be modelled. The remaining orifice areas could not be modeled due the fast mass flow rate and difficulties in data processing. Therefore, main control valve has been selected as proportional PRV due to its easier control over proportional DCV. The modelling of proportional DCV for the full range of control signal may yield different type control configurations and performances.

- The friction torque identification of vane type semi-rotary cylinder has been conducted under changing working pressures. Nevertheless, the friction force identification of double-acting linear pneumatic cylinder has just been identified under a specific working pressure. When the wrist movements are studied on the robot, the friction force identification of linear cylinder should be carried out under changing working pressures for more comprehensive friction behavior.
- In this study, Stribeck friction model has been selected. The effect of more complex friction model could not be analyzed.
- The main controller's PID gains have been successfully optimized, yet the PID gains in TDRA algorithm have just optimized with experimental trial-error method. The more efficient either off-line or on-line tuning methodology may yield more promising results.
- The PDRA structure can just only consider the pressure disturbances (i.e. losses) in counter chamber. The disturbances in respective chamber could not be evaluated.
- The fuzzy logic algorithm has been designed based on specific rules, type of input/output membership functions and defuzzification method. The effect of different rule structure, type of membership functions or defuzzification methods could not be investigated.

As a result, pneumatic actuation methodology is good candidate for rehabilitation robotics, however it requires correct identification of its non-linear behaviors. The information given in this study can be taken as reference in future and related studies to get efficient control outputs.

As a future work, the following studies will be considered:

- Wrist flexion/extension and radial/ulnar deviation will be controlled based on interaction torque.
- Different exercise modes will be tested such as isometric, assistive/resistive.
- Different types of control approaches will be investigated.
- A game interface will be designed.

REFERENCES

Akdoğan, E., Aktan, M. E., Koru, A. T., Selçuk Arslan, M., Atlıhan, M., and Kuran, B., 2018. Hybrid impedance control of a robot manipulator for wrist and forearm rehabilitation: Performance analysis and clinical results. Mechatronics, 49(December 2017): 77–91.

Aktan, M. E., and Akdoğan, E., 2021. Development of an intelligent controller for robot-aided assessment and treatment guidance in physical medicine and rehabilitation. Turkish Journal of Electrical Engineering and Computer Sciences, 29(1): 403–420.

Allington, J., Spencer, S. J., Klein, J., Buell, M., Reinkensmeyer, D. J., and Bobrow, J., 2011. Supinator extender (SUE): A pneumatically actuated robot for forearm/wrist rehabilitation after stroke. Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS: 1579–1582.

Amiri, M. S., Ramli, R., and Aliman, N., 2022. Adaptive Swarm Fuzzy Logic Controller of Multi-Joint Lower Limb Assistive Robot. Machines, 10(6): 1–20.

Arbab, N. H., and Najafi, F., 2014. Stability and performance of pneumatic actuators in impedance control. Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering, 228(7): 476–485.

Ayas, M. S., and Altas, I. H., 2017. Fuzzy logic based adaptive admittance control of a redundantly actuated ankle rehabilitation robot. Control Engineering Practice, 59(November 2016): 44–54.

Beerens, R., Bisoffi, A., Zaccarian, L., Nijmeijer, H., Heemels, M., and Van De Wouw, N., 2022. Reset PID Design for Motion Systems with Stribeck Friction. IEEE Transactions on Control Systems Technology, 30(1): 294–310.

Belforte,G.,D'Alfio,N., and Raparelli,T.,1989. Experimental Analysis of Friction Forces in Pneumatic Cylinders. The Journal of Fluid Control, 20: 42-60

Belforte, G., and Raparelli, T., 1997. Friction analysis of pneumatic semi-rotary actuators©. Tribology Transactions, 40(1): 57–62.

Bobrow, J. E., and McDonell, B. W., 1998. Modeling, identification, and control of a pneumatically actuated, force controllable robot. IEEE Transactions on Robotics and Automation, 14(5): 732–742.

Bouteraa, Y., Abdallah, I. Ben, Ibrahim, A., and Ahanger, T. A., 2021. Fuzzy logic-based connected robot for home rehabilitation. Journal of Intelligent and Fuzzy Systems, 40(3): 4835–4850.

Brahmi, B., Driscoll, M., El Bojairami, I. K., Saad, M., and Brahmi, A., 2020. Novel adaptive impedance control for exoskeleton robot for rehabilitation using a nonlinear time-delay disturbance observer. ISA Transactions, 108: 381–392.

Brun, X., Belgharbi, M., Sesmat, S., Thomasset, D., and Scavarda, S., 1999. Control of an electropneumatic actuator: Comparison between some linear and non-linear control laws. Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering, 213(5): 387–406.

Cao, J., Xie, S. Q., Das, R., and Zhu, G. L., 2014. Control strategies for effective robot assisted gait rehabilitation: The state of art and future prospects. Medical Engineering and Physics, 36(12): 1555–1566.

Carneiro, J. F., and De Almeida, F. G., 2007. Heat transfer evaluation of industrial pneumatic cylinders. Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering, 221(1): 119–128.

Chen, T., Casas, R., and Lum, P. S., 2019. An Elbow Exoskeleton for Upper Limb Rehabilitation with Series Elastic Actuator and Cable-Driven Differential. IEEE Transactions on Robotics, 35(6): 1464–1474.

Dağdelen, M., 2017. Development of a Conceptual Model for Wrist and Forearm Rehabilitation Robot with Two Degrees of Freedom. Master Degree Thesis, Çukurova University, 102 pages.

Dağdelen, M., and Sarıgeçili, M. İ., 2020. Estimation of the Friction Parameters of Linear Pneumatic Cylinders. Mühendislik ve Tasarım Bilimleri Dergisi, 8(2): 397-406.

Dağdelen, M., and Sarıgeçili, M. İ., 2021. Evaluation of Cracking Pressure Ratio at Proportional Directional Control Valves for Different Valve Control Signals and Supply Pressures. III. International Turkic World Congress on Science and Engineering, Niğde, Turkey, 849-856.

Dağdelen, M., Sarıgeçili, M. İ., and Özbek, N. S., 2022. Interaction Torque Control of Pneumatically Actuated Wrist/Forearm Rehabilitation Robot. II. International Antalya Scientific Research and Innovative Studies Congress, Antalya, Turkey, 546-555.

Dağdelen, M., Sarıgeçili, M. İ., and Ozbek, N. S., 2023a. A new compressible flow model for pneumatic directional control valves. Proceedings of the Institution of Mechanical Engineers Part I-Journal of Systems and Control Engineering, 237(2): 179-195.

Dağdelen, M., Sarigeçili, M. İ., and Özbek, N. S., 2023b. A cascade fuzzy adaptive based interaction torque control of a pneumatically actuated forearm rehabilitation robot under disturbance effects. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science "in press".

Dağdelen, M., and Sarıgeçili, M. İ., 2023a. Automation of friction torque identification for vane-type semi-rotary pneumatic actuators. Journal of the Brazilian Society of Mechanical Sciences and Engineering, 45(2023): 1-17.

Dağdelen, M., and Sarıgeçili, M. İ., 2023b. Friction force estimation in pneumatic cylinders by full automation of experimental procedures. Transactions of the Institute of Measurement and

Control, 45(5): 986-997.

de Gaitani, F. H. M., dos Santos, W. M., and Siqueira, A. A. G., 2022. Design and Performance Analysis of a Compact Series Elastic Actuator for Exoskeletons. Journal of Control, Automation and Electrical Systems, 33(3): 1012–1021.

de Wit, C. C., Olsson, H., Astrom, K. J., and Lischinsky, P., 1995. A New Model of Control Systems with Friction. IEEE Transactions On Automatic Control, 40(3): 419–425.

dos Santos, W. M., and Siqueira, A. A. G., 2019. Optimal impedance via model predictive control for robot-aided rehabilitation. Control Engineering Practice, 93(October): 104177.

Gopura, R. A. R. C., and Kiguchi, K., 2008. A human forearm and wrist motion assist exoskeleton robot with EMG-based fuzzy-neuro control. Proceedings of the 2nd Biennial IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, BioRob, 2008: 550–555.

Guo, W., Wang, T., and Wang, Q., 2013. A passive dynamic walking model with Coulomb friction at the hip joint. Robotica, 31(8): 1221–1227.

Gupta, A., and O'Malley, M. K., 2006. Design of a haptic arm exoskeleton for training and rehabilitation. IEEE/ASME Transactions on Mechatronics, 11(3): 280–289.

Gupta, A., O'Malley, M. K., Patoglu, V., and Burgar, C., 2008. Design, control and performance of RiceWrist: A force feedback wrist exoskeleton for rehabilitation and training. International Journal of Robotics Research, 27(2): 233–251.

Harris, P. G., O'Donnell, G. E., and Whelan, T., 2012. Modelling and identification of industrial pneumatic drive system. International Journal of Advanced Manufacturing Technology, 58(9–12): 1075–1086.

Huang, Y. C., Hu, H., Chen, C. H., Chen, Y. J., and Chi, K. C., 2021. Assistive Robot Design for Lower Limbs Rehabilitation Using Fuzzy Control. International Journal of Fuzzy Systems, 23(8): 2384–2395.

Kalita, B., and Dwivedy, S. K., 2020. Nonlinear dynamic response of pneumatic artificial muscle: A theoretical and experimental study. International Journal of Non-Linear Mechanics, 125(June): 103544.

Kawashima, K., Ishii, Y., Funaki, T., and Kagawa, T., 2004. Determination of flow rate characteristics of pneumatic solenoid valves using an isothermal chamber. Journal of Fluids Engineering, Transactions of the ASME, 126(2): 273–279.

Kawashima, K., Kagawa, T., and Fujita, T., 2000. Instantaneous flow rate measurement of ideal gases. Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME, 122(1): 174–178.

Keck, A., Zimmermann, J., and Sawodny, O., 2017. Friction parameter identification and compensation using the ElastoPlastic friction model. Mechatronics, 47: 168–182.

Kosari, H., and Moosavian, S. A. A., 2015. Friction compensation in a pneumatic actuator using

recursive least square algorithm. 2015 Australian Control Conference, AUCC 2015: 81–86.

Krebs, H. I., Volpe, B. T., Williams, D., Celestino, J., Charles, S. K., Lynch, D., and Hogan, N. ,2007. Robot-aided neurorehabilitation: A robot for wrist rehabilitation. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 15(3): 327–335.

Liang, J., Fillmore, S., and Ma, O., 2012. An extended bristle friction force model with experimental validation. Mechanism and Machine Theory, 56: 123–137.

Lin, W., Guan, R., Yuan, L., Li, Z., and Tong, M., 2018. Position feedback dynamic surface control for pneumatic actuator position servo system. Systems Science and Control Engineering, 6(1): 388–397.

Lin, Z., Zhang, T., Xie, Q., and Wei, Q., 2018. Electro-pneumatic position tracking control system based on an intelligent phase-change PWM strategy. Journal of the Brazilian Society of Mechanical Sciences and Engineering, 40(11): 1–17.

Liu, Q., Zuo, J., Zhu, C., and Xie, S. Q., 2020. Design and control of soft rehabilitation robots actuated by pneumatic muscles: State of the art. Future Generation Computer Systems, 113: 620–634.

Martinez-Peon, D., Olguín-Díaz, E., Muñoz-Vázquez, A. J., Francisco, P. C., and Méndez, D. S., 2021. Modeling and control of exoskeleton for wrist and forearm rehabilitation. Biomedical Signal Processing and Control, 70(August): 103022.

Martinez, J. A., Ng, P., Lu, S., Campagna, M. S., and Celik, O., 2013. Design of Wrist Gimbal: A forearm and wrist exoskeleton for stroke rehabilitation. IEEE International Conference on Rehabilitation Robotics.

Mashayekhi, A., Behbahani, S., Nahvi, A., Keshmiri, M., and Shakeri, M., 2022. Analytical describing function of LuGre friction model. International Journal of Intelligent Robotics and Applications, 6: 437-448.

Masia, L., Rodriguez, N. N., Casadio, M., Morasso, P., Sandini, G., and Giannoni, P., 2009. Adaptive training strategy of distal movements by means of a wrist-robot. Proceedings of the 2nd International Conferences on Advances in Computer-Human Interactions, ACHI 2009: 227–233.

Mayetin, U., and Kucuk, S. (2022). Design and Experimental Evaluation of a Low Cost, Portable, 3-DOF Wrist Rehabilitation Robot with High Physical Human–Robot Interaction. Journal of Intelligent and Robotic Systems: Theory and Applications, 65(2022).

Mekki, M., Delgado, A. D., Fry, A., Putrino, D., and Huang, V., 2018. Robotic Rehabilitation and Spinal Cord Injury: a Narrative Review. Neurotherapeutics, 15(3): 604–617.

Meng, D., Tao, G., Chen, J., and Ban, W., 2011. Modeling of a pneumatic system for high-accuracy position control. Proceedings of 2011 International Conference on Fluid Power and Mechatronics, FPM 2011: 505–510.

Miao, Q., Zhang, M., Cao, J., and Xie, S. Q., 2018. Reviewing high-level control techniques on

robot-assisted upper-limb rehabilitation. Advanced Robotics, 32(24): 1253–1268.

Ning, S., and Bone, G. M., 2005. Development of a nonlinear dynamic model for a servo pneumatic positioning system. IEEE International Conference on Mechatronics and Automation, ICMA 2005, July: 43–48.

Nouri, B. M. Y., 2004. Friction identification in mechatronic systems. ISA Transactions, 43(2): 205–216.

Nouri, B. M. Y., and Saudi, M. B. Y., 2014. Experimental Modelling and Identification of Compressible Flow through Proportional Directional Control Valves. Universal Journal of Control and Automation, 2(1): 4–13.

Oblak, J., Cikajlo, I., and Matjačić, Z., 2009. A universal haptic device for arm and wrist rehabilitation. 2009 IEEE International Conference on Rehabilitation Robotics, ICORR 2009, 1: 436–441.

Odabaş, C., and Morgül, Ö., 2020. Adaptive friction compensations for mechanical systems with measurement delay. Transactions of the Institute of Measurement and Control, 43(8): 1745–1759.

Ohara, K., Iwazawa, R., and Kaneko, M., 2022. Modeling and Analysis of a High-Speed Adjustable Grasping Robot Controlled by a Pneumatic Actuator. Robotics, 11(1): 1–16.

Özbek, N. S., and Eker, İ., 2019. Design of an optimal fractional fuzzy gain-scheduled Smith Predictor for a time-delay process with experimental application. ISA Transactions, 97: 14–35.

Pehlivan, A. U., Celik, O., and O'Malley, M. K., 2011. Mechanical design of a distal arm exoskeleton for stroke and spinal cord injury rehabilitation. IEEE International Conference on Rehabilitation Robotics, 1–5.

Pehlivan, A. U., Lee, S., and O'Malley, M. K., 2012. Mechanical design of RiceWrist-S: A forearm-wrist exoskeleton for stroke and spinal cord injury rehabilitation. Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics, 1573–1578.

Piatkowski, T., 2014. Dahl and LuGre dynamic friction models - The analysis of selected properties. Mechanism and Machine Theory, 73: 91–100.

Qian, P., Pu, C., He, D., Lv, P., and Ruiz Páez, L. M., 2022. A method to improve the motion trajectory tracking accuracy of pneumatic servo system—by exciting longitudinal resonance. Journal of the Brazilian Society of Mechanical Sciences and Engineering, 44: 376(2022).

Rad, C. R., and Hancu, O., 2017. An improved nonlinear modelling and identification methodology of a servo-pneumatic actuating system with complex internal design for high-accuracy motion control applications. Simulation Modelling Practice and Theory, 75: 29–47.

Rad, C. R., Hancu, O., Maties, V., and Lapusan, C., 2014. Parameter identification and modeling of

a pneumatic proportional valve with applicability in control design of servo-pneumatic systems. Applied Mechanics and Materials, *658*: 700–705.

Rahman, H. A., Fai, Y. C., and Ming, E. S. L., 2014. Analysis of human hand kinematics: Forearm pronation and supination. Journal of Medical Imaging and Health Informatics, 4(2): 245–249.

Rao, Z., and Bone, G. M., 2008. Nonlinear modeling and control of servo pneumatic actuators. IEEE Transactions on Control Systems Technology, 16(3): 562–569.

Richer, E., and Hurmuzlu, Y., 2000. A high performance pneumatic force actuator system: Part I—nonlinear mathematical model. Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME, 122(3): 416–425.

Saha, A., Wahi, P., Wiercigroch, M., and Stefański, A., 2016. A modified LuGre friction model for an accurate prediction of friction force in the pure sliding regime. International Journal of Non-Linear Mechanics, 80: 122–131.

Saleem, A., Al Ratrout, S., and Wong, C. B., 2017. Nonlinear component-based modelling and hardware-in-the loop simulation of servo-pneumatic systems. International Journal of Modelling, Identification and Control, 28(1): 40–57.

Sanjeevi, N. S. S., and Vashista, V., 2021. Stiffness modulation of a cable-driven leg exoskeleton for effective human-robot interaction. Robotica, 39(12): 2172–2192.

Sarhan, S. M., Al-Faiz, M. Z., and Takhakh, A. M., 2023. A review on EMG/EEG based control scheme of upper limb rehabilitation robots for stroke patients. Heliyon, *9*(8): e18308.

Schlüter, M. S., and Perondi, E. A., 2018. Mathematical modeling of pneumatic semi-rotary actuator with friction. Journal of the Brazilian Society of Mechanical Sciences and Engineering, 40(11): 1–17.

Sharifi, M., Azimi, V., Mushahwar, V. K., and Tavakoli, M., 2022. Impedance Learning-Based Adaptive Control for Human-Robot Interaction. IEEE Transactions on Control Systems Technology, 30(4): 1345–1358.

Sharifi, M., Behzadipour, S., Salarieh, H., and Tavakoli, M., 2017. Cooperative modalities in robotic tele-rehabilitation using nonlinear bilateral impedance control. Control Engineering Practice, 67(August): 52–63.

Sharifi, M., Behzadipour, S., and Vossoughi, G., 2014. Nonlinear model reference adaptive impedance control for human-robot interactions. Control Engineering Practice, 32: 9–27.

Sharma, R., Gaur, P., Bhatt, S., and Joshi, D., 2021. Optimal fuzzy logic-based control strategy for lower limb rehabilitation exoskeleton. Applied Soft Computing, 105: 107226.

Shi, D., Zhang, W., Zhang, W., Ju, L., and Ding, X., 2021. Human-centred adaptive control of lower limb rehabilitation robot based on human–robot interaction dynamic model. Mechanism and Machine Theory, 162: 104340.

Tran, X. B., and Yanada, H., 2013. Dynamic Friction Behaviors of Pneumatic Cylinders. Intelligent

Control and Automation, 04(02): 180–190.

Valdiero, A. C., Ritter, C. S., Rios, C. F., and Rafikov, M., 2011. Nonlinear mathematical modeling in pneumatic servo position applications. Mathematical Problems in Engineering, 2011.

Van Der Merwe, J., Muller, J. H., and Scheffer, C., 2013. Parameter Identification and Evaluation of a Proportional Directional Flow Control Valve Model. Journal, of the South African Institution of Mechanical Engineering, 29(September): 18–25.

Wang, J., Wang, J. D., Daw, N., Wu, Q. H., and Member, S., 2004. Identification of Pneumatic Cylinder Friction Parameters Using Genetic Algorithms. IEEE/ASME Transactions on Mechatronics, 9(1): 100–107.

Wang, Y., Wang, H., and Tian, Y., 2022. Adaptive interaction torque-based AAN control for lower limb rehabilitation exoskeleton. ISA Transactions, 128: 184–197.

Wendong, W., Hanhao, L., Menghan, X., Yang, C., Xiaoqing, Y., Xing, M., and Bing, Z., 2020. Design and verification of a human–robot interaction system for upper limb exoskeleton rehabilitation. Medical Engineering and Physics, 79: 19–25.

Wu, K. Y., Su, Y. Y., Yu, Y. L., Lin, C. H., and Lan, C. C., 2019. A 5-Degrees-of-Freedom Lightweight Elbow-Wrist Exoskeleton for Forearm Fine-Motion Rehabilitation. IEEE/ASME Transactions on Mechatronics, 24(6): 2684–2695.

Xiangrong, S., and Goldfarb, M., 2007. Simultaneous force and stiffness control of a pneumatic actuator. Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME, 129(4): 425–434.

Xu, W., Chu, B., and Rogers, E., 2014. Iterative learning control for robotic-assisted upper limb stroke rehabilitation in the presence of muscle fatigue. Control Engineering Practice, 31: 63–72.

Yellewa, M. E., Mohamed, A., Ishii, H., and Assal, S. F. M., 2022. Design and Hybrid Impedance Control of a Compliant and Balanced Wrist Rehabilitation Device. IECON Proceedings (Industrial Electronics Conference), 2022-Octob: 1–6.

Yerlikaya, U., and Balkan, T., 2018. Identification of viscous and coulomb friction in motion constrained systems. IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM, 2018-July: 91–96.

Yu, H., Huang, S., Chen, G., Pan, Y., and Guo, Z., 2015. Human-Robot Interaction Control of Rehabilitation Robots with Series Elastic Actuators. IEEE Transactions on Robotics, 31(5): 1089–1100.

Yue, F., Li, X., Chen, C., and Tan, W., 2017. Adaptive integral backstepping sliding mode control for opto-electronic tracking system based on modified LuGre friction model. International Journal of Systems Science, 48(16), 3374–3381.

Zhang, L., Guo, S., and Sun, Q., 2020. Development and assist-as-needed control of an end-effector upper limb rehabilitation robot. Applied Sciences (Switzerland), 10(19).

CURRICULUM VITAE

Mustafa DAĞDELEN is currently working as a research assistant at Çukurova University Faculty of Engineering, Department of Mechanical Engineering. His areas of interest are Mechanisms, Machine Theory and Dynamics, Control Systems, Simulation and also Pneumatic Actuator Systems.

Education

2017- : Çukurova University (PhD-Mechanical Engineering)
2014-2017 : Çukurova University (MSc-Mechanical Engineering)
2009-2014 : Gaziantep University (BSc-Mechanical Engineering)

Working Experience

2014- : Çukurova University, Engineering Faculty, Research assistant

APPENDICES

APPENDIX 1

The script code written for the automation of friction torque identification process for vane-type semi rotary pneumatic cylinder:

```
function StartButtonPushed(app, event)
    set_param('rotary_friction_updated', 'SimulationCommand', 'start')
    value_1=app.RangeofmotionROMEditField.Value;
    set_param('rotary_friction_updated/rom','Value',num2str(value_1));
    value_2=app.VaneareaA_vEditField.Value;
    value_3=app.TorqueradiusrEditField.Value;
    set_param('rotary_friction_updated/vane area','Value',num2str(value_2));
    set_param('rotary_friction_updated/torque radius','Value',num2str(value_3));
    value_4=app.oftestsforstaticexpNsEditField.Value;
    set_param('rotary_friction_updated/static experiment no','Value',num2str(value_4));
    value_5=app.oftestsfordynamicexpNdEditField.Value;
    set_param('rotary_friction_updated/dynamic experiment no','Value',num2str(value_5));
    value_6=app.AbsoluteworkingpressureP_wEditField.Value;
    set_param('rotary_friction_updated/working pressure','Value',num2str(value_6));

    app.StopButton.Enable = 'on';
    app.StartButton.Enable = 'off';
    app.DisconnectButton.Enable = 'off';
    app.ConnectButton.Enable='off';
    app.CleardefinedparametersButton.Enable='off';
    app.EditField.Value="Experiments are started, please wait to complete!";

  end

function StopButtonPushed(app, event)
    set_param('rotary_friction_updated', 'SimulationCommand', 'stop');
    app.EditField.Value="Experiment process is ended and system is disconnected!";
    app.CleardefinedparametersButton.Enable='on';
    app.clicktoESTIMATEButton.Enable = 'on';
    app.DisconnectButton.Enable = 'off';
    app.StartButton.Enable = 'off';
    app.StopButton.Enable = 'off';
```

```
            app.ConnectButton.Enable='off';
            app.EditField.BackgroundColor = [1.00,1.00,1.00];
        end
function ConnectButtonPushed(app, event)
            set_param('rotary_friction_updated', 'SimulationCommand', 'connect');
            app.DisconnectButton.Enable = 'on';
            app.StartButton.Enable = 'on';
            app.StopButton.Enable  = 'off';
            app.ConnectButton.Enable='off';
            app.CleardefinedparametersButton.Enable='off';
            app.EditField.Value= "Connection is successfull, Press Start button to continue!";
            app.EditField.BackgroundColor = [1.00,1.00,1.00];
        end

function DisconnectButtonPushed(app, event)
            set_param('rotary_friction_updated', 'SimulationCommand', 'disconnect')
            app.EditField.BackgroundColor = [1.00 1.00 1.00];
            app.EditField.Value="System is disconnected!";
            app.CleardefinedparametersButton.Enable='on';
            app.DisconnectButton.Enable = 'off';
            app.StartButton.Enable = 'off';
            app.StopButton.Enable = 'off';
        end

function clicktoESTIMATEButtonPushed(app, event)

clc;

load('Static_CCW.mat');
load('Static_CW.mat');
load('Dynamic_CCW.mat');
load('Dynamic_CW.mat');
load('Speed_CCW.mat');
load('Speed_CW.mat');
```

```
a=size(T_f_max_CCW);
b=size(T_f_max_CW);
c=size(T_f_CCW_dyn);
w=size(T_f_CW_dyn);

speed_size_CCW=size(S_CCW);
speed_size_CW=size(S_CW);

CCW_T_s=zeros(1,a(2));
CW_T_s=zeros(1,b(2));

CCW_T_s=T_f_max_CCW(2,:);
CW_T_s=T_f_max_CW(2,:);

q=0;
for i=1:c(2)
   if(T_f_CCW_dyn(2,i)~=0 && isnan(T_f_CCW_dyn(2,i))==0)
      q=q+1;
      Dyn_CCW(q)=T_f_CCW_dyn(2,i);
   end
end

p=0;
for i=1:w(2)
   if(T_f_CW_dyn(2,i)~=0 && isnan(T_f_CW_dyn(2,i))==0)
      p=p+1;
      Dyn_CW(p)=T_f_CW_dyn(2,i);
   end
end

d=size(Dyn_CCW);
e=size(Dyn_CW);
speed_CCW=zeros(1,d(2));
speed_CW=zeros(1,e(2));
```

```
v=0;
for i=1:speed_size_CCW(2)
   if(S_CCW(2,i)~=0 && isnan(S_CCW(2,i))==0)
      v=v+1;
      speed_CCW(v)=S_CCW(2,i);
   end
end

z=0;
for i=1:speed_size_CW(2)
   if(S_CW(2,i)~=0 && isnan(S_CW(2,i))==0)
      z=z+1;
      speed_CW(z)=S_CW(2,i);
   end
end

j=0;
for i=1:a(2)
   if CCW_T_s(1,i)>0
      j=j+1;
      Real_T_s_CCW(1,j)=CCW_T_s(i);
   end
end

k=0;
for t=1:b(2)
   if CW_T_s(1,t)<0
      k=k+1;
      Real_T_s_CW(1,k)=CW_T_s(t);
   end
end

%sp_CCW=zeros(1,j);
%speed_CW=zeros(1,k);
CCW_T_s_total=sum(T_f_max_CCW(2,:));
CW_T_s_total=sum(T_f_max_CW(2,:));
average_CCW_T_s=CCW_T_s_total/j;
average_CW_T_s=CW_T_s_total/k;
```

```
sum_CCW=0;
dif_CCW_2=0;
for i=1:j
   dif_CCW_2= (Real_T_s_CCW(i)-average_CCW_T_s)^2;
   sum_CCW=sum_CCW+dif_CCW_2;
   s_d_CCW=((sum_CCW/j))^0.5;
end

sum_CW=0;
dif_CW_2=0;
for i=1:k
   dif_CW_2= (Real_T_s_CW(i)-average_CW_T_s)^2;
   sum_CW=sum_CW+dif_CW_2;
   s_d_CW=((sum_CW/k))^0.5;
end

sum_CCW=0;
dif_CCW_2=0;
for i=1:j
   dif_CCW_2= (Real_T_s_CCW(i)-average_CCW_T_s)^2;
   sum_CCW=sum_CCW+dif_CCW_2;
   s_d_CCW=((sum_CCW/j))^0.5;
end

sum_CW=0;
dif_CW_2=0;
for i=1:k
   dif_CW_2= (Real_T_s_CW(i)-average_CW_T_s)^2;
   sum_CW=sum_CW+dif_CW_2;
   s_d_CW=((sum_CW/k))^0.5;
end

t=1;
Real_Dyn_CCW(1)=Dyn_CCW(1);
for y=1:(d(2)-1)
   if ((Dyn_CCW(y+1)~=Dyn_CCW(y)))
      t=t+1;
```

```
        Real_Dyn_CCW(t)= Dyn_CCW(y+1);

    end
end

n=1;
Real_Dyn_CW(1)=Dyn_CW(1);

for y=1:(e(2)-1)
    if ((Dyn_CW(y+1)~=Dyn_CW(y)))
        n=n+1;
        Real_Dyn_CW(n)= Dyn_CW(y+1);
     end
end

g=1;
Real_Speed_CCW(1)=speed_CCW(1);

for y=1:(d(2)-1)
    if ((speed_CCW(y+1)~=speed_CCW(y)))
        g=g+1;
        Real_Speed_CCW(g)=speed_CCW(y+1);
    end
end

s=1;
Real_Speed_CW(1)=speed_CW(1);

for y=1:(e(2)-1)
    if ((speed_CW(y+1)~=speed_CW(y)))
        s=s+1;
        Real_Speed_CW(s)=speed_CW(y+1);
    end
end
```

```
T_s_CCW=average_CCW_T_s;
T_s_CW=average_CW_T_s;
tt=size(Real_Speed_CCW)+1;
ttl=size(Real_Speed_CW)+1;
linear_speed_CCW = zeros(1,tt(2));
linear_speed_CW= zeros(1,ttl(2));
T_f_CCW=zeros(1,tt(2));
T_f_CW=zeros(1,ttl(2));
linear_speed_CCW(1)=0;
linear_speed_CW(1)=0;
T_f_CCW(1)=T_s_CCW;
T_f_CW(1)=T_s_CW;
linear_speed_CCW(2:tt(2))= Real_Speed_CCW();
linear_speed_CW(2:ttl(2))=Real_Speed_CW();
T_f_CCW(2:tt(2))=Real_Dyn_CCW;
T_f_CW(2:ttl(2))=Real_Dyn_CW;

my_fun_CCW=@(alfa,linear_speed_CCW) (alfa(1)+((T_s_CCW-
alfa(1))./exp((linear_speed_CCW/alfa(2)).^2))+linear_speed_CCW.*alfa(3));
intial_values_CCW=[1,1,2];
lb_CCW=[0,0,0];
ub_CCW=[T_s_CCW,1,2];
options_CCW=optimoptions('lsqcurvefit','Algorithm','levenberg-marquardt');
new_parameters_CCW=lsqcurvefit(my_fun_CCW,intial_values_CCW,linear_speed_CCW,T_f_C
CW,lb_CCW,ub_CCW,options_CCW);

my_fun_CW=@(beta,linear_speed_CW) (beta(1)+(T_s_CW-
beta(1))./exp((linear_speed_CW/beta(2)).^2)+linear_speed_CW.*beta(3));
intial_values_CW=[-1, -1, 2];
lb_CW=[T_s_CW, -1, 0];
ub_CW=[0, 0, 2];
options_CW=optimoptions('lsqcurvefit','Algorithm','levenberg-marquardt');
new_parameters_CW=lsqcurvefit(my_fun_CW,intial_values_CW,linear_speed_CW,T_f_CW,lb_
CW,ub_CW,options_CW);
```

```
app.TsNmEditField.Value=num2str(T_s_CCW);
app.SDforTsdataEditField.Value=num2str(s_d_CCW);
app.TcNmEditField.Value=num2str(new_parameters_CCW(1));
app.BNsmradEditField.Value=num2str(new_parameters_CCW(3));
app.wsradsEditField.Value=num2str(new_parameters_CCW(2));

app.TsNmEditField_2.Value=num2str(T_s_CW);
app.SDforTsdataEditField_2.Value=num2str(s_d_CW);
app.TcNmEditField_2.Value=num2str(new_parameters_CW(1));
app.BNsmradEditField_2.Value=num2str(new_parameters_CW(3));
app.wsradsEditField_2.Value=num2str(new_parameters_CW(2));

sp_CCW=zeros(1,j);
sp_CW=zeros(1,k);

figure=app.UIAxes;
plot(figure,sp_CCW,Real_T_s_CCW,'k^');
hold(figure,'on');
plot(figure,sp_CW,Real_T_s_CW,'k^');

plot(figure,0,average_CCW_T_s,'ko');
plot(figure,0,average_CW_T_s,'ko');
grid(figure,'on');
ylim(figure, [(min(T_f_CW)-1.2) (max(T_f_CCW)+1.2)]);
xlim(figure,  [-12 12]);

d=0:0.001:max(linear_speed_CCW);
t=0:-0.001:min(linear_speed_CW);
plot(figure,Real_Speed_CCW,Real_Dyn_CCW,'k*',Real_Speed_CW, Real_Dyn_CW,'k*');
new_T_f_CCW=my_fun_CCW(new_parameters_CCW, d);
new_T_f_CW=my_fun_CW(new_parameters_CW, t);

plot(figure,d,new_T_f_CCW,'k-','LineWidth',1.1);
hold(figure,'on');
plot(figure,t,new_T_f_CW,'k-','LineWidth',1.1);
```

```
    end
function RangeofmotionROMEditFieldValueChanged(app, event)
        value = app.RangeofmotionROMEditField.Value;

        if value>0 && value<=360
            app.VaneareaA_vEditField.Editable="on";
            app.EditField.Value="Enter the vane area of the actuator";
            app.RangeofmotionROMEditField.BackgroundColor=[0 1 0];
            app.EditField.FontColor=[0 0 0];
            app.RangeofmotionROMEditField.Editable="off";
        else if value<0 || value>360
            app.VaneareaA_vEditField.Editable="off";
            app.EditField.Value="Please enter a reasonable value! It should be between 0-360
degrees.";
            app.EditField.FontColor=[1 0 0];
            app.RangeofmotionROMEditField.Editable="on";
        end
        end
    end
function VaneareaA_vEditFieldValueChanged(app, event)
        value = app.VaneareaA_vEditField.Value;

        if value>0
            app.TorqueradiusrEditField.Editable="on";
            app.EditField.Value="Enter the torque radius of the actuator";
            app.VaneareaA_vEditField.Editable="off";
            app.EditField.FontColor=[0 0 0];
            app.VaneareaA_vEditField.BackgroundColor=[0 1 0];
        else if value<0
            app.TorqueradiusrEditField.Editable="off";
            app.EditField.Value="Please enter a reasonable value!";
            app.EditField.FontColor=[1 0 0];
            app.VaneareaA_vEditField.Editable="on";
```

```
        end
        end
    end
function TorqueradiusrEditFieldValueChanged(app, event)
        value = app.TorqueradiusrEditField.Value;

        if value>0
            app.oftestsforstaticexpNsEditField.Editable="on";
            app.EditField.Value="Specify the number of static experiments [suggested as >=30]";
            app.TorqueradiusrEditField.Editable="off";
            app.EditField.FontColor=[0 0 0];
            app.TorqueradiusrEditField.BackgroundColor=[0 1 0];
        else if value<0
                app.oftestsforstaticexpNsEditField.Editable="off";
                app.EditField.Value="Please enter a reasonable value!";
                app.EditField.FontColor=[1 0 0];
                app.TorqueradiusrEditField.Editable="on";
        end
        end
    end
function CleardefinedparametersButtonPushed(app, event)
        app.RangeofmotionROMEditField.Value=0;
        app.VaneareaA_vEditField.Value=0;
        app.TorqueradiusrEditField.Value=0;
        app.oftestsforstaticexpNsEditField.Value=0;
        app.oftestsfordynamicexpNdEditField.Value=0;
        app.AbsoluteworkingpressureP_wEditField.Value=0;

        app.UIAxes.cla;

        app.TsNmEditField.Value=['-'];
        app.SDforTsdataEditField.Value=['-'];
```

```
app.TcNmEditField.Value=['-'];
app.wsradsEditField.Value=['-'];
app.BNsmradEditField.Value=['-'];

app.TsNmEditField_2.Value=['-'];
app.SDforTsdataEditField_2.Value=['-'];
app.TcNmEditField_2.Value=['-'];
app.wsradsEditField_2.Value=['-'];
app.BNsmradEditField_2.Value=['-'];

app.EditField.Value="Please enter the required parameters! Start with range of motion
(ROM).";
app.RangeofmotionROMEditField.Editable="on";
app.VaneareaA_vEditField.Editable="off";
app.TorqueradiusrEditField.Editable="off";
app.oftestsforstaticexpNsEditField.Editable="off";
app.oftestsfordynamicexpNdEditField.Editable="off";
app.AbsoluteworkingpressureP_wEditField.Editable="off";

app.ConnectButton.Enable="off";
app.StartButton.Enable="off";
app.DisconnectButton.Enable="off";
app.StopButton.Enable="off";
app.clicktoESTIMATEButton.Enable="off";

app.RangeofmotionROMEditField.BackgroundColor=[1 1 1];
app.VaneareaA_vEditField.BackgroundColor=[1 1 1];
app.TorqueradiusrEditField.BackgroundColor=[1 1 1];
app.EditField.BackgroundColor=[1 1 1];
app.oftestsforstaticexpNsEditField.BackgroundColor=[1 1 1];
app.oftestsfordynamicexpNdEditField.BackgroundColor=[1 1 1];
app.AbsoluteworkingpressureP_wEditField.BackgroundColor=[1 1 1];
```

```
    end
function oftestsforstaticexpNsEditFieldValueChanged(app, event)
        value = app.oftestsforstaticexpNsEditField.Value;

        if value>=30
            app.oftestsforstaticexpNsEditField.Editable="off";
            app.oftestsforstaticexpNsEditField.BackgroundColor=[0 1 0];
            app.oftestsfordynamicexpNdEditField.Editable="on";
            app.EditField.FontColor=[0 0 0];
            app.EditField.Value="Specify the number of dynamic experiments [suggested as >=30]";
        else
            app.EditField.Value="Value cannot be lower than 30, login new value!";
            app.EditField.FontColor=[1 0 0];
            app.oftestsfordynamicexpNdEditField.Editable="off";
        end

    end
    function oftestsfordynamicexpNdEditFieldValueChanged(app, event)
        value = app.oftestsfordynamicexpNdEditField.Value;

         if value>=30
            app.oftestsfordynamicexpNdEditField.Editable="off";
            app.oftestsfordynamicexpNdEditField.BackgroundColor=[0 1 0];
            app.EditField.FontColor=[0 0 0];
            app.EditField.Value= "Specify the working pressure [cannot be lower than 1]";
            app.AbsoluteworkingpressureP_wEditField.Editable="on";
        else
            app.EditField.Value="Value cannot be lower than 30, login new value!";
            app.EditField.FontColor=[1 0 0];
            app.AbsoluteworkingpressureP_wEditField.Editable="off";
        end
```

```
    end
function AbsoluteworkingpressureP_wEditFieldValueChanged(app, event)
        value = app.AbsoluteworkingpressureP_wEditField.Value;

        if value>=1 && value<=8
            app.AbsoluteworkingpressureP_wEditField.Editable="off";
            app.AbsoluteworkingpressureP_wEditField.BackgroundColor=[0 1 0];
            app.EditField.FontColor=[0 0 0];
            app.EditField.Value="Parameters has been defined successfully, please connect to
proceed.";
            app.ConnectButton.Enable="on";
        else
           app.EditField.Value="Working pressure can be min. 1 bar and max. 8 bar!";
            app.EditField.FontColor=[1 0 0];
           app.ConnectButton.Enable="off";
        end
    end
```

APPENDIX 2

The script code written for the automation of friction force identification process for double-acting linear pneumatic cylinder:

function STARTButtonPushed(app, event)

```
set_param('Automatic_friction_estimation_with_verification', 'SimulationCommand', 'start')
        value_1=app.CylinderStrokeLEditField.Value;

set_param('Automatic_friction_estimation_with_verification/stroke','Value',num2str(value_1));
        value_2=app.PistondiameterPDEditField.Value;
        set_param('Automatic_friction_estimation_with_verification/pd','Value',num2str(value_2));
        value_3=app.RoddiameterRDEditField.Value;
        set_param('Automatic_friction_estimation_with_verification/rd','Value',num2str(value_3));
        value_4=app.ofstaticexperimentEditField.Value;

set_param('Automatic_friction_estimation_with_verification/interval','Value',num2str(value_4));
        value_6=app.ofdynamicexperimnetEditField.Value;

set_param('Automatic_friction_estimation_with_verification/repetition','Value',num2str(value_6));


        app.STOPButton.Enable = 'on';
        app.STARTButton.Enable = 'off';
        app.DISCONNECTButton.Enable = 'off';
        app.CONNECTButton.Enable='off';
        app.CLEARALLButton.Enable='off';
        app.EditField.Value=['Experiments are started, please wait to complete!'];
         app.EditField.BackgroundColor = [1.00,1.00,1.00];
    end
```

function STOPButtonPushed(app, event)

```
         set_param('Automatic_friction_estimation_with_verification', 'SimulationCommand',
'stop');
         app.EditField.Value=['Experiment process is ended and system is disconnected!'];
         app.CLEARALLButton.Enable='on';
```

```
            app.CALCULATEButton.Enable = 'on';
            app.DISCONNECTButton.Enable = 'off';
            app.STARTButton.Enable = 'off';
            app.STOPButton.Enable = 'off';
            app.CONNECTButton.Enable='off';
            app.EditField.BackgroundColor = [1.00,1.00,1.00];
        end
    function CALCULATEButtonPushed(app, event)
        app.CALCULATEButton.Enable = 'off';
        clc;

load('Static_ext.mat');
load('Static_ret.mat');
load('Dynamic_ext.mat');
load('Dynamic_ret.mat');
load('Speed_ext.mat');
load('Speed_ret.mat');

a=size(F_s_ext);
b=size(F_s_ret);
c=size(D_e);
w=size(D_r);

speed_size_e=size(S_e);
speed_size_r=size(S_r);

Ext_F_s=zeros(1,a(2));
Ret_F_s=zeros(1,b(2));



Ext_F_s=F_s_ext(2,:);
Ret_F_s=F_s_ret(2,:);
```

```
q=0;
for i=1:c(2)
   if(D_e(2,i)~=0 && isnan(D_e(2,i))==0)
      q=q+1;
      Dyn_ext(q)=D_e(2,i);
   end
end

p=0;
for i=1:w(2)
   if(D_r(2,i)~=0 && isnan(D_r(2,i))==0)
      p=p+1;
      Dyn_ret(p)=D_r(2,i);
   end
end

d=size(Dyn_ext);
e=size(Dyn_ret);
speed_e=zeros(1,d(2));
speed_r=zeros(1,e(2));

v=0;
for i=1:speed_size_e(2)
   if(S_e(2,i)~=0 && isnan(S_e(2,i))==0)
      v=v+1;
      speed_e(v)=S_e(2,i);
   end
end

z=0;
for i=1:speed_size_r(2)
   if(S_r(2,i)~=0 && isnan(S_r(2,i))==0)
```

```
        z=z+1;
        speed_r(z)=S_r(2,i);
    end
end

j=0;
for i=1:a(2)
    if Ext_F_s(1,i)>0
        j=j+1;
        Real_F_s_ext(1,j)=Ext_F_s(i);
    end
end

k=0;
for t=1:b(2)
    if Ret_F_s(1,t)<0
        k=k+1;
        Real_F_s_ret(1,k)=Ret_F_s(t);
    end
end

sp_e=zeros(1,j);
speed_ret=zeros(1,k);
E_F_s_total=sum(F_s_ext(2,:));
R_F_s_total=sum(F_s_ret(2,:));
average_E_F_s=E_F_s_total/j;
average_R_F_s=R_F_s_total/k;

sum_ext=0;
dif_ext_2=0;
for i=1:j
    dif_ext_2= (Real_F_s_ext(i)-average_E_F_s)^2;
```

```
    sum_ext=sum_ext+dif_ext_2;
    s_d_ext=((sum_ext/j))^0.5;
end

sum_ret=0;
dif_ret_2=0;
for i=1:k
    dif_ret_2= (Real_F_s_ret(i)-average_R_F_s)^2;
    sum_ret=sum_ret+dif_ret_2;
    s_d_ret=((sum_ret/k))^0.5;
end

t=1;
Real_Dyn_Ext(1)=Dyn_ext(1);
for y=1:(d(2)-1)
    if ((Dyn_ext(y+1)~=Dyn_ext(y)))
        t=t+1;
        Real_Dyn_Ext(t)= Dyn_ext(y+1);

    end
end

n=1;
Real_Dyn_Ret(1)=Dyn_ret(1);
for y=1:(e(2)-1)
    if ((Dyn_ret(y+1)~=Dyn_ret(y)))
        n=n+1;
        Real_Dyn_Ret(n)= Dyn_ret(y+1);

    end
end
```

```
g=1;
Real_Speed_Ext(1)=speed_e(1);

for y=1:(d(2)-1)
   if ((speed_e(y+1)~=speed_e(y)))
      g=g+1;
      Real_Speed_Ext(g)=speed_e(y+1);
   end
end

s=1;
Real_Speed_Ret(1)=speed_r(1);

for y=1:(e(2)-1)
   if ((speed_r(y+1)~=speed_r(y)))
      s=s+1;
      Real_Speed_Ret(s)=speed_r(y+1);
   end
end

F_s_ext=average_E_F_s;
F_s_ret=average_R_F_s;
tt=size(Real_Speed_Ext)+1;
ttl=size(Real_Speed_Ret)+1;
linear_speed_ext = zeros(1,tt(2));
linear_speed_ret= zeros(1,ttl(2));
F_f_ext=zeros(1,tt(2));
F_f_ret=zeros(1,ttl(2));
linear_speed_ext(1)=0;
linear_speed_ret(1)=0;
F_f_ext(1)=F_s_ext;
F_f_ret(1)=F_s_ret;
```

```
linear_speed_ext(2:tt(2))= Real_Speed_Ext();

linear_speed_ret(2:ttl(2))=Real_Speed_Ret();

F_f_ext(2:tt(2))=Real_Dyn_Ext;

F_f_ret(2:ttl(2))=Real_Dyn_Ret;


my_fun_ext=@(alfa,linear_speed_ext) (alfa(1)+((F_s_ext-
alfa(1))./exp((linear_speed_ext/alfa(2)).^2))+linear_speed_ext.*alfa(3));

intial_values_ext=[5,0.1,50];

lb_ext=[0,0,0];

ub_ext=[F_s_ext, 0.1, 100];

options_ext=optimoptions('lsqcurvefit','Algorithm','levenberg-marquardt');

new_parameters_ext=lsqcurvefit(my_fun_ext,intial_values_ext,linear_speed_ext,F_f_ext,lb_ext,ub
_ext,options_ext);




my_fun_ret=@(beta,linear_speed_ret) (beta(1)+(F_s_ret-
beta(1))./exp((linear_speed_ret/beta(2)).^2)+linear_speed_ret.*beta(3));

intial_values_ret=[-5, -0.1, 50];

lb_ret=[F_s_ret, -0.1, 0];

ub_ret=[0, 0, 100];

options_ret=optimoptions('lsqcurvefit','Algorithm','levenberg-marquardt');

new_parameters_ret=lsqcurvefit(my_fun_ret,intial_values_ret,linear_speed_ret,F_f_ret,lb_ret,ub_r
et,options_ret);


tf_1=isreal(new_parameters_ext(1));

tf_2=isreal(new_parameters_ext(2));

tf_3=isreal(new_parameters_ext(3));


tf_4=isreal(new_parameters_ret(1));

tf_5=isreal(new_parameters_ret(2));

tf_6=isreal(new_parameters_ret(3));
```

```
if tf_1==0 F_c_result_ext=real(new_parameters_ext(1));
else F_c_result_ext=new_parameters_ext(1);
end
if tf_2==0 stribeck_result_ext=real(new_parameters_ext(2));
else stribeck_result_ext=new_parameters_ext(2);
end
if tf_3==0  viscous_result_ext=real(new_parameters_ext(3));
else viscous_result_ext=new_parameters_ext(3);
end

if tf_4==0 F_c_result_ret=real(new_parameters_ret(1));
else F_c_result_ret=new_parameters_ret(1);
end
if tf_5==0 stribeck_result_ret=real(new_parameters_ret(2));
else stribeck_result_ret=new_parameters_ret(2);
end
if tf_6==0 viscous_result_ret=real(new_parameters_ret(3));
else viscous_result_ret=new_parameters_ret(3);
end

app.EditField_2.Value='Stribeck friction model more suitable for extension!';
app.FsNEditField.Value=num2str(F_s_ext);
app.FcNEditField.Value=num2str(F_c_result_ext);
app.BNsmEditField.Value=num2str(viscous_result_ext);
app.StrspeedmsEditField.Value=num2str(stribeck_result_ext);
app.NumberofcollectedstaticdataEditField.Value=num2str(j);
app.StandarddeviationofstaticdataEditField.Value=num2str(s_d_ext);
app.NumberofcollecteddynamicdataEditField.Value=num2str(n);
```

```
a=isequal(app.FsNEditField.Value,app.FcNEditField.Value);

 if a==1
    app.EditField_2.Value=['Coulomb+Viscous friction model more suitable for extension!'];
    app.StrspeedmsEditField.Value='not available!';
    app.FsNEditField.Value='not available!';
    app.FcNEditField.Value=num2str(F_c_result_ext);
    app.BNsmEditField.Value=num2str(viscous_result_ext);
 end
 if stribeck_result_ext==0.1
    app.EditField_2.Value='Coulomb+Viscous friction model more suitable for extension!';
    app.StrspeedmsEditField.Value='not available!';
    app.FsNEditField.Value='not available!';
    app.FcNEditField.Value=num2str(F_c_result_ext);
    app.BNsmEditField.Value=num2str(viscous_result_ext);
 end


     app.EditField_3.Value='Stribeck friction model more suitable for retraction!';
     app.FsNEditField_2.Value=num2str(F_s_ret);
     app.FcNEditField_2.Value=num2str(F_c_result_ret);
     app.BNsmEditField_2.Value=num2str(viscous_result_ret);
     app.StrspeedmsEditField_2.Value=num2str(stribeck_result_ret);
     app.NumberofcollectedstaticdataEditField_2.Value=num2str(k);
app.StandarddeviationofstaticdataEditField_2.Value=num2str(s_d_ret);
app.NumberofcollecteddynamicdataEditField_2.Value=num2str(n);

b=isequal(app.FsNEditField_2.Value,app.FcNEditField_2.Value)

 if b==1;
    app.EditField_3.Value='Coulomb+Viscous friction model more suitable for retraction!';
    app.StrspeedmsEditField.Value_2='not available!';
```

```
    app.FsNEditField.Value_2='not available!';
    app.FcNEditField.Value=num2str(F_c_result_ret);
    app.BNsmEditField.Value_2=num2str(viscous_result_ret);
 end
if stribeck_result_ret==-0.1
    app.EditField_3.Value='Coulomb+Viscous friction model more suitable for retraction!';
    app.StrspeedmsEditField.Value_2='not available!';
    app.FsNEditField.Value_2='not available!';
    app.FcNEditField.Value_2=num2str(F_c_result_ret);
    app.BNsmEditField.Value_2=num2str(viscous_result_ret);
end

figure=app.UIAxes;
plot(figure,sp_e,Real_F_s_ext,'k^');
hold(figure,'on');
plot(figure,speed_ret,Real_F_s_ret,'k^');
plot(figure,0,average_E_F_s,'ko');
plot(figure,0,average_R_F_s,'ko');
grid(figure,'on');
ylim(figure, [(min(F_f_ret)-1.2) (max(F_f_ext)+1.2)]);
xlim(figure,  [-0.85 0.85]);

d=0:0.001:max(linear_speed_ext);
t=0:-0.001:min(linear_speed_ret);
plot(figure,Real_Speed_Ext,Real_Dyn_Ext,'k*',Real_Speed_Ret, Real_Dyn_Ret,'k*');
new_F_f_ext=my_fun_ext(new_parameters_ext, d);
new_F_f_ret=my_fun_ret(new_parameters_ret, t);
plot(figure,d,new_F_f_ext,'k-','LineWidth',1.1);
hold(figure,'on');
plot(figure,t,new_F_f_ret,'k-','LineWidth',1.1);
```

end

function CONNECTButtonPushed(app, event)

```
        set_param('Automatic_friction_estimation_with_verification', 'SimulationCommand',
'connect');
        app.DISCONNECTButton.Enable = 'on';
        app.STARTButton.Enable = 'on';
        app.STOPButton.Enable  = 'off';
        app.CONNECTButton.Enable='off';
         app.CLEARALLButton.Enable='off';
        app.EditField.Value= ['Connection is successfull, Press Start button to continue!'];
        app.EditField.BackgroundColor = [1.00,1.00,1.00];
    end
function DISCONNECTButtonPushed(app, event)
        set_param('Automatic_friction_estimation_with_verification', 'SimulationCommand',
'disconnect')
         app.EditField.BackgroundColor = [1.00 1.00 1.00];
         app.EditField.Value=['System is disconnected!'];
         app.CLEARALLButton.Enable='on';
         app.DISCONNECTButton.Enable = 'off';
         app.STARTButton.Enable = 'off';
         app.STOPButton.Enable = 'off';
    end
function CLEARALLButtonPushed(app, event)
        app.CylinderStrokeLEditField.Value=0;
        app.PistondiameterPDEditField.Value=0;
        app.RoddiameterRDEditField.Value=0;
        app.ofstaticexperimentEditField.Value=0;
        app.ofdynamicexperimnetEditField.Value=0;

%This code clear friction force-speed curve for later experiments.
app.UIAxes.cla;

app.EditField_2.Value=['Comment on output results'];
app.FsNEditField.Value=['-'];
app.FcNEditField.Value=['-'];
```

```
app.BNsmEditField.Value=['-'];
app.StrspeedmsEditField.Value=['-'];
app.NumberofcollectedstaticdataEditField.Value=['-'];
app.StandarddeviationofstaticdataEditField.Value=['-'];
app.NumberofcollecteddynamicdataEditField.Value=['-'];

app.EditField_3.Value=['Comment on output results'];
app.FsNEditField_2.Value=['-'];
app.FcNEditField_2.Value=['-'];
app.BNsmEditField_2.Value=['-'];
app.StrspeedmsEditField_2.Value=['-'];
app.NumberofcollectedstaticdataEditField_2.Value=['-'];
app.StandarddeviationofstaticdataEditField_2.Value=['-'];
app.NumberofcollecteddynamicdataEditField_2.Value=['-'];

        app.EditField.Value=['Please enter the input parameters on the left!'];
        app.EditField.FontColor=[0.64,0.08,0.18];
        app.CylinderStrokeLEditField.Editable= 'on';
        app.PistondiameterPDEditField.Editable= 'off';
        app.RoddiameterRDEditField.Editable='off';
        app.ofstaticexperimentEditField.Editable='off';
        app.ofcylclesforstaticEditField.Editable='off';
        app.ofdynamicexperimnetEditField.Editable='off';

        app.Image2.Visible='off';
        app.Image4.Visible='on';
        app.Image5.Visible='off';
        app.Image6.Visible='off';

        app.CONNECTButton.Enable='off';
    end
function CylinderStrokemmEditFieldValueChanged(app, event)
```

```
value=app.CylinderStrokeLEditField.Value;
value_1=app.ofstaticexperimentEditField.Value;
value_2=app.PistondiameterPDEditField.Value;
value_3=app.RoddiameterRDEditField.Value;
value_4=app.ofdynamicexperimnetEditField.Value;

app.Image4.Visible = 'on';

   if value>0
   app.CylinderStrokeLEditField.Editable= 'off';
    app.Image4.Visible = 'off';
   app.EditField.Value=['Enter piston diameter'];
   app.EditField.FontColor = [0.64,0.08,0.18];
   app.PistondiameterPDEditField.Editable= 'on'
    app.Image5.Visible='on';
   end

   value_2=app.PistondiameterPDEditField.Value;

   if value_2>0
      app.PistondiameterPDEditField.Editable= 'off';
      app.EditField.Value=['Enter rod diameter'];
      app.RoddiameterRDEditField.Editable='on';
        app.Image5.Visible='off';
        app.Image6.Visible='on';
   end
   value_3=app.RoddiameterRDEditField.Value;

   if value_3>0
      if value_3<value_2
      app.RoddiameterRDEditField.Editable='off';
      app.Image6.Visible='off';
```

```
            app.Image2.Visible='on';
            app.EditField.Value=['Enter number of static experiments'];
            app.ofstaticexperimentEditField.Editable='on';
            else app.RoddiameterRDEditField.Value=0;
               app.EditField.Value=['Please enter correct value!'];
            end
          end

        value_1=app.ofstaticexperimentEditField.Value;

        if value_1>0
            app.ofstaticexperimentEditField.Editable='off';
            app.EditField.Value=['Enter number of dynamic experiments'];
            app.ofdynamicexperimnetEditField.Editable='on';
        end

        value_4=app.ofdynamicexperimnetEditField.Value

        if value_4>0

             app.ofdynamicexperimnetEditField.Editable='off';
              app.EditField.Value=['Parameters are entered successfully, system is ready for
connection.'];
              app.CONNECTButton.Enable='on';
              app.EditField.FontColor=[0.64,0.08,0.18]
                 end


    end
```