Towards Developing Gripper to obtain Dexterous Manipulation

Nahian Rahman

Advanced Robotics Department
Istituto Italiano di Tecnologia - Università Degli Studi di Genova

This dissertation is submitted for the degree of
Doctor of Philosophy

Istituto Italiano di Tecnologia February 2018

Thesis Supervisors

- **Dr. Luca Carbonari**
  Post Doc
  Department of Industrial Engineering
  and Mathematical Sciences Faculty of Engineering Università Politecnica delle Marche

- **Dr. Marco Gabiccini**
  Assistant Professor
  Department of Mechanical, Nuclear and Production Engineering
  (DIMNP) University of Pisa

- **Dr. Matteo Palpacelli**
  Assistant Professor
  Department of Industrial Engineering
  and Mathematical Sciences Faculty of Engineering Università Politecnica delle Marche

- **Dr. Ferdinando Cannella**
  Header
  Advanced Industrial Automation Lab
  Advanced Robotics Department
  Istituto Italiano di Tecnologia (IIT)

- **Prof. Darwin G. Caldwell**
  Director
  Advanced Robotics Department
  Istituto Italiano di Tecnologia (IIT)

I would like to dedicate my thesis to the memories of my mom . . .

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Nahian Rahman
February 2018

Acknowledgements

First of all I am grateful to the Almighty, and I would then like to express my deepest gratitude to all of my childhood/University teachers, mentors from my bottom of my heart, without whose immense support, I couldn't reach at that point truly. It has been a tremendous mission from the beginning, and I am glad that I have made this finally. It is impossible to put all the names to whom I am grateful to. I must thank to my PhD mentor, Dr. Ferdinando Cannella, specially for trusting me and allowing me to conduct my research in a way that I was able to develop myself. Working with him is a new experience, which I will never forget for the rest of my life. I must not forget to thank Dr. Luca Carbonari, for his strong guidance at the starting point when I was almost clueless, and for his critical remarks throughout my progress as well. I am grateful to Prof. Darwin G. Caldwell for providing me this great opportunity to be part of ADVR, IIT at the first place and I am thankful to him for his generous attitude, selfless attention and granting my numerous missions in the past three years. I am also thankful to all ADVR technicians and colleagues of admin office for their spontaneous support, and to facilitate all my demands to conduct my research activity.

I would like to recall all the past and present members of AIAL for their remarkable hospitality, jovial attitude towards me; more importantly for being my friend rather than colleague. I must say a special thanks to the Loccioni members of our team "LAIAL", for their thorough co-operation, contribution and the outcome of the Bayer robotic challenge 2017, a notable memory indeed at Berlin!

It is difficult to put into words how much I miss my father, two lovely sisters and other family members from so far away. It would not have been possible without their patience, sacrifice, vivid presence on Skype allowed me to stay away for the last five years. And of course I am missing my mother as always, at this significant point of my life.

At last, but not the least, I would like to say a warm *thanks* to all my relatives, friends, colleagues, well-wishers for offering generous support, endless appreciations for free of cost, since from the beginning of my consciousness...

Abstract

Artificial hands or grippers are essential elements in many robotic systems, such as, humanoid, industry, social robot, space robot, mobile robot, surgery and so on. As humans, we use our hands in different ways and can perform various maneuvers such as writing, altering posture of an object in-hand without having difficulties. Most of our daily activities are dependent on the prehensile and non-prehensile capabilities of our hand. Therefore, the human hand is the central motivation of grasping and manipulation, and has been explicitly studied from many perspectives such as, from the design of complex actuation, synergy, use of soft material, sensors, etc; however to obtain the adaptability to a plurality of objects along with the capabilities of in-hand manipulation of our hand in a grasping device is not easy, and not fully evaluated by any developed gripper.

Industrial researchers primarily use rigid materials and heavy actuators in the design for repeatability, reliability to meet dexterity, precision, time requirements where the required flexibility to manipulate object in-hand is typically absent. On the other hand, anthropomorphic hands are generally developed by soft materials. However they are not deployed for manipulation mainly due to the presence of numerous sensors and consequent control complexity of under-actuated mechanisms that significantly reduce speed and time requirements of industrial demand. Hence, developing artificial hands or grippers with prehensile capabilities and dexterity similar to human like hands is challenging, and it urges combined contributions from multiple disciplines such as, kinematics, dynamics, control, machine learning and so on. Therefore, capabilities of artificial hands in general have been constrained to some specific tasks according to their target applications, such as grasping (*in biomimetic hands*) or speed/precision in a pick and place (*in industrial grippers*).

Robotic grippers developed during last decades are mostly aimed to solve grasping complexities of several objects as their primary objective. However, due to the increasing demands of industries, many issues are rising and remain unsolved such as *in-hand manipulation* and *placing object with appropriate posture*. Operations like *twisting, altering orientation of object* within-hand, require significant dexterity of the gripper that must be achieved from a compact mechanical design at the first place. Along with manipulation, speed is also required in many robotic applications. Therefore, for the available speed and

design simplicity, nonprehensile or dynamic manipulation is widely exploited. The nonprehensile approach however, does not focus on stable grasping in general. Also, nonprehensile or dynamic manipulation often exceeds robot's kinematic workspace, which additionally urges installation of high speed feedback and robust control. Hence, these approaches are inapplicable especially when, the requirements are grasp oriented such as, *precise posture change of a payload* in-hand, *placing payload afterward according to a strict final configuration*. Also, addressing critical payload such as egg, contacts (between gripper and egg) cannot be broken completely during manipulation. Moreover, theoretical analysis, such as contact kinematics, grasp stability cannot predict the nonholonomic behaviors, and therefore, uncertainties are always present to restrict a maneuver, even though the gripper is capable of doing the task.

From a technical point of view, *in-hand manipulation* or *within-hand dexterity* of a gripper significantly isolates grasping and manipulation skills from the dependencies on contact type, a priory knowledge of object model, configurations such as initial or final postures and also additional environmental constraints like disturbance, that may causes breaking of contacts between object and finger. Hence, the property (*in-hand manipulation*) is important for a gripper in order to obtain human hand skill.

In this research, these problems (to obtain speed, flexibility to a plurality of grasps, within-hand dexterity in a single gripper) have been tackled in a novel way. A gripper platform named *Dexclar* (DEXterous reConfigurable moduLAR) has been developed in order to study in-hand manipulation, and a generic spherical payload has been considered at the first place. *Dexclar* is mechanism-centric and it exploits modularity and reconfigurability to the aim of achieving within-hand dexterity rather than utilizing soft materials. And hence, precision, speed are also achievable from the platform. The platform can perform several grasps (*pinching, form closure, force closure*) and address a very important issue of *releasing payload with final posture/ configuration* after manipulation. By exploiting 16 degrees of freedom (DoF), *Dexclar* is capable to provide 6 DoF motions to a generic spherical or ellipsoidal payload. And since a mechanism is reliable, repeatable once it has been properly synthesized, precision and speed are also obtainable from them. Hence Dexclar is an ideal starting point to study within-hand dexterity from kinematic point of view.

As the final aim is to develop specific grippers (having the above capabilities) by exploiting *Dexclar*, a highly dexterous but simply constructed reconfigurable platform named VARO-fi (VARiable Orientable fingers with translation) is proposed, which can be used as an industrial end-effector, as well as an alternative of bio-inspired gripper in many robotic applications. The robust four fingered VARO-fi addresses grasp, in-hand manipulation and release (payload with desired configuration) of plurality of payloads, as demonstrated in this thesis.

Last but not the least, several tools and end-effectors have been constructed to study prehensile and non-prehensile manipulation, thanks to Bayer Robotic challenge 2017, where the feasibility and their potentiality to use them in an industrial environment have been validated.

The above mentioned research will enhance a new dimension for designing grippers with the properties of dexterity and flexibility at the same time, without explicit theoretical analysis, algorithms, as those are difficult to implement and sometime not feasible for real systems.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

Robotics is an advanced engineering discipline, an ever emerging field and is being inspired from the mother nature continuously. As numerous vivid activities of living beings keep the earth alive everyday, at the same time, they also capture the interest of researchers as well. And by exploiting these inspirations or biomimetic activities into artificial machines, many problems are being solved. Among these inspirations of living being, the human hand and its dexterity are the major attractions to robotic community, since it is the primary way to interact with environment, such as touch, sense, grasp, manipulation; and a large portion of our daily activities is governed by hand's prehensile and non-prehensile capabilities. And so, artificial gripper is required in many robotic applications such as, humanoid, space-robots, industrial manipulators or social interactive robots. Another fascinating fact of artificial machines is they often surpass human-limit in terms of speed, precision, reliability, repeatability. This is mainly because of our large and slow actuators, which constrain the range of bandwidth, that increases the response time to an event.

Due to the technological revolution in the last few decades, robotics now-a-days, is a fusion of multiple engineering disciplines, such as mechanics, dynamics, control, vision, artificial intelligence, deep learning and so on. However even said so, robots are not fully capable to conduct maneuvers, to the aim they are built for. One example could be artificial, anthropomorphic hand, that intends to mimic human hand's capability [47]. Numerous artificial grippers and hands have been developed by robotic community in the last decades, however, merely those hands can perform in-hand manipulation or address within hand dexterity similar to human hand (A literature review on grippers is presented in Chapter 2).

Figure 1.1 depicts several dexterous capabilities of our hand. We, human are able to perform plurality of tasks by exploiting multiple degrees of freedom of our hand. Most of these maneuvers are possible due to the complex combinations of skin sensor, powerful vision system and, last but not least, our brain. However in practice, these tasks are very

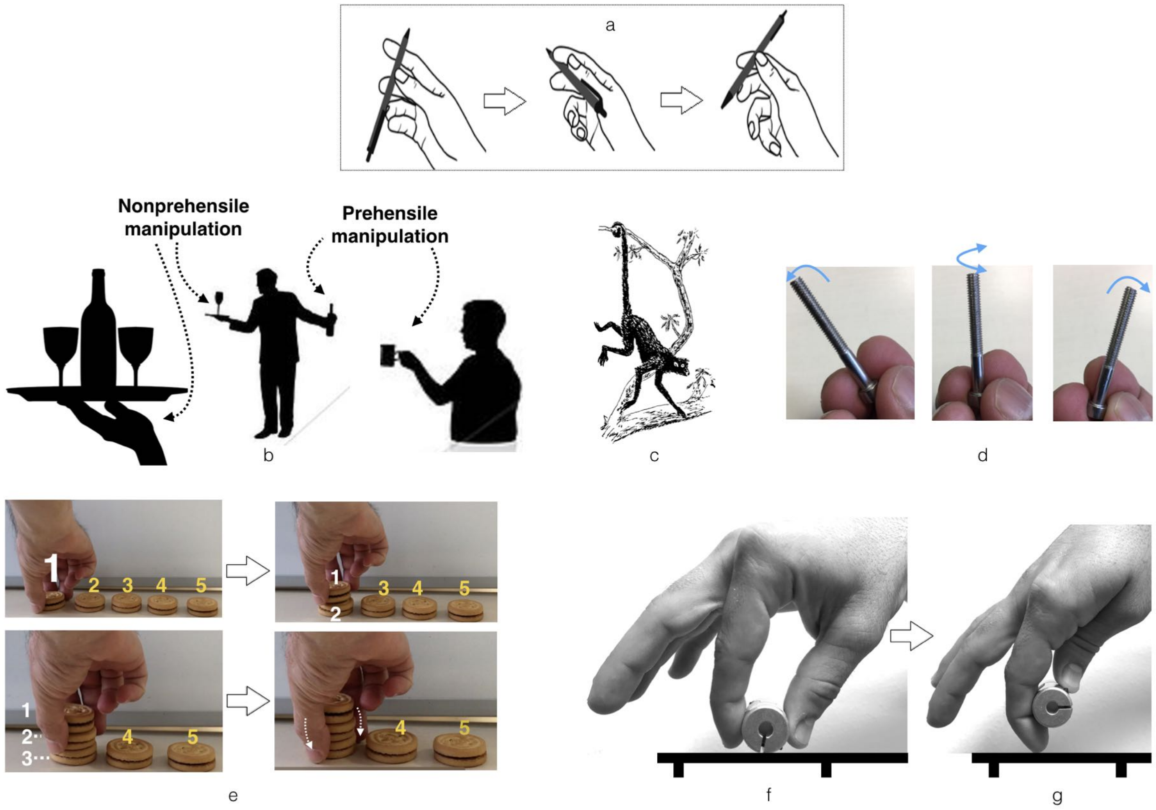


Fig. 1.1 a) typical human hand skill, in-hand twist of a pen, b) examples of non-prehensile and prehensile manipulation, c) prehensile tail maneuver of monkey, d) prehensile manipulation of a screw by fingertip, e) picking biscuit-1 using thumb and index and placing biscuit-1 over biscuit-2. repeating the same process for biscuit-3, finally grab them (biscuit-1,2,3) in a form closure grasp, f) picking a mechanical joint by index and thumb, g) twisting the joint by moving the thumb: at that state it is difficult to place the joint with the final configuration

difficult to obtain from an artificial gripper. Moreover, skills that we do by our synchronous fusion of two hands, such as shuffling cards, solving rubik's cube as illustrated in Fig. 1.2 are remarkable compared to the capabilities of artificial hand or gripper. It is obvious that several designs could be found in literature that are able to address particular problems such as solving rubik's cube even faster than human hand however, such task specific device cannot be used for general purpose grippers.

1.1 In-hand Manipulation and Why is it Important?

The ability to change configuration of a grasped object without external support, constraint, force is called in-hand manipulation or within hand dexterity. This maneuver or skill is useful, because it helps to be independent from initial configurations of the gripper and grasped

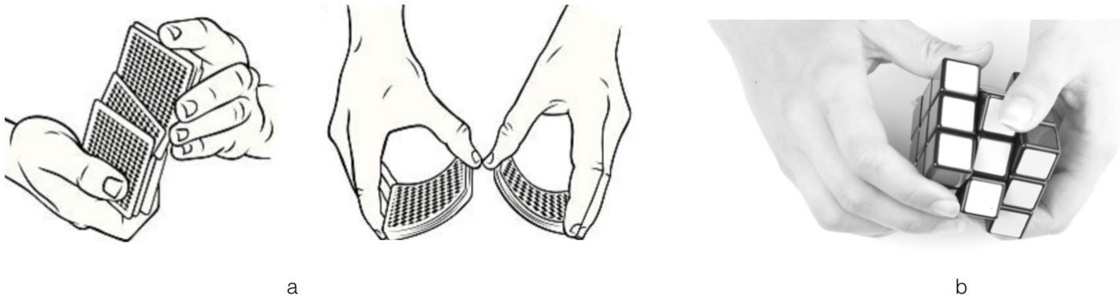


Fig. 1.2 Dexterity of two hands: a) shuffling of playing cards, b) solving rubik's cube

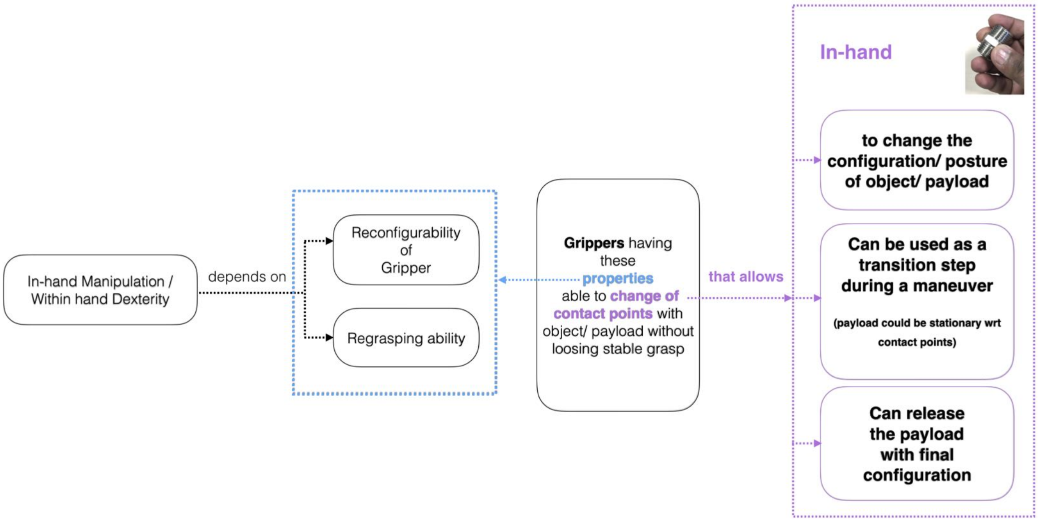


Fig. 1.3 Required properties of in-hand manipulation and the advantages

object. That also sometime means, the gripper can perform "regrasping" [83] and capable to create suitable contact set with the object (with desired posture) regardless of any additional constraint/ support. Hence, the gripper is also capable to recover from a poor initial grasp (from grasp stability point of view [29]) or a from a contact set which is not suitable for the desired task. Moreover a bunch of prehensile, non-prehensile manipulation (fig. 1.1) requires a series of in-hand posture/configuration change of the grasped object.

From a design point of view, grippers which are reconfigurable and capable to regrasp payloads without loosing, they can perform in-hand manipulation tasks. Figure 1.3 illustrates several outcomes that can be obtained from a gripper having the properties. By exploiting "the change of contact points" with payload, it is also possible to release the payload with the final or desired configuration. The difficulties of release or placing payload with the final/desired configuration is also a problem for human hand, one example has been shown in Fig. 1.1g.

Table 1.1 A comparison between several grippers in terms of DoF, number of fingers & ability

Brand	**Model**	**Finger + thumb**	**DoF**	**Manipulation**	**Release focused?**
Schunk GmbH & Co. KG	SDH	3 + 0	7	capable	no
EMI Corp.	PN-040-3	3 + 0	3	not good	no
GWU	GWU Hand-I	3 + 0	3	not good	no
Barrett	Barrett [84]	3 + 0	4	capable	no
NASA	Robonaut Hand	4 + 1	14	capable	no
Shadow Robot & Co	Shadow hand	4 + 1	20	capable	no
Genoa University	DIST hand [14]	3 + 1	16	capable	no
BIC [19]	TP gripper	2 + 0	6	only in one plane	yes
Velvet fingers [81]	Velvet fingers	2 + 0	3	only in one plane	yes
Salisbury hand [68]	Salisbury hand	3 + 0	9	capable	no

1.2 A Very Brief Description: State of the art

During the last few decades, robotic grippers have been developed by research community to solve grasping complexities of several objects as their primary objective [47]. The reason is obvious: due to the differences in geometry, size and shape (of the target object), the prior requirement is to grasp the object at the first place. Hence, grasp synthesis algorithm [73, 67], grasp stability [29], contact kinematics [68] have been explicitly studied by many researchers. However, due to the increasing demands of industries, many issues are rising and remain unsolved such as in-hand manipulation and placing object with appropriate posture. Operations like twisting and altering orientation of object in a hand requires significant dexterity of the gripper that must be achieved from a compact mechanical design at the first place. Table 1.1 shows a very brief state of art of some industrial and dexterous grippers/hands. The payload release issue is hardly been addressed in the early stage such as in Salisbury hand [68] or Barret hand [84]. In the recent years, some efforts, velvet fingers [81], TP gripper [19] addressed some issues of payload release with final configuration. However the limitations of such efforts are: the solution is not adequate for a plurality of payloads, the

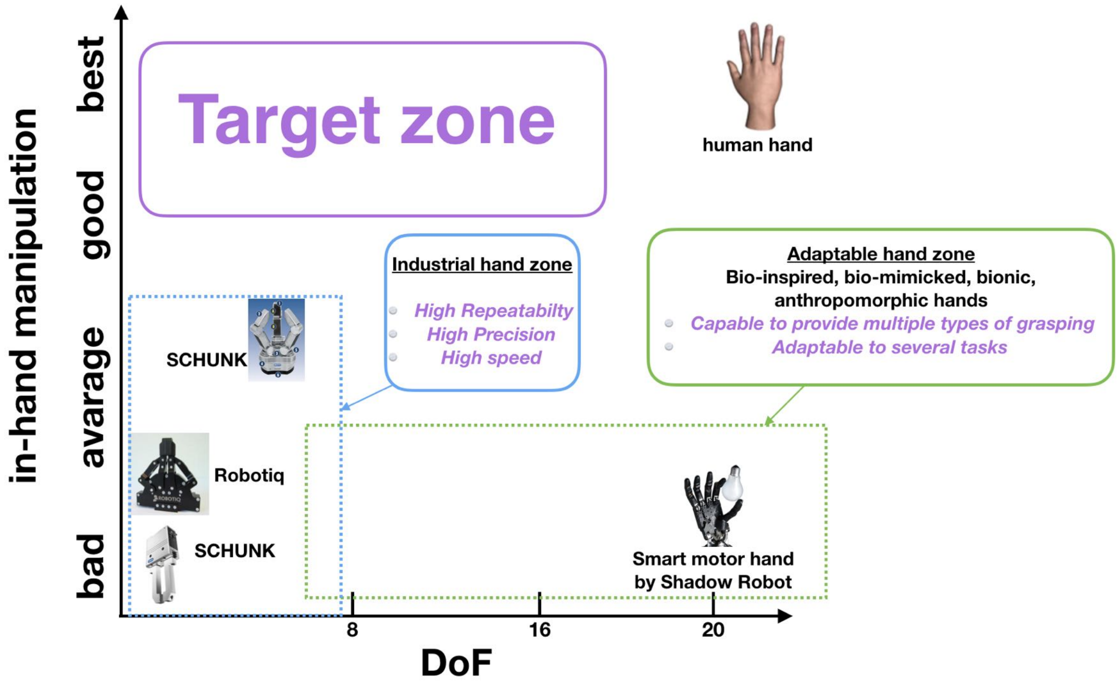


Fig. 1.4 Target zone of this research

dexterity of the gripper is also constrained into a particular plane. Hence a generic gripper platform is required to study in-hand manipulation.

1.3 Contribution and Outline

This work is first and foremost driven by the idea of developing a general platform to obtain posture change of payload in-hand. In order to build a gripper that is able to perform grasp, in-hand manipulation and release payload with the final / desired configuration, it is required to verify the kinematic feasibility of those tasks in a platform first before developing a prototype. The idea is to study the required motions for a given / desired manipulation of payloads in the platform, hence the platform could act as a starting point for developing grippers. This thesis is organized as follows:

- **Chapter 2: State of the Art**

  Chapter 2 presents the state of the art of grippers. The description includes developed anthropomorphic, bio-mimicked, bio-inspired and industrial hands by research

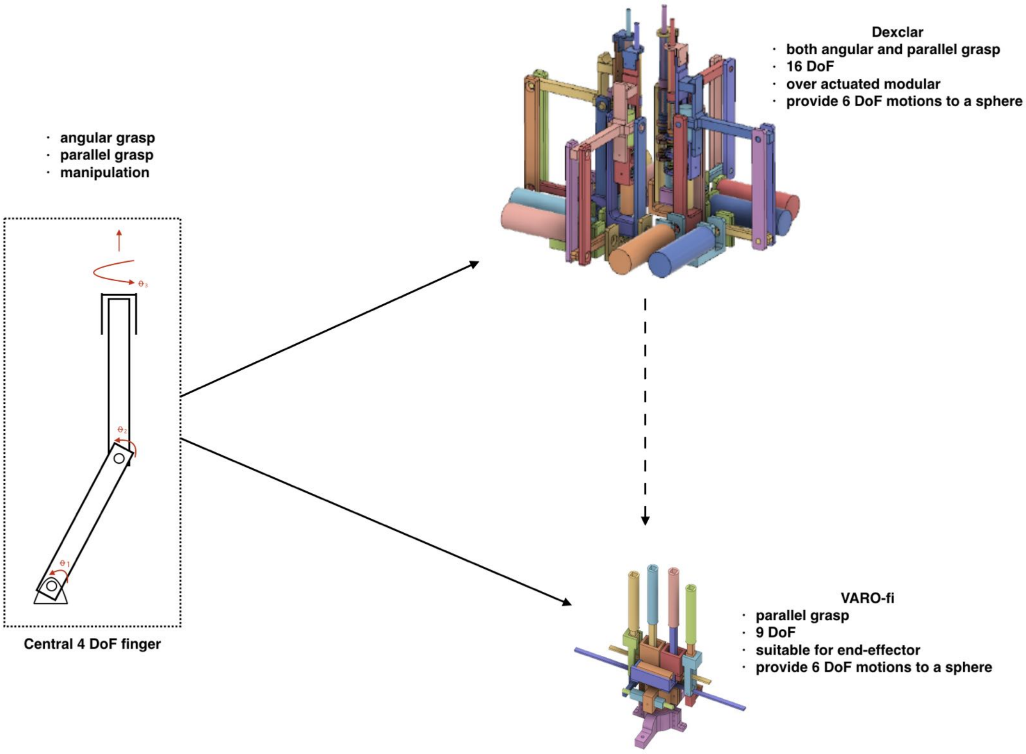


Fig. 1.5 The central 4 DoF finger and the derivation of Dexclar and VARO-fi

community in the last few decades in order to demonstrate their capabilities and limitations.

- **Chapter 3: Towards Developing a Generic Platform for In-hand Manipulation**

  Chapter 3 focuses on the development of a generic platform for in-hand manipulation. A 4 DoF finger is considered as a central point at the initial stage and the platform named *Dexclar* was built using the concept of the central finger. *Dexclar* exploits mechanisms, modularity and reconfigurability to achieve manipulation. A detailed analysis of required motions (for manipulation) have been shown and a several solutions are taken under consideration in the development phase. The chapter illustrates an explicit study for required number of *fingers*, *DoF* and their arrangement. The problem of *release after final configuration* is considered in the design process and has been solved. The concepts are first investigated in a multibody simulated environment and some prototypes have been developed further to conduct experiments. And through these investigations, the final platform *Dexclar* has been developed and several experiments have conducted to validate its feasibility and purpose. A 4 DoF finger has been developed by implementing two well known mechanisms, and *Dexclar* utilizes

4 identical fingers of such type. The synthesis process of the proposed finger and the prehensile capabilities of the gripper platform have been explained thoroughly. A control method is also proposed and validated through experiments.

Besides in short, the 16 DoF *Dexclar* considers mechanism centric approach, and it uses modularity and reconfigurability to obtain both within-hand dexterity and flexibilities to plurality of payloads at the same time. The number of fingers of this platform is chosen as 4 with the aim of achieving reconfigurability and re-grasping capability. By exploiting these properties (*reconfigurability* and *re-grasp*), *Dexclar* is capable to *release payload with final configuration* and *re-manipulate* payload in-hand.

- **Chapter 4: End-effector Design**

  By exploiting the formulation/methodology proposed and validated in Chapter 3, Chapter 4 develops dexterous end-effector for in-hand manipulation. In general, industrial and bio-inspired contributions in literature are considered as two separate domains. VARO-fi has been proposed which is a feasible solution for both trends of research lines and it surpass many limitations of the capabilities of existing grippers of both fields. The self reconfigurable VARO-fi is capable of satisfying the three major challenges; grasp, manipulation and release object with desired posture of plurality of payload, also minimizes grasp synthesis problem at some margin. Besides, the key advantages of such device are dexterity at any given plane, robustness such as, scalability and it's construction simplicity in particular. One of the major novelty of this VARO-fi is, it can work as two hands at a time since it exploited the concept of two pair of *index-thumb* (of our hand which is explained in this chapter). A prototype of VARO-fi has been built and the validity of the claims have proved through experiments.

- **Chapter 5: Bayer Robotic Challenge**

  Chapter 5 describes a robotic competition organized by Bayer pharmaceutics company. Although the focus of the robotic challenge is slightly offset from in-hand manipulation, however it facilitates to validate both the prehensile and non-prehensile capabilities of some developed tools, end-effectors in an industrial environment.

- **Chapter 6: Conclusion**

  Chapter 6 concludes the thesis by explaining an overview of the whole work and describes the future tasks required to be done.

Chapter 2 describes the state of the art of grippers. It aims to substantiate the fact that merely a gripper has addressed the three major challenges: Grasp, In-hand manipulation and Release payload with desired posture.

Chapter 2

State of the Art

2.1 Grippers in Literature

In the last few decades, numerous studies have been conducted by the robotic community on grippers or artificial hands primarily to the aim of solving grasping problems [5], stable grasping [29], kinematics of the contact between gripper and target object [68], contact stability [51] and force-form closure approaches of grasping [75]; all of these ingredients [5, 29, 68, 51, 75, 70, 50, 33, 45] are often used to develop grasping devices such as industrial grippers [9, 16] and anthropomorphic hands [47]. Although, technically all grippers are somehow either bio-inspired or bio-mimicked, however the following description is divided into two parts, with respect to their target use and application

- Bio-inspired and Bio-mimicked grippers (primarily for grasping)
- Grippers that focused on Manipulation

2.1.1 Bio-inspired and Bio-mimicked grippers

Anthropomorphic robotic hands are usually developed for tasks different from dexterous manipulation: the DLR [13] and the dexterous Barrett hand [84] are developed for secure grasp of different sized objects; some other bio-inspired hands those presented such as i-Limb in [20], proposals those mentioned in [42], biomimetic hands by parallel mechanisms in [35] softhand [72] and bio-inspired grippers [47, 32] are extremely flexible but not suitable for dexterous applications. In general, the presence of under actuated mechanisms, the huge number of serial-parallel connections to achieve biomimetic motions, the large number of

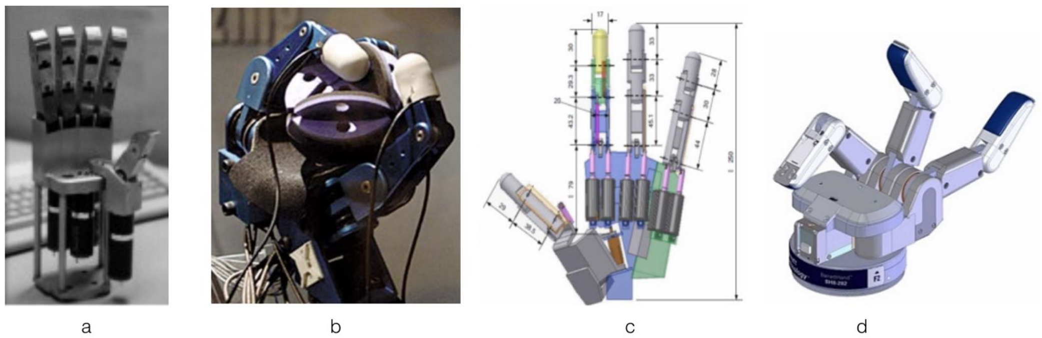


Fig. 2.1 a) Belgrade/USC Hand [42] b) Salisbury hand [68], c) Biomimetic robot hand [35], d) BarretHand [84]

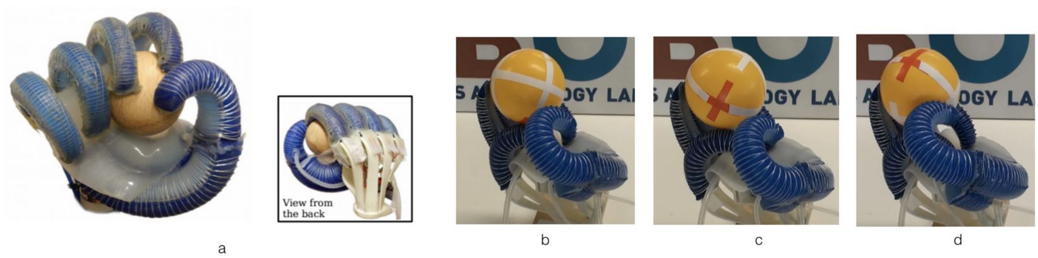


Fig. 2.2 softhand developed in [23, 22]

sensors, the consequent complexity of control, the use of soft or elastic materials make these devices not feasible for fast and reliable manipulation, as required by industry. Bio-inspired grippers are in most cases used to achieve very specific goals and to this aim the limits given by rigid links must be overpassed in order to conceive such biological inspiration. An interesting example for manipulation using shape memory alloys based actuation is proposed by Price et al. in [60], which is a smart solution to the heavy actuator driven grippers. Unfortunately, manipulation dexterity depends on series of transition steps of the fingers and since SMA based design requires thermal activities, it is difficult to obtain consequent control in order to meet the manipulation requirement.

Human hands, which represent the ideal behavior ambition while designing grippers, are superior [56] than both trends of developed artificial hands (anthropomorphic and industrial) [47, 9] under several aspects, such as adaptability to plurality of objects and versatile in-hand manipulation (an example is shown in fig. 1.1). However in terms of control and manipulation mimicking human hand is difficult and often not fully understandable [6]. Although anthropomorphic robotic hands [47] are usually explored for tasks different from manipulation, significant research have been carried out on human like hands [2, 23] under the dexterity point of view. As mentioned, despite the adaption capabilities of these hands

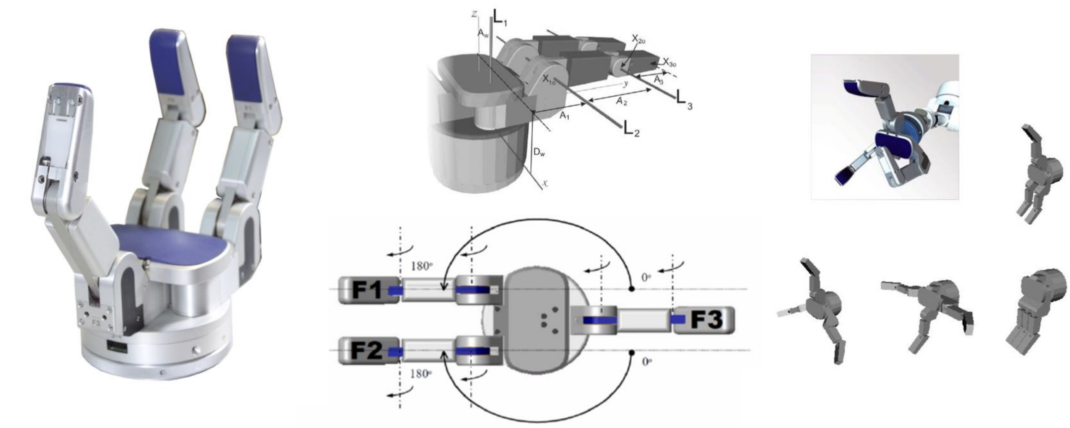


Fig. 2.3 Detail of BarretHand and its DoF, dexterity, workspace [84]

[47, 2, 28], precision and speed are hardly achievable by artificial human inspired hands, due to the use of flexible materials and under-actuated mechanisms.

Figure 2.1 shows some anthropomorphic hands those are flexible to several payloads for grasping. A softhand with pneumatic control has been shown in Fig. 2.2; although the gripper [23, 22] is compliant, flexible and capable to do manipulation, however it requires a complicated control. Moreover as mentioned previously, *precision, speed, release with final posture* are difficult to obtain from such design.

Barrethand (fig. 2.3) in literature obtained a significant attention for its dexterity. The concept of moving the finger location relative to the base frame offers robustness on grasping an arbitrary payload, also manipulation to some extent. Since it is under-actuated, *precision* is difficult to achieve and the *release* issue is not addressed.

2.1.2 Grippers that focus Manipulation

Manipulation has been studied in particular in many literature by implementing non-prehensile approaches such as [10, 46], dual arm [76] or using additional surface constraints [78] which typically requires space and consequent dependency on environment or unknown friction force. However, manipulation is also addressed by using palm [2], using additional surface constraints [78]. Author of [43, 44] exploited other common strategies such as pushing against, following a specified path or using the advantage of gravity. Re-grasp technique is also adopted to some circumstances [83]. However, most of these research [2, 78] are difficult to replicate in industrial gripper due to the mentioned constraints like unknown workspace and additional dependences. Also gripper's workspace and dimensional synthesis have been addressed in [12, 8]; and some researchers particularly addressed the problem of

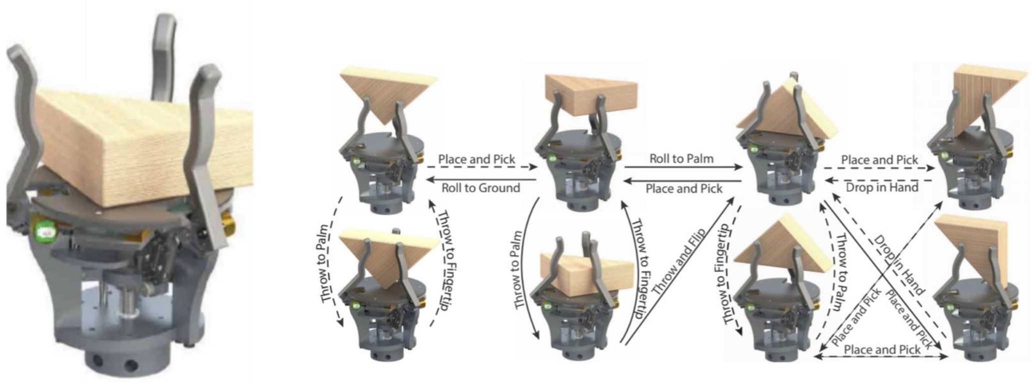


Fig. 2.4 A new approach called extrinsic dexterity in [18]

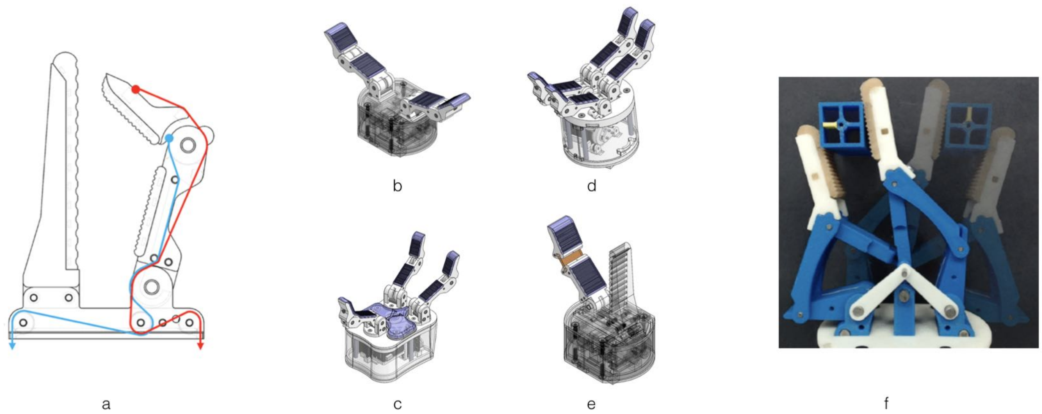


Fig. 2.5 Some grippers developed by Yale GRAB lab: a) a typical under-actuated principle [39] b) Model T42 [58] c) Model O [57] d) Model T [38] e) Model M2 [39] f) an in-hand posture change approach by two fingers ([7])

industrial manipulation [19, 81, 64, 65] from design perspective. Almost all aspects related to these issues have been tackled in the recent past, from structure of robotic arms which are demanded to carry on grippers [77, 41], to the shape and the kinematics of the wrists aimed at their orientation [25, 15]. Also, the possibility of using reconfigurable devices has been investigated [59, 17, 87], in order to accomplish positioning and orientation tasks with the same mechanical structure with a consequent economy in terms of cost, weight and even control burden. However, the dynamic or non prehensile approaches mentioned above are difficult to replicate in a gripper, in case the target payload is not allowed to move away from the workspace (of a gripper), or fixed payload posture (fig. 1.1g) is desired after the manipulation. In-hand manipulation of several target objects require significant flexibility or compliance in the gripper design which may reduce performances (*speed, precision*) of the industrial grippers. Since the prime target of industrial grippers is production, the

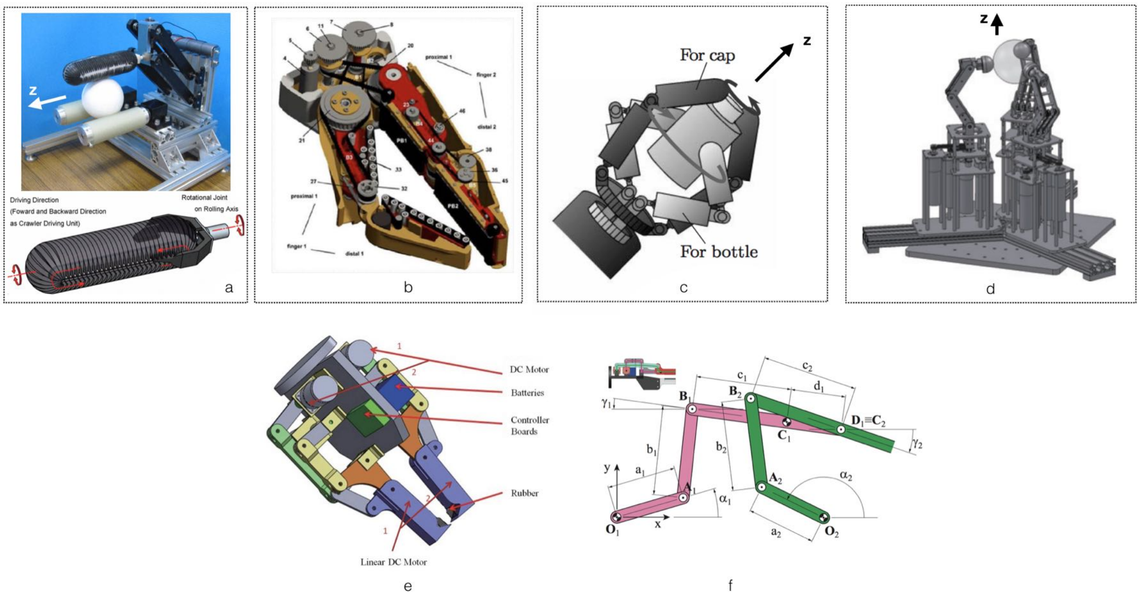


Fig. 2.6 a) gripper proposed in [79] b) gripper proposed in [82] c) gripper proposed in [27] d) gripper proposed in [80] e) TP gripper proposed in [19] f) kinematic chain of one finger [19]

researchers on this field are more interested on rigid body structures to obtain *repeatability, precision, dexterity, control* and hence, it is difficult to meet adaptive capabilities such as in-hand manipulation from those designs. A new interesting non-prehensile and prehensile manipulation approach have been proposed in [18] which is illustrated in Fig. 2.4, however this approach is difficult to convey for brittle payloads.

In recent times, Yale GRAB lab in particular has contributed a significant margin in developing under-actuated grippers, and some of them have been shown in Fig. 2.5. A large object re-orientation method (within-hand) has been proposed in [7]. However under-actuated hands in general are explored for other than *precision manipulation* by their inherent construction as stated several times previously. Another variation of those studies (fig. 2.5) with respect to the goal of this work is, *the release of payload once the final configuration has been achieved by the gripper*. The two fingered gripper that has been studied in [7], is capable to offer a large re-orientation of payload, at a *fixed geometric configuration of the gripper*. However, if the *release* issue has to be addressed, the kinematic chain of such type [7] would only able to *place* the payload at a particular space due to its geometry.

Some researchers addressed manipulation from mechanical design perspective in [19, 81, 64]. However, in those efforts, the manipulation dexterity is bounded either into single plane [19, 81] or two planes [64]; which infers that, a payload cannot be twisted about a arbitrary axis by using them. Grippers proposed in [27, 80, 79] are intelligent solutions to achieve in-hand manipulation. However, their [27, 80, 79] dexterity is also limited and the *release* issue is not addressed.

2.2 Future or Target Grippers in this study

The research mentioned in the previous subsections, are focused either on solely grasping or both the grasping and manipulation at some degree. In a broad point of view, most of those grippers have been built for specific tasks, and hence they have kinematic limitations for doing *in-hand manipulation* and *release payload with final posture*. In this regard, it is required to develop novel concepts of gripper, by exploiting *degrees of freedom*, *number of fingers*, *contact type*, which means a deep study at a kinematic level. Of course control, vision, methodology of grasping and planning are also required. However a study in kinematic domain will enhance researchers to build more dexterous gripper at the first place, or at least it will include obtainable dexterity from a design in the state of the art. Hence more optimal grippers can be synthesized using the kinematic ingredients along with the synergy between other fields.

The next chapter shows the details of developing such concepts.

Chapter 3 illustrates the development process of a generic gripper platform (4 fingered, 16 DoF) for studying in-hand manipulation in particular, as well as grasp and release problem. A central 4 DoF finger has been conceptualized at the beginning and finally a platform called Dexclar has been built by through results of many conducted experiments. This chapter also lights some issues such as degrees of freedom, number of finger and their arrangements in a gripper.

Chapter 3

Towards Developing a Generic Platform for In-hand Manipulation

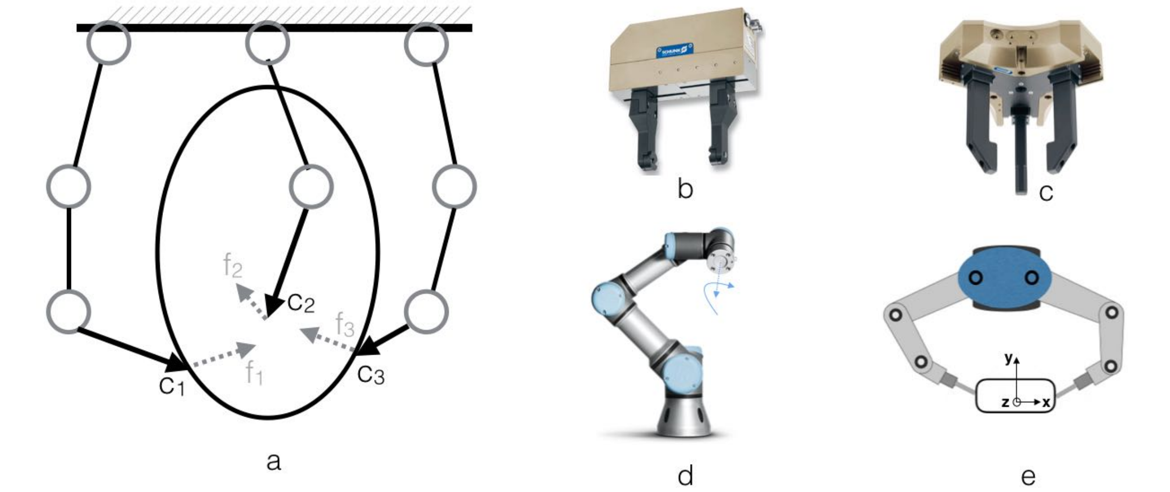


Fig. 3.1 a) The classical approach of developing multi-fingered grippers in academy, b) a two fingered SCHUNK WSG, [71] c) Three fingered SCHUNK PZH-plus [71] d) universal robot UR3 e) a typical dual arm arrangement for manipulation

This chapter is focused on the development of a generic platform to the aim of satisfying successful grasp, in-hand manipulation and release payload with final posture for plurality of payloads. As it was mentioned in the previous chapter, a general platform could be a starting point to develop an application specific gripper and study required motions for manipulation and grasping of an arbitrary payload. In literature, grasping is a primary concern and have been studied explicitly as described in the previous Chapter 2. Figure 3.1a illustrates the

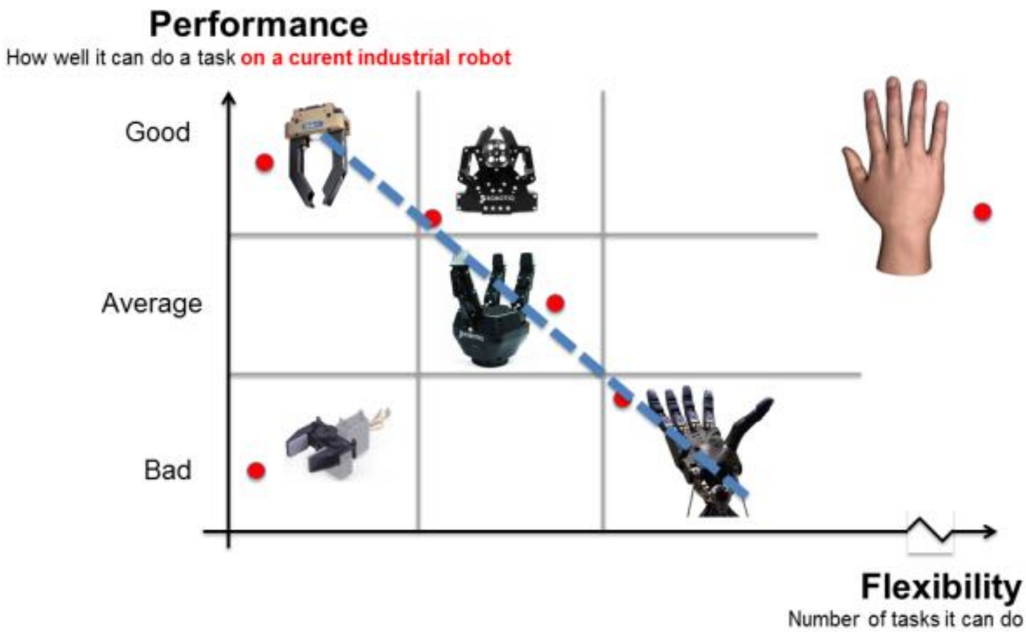


Fig. 3.2 a comparison between artificial hands and human hand. (figure source: https://blog.robotiq.com/bid/63445/What-is-the-best-robot-hand-on-the-market)

classical approach of developing multi-fingered grippers. To exploit the approach shown in Fig. 3.1a requires an extensive study of control, motion plan, contact type, friction & contact force, DoF, number and arrangement of fingers, grasp stability, grasp planning and so on. It is no wonder, the primary aim of anthropomorphic or bio-inspired grippers is grasping because of these mentioned complexities; dexterous manipulation such as in-hand manipulation is also not obtainable very easily for these very same reasons. Figure 3.2 depicts a comparison between artificial hands and human hand.

Industrial grippers exploit simple structures rather than the classical approach (fig. 3.1a). And industrial grippers are generally built by specific types of method such as, parallel type (fig. 3.1b), centric type (fig. 3.1c) and angular type. And they (fig. 3.1b,c) exploit the flexibilities of the manipulators (fig. 3.1d) to whom they are normally attached with. Dual arm is also an interesting approach that has been adopted in industry for many applications.

3.1 Methodology of developing a generic platform

Let us begin with few questions to organize the problems in an engineering relevance:

- How many fingers are necessary to perform in-hand manipulation of a generic convex payload? what should be the finger arrangement?

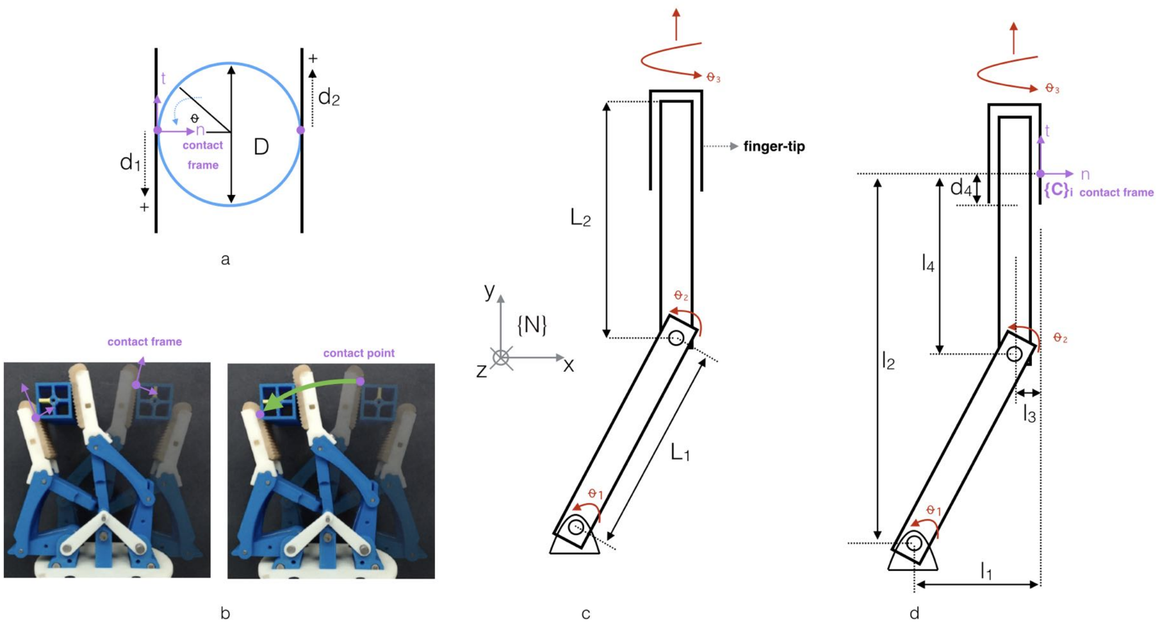


Fig. 3.3 a) A sphere is being grasped by two contact points, b) twisting object in-hand requires evolutions or changes of contact points according to the geometry of the object (gripper : [7]), c) the central 4 DoF finger and d) screw parameters of the finger

- What kind of contact between payload and finger is desirable. What will be the finger shape?
- How many Degrees of Freedom are required in the gripper platform (or finger) to the aim of such manipulation?
- What should be the driving method and material (rigid / flexible) of the platform. Will it be under-actuated or not?
- What should be the methods of grasping, manipulation and release?

Assumptions

For the sake of simplicity, spherical payload is chosen as a generic convex payload at initial point. Later, ellipsoidal and rectangular payloads are considered. The designs conceived in this dissertation are focused to be implemented by rigid material. Flexible materials and under-actuated approach have been avoided since precision, speed, repeatability, reliability can be obtained from a well design gripper constructed by rigid bodies.

Multiple tasks such as grasping, posture change and placing according to the final posture of a payload are difficult. Therefore the raised questions above are not entirely independent to each other. In this study, a conceptual finger module (fig. 3.3c,d) is considered at the first place and the generic platform (Dexclar) and gripper (VARO-fi) have been developed based on this central finger module.

Development of the 4 DoF central finger

A spherical payload experiences pure rotation about its center when the two opposite contact points (of the surface constraints of two side) translate with equal and opposite motion (fig. 3.3a). Except for this particular combination (equal and opposite motion), the sphere must experience translation or pure translation in any other combinations. On the other hand, in order to rotate an arbitrary geometric payload, the required motions of two contact points are not that straight forward. One such example is shown in Fig. 3.3b. Where both the rotation and rotation degree of the rectangular shaped payload are dependent on the chosen kinematic loop and their link dimensions. Moreover such payload specific design (fig. 3.3b) may not be suitable for applying rotation to an arbitrary payload.

Hence the finger is illustrated in Fig. 3.3c and 3.3d details its screw parameters. It has 4 DoF and 2 of the rotations are planar (θ_1 and θ_2 in fig. 3.3c). The finger-tip (fig. 3.3c) can rotate and translate with respect to L_2 since they have cylindrical joint between them. In this finger design, it is supposed that the finger-tip will create higher pair of contact with the payload.

The rotations of L_1 and L_2 will allow achieving both the parallel and angular alignments (fig. 3.5) with arbitrary payloads. Therefore the primary aim of these two rotations is to achieve grasping however, their usefulness for manipulation will be explained in later sections. Since the finger-tip is supposed to create contact with the payload, its rotation and translation are expected to apply twist to the payloads. Hence the purpose of this 2 DoF of finger-tip is manipulation and in-hand manipulation.

Twist of the finger with respect to contact frame

Since one of the purpose of this finger is manipulation hence, a kinematic measure is required to achieve that. One efficient way can be to measure the finger twist with respect to contact frame because, the finger is required to move with respect to the contact frame in order to conduct manipulation or break and re-contact with the payload. Therefore the components of finger twist can be exploited for choosing the appropriate dimensions in order to meet user specific choice. The finger (3.3c) is consist of 4 DoF and 4 joints. Let us consider a simple configuration shown in Fig. 3.3d, the frame $\{C\}$ is defined at the contact with axes $(\hat{n}, \hat{t}, \hat{o})$. Let $c \in \Re^3$ be the position of contact defined in a global frame $\{N\}$. While $l_1, .., l_4$ are screw parameters [49, 54, 30] that can describe the distance between joints and contact frame. Also note that, knowing $\theta_1, \theta_2, \theta_3$ and d_4, the screw parameters l_1 to l_4 can be estimated for a given contact. According to the illustration of hand Jacobian described in [75], the partial hand Jacobian $\widetilde{J}_i \in \Re^{6\times n}$ where n is the number of joints.

$$\widetilde{J}_i = \bar{R}_i^T Z_i \tag{3.1}$$

where $\bar{R}_i$ = Blockdiag(R_i, R_i)= $\begin{pmatrix} R_i & 0 \\ 0 & R_i \end{pmatrix} \in \Re^{6\times 6}$ and $R_i \in \Re^{3\times 3}$ represents the orientation of i-th contact frame $\{C\}_i$ w.r.t. $\{N\}$. While $Z_i \in \Re^{6\times n}$ defined as

$$Z_i = \begin{pmatrix} d_{i,1} & \dots & d_{i,n} \\ l_{i,1} & \dots & l_{i,n} \end{pmatrix} \tag{3.2}$$

where both $d_{i,j}$ and $l_{i,j} \in \Re^{3\times 1}$ such that they are $[0,0,0]^T$ while contact i does not affect joint j. Otherwise,

$$d_{i,j} = \begin{cases} \hat{z}_j & \text{if joint j is prismatic,} \\ S(c_i - \zeta_j)^T \hat{z}_j & \text{if joint j is revolute.} \end{cases}$$

$$l_{i,j} = \begin{cases} 0_{3\times 1} & \text{if joint j is prismatic,} \\ \hat{z}_j & \text{if joint j is revolute.} \end{cases}$$

where ζ_j is the origin of coordinate frame associated with j-th joint and $\hat{z}_j$ is the unit vector of z-axis of the same frame. While $S(r)$ is the skew symmetric matrix defined as

$$S(r) = \begin{pmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{pmatrix}$$

Using (3.2), estimated Z_i for the finger shown in Fig. 3.3d

$$Z_i = \begin{pmatrix} l_2 & l_4 & 0 & 0 \\ -l_1 & -l_3 & 0 & 1 \\ 0 & 0 & l_3 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \tag{3.3}$$

Now the twist of hand with respect to i-th contact frame is defined as,

$$t_{i,finger} = \widetilde{J}_i \dot{q} \tag{3.4}$$

where $t_{i,finger} \in \Re^6$ is the twist of the hand expressed in $\{C\}_i$ and $q = [q_1....q_n]^T$(here $\theta_1, \theta_2, \theta_3, d_4$) is vector of joint positions. Now the twist of the finger using (3.4):

$$t_{i,finger} = \begin{pmatrix} v_x \\ v_y \\ l_3\dot{\theta}_3 \\ s_i\dot{\theta}_3 \\ c_i\dot{\theta}_3 \\ \dot{\theta}_1 + \dot{\theta}_2 \end{pmatrix} \tag{3.5}$$

where

$$v_x = \dot{\theta}_1(l_2c_i - l_1s_i) + \dot{\theta}_2(l_4c_i - l_3s_i) + \dot{d}_4s_i$$

$$v_y = -\dot{\theta}_1(l_1c_i + l_2s_i) - \dot{\theta}_2(l_3c_i + l_4s_i) + \dot{d}_4c_i$$

And s_i, c_i represent $sin\theta_i, cos\theta_i$ respectively and the θ_i is the orientation of the contact frame with respect to global frame. Notice that, the required twist stated in (Eq. 3.5) is not independent from dimensions of the target payload since the purpose of synthesis is manipulation. Evolution of contact frames through a given trajectory may also required during a manipulation that can be satisfied using chosen dimension constraints.

However to conduct grasp, manipulation and release payload, the following sections address to find the answers of the initial raised questions.

3.1.1 Number of fingers

Number of finger in a gripper system is the first question which depends on the target tasks. In literature, the most efficient and successful designs are consisted of three fingers such as the Barret hand [84], Salisbury hand [68]. Two fingered grippers are most widely used because of the simplicity and cost effectiveness. However, they are not flexible to multiple applications as shown in Fig. 3.2. On the other hand, grippers having more than three fingers are generally multiple purpose oriented; primarily they are more flexible to numerous grasping tasks of plurality of payloads by utilizing the DoF and sensors. They also able to perform different types of grasping such as pinching, gaiting and so on. However, they lack speed, precision, and require a complicated control scheme to conduct a simple maneuver.

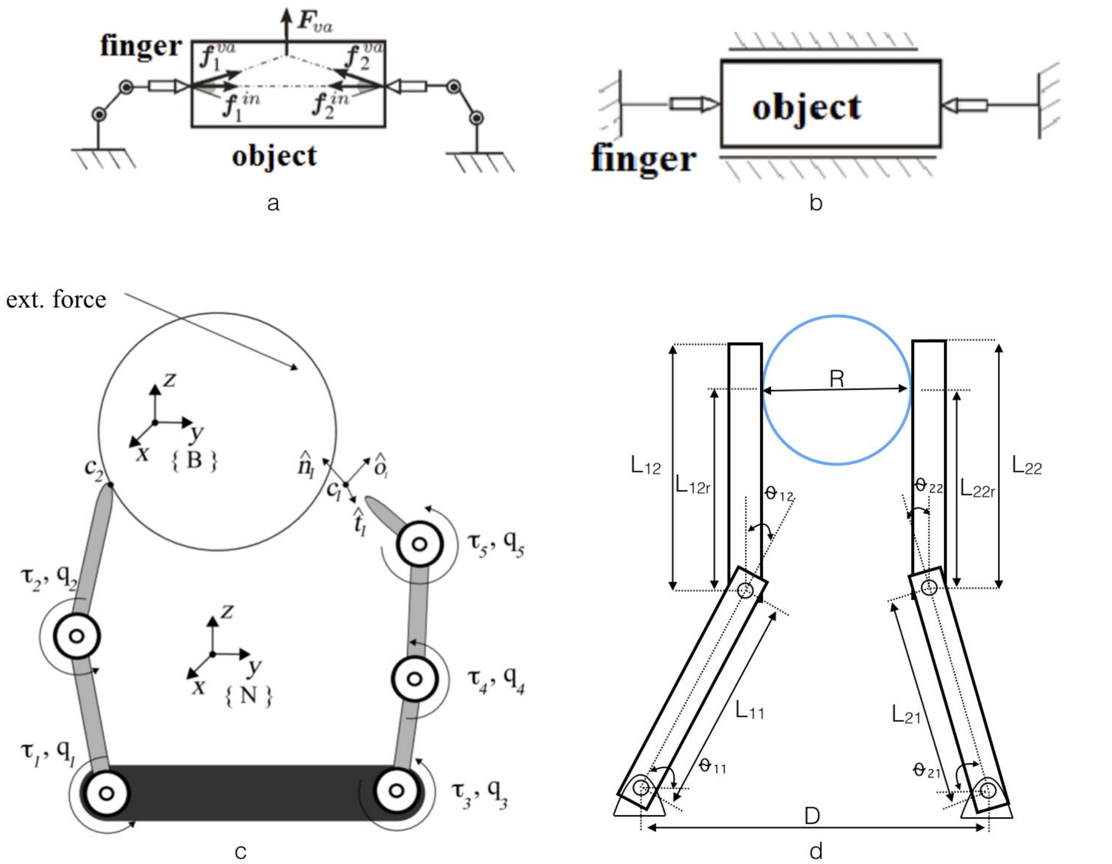


Fig. 3.4 force (a) and form closure (b) approach of grasping (c) and manipulation (d)

Hence "3" is being used as optimum number for finger in many successful designs, primarily because:

- Three contacts can encompass an arbitrary 3D payload better than 2 contacts most likely, without having an additional constraint such as palm.
- A good choice between cost effectiveness and performance (less number of actuator, so the DoF with flexibility)

An arrangement of three fingers in 120 degree (fig. 3.1c) angular orientation is common and exploited in industry.

As mentioned, number of finger is not an independent parameter, and it is related to the contact type that fingers are going to make with payload, arrangement of fingers & grasp stability and last but not least the degree of freedom of the finger; hence, it is important to take into consideration of the other requirements.

Since the aim of the platform is not limited to grasping, it is required to study the best approaches of manipulation as well in this stage, for considering the number of finger.

For in-hand manipulation, human hand uses thumb and index primarily as depicted in Fig. 1.1f-g. Manipulation, in a gripper is governed by force closure method as shown in Fig. 3.4c. However, this approach is difficult for maintaining secure grasp, force control and for complex joint motions during manipulation. Also, this method constrains twist range of a payload, dexterity is limited in specific plane. And hence, this approach (fig. 3.4c) is generally adopted for grasp stability analysis. On the other hand, form closure (fig. 3.4b) in theory constrains the motion of the payload. However, the situation may slightly different, if the contacts between payload and finger allow some degrees of freedom.

Also the release problem after manipulation is an issue while manipulating object by two fingers as explained in Fig. 1.1g, hence it is better to implement additional contact(s) or exploit ingenious mechanism to conceive [31] in the design.

3.1.2 Type of contact

The issue of contact type [1] is another critical constraint for manipulation and grasping [40]. Since dexterous manipulation is object / payload centered, problems regarding posture changes have to be considered from payload-geometry point of view. However, to ensure a stable grasp, common trend is to use parallel jaws (fig. 3.1b) or centric type (fig. 3.1c), and those are typically opposed feasible alterations or movement between payload and finger. And, on the other hand, higher pair of contacts (curve, line, point) such as point contact with friction acts as an additional joint that provides mobility in the contact between finger and payload.

Hence, in order to manipulate payload in-hand, higher pair contacts are more feasible as long as a stable grasp can be obtained.

3.1.3 Method of grasping

For a 2 finger gripper, it is common that either the fingers move parallels or sideways (fig. 3.1b, fig. 3.5b) neither they move towards payload at an angle (fig. 3.5a). However, in a generic platform, it is better to obtain both features of grasps for arbitrary payloads. Manipulation can also be exploited from parallel and angular motions of the finger: those are addresses in the later sections.

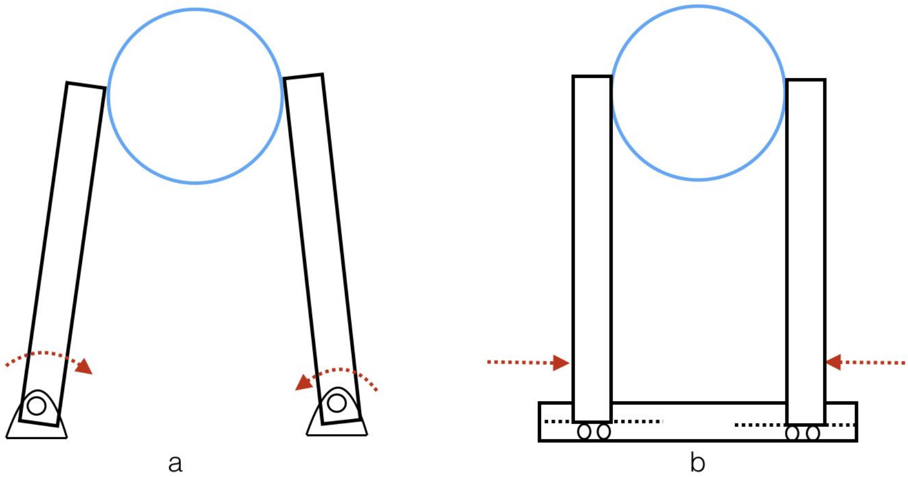


Fig. 3.5 a) angular grasping, b) parallel grasping

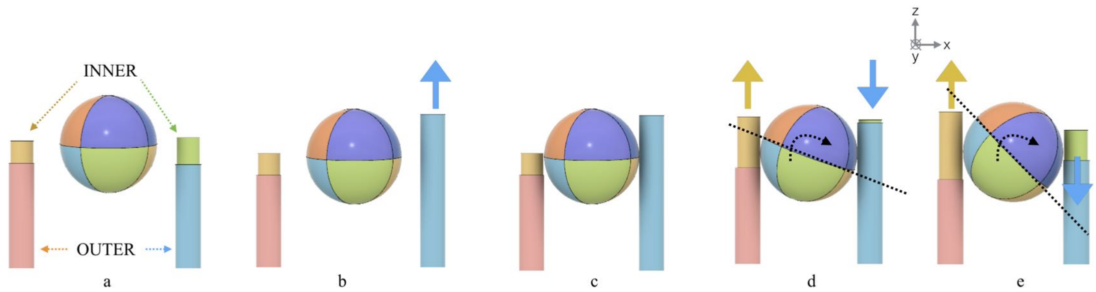


Fig. 3.6 a) INNER, OUTER of two fingers and spherical payload, b) one OUTER moves up, c) grasping of the sphere, d) equal and bi-directional translations cause the payload rotated without moving the center, e) at some point of translations, the INNER got unfolded (right) and maintains the sphere grasped for release

3.2 A two fingered approach for grasping, manipulation and release

Let us begin with considering a possible design by exploiting 2 fingers. Each finger consists of two parts, INNER and OUTER (fig. 3.6a). The INNER and OUTER are hollow cylindrical in shape, however the INNER is slightly shorter in radius compared to OUTER. Both are concentric and the INNER can translate freely inside the OUTER. Also the OUTER can translate with respect to INNER. Let us assume, a parallel grasping method is present in the system, and the gripper is capable to grasp a spherical payload by making contacts of INNER from one finger (left) and OUTER from the other one (right) as shown Fig. 3.6c. At that point, an equal and bi-direction translations of INNER and OUTER as illustrated in Fig.

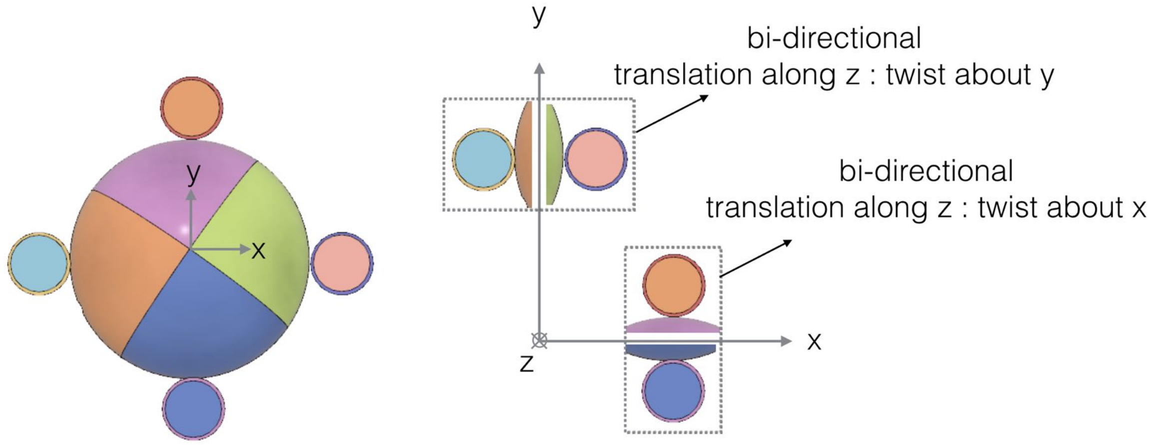
3.6d cause a twist on to the payload. Note that, the center of the payload is unchanged after the twist, since the speed of translations is the same at both contact points.

However, at Fig. 3.6e, the sphere has been rotated a bit further, and the INNER of right finger appears. Let us assume, the configuration of the sphere is the desired posture, and notice that, it is very easy to release the payload without losing that posture, since the tricky appearance of INNER (right) (fig. 3.6e).

Hence, by exploiting this approach, it is possible to grasp, manipulate in-hand and the release issue that has been explained in Fig. 1.1g.

3.3 A four fingered approach to obtain dexterity at two axes

In the last part, an idea has been developed for in-hand manipulation and release of a sphere. The 2 fingered system was capable to twist a sphere, however, the dexterity is limited about the *y* axis (fig. 3.6) only. If an additional pair of fingers is exploited from another plane, the system would get additional twist about *x* axis. This phenomenon has been explained in Fig. 3.7. A feasible design is shown in this section by implementing this concept.



a b

Fig. 3.7 two opponent pairs for improving dexterity: a) top view of the system and sphere, b) finger pairs and associate axis of dexterity

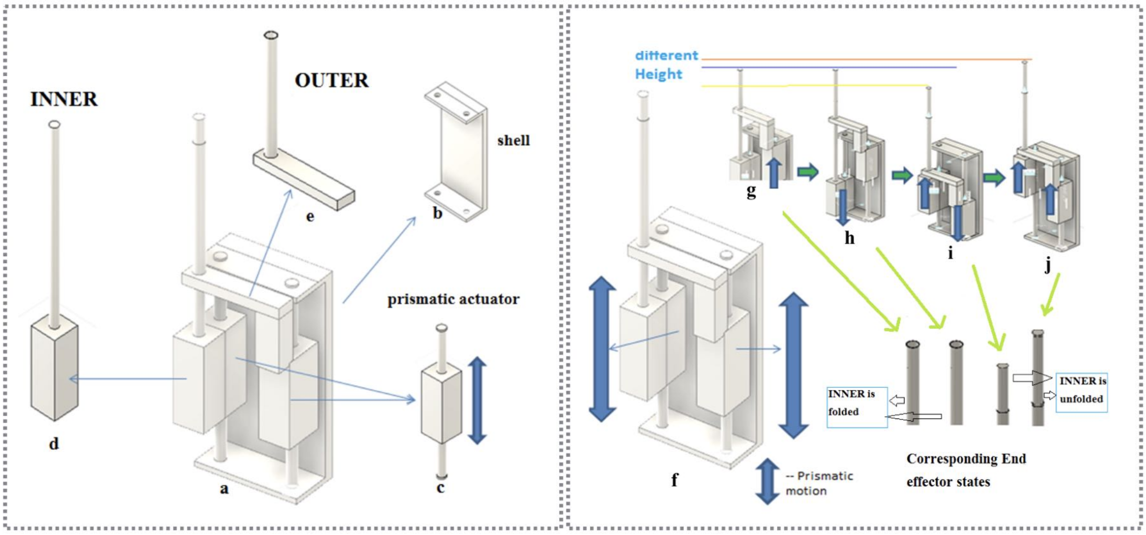


Fig. 3.8 details of modular two DoF finger (leftside) and on right, the actuations of modular finger cause various states at the end effector

3.3.1 Two DoF modular finger

The finger is designed for being used as a modular unit. It consists of a shell where two linear motors are mounted as schematically shown in Fig. 3.8 (letfside). One of the output shaft is a rigid cylinder hereby named INNER and another one is a hollow cylinder named OUTER, shown in Fig. 3.8d and Fig. 3.8e respectively. The parts are designed and mounted such a way that they can co-axially translate with respect to each other and to the shell. Figure 3.8a shows the assembled modular unit. Figure 3.8g-j illustrate the some end effector positions and the actual mobility offered by each modular unit. The INNER can be folded completely by OUTER for several configurations of the finger such as Fig. 3.8g-h. Again, it is visible for the configurations shown in Fig. 3.8i and 3.8j. For the given design, Fig. 3.8j shows the configuration where the maximum stroke is achieved.

3.3.2 Fingers assembly and grasp mechanism

Four modular fingers are assembled as shown in Fig. 3.9a. The four modules are arranged so as to realize two identical couples of fingers, each one able to grasp an object and perform a rotation about two individual axis. While the in-hand operation is performed thanks to the translation of the linear motors, the grasp of the payload requires the mutual motion of the fingers to be done. Due to that, one of the modular fingers of each couple is provided with a properly conceived 2 DoF parallel mechanism able to accomplish this task. The mechanism consists of one rotational and one prismatic actuator as shown in Fig. 3.9b. Some achievable

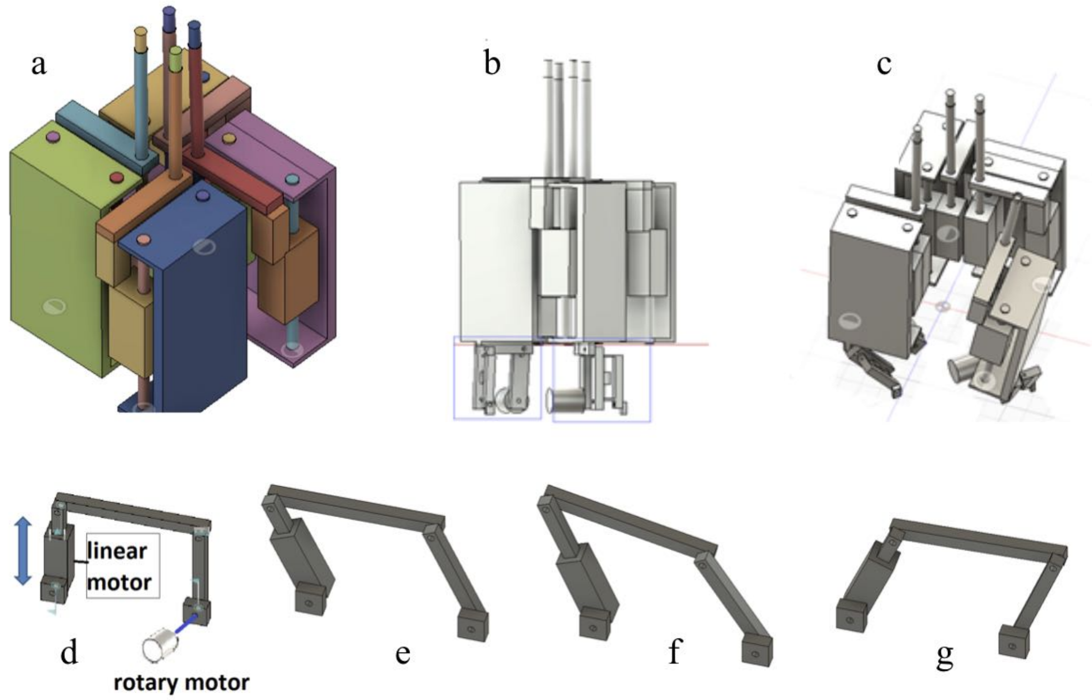


Fig. 3.9 a) 4 fingered arrangement, b) front view of the gripper, each module consists of additional 2 DoF mechanism for grasping, c) a possible appearance of the gripper by using 2 DoF mechanism, d) the 2 DoF grasping mechanism, e)-f) possible configurations of the 2 DoF mechanism

configurations are also shown in Fig. 3.9d-g, which may be required to deal with different shaped objects or various manipulation transition states. The two motors of the mechanism allow the finger to align and grip the object surface with proper alignment for manipulations. Hence, each of the modular unit is capable to provide both the angular and parallel grasping.

3.3.3 In-hand Manipulation

As already mentioned, this goal of the gripper is to obtain dexterous manipulation. Hence, it is able to twist, re-grasp, re-orientate objects by means of complex combinations of four modular units. Twist can even be achieved without altering initial position of a spherical payload. Some manipulations are illustrated in the following. Considering a grasp as illustrated in Fig. 3.10a-d, the fingers are in open initial configuration (fig. 3.10a); the grasping mechanisms can then be actuated to ensure the grasping by creating 2 or 4 contact points (fig. 3.10b). At this point, the gripper can perform two different rotations by actuating the two couple of fingers (fig. 3.10c and 3.10d) one at a time. Successive rotations about

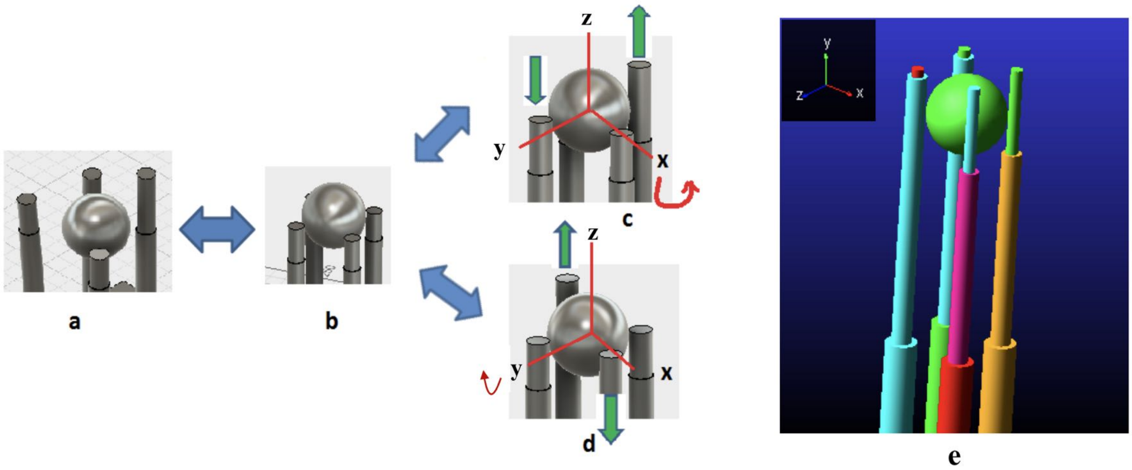


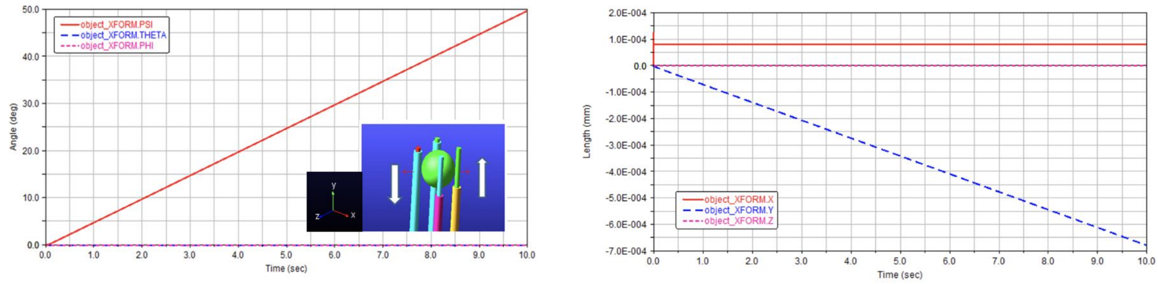
Fig. 3.10 a)- d) some possible manipulations, e) simplified ADAMS [53] model

different axes can be performed by re-creating contact points when needed, i.e. going from configuration of Fig. 3.10c back to 3.10b and then to 3.10d (or vice versa).

3.3.4 Feasibility test by simulation

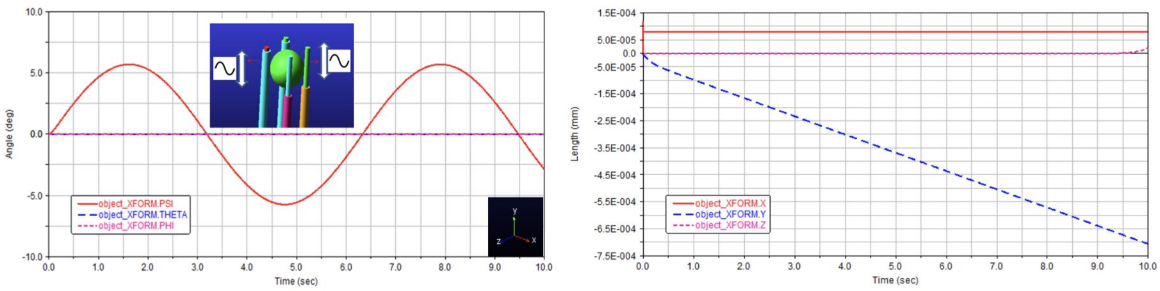
The feasibility of the modular gripper was investigated by means of multibody simulations, carried on with the software MSC ADAMS [53]. The particular case of a spherical payload was considered for this test. Figure 3.10e shows the gripper simplified model and the sphere which have been used to such aim. As mentioned, the grasping configuration can be chosen among the several possibilities provided by the particular kinematic structure of the gripper, the arrangement of the contact points can also be varied during an in-hand manipulation. This peculiarity allows the gripper, as well as, for example, the human hand, to provide the payload with several different motions. Of course, the complexity of this task requires a deep analysis in order to guarantee an efficient motion planning which goes further the intention of this trials, the idea is to verify the kinematic feasibility of in-hand manipulation of the proposed system. Hence, a proper contact model (static friction coefficient was chosen as 0.5 for the finger and payload, and dynamic friction coefficient was 0.2)

For the particular case study of a spherical payload, a twisting without translations can be obtained by providing a couple of fingers with the same speed in opposite directions; in case of not regular objects, such velocities should be analyzed and properly chosen in order to maintain the instantaneous center of rotation constant in space. Figure 3.11a shows a result of a sphere twisting by the gripper, where ϕ, θ, ψ are the Euler angles. The rotation is achieved here by two points of contact, by one downward motion of OUTER and upward motion



a **b**

Fig. 3.11 a) Twisting angles of the object along three axes for bi directional motions of INNER and OUTER at two point contacts b) Change in position of co-ordinate of sphere due to the manipulation



a **b**

Fig. 3.12 a) Twisting angles of the object along three axes for sinusoidal motions of INNER and OUTER at two point contacts b) Change in position of co-ordinate of sphere due to the manipulation

of one INNER. The variation of θ and ϕ during such motion is null due to the absence of torques created by the gripper around such axes. However, the other pair of finger can be in static contacts with object without applying any force on the object. This idea restricts the random translations and rotations of the object except about the permitted direction. In a real application, of course such contact would affect the whole operation with same friction forces. Figure 3.11b shows the trajectory of the center of the sphere during the motion. As visible, the changes are negligible with respect to the radius of the sphere ($r = 10$ mm), denoting a substantial lack of motion of Its center. In order to realize contact-constraint and to further investigating, a sinusoidal motion was given to INNER and OUTER with relatively small magnitude. The twisted angles of the object along three axes are shown in Fig. 3.12a.

Figure 3.12b shows the changes in center position of the manipulated object during the sinusoidal twisting. Even in this case the center of mass is devoid of significant motions, indicating the applicability of the gripper in terms of precise manipulation.

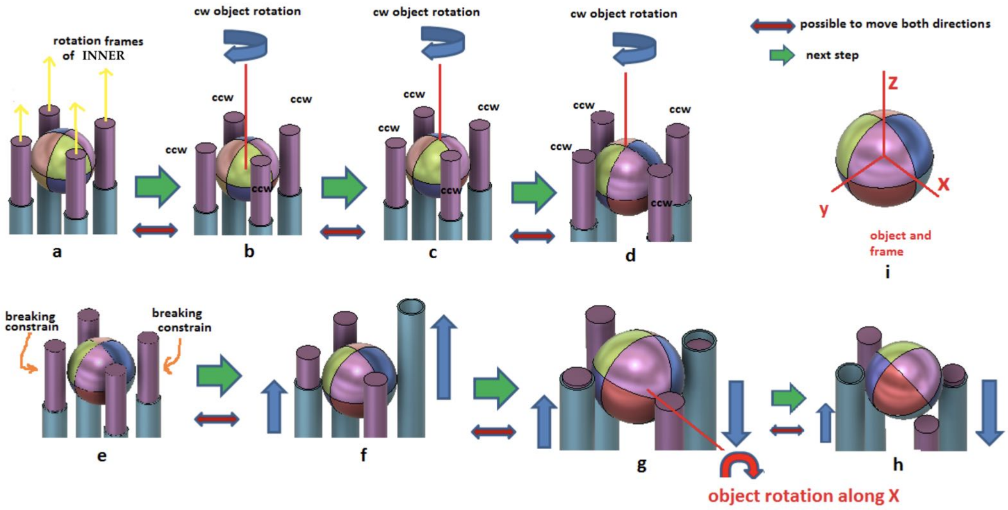


Fig. 3.13 a) the rotation axis of INNER, b)-d) equal speed CCW rotations of INNER cause CW payload rotation, e) One pair is breaking contacts since the rest maintains the grasp, f) the pair re-grasped the payload after changing the OUTER positions, g)-h) bi-directional OUTER translations cause rotation, i) object frame

3.4 A four fingered approach to obtain dexterity in three axes

In the last sections, it has been shown that, the proposed solution can offer two individual twists of a spherical payload about two independent axes. The only twist which was missing, is about z axis (shown in fig. 3.13i). However, this also can be addressed by altering one translation into rotation.

Figure 3.13a-d illustrated the possibilities of obtaining the rest rotation about z axis. Consider that, if the INNER is directly coupled with a rotary actuator despite of a linear actuator, and the OUTER remains with Its previous actuator, the INNER then would able to rotate about the shown axis in Fig. 3.13a. The clockwise rotations of 4 INNER cause a counter clockwise rotation of the sphere as described in Fig. 3.13c-d, while effect of OUTERs will remain same. Both pair of the OUTERs can provide the previous twists about the payload x and y axes (fig. 3.13i)

3.5 Required finger motions and Release analysis

Until now, some feasible designs for obtaining twist and release with final posture of a spherical payload have been presented. In the last part in particular, forecasts a design to

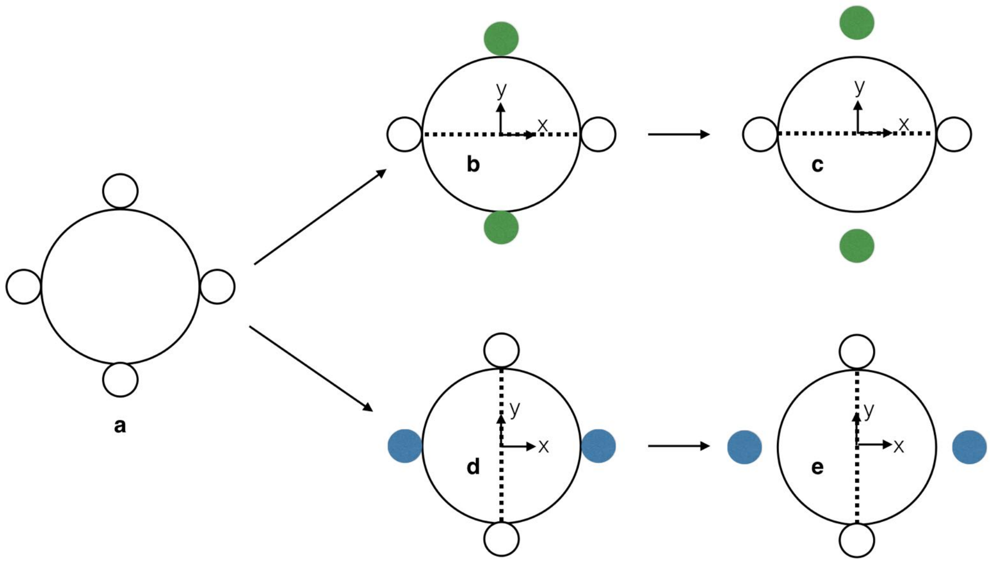


Fig. 3.14 Using 4 contact points by decoupling manipulation and release: a) top view of a sphere which is being grasped by 4 contact points, b) assume that, green pair causes twist on to the sphere, and c) green pair broke the contacts and the other pair maintains the sphere with final posture for release, d) assume that, blue pair causes twist on to the sphere, and e) blue pair broke the contacts and the other pair maintains the sphere with final posture for release

achieve twist about three planes. However, in order to build a generic platform, it is required to analyze the motion of the finger to address twist of several payloads other than sphere.

Each finger in Fig. 3.13 is consists of 2 DoF, but the module has two parts named INNER and OUTER, where the motions are decoupled. Note that, the prismatic motion of OUTER over the INNER was conceived to grab payload after the manipulation, in order to release and keep the final posture. However, this can also be done by the method illustrated in Fig. 3.14. Consider a grasp utilizing 4 contact points as shown in Fig. 3.14a. At that point, one opponent pair can perform bi-directional motions in order to create twist on to the payload such as Fig. 3.14b w.r.t x or Fig. 3.14d w.r.t y. While the remain pairs in both cases (fig. 3.14b,d) are providing the necessary axes, about which the payload are being twisted. And once the twisting is done, the green pair (fig. 3.14c) and blue pair (fig. 3.14d) can break the contacts, while the remain pair able to keep the final posture also to release. Hence, the OUTER translation over the INNER can be avoided; which is more convenient way to address arbitrary payloads in the gripper system.

3.5.1 Design of the modular finger and assembly avoiding mutual translation

In the last section, required motions for manipulating a sphere have been evaluated. This section focuses to develop some feasible designs based on the conceived idea.

From manipulation point of view, in the current concept, a cylindrical finger requires

- translation along an axis
- rotation about the same axis

From grasping point of view,

- parallel grasp
- angular grasp

are required. However several mechanisms can be exploited to achieve the mentioned motions above. And, it is difficult to be convinced on one particular design. Hence, some designs have been proposed in the following sections.

3.6 Case 1

A gripper is shown in Fig. 3.15a, which consists four identical modules. Each module has two frames; module frame (3.15c) and base frame (3.15e). The module frame consists of two DoF, one is rotary and the other is prismatic actuator (3.15b). The rotary actuator is directly coupled with a cylindrical object, titled as finger-tip, (3.15b, name is arbitrarily chosen in this paper) which is supposed to use as finger for the gripper system. The prismatic actuator is mounted with the base frame in such a way that, Its linear motion creates translation to the entire module frame including finger-tip (3.15d). Hence, the finger-tip has both independent rotation and linear motion governed by the two actuators installed in module frame.

3.6.1 Significance of finger-tip

As already stated, the finger-tip is cylindrical in shape and it has two degrees of freedom; due to Its shape, the finger-tip is able to create higher pair of contact with most regular shaped objects such as spherical, ellipsoidal payloads. Since the higher pair of contact allows

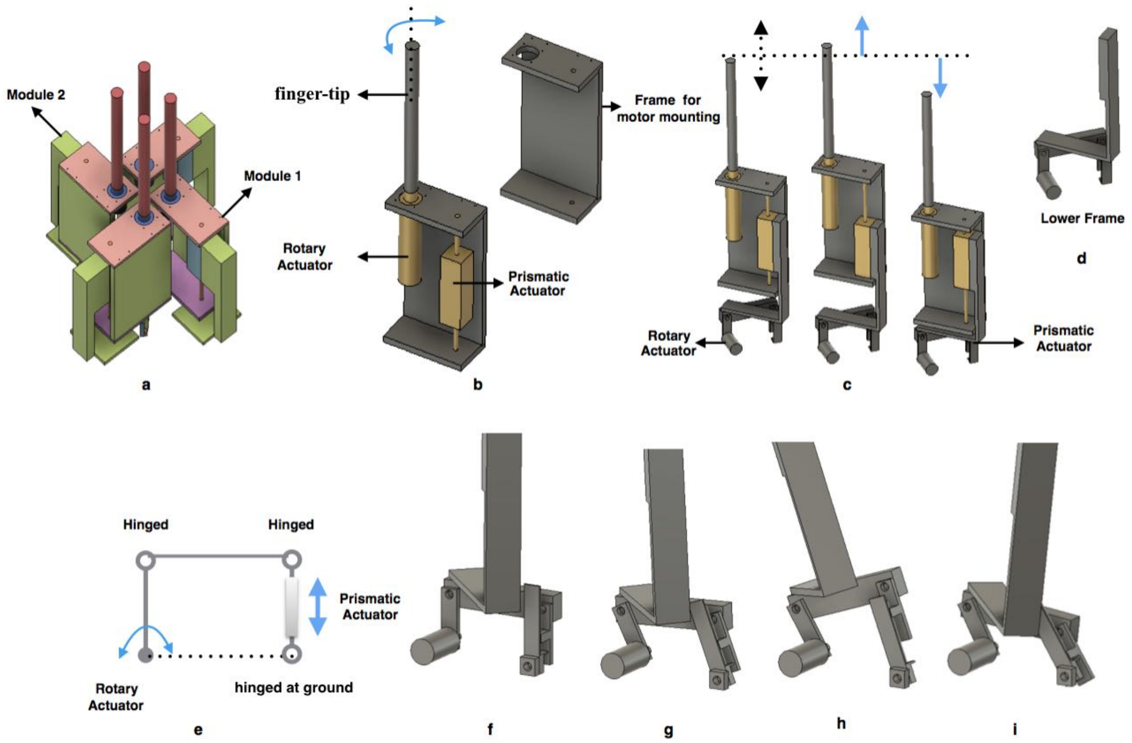


Fig. 3.15 the detail of the modular gripper without INNER-OUTER mutual translation

additional DoF in a kinematic pair, it is convenient to choose such shape (cylindrical shape of the finger-tip) to exploit this advantages during manipulation. The translation and rotation capabilities of finger-tip apply forces at the contact point between object and finger-tip, hence several movements can be achieved. However, one higher pair contact between payload and finger-tip might reduce secure grasping, but by using four finger modules, (3.15a) the issue of stable grasping can be solved.

3.6.2 Fingers assembly and grasp mechanism

The modular finger can be used in several ways to construct a gripper. To obtain three axes dexterity, four modules are considered as shown in Fig. 3.15a. The four modules are arranged so as to realize two identical couples of fingers (module 1 - module 2 and module 3 - module 4), each pair is able to grasp a regular shaped object independently and to apply twists about two different axes, thanks to the prismatic motions of each finger-tip. The rotations of the finger-tips can apply another twist about the vertical axis. Hence the gripper is feasible for three axis manipulation.

While the grasp of a payload needs the mutual motion of the fingers to be done. To this aim, each base frame (3.15e) consists of additional two DoF mechanism similar to previous

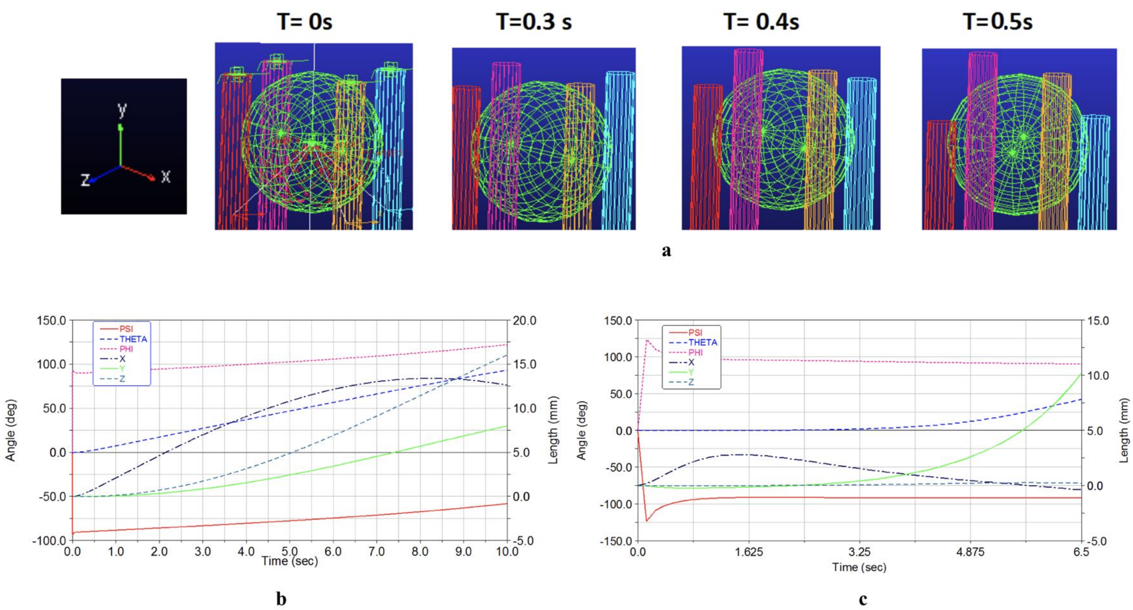


Fig. 3.16 a) in-hand manipulation and release preparation of a sphere, b) twisting angles and center co-ordinates of the object for the spiral motions generated at contact, c) twisting angles and center co-ordinates of the object during manipulation and release

one Fig. 3.9d. The mechanism consists of one rotational actuator and one prismatic actuator mounted below the frame explained in Fig. 3.15f. These two motors allow the finger to align and grip the different shaped object surface with proper alignment for manipulations, which is further illustrated in Fig. 3.15g-j. Due to the arrangement of the pairs (module 1 - module 2 and module 3 - module 4 as shown in Fig. 3.15a) the grasping mechanism (3.15e) is installed in an angular fashion below each base frame (3.15d), so that the fingers of the pair can address both parallel and angular grasping separately.

3.6.3 Feasibility verification of the concept in simulated environment

The feasibility of the new modular gripper was investigated by means of multibody simulations, carried on with the software MSC ADAMS [53]. The particular case of a spherical payload was considered for this test. The twisting capabilities and release of the sphere has been investigated. Figure 3.16b shows a result of a sphere twisted and translated synchronously by the gripper, where ϕ, θ, ψ are the angles with respect to x,y,z. (please note that, the frame is different in simulated environment) In this process, one finger-tip rotates and translates at the same time in a spiral manner, while the conjugate of Its pair does the same but in opposite translation direction (both finger-tips rotate clockwise in a same speed, they also have translation at same speed but different direction). In the whole process the finger-tips of the other pair remain in contact to provide a stable support or in static without applying any force on the sphere. Note that, the inactive pair can also wait and prepare for re-grasp; once the manipulation is done, it can grab the sphere with the final posture of the payload and release it. However, Fig. 3.16b indicates an achievable manipulation possibilities of the gripper.

As mentioned, the grasping configuration can be chosen among the several possibilities provided by the particular kinematic structure of the gripper. The arrangement of the contact points can also be varied during an in-hand manipulation. This peculiarity allows the gripper, as well as, for example, the human hand, to provide the payload with several different motions. Of course, the complexity of this task requires a deep analysis in order to guarantee an efficient motion planning. Figure 3.16a illustrates a fast maneuver, where the red-sky and pink-yellow are two pair of finger-tips. The first pair (red-sky) are applying twist while the other pair applies force and translate the object upward simultaneously. This concept allows to perform manipulation and release preparation simultaneously, which can reduce significant energy and time of production. Figure 3.16c shows one such process, where the release direction is set to y direction. Consider the shown angular values (ϕ, θ, ψ) at $t = 6.5s$ are required as final posture. The process can be done such a manner that, the payload is also ready for release at the time of manipulation. The motion planning of all finger-tips are needed to be exploited such a way that, they can address the requirement in minimum time, working simultaneously as required.

The form closure manipulation restricts random translations and rotations of the object, except in the permitted direction. However, the inactive pair contacts with object during manipulation would affect the whole operation with same friction forces in a real application. Thus, it is worth mentioning that, providing extra constraints to the motion is possible even though the applicability of this strategy should be analyzed case by case depending on the mutual friction substituting among the surfaces of the finger and the target object. From performance perspective, it is worth mentioning that, grasping and twisting motions are depended on friction between and fingers and the target object shape/surfaces, and hence, relative motion may occur during the process that may cause manipulation deficiency. This fact is true for all grippers, and since dexterous manipulation generally uses higher pairs of contact (point, line, curves) with the object, the error cannot be vanished in practical case. The proposed gripper manipulates objects by means of form closure (in an envelope grasp), where the inactive pair also creates constrain with the object without applying force onto the object in ideal case. If any relative motion occurs in that direction (namely in the direction of inactive pair) the deficiency might be reduced.

Table 3.1 Details of the finger prototype

Component	Details
Linear Actuator	Faulhaber LM 2070 040 01, continuous force 9.2N, stroke length 40 mm, max speed 1.9 m/s.
Rotary Actuator	Maxon DC 365511, gear ratio 86:1, optical encoder: HEDS-5540.

3.6.4 Experimental test of finger module

This section shows the results obtained for a set of experiments conducted on the device prototype. The main aim is to investigate the behaviour of the modular finger for in hand manipulation. A prototype finger has been built as shown in Fig. 4.9a, while the test rig is illustrated in Fig. 4.9b. The prismatic actuator of the finger is mounted with a slider such that the finger can translate along it (4.9b).

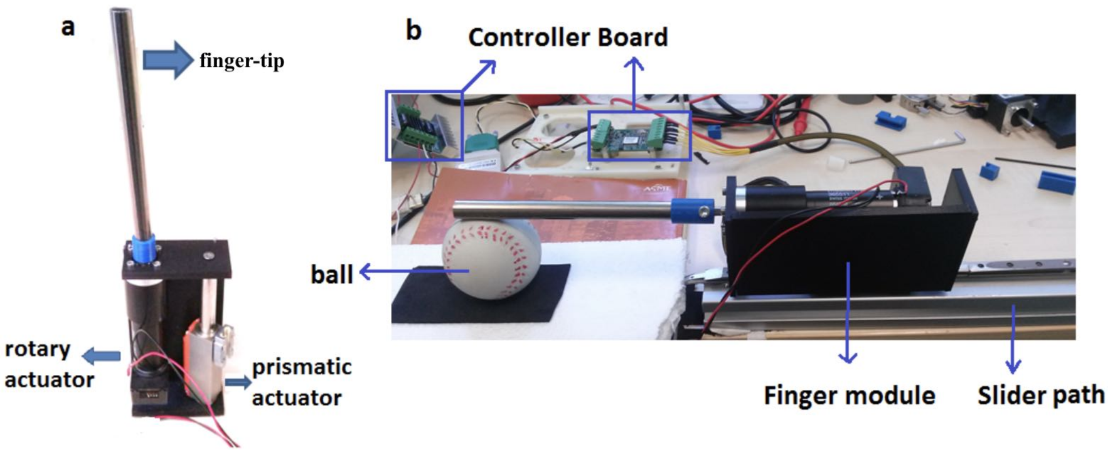


Fig. 3.17 Experiment setup, a) modular finger b) setup details

Twisting experiments

Since the finger-tip of modular finger has two DoF (i.e. rotation and translation) therefore it is able to apply twist about two axes separately to the object and also a combined transformation (due to both motions simultaneously) is achievable onto the object. A sphere 60 mm diameter

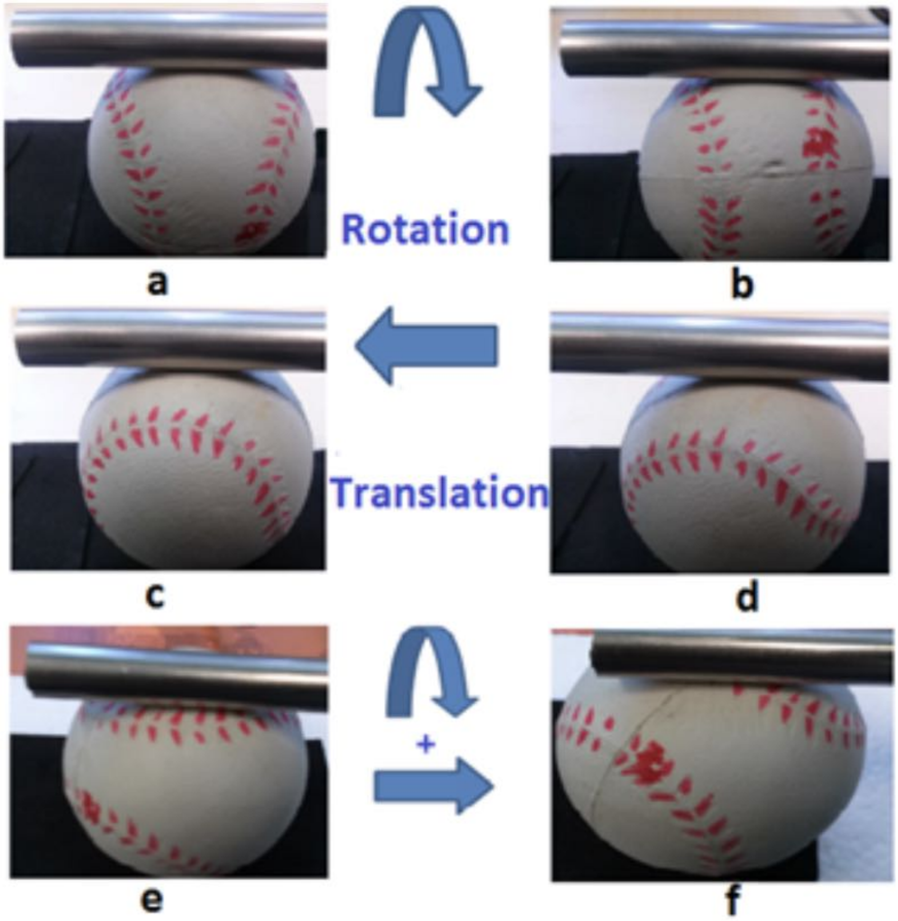


Fig. 3.18 rotation (a-b), translation (c-d) and hybrid motion (e-f) of the finger-tip

was chosen as manipulated object in this experiment. As shown in the setup (fig. 4.9b), one contact point where the ball is grasped is fixed, the translation of the finger-tip applies a rotation and translation to the object, as illustrated in Fig. 3.18c to Fig. 3.18d. However, the translation can be nullified in the actual gripper according to the manipulation requirement, thanks to the possibility to achieve bidirectional translation of the contact pair of fingers. The same phenomenon is applicable for the rotation of the finger-tip as illustrated in Fig. 3.18a to Fig. 3.18b. Using both the translation and rotation simultaneously, it is possible to achieve many complicated object manipulation requirements by exploiting proper motion planning and synchronous control; and hence the complete gripper of four fingers will enhance the speed for a given manipulation and also a robust platform for complex requirements.

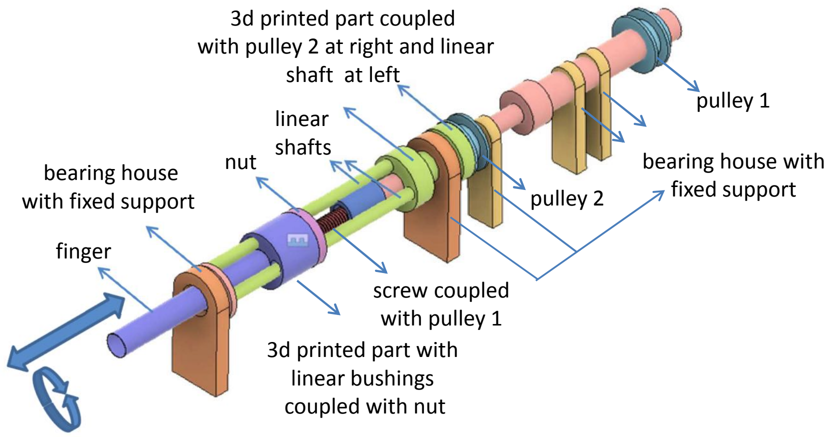


Fig. 3.19 CAD of the 2 DoF ball-screw finger

3.7 Case 2

As mentioned, required motions for manipulation can be transferred using many mechanisms. Here a finger has been developed, which is conceived from 2 DoF ball-screw mechanism.

3.7.1 Inspiration from ball-screw mechanism

The detail of each finger module is illustrated in Fig. 3.19. Pulley 1 is coupled with a lead screw through proper bearings. The nut of the screw is mounted with a cylindrical part that contains three linear bushings; so that the cylindrical part can slide along three linear shafts. The other side of the cylindrical part is mounted with finger (left side at fig. 3.19). Pulley 2 is coupled with another cylindrical part consisting three linear shafts. The arrangement of screw and pulley 2 is such that both are concentric and both can rotate independently. Since, if pulley 1 rotates, and pulley 2 being fixed, the finger experiences translation. While, the finger rotates (without translation) when pulley 1 and pulley 2 have same speed of rotation

Table 3.2 2 DoF finger tip movements

Motion of pulleys	Movements
Pulley 1 rotates, pulley 2 stationary	pure translation
Both pulleys at same speed & direction	pure rotation
Pulley 2 rotates, pulley 1 stationary	constant spiral motion
Other cases	variable spiral motion

and direction. Hence, driving pulley 1 and 2 at different speeds, translation and rotation at same time such as spiral motion is achievable at finger.

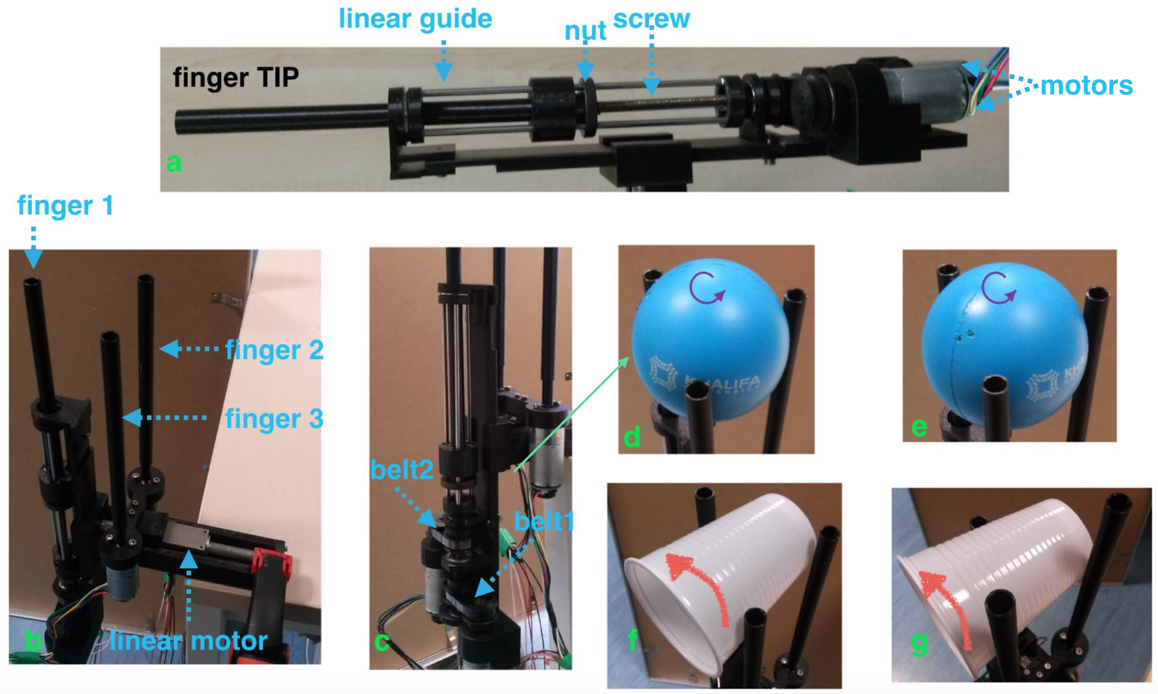


Fig. 3.20 Details of ball-screw finger prototype and experiment

3.7.2 Experimental validation of ball-screw and Three fingered system

An optimized gripper (5 DoF) is developed to conduct experiments to verify the feasibility of the ball-screw conceived finger for manipulation. Also, the system was chosen as a three fingered platform, to the aim of evaluating the feasibility of release in a 3 contacts system, as shown in Fig. 3.20b.

The platform (fig. 3.20b) consists of three fingers; one finger (2 DoF) of them is similar to ball screw, whose detail is shown in Fig. 3.20a and titled as "finger 1" in the platform (fig. 3.20b). Finger 2 and finger 3 (fig. 3.20b) both are directly coupled with two independent rotational actuators and hence, they both are 1 DoF each. Both finger 2 and 3 are mounted on a base, which slides towards and backwards with respect to the finger 1, with the help of an additional linear motor. Hence the platform is capable to grasp and manipulate several payloads. Various motions have been generated by the fingers and expected postures achieved for different shaped objects as shown in Fig. 3.20 d-g. However, the difficulties occurred obtaining release with the final posture, and hence it is obvious that, one more support or contact point is required to achieve that requirement.

3.8 Case 2

Until now, a spherical payload has been considered. However, since the aim for the platform is to manipulate different geometric payloads should be addressed hence, two other regular shaped payloads (ellipsoidal, rectangular) are chosen in this phase. And the assumptions in the design process are taken such a manner that, the gripper platform will also address payloads regardless of ellipsoidal and rectangular shapes.

3.8.1 Number of contact points / fingers for ellipsoidal payload

An example of planar repositioning of an ellipsoidal payload is shown in Fig. 3.21a. It is possible for a three fingered gripper arrangement shown in Fig. 3.21a, to alter the payload to the desired posture with the help of appropriate dimension synthesis and control, however the final posture may require three fingers are in contact with the payload after the manipulation. And in the final configuration of the ellipsoid, it is also possible that, any two of the fingers (out of three) might not guarantee a stable grasp. Hence the system might have found itself in an awkward position to release the ellipsoid, if additional movement is required to release/place the payload after manipulation. Also desired posture of the payload may demand to be such that, it requires the fingers are needed to be placed very closed to their singularity conditions (in order to obtain the desired posture of payload). In such cases, the gripper fingers cannot have the ability to reconfigure themselves (with maintaining stable grasp). These problems (release and singularity) can be solved by increasing the number of contact point by one, which basically implies the use of an additional finger as shown in Fig. 3.21b.

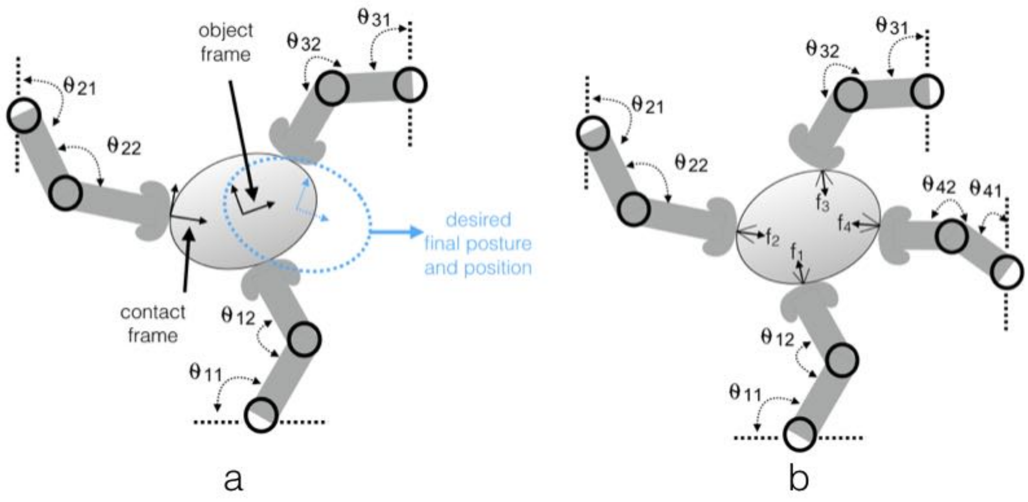


Fig. 3.21 a) An example of posture change of an ellipsoidal payload by three fingers, b) Manipulation by four fingers

Additionally, it can be noticed that, four contact points can be used as two individual pairs of opponent fingers (opponent contacts shown in 3.23a). Which essentially means, each pair is solely capable to maintain grasp due to their arrangement (of fingers). Hence, one pair can maintain a stable grasp once the posture change is done, while the rest pair can reconfigure themselves and perform the remaining task for release, maintaining the desired posture of payload.

3.8.2 Planar transformation

DoF for translation component & contact arrangement

The transformation between given and final posture shown in Fig. 3.21a of the payload can be distinguished as a summation of separate rotation and translation components as shown in Fig. 3.22a. Figure 3.22b describes the necessary numbers of DoF for current system of Fig. 3.21b. Each of the finger needs one translation and one rotation, therefore the total eight DoF are required. However, as mentioned, the desired requirement (both the rotation and translation) can be achieved by using only four translational joints and repositioning different location of contacts as well, as illustrated in Fig. 3.22c. The drawback of this principle (fig. 3.22c) is that, it is not a feasible solution for the case when the i and j become equal, (i, j shown in Fig. 3.22d), i.e. a spherical payload, or other shaped payloads such as rectangular solid due to the geometrical constraints. Moreover, since higher pair of contacts between finger and payload are being considered, the arrangement (in terms of finger positions) shown

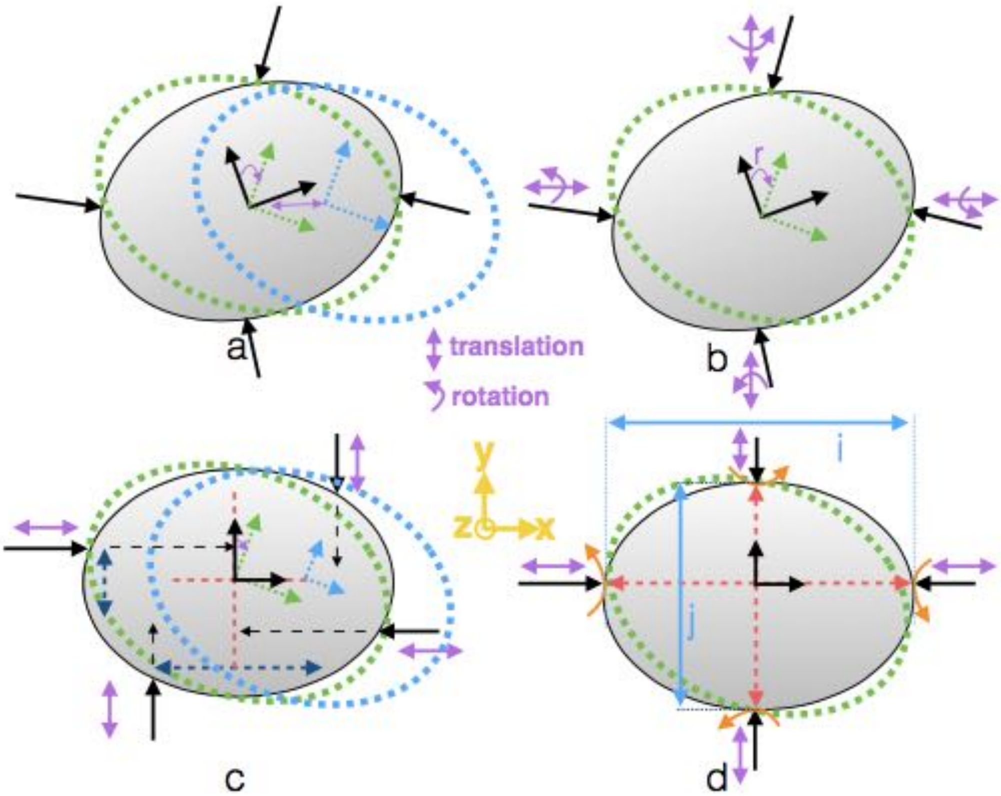


Fig. 3.22 planar transformation of the payload in terms of rotation and translation

in Fig. 3.22c might not be able to maintain stable grasp for several payloads. Ideally, the contact arrangements by fingers shown in 3.22d should provide better grasp stability, as both the vectors joining two contacts of each pair pass through the payload center. Hence, two opponent pairs (placed at perpendicular to each other, named q-r and s-w shown in Fig. 3.23c) can generate several orientations to address grasping arbitrary payloads.

DoF for rotation component

It has stated several times that, higher pair of contacts (curve, line, point) provides additional degree of freedom at contact point, now this advantage can be utilized to implement payload rotation about the z axis (frame shown in Fig. 3.22) at the current finger arrangement (3.22d).

Twist or rotation needs a continuous relative motion between the contact points and payload (ellipsoid in that case), hence, circular shaped finger-tips have been chosen (3.23a) and they are capable of providing rotation (1 DoF about z of Fig. 3.23a, additionally translation towards payload center to maintain stable grasp) such that, the payload will experience twist. Figure 3.23a shows one such example, where each circular shaped finger assumed to have

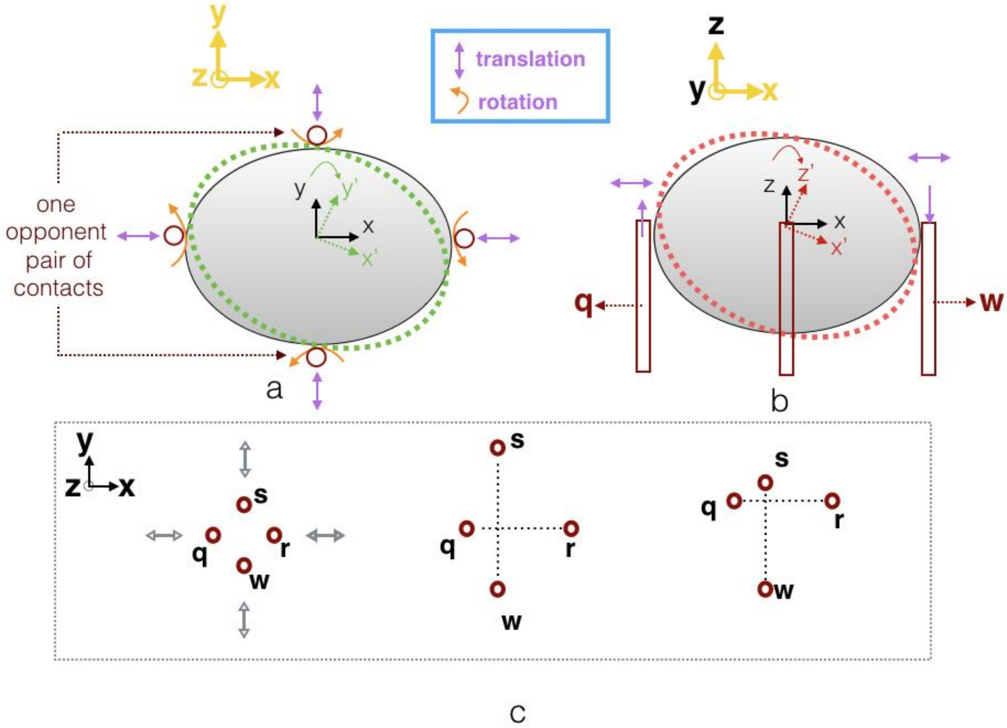


Fig. 3.23 Evaluation of necessary degree of freedoms and possible arrangement of contacts

translation towards payload center and also can rotate around z axis (3.23a). Hence, for a planar manipulation of a payload, eight DoF ($4 finger \times 2$ DoF) are essential to address the maneuver smoothly.

3.8.3 6 DoF transformation of payload (spherical / ellipsoidal)

Until now, the required DoF of gripper for planner transformation of payload have been analyzed. The rest DoF of payload manipulation are,

- Rotation about x (frame shown in 3.23)
- Rotation about y (frame shown in 3.23)
- Translation about z (frame shown in 3.23)

These two rotations and one translation can be obtained in the current developed gripper (of eight DoF). Consider that, circular finger-tips (fig. 3.23a) are cylindrical in shape in three dimensions, as illustrated in Fig. 3.23b. Now, if the fingers has additional translations in the z direction, the remaining three motions of payload can be addressed by

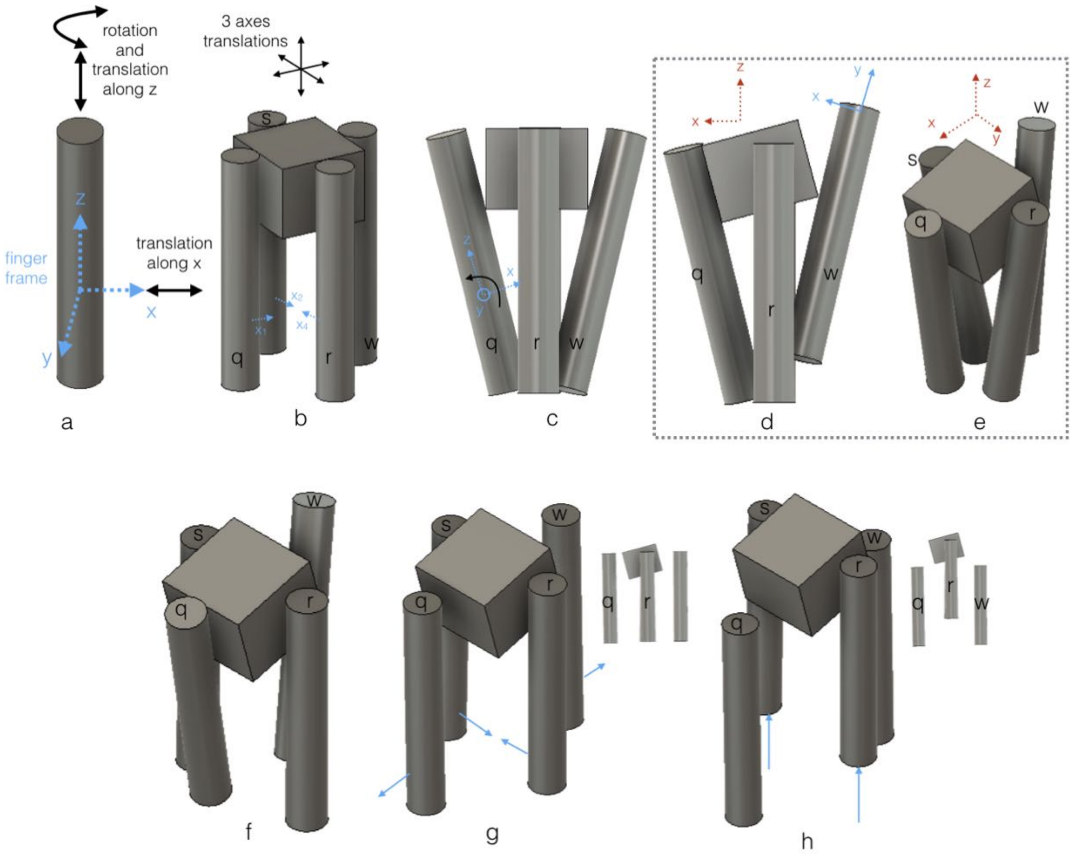


Fig. 3.24 a) 3 DoF cylindrical finger, b) A four fingered stable grasp of rectangular payload, where 3 translations are possible, c) Using the additional DoF (rotation about *y* axis of each finger-body-frame, each finger is 4 DoF here) the translation (along *x* of each fingers body frame) grasp of (b) can be evolved into this shown one here, d) and e) Rotation of rectangular payload, due to the controlled translations toward *y* and *x* of the body frame of finger titled *w*. f) to the aim of release with the current posture, *s*-*r* pair reconfigured and grasp the payload in proper way, g) *q*-*w* pair broke the contacts and moved away from payload, h) *s*-*r* pair moves up maintaining the final posture of payload for release

- Rotations about x and y: equal and bi-directional translations of one pair of opponent fingers in z direction
- Translation along z: equal and same direction translations of one or both pairs fingers z direction

One example is illustrated in Fig. 3.23b, where the q and w are two cylindrical fingers (having translation along z of each) of one opponent pair. The equal and opposite translations of q and w will apply a twist onto payload about the y axis.

On the other hand, the translation along z of payload can be obtained by actuating the finger pairs-translation in same direction. Hence, until now, it can be stated that the four cylindrical shaped fingers (of three degree of freedom each, one rotation and two translations) in the chosen arrangement, are capable to provide six DoF manipulation to a spherical/ ellipsoidal payload.

3.8.4 Manipulation of rectangular payload

The DoF of finger (described in Fig. 3.24a) are already capable to provide three translations into a rectangular payload. Assuming the rectangular payload is in a stable grasp:

- s and r can move towards x and -x respectively (3.24b)
- q and w can move towards x and -x respectively (3.24b)
- both pairs can translate towards the z direction (3.24b)

To obtain rotation, it can be useful to add another DoF to each finger about It's y (shown in 3.24a). This phenomena is shown in Fig. 3.24c-e. The new DoF will allow the fingers to re-orientate around the payload shown in Fig. 3.24c, and a controlled translations along x (to maintain grasp) and y (for manipulation) of body frame of one finger (the finger named w shown in 3.24d in this case) will create a twist shown in Fig. 3.24d, while 3.24e describes the isometric view of 3.24d. Hence, the rectangular payload can achieve two additional rotations; any finger q or w at the orientation Fig. 3.24c can trigger a twist, and this fact is also true for any finger of opposite pair r-s. Hence, the total number of DoF of the four fingered system currently becomes sixteen (4 DoF $\times$ $4 finger$).

3.9 The Final Platform: Dexclar

By assessing all the conceived ideas from the previous sections, this part develops the final platform called Dexclar (*DEX*terous re*C*onfigurable modu*LAR*). The synthesis process of the mechanism of grasping and manipulation, release has been portrayed in the following sections. It is worth mentioning that, Dexclar is capable to provide 6 degrees of freedom to spherical and ellipsoidal payload, and 5 degrees of freedom to a rectangular shaped payload. By these maneuvers, Dexclar is also capable to manipulate different shaped payloads, or at least the required motions for manipulation can be obtained studying the methodology that has been used in the formation of this platform.

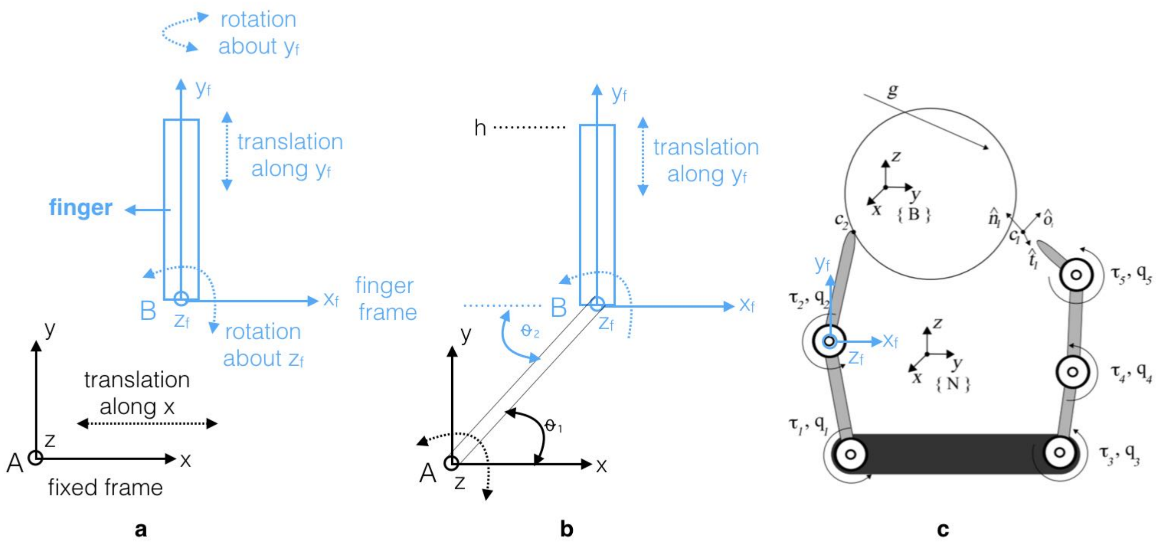


Fig. 3.25 a) Required degree of freedom of cylindrical finger, b) A solution to avoid one translational joint in the finger, c) Classical approach to grasp synthesis

3.9.1 Mechanism Synthesis for Dexclar Finger

This section has been exploited to synthesize Dexclar platform by choosing appropriate mechanism. Figure 3.25a details the required motions of each finger, where three DoF are required w.r.t $finger\ frame$ (two rotations about y_f and z_f, one translation along y_f) and one translation along x w.r.t $fixed\ frame$, that is one DoF motion. The three DoF w.r.t $finger\ frame$ are mainly needed for manipulation, although the finger rotation about z_f contributes for grasping as well, one such example is shown in Fig. 3.25c. While the translational motion along x w.r.t $fixed\ frame$ is mainly required for confirming grasping. Although the rotation about z_f contributes to both in grasping and manipulation, in this synthesis process, it is addressed in the grasping section.

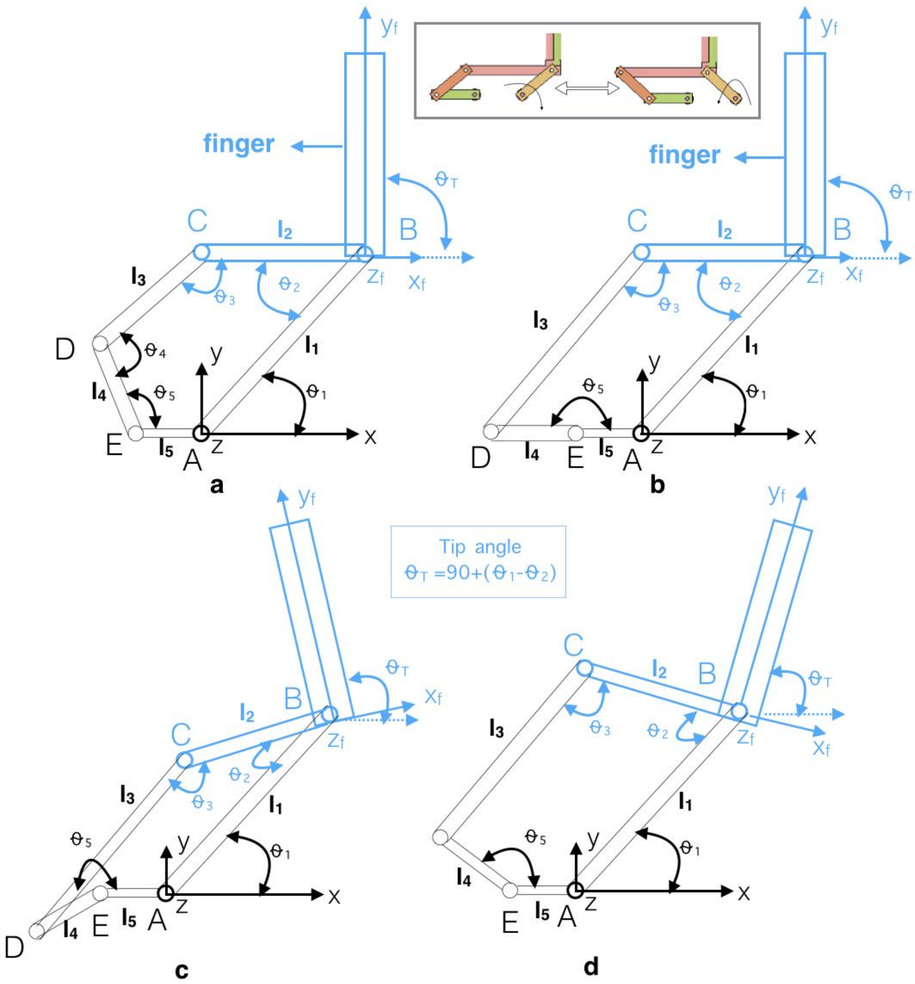


Fig. 3.26 a) Five bar linkage mechanism with finger b) a chosen initial configuration of fivebar mechanism c) and d) actuating θ_5 to will cause finger rotation about z_f

3.9.2 Mechanism for Grasping

For grasping, the necessary translation along x can be achieved assuming a link rotation (which connects A and B shown in Fig. 3.25b); in that case, the x_f of $finger\ frame$ must to be maintained an angle of θ_2 with the link such that, the rotation of link about A, that is θ_1 is always equal to θ_2. It should be stated that, the height h (shown in Fig. 3.25b) will vary as the link rotates about A, however, before making a contact with payload, the changes in h has no effect on manipulation, or grasping; also the effect can be nullified by translating the finger towards y_f. Besides, a rotation about z_f is required for manipulation, irrespective of the value of θ_1. To meet these requirements, planar linkage mechanisms are preferable, hence five bar linkage is chosen.

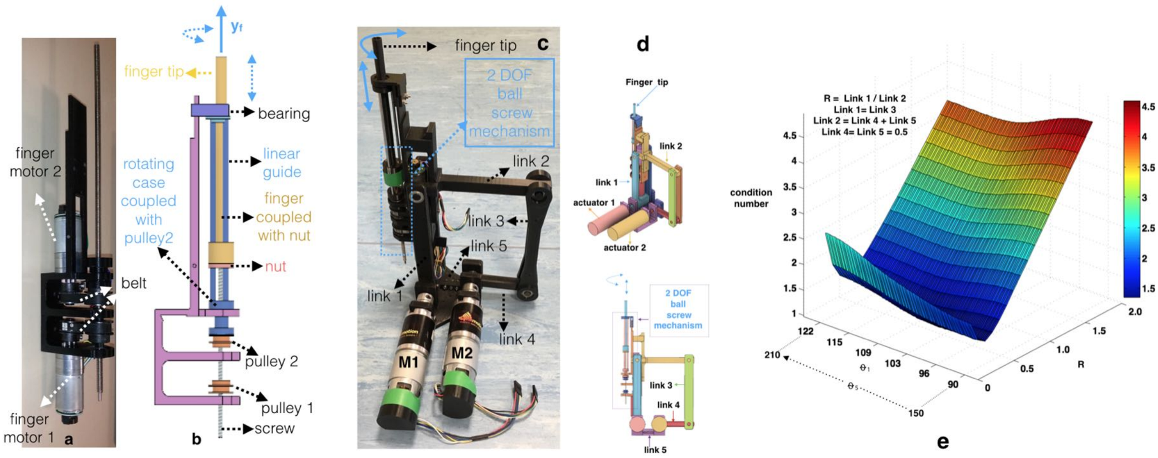


Fig. 3.27 a) Dismantled 3D printed ball spline inspired finger module, b) detail CAD of ball screw mechanism, c) Developed finger prototype that consists of fivebar and ball screw mechanism d) CAD of the finger, e) Mobility [48] of fivebar at different R, for a given joint motions at θ_1 and θ_5.

As the five bar linkage is a 2 DoF mechanism, it is possible to obtain the desired translation (along x) and rotation (about z_f) requirements of the finger from the synthesis shown in Fig. 3.26a; where the finger is mounted with link2, named as l_2. While, θ_T is defined as $Tip\ angle$ and l_5 is fixed at ground. θ_1 and θ_5 are required to be controlled to achieve desired position or motion at the finger. To avoid complexities in the motion planning of θ_1 and θ_5 joints, the initial configuration of five bar is chosen as Fig. 3.26b, where $l_1 = l_3$ and $l_2 = l_4 + l_5$ are maintained. Choosing such conditions will allow $\theta_2(t) = \theta_1(t)$ and $\dot{\theta}_2(t) = \dot{\theta}_1(t)$ while varying the θ_1 at $\theta_5(t) = 180$ and $\dot{\theta}_5(t) = 0$ conditions. Hence, such dimensional choice (fig. 3.26b) will enhance the motion planning (of θ_1 and θ_5) in terms of $Tip\ angle$ very easily, because it ($Tip\ angle$) can be maintained 90 $degree$ while actuating θ_1 at $\theta_5 = 180\ degree$ and $\dot{\theta}_5(t) = 0$ conditions (shown in inset of Fig. 3.26), also it can be varied by actuating the θ_5, ($\dot{\theta}_5(t) \neq 0$) if necessary. Figure 3.27e illustrates mobility criteria at several link ratios ($R = Link1/Link2$) for a given joint motions $\theta_1 = 90\ to\ 130$ and $\theta_5 = 150\ to\ 220$ and the ideal range of link synthesis is 0.4 to 0.6 for these conditions.

Figure 3.28a describes the design parameters of each finger. θ_T is defined as finger-tip angle and It's maximum and minimum values are $\theta_{Tmax}, \theta_{Tmin}$ at a given θ_1. Notice that, θ_T is a function of $l_1, l_2, l_3, l_4, l_5, \theta_1, \theta_5$; and the grippers ability of grasping and manipulation dependent on It's range ($\theta_{Tmax}, \theta_{Tmin}\ @\theta_1$). Using inverse kinematics of five bar linkage,

$$\theta_3 = 2\arctan((A \pm \sqrt{(}A^2 + B^2 - C^2))/(B - C)) \tag{3.6}$$

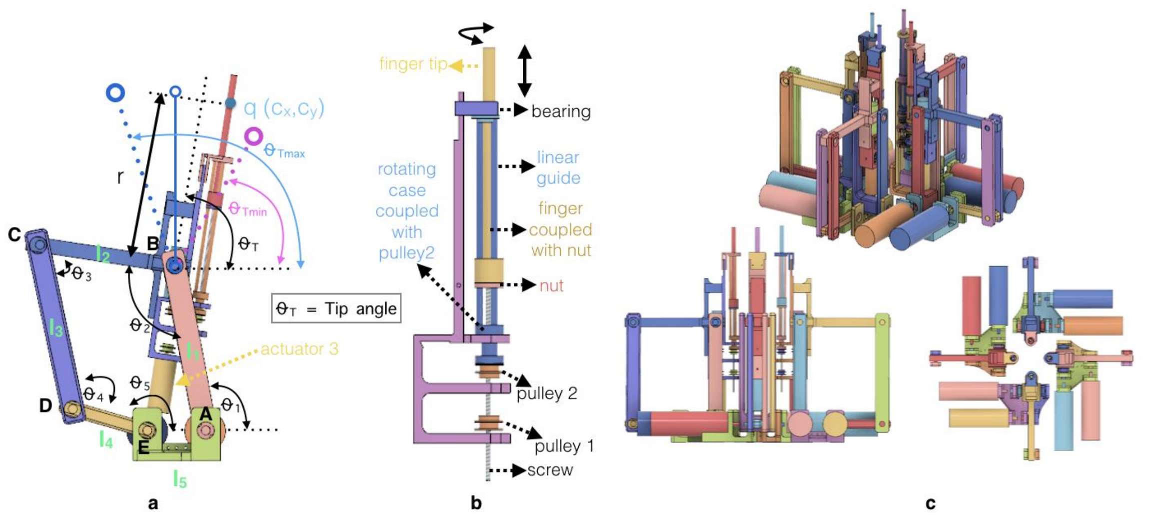


Fig. 3.28 a) design parameters of a finger & b) ball screw mechanism, c) the proposed gripper arrangement named Dexclar (consists of four modular fingers of type (a)

$$\theta_4 = \arcsin(D/l_3) \tag{3.7}$$

where

$$A = 2l_2l_1 \sin\theta_1 - 2l_4l_2 \cos\theta_5$$

$$B = 2l_2l_5 - 2l_4l_2 \cos\theta_5 + 2l_2l_1 \cos\theta_1$$

$$C = l_4^2 - l_3^2 + l_2^2 + l_1^2 + l_5^2 - l_4l_1 \sin\theta_5 \sin\theta_1$$
$$-2l_4l_5 \cos\theta_5 + 2l_1l_5 \cos\theta_1 - 2l_4l_1 \cos\theta_5 \cos\theta_1$$

$$D = l_2 \sin\theta_3 + l_1 \sin\theta_1 - l_4 \sin\theta_5$$

Also from Fig. 3.28a,

$$\theta_T = 90 + (\theta_1 - \theta_2) \tag{3.8}$$

Now, the Jacobian matrix of point C can be written as

$$J_c = \begin{pmatrix} -l_1 \sin\theta_1 & -l_4 \sin\theta_5 \\ l_1 \cos\theta_1 & l_4 \cos\theta_5 \end{pmatrix} \tag{3.9}$$

which consists of the two independent variables θ_1 and θ_5. However, θ_5 can be considered and estimated as a dependent variable in terms of θ_1 and θ_T, just like θ_3 is estimated in terms

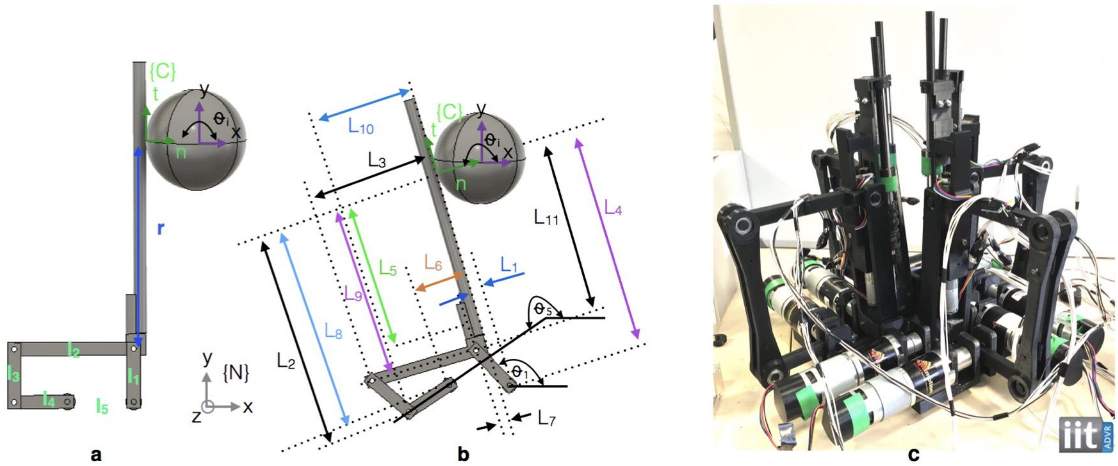


Fig. 3.29 a) A finger and sphere in a point contact with friction and b) screw parameter detail, c) prototype of Dexclar

of θ_1 and θ_5 in (1). Thus, θ_5 can be generated for desired values of $\theta_{Tmax}, \theta_{Tmin}$ at a given θ_1 with predefined link dimensions. Obviously, the choice of θ_T at a given θ_1 must not be closed to singularity condition. To evaluate the mobility at some given states (θ_1, θ_T), the condition numbers [48] of (4) have generated at several cases illustrated in Fig. 3.30.

Now consider, $Ratio$ is a variable such that $Ratio = l_1/l_2 = l_3/l_2$. $Ratio$ and θ_T are used as variable in Fig. 3.30. Now, Fig. 3.30a shows the mobility of the mechanism when θ_T moves from 60 to 120 degree and θ_1 remains constant at 90 degree. $l_5 = 0$ indicates that, $l_2 = l_4$, which physically means, it is a four link parallel mechanism, where the point E shown in Fig. 3.28a moved to point A. Figure 3.30b plots the mobility at the same case except when θ_1 has increased to 130 degree. The effect of rising θ_1, increases the condition numbers specifically in low $Ratio$ region.

Figure 3.30c-f, show the effects of increasing l_5/l_4 on mobility at the same described condition previously. It is clear that, as l_5 increases, the $Ratio$ region with high value starts getting worse. It is also noticeable from Fig. 3.30 that, for the low values of l_5/l_4 the overall mobility is better for wider span of $Ratio$. For a requirement of maneuvering θ_T =60 to 120 degree at $\theta_1 = 130$ degree, roughly $l_5/l_4 = 0.2$ and $Ratio$ = 0.4∼0.7 (fig. 3.30c) can be considered. Other values of $Ratio$ such as 0.25 can also be chosen at $l_5/l_4 = 1$ (fig. 3.30e) for the same requirement. The choice can also be made from the plot $l_5 = 0$ (fig. 3.30a), however having an additional link l_5 will amplify θ_5 profile slightly for a given profile of θ_T; which is an advantage in terms of control, when θ_T needs to move precisely.

The multiple choices for synthesis provide robustness to meet user specific applications, where, additional constraints may specify the choice more distinctly, one such case has illustrated in next part.

3.9.3 Synthesis for a given constraint

Let us consider a manipulation, where a finger-tip needs to push an object or move away due to synchronous fingers action (fig. 3.29b), or to break and avoid further contact instantly. Many such cases like these the contact point between object and finger-tip needs to evolve or break completely at a given time. In those cases a measure of twist of finger with respect to contact frame is required.

As shown in Fig. 3.29, each finger consists of 4 DoF and 7 joints, hence the four fingered gripper has 28 joints controlled by 16 actuators. Generally manipulation is done by gripper making higher pair of contacts with the object, where the contact points are also act as additional joints in the system. Let us consider a simple configuration shown in Fig. 3.29a, where a sphere made a point contact with friction with one of the modular finger. The frame $\{C\}$ is defined at the contact with axes $(\hat{n}, \hat{t}, \hat{o})$. Let $c \in \Re^3$ be the position of contact defined in a global frame $\{N\}$. While $L_1, .., L_{11}$ (3.29b) are screw parameters [49, 54, 30] that can describe the distance between joints and contact frame. Notice that, L_5, L_7 describe the distance from both the two joints; the ones responsible for finger-tip translation and rotation. Also note that, knowing θ_1, θ_5 and r, all L_1 to L_{11} can be estimated. Let p represents a vector describing a fixed frame of object, expressed in $\{N\}$. According to the illustration of hand Jacobian described in [75], the partial hand Jacobian $\widetilde{J}_i \in \Re^{6\times n}$ where n is the number of joints.

$$\widetilde{J}_i = \bar{R}_i^{\,T} Z_i \tag{3.10}$$

where $\bar{R}_i$ = Blockdiag(R_i, R_i)=$\begin{pmatrix} R_i & 0 \\ 0 & R_i \end{pmatrix} \in \Re^{6\times 6}$ and $R_i \in \Re^{3\times 3}$ represents the orientation of i-th contact frame $\{C\}_i$ wrt $\{N\}$. While $Z_i \in \Re^{6\times n}$ defined as

$$Z_i = \begin{pmatrix} d_{i,1} & \dots & d_{i,n} \\ l_{i,1} & \dots & l_{i,n} \\ & & \end{pmatrix} \tag{3.11}$$

where both $d_{i,j}$ and $l_{i,j} \in \Re^{3\times 1}$ such that they are $[0,0,0]^T$ while contact i does not affect joint j. Otherwise,

$$d_{i,j} = \begin{cases} \hat{z}_j & \text{if joint j is prismatic,} \\ S(c_i - \zeta_j)^T \hat{z}_j & \text{if joint j is revolute.} \end{cases}$$

$$l_{i,j} = \begin{cases} 0_{3\times 1} & \text{if joint j is prismatic,} \\ \hat{z}_j & \text{if joint j is revolute.} \end{cases}$$

where ζ_j is the origin of coordinate frame associated with j-th joint and $\hat{z}_j$ is the unit vector of z-axis of the same frame. While $S(r)$ is the skew symmetric matrix defined as

$$S(r) = \begin{pmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{pmatrix}$$

Using (3.11), estimated Z_i for the finger shown in Fig. 3.29b

$$Z_i = \begin{pmatrix} -L_4 & -L_{11} & -L_9 & -L_2 & -L_8 & 0 & 0 \\ -L_1 & L_7 & L_3 & L_{10} & L_6 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -L_7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 \end{pmatrix} \tag{3.12}$$

Hence the hand Jacobian of the four fingered gripper will $\widetilde{J} \in \Re^{24\times 28}$. Incase all dimensions and configuration of the fingers are same, ($\bar{R_1}^T Z_1 \in \Re^{6\times 7}$) the complete Jacobian,

$$\widetilde{J} = \begin{pmatrix} \bar{R_1}^T Z_1 & 0 & 0 & 0 \\ 0 & \bar{R_1}^T Z_1 & 0 & 0 \\ 0 & 0 & \bar{R_1}^T Z_1 & 0 \\ 0 & 0 & 0 & \bar{R_1}^T Z_1 \\ & & & \end{pmatrix} \tag{3.13}$$

$$t_{i,hand} = \widetilde{J}_i \dot{q} \tag{3.14}$$

where $t_{i,hand} \in \Re^6$ is the twist of the hand expressed in $\{C\}_i$ and $q = [q_1....q_n]^T$(here $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, r_7$) is vector of joint positions. Using complete Jacobian (Equ. 3.13) twist components wrt all contact frames constructed by all fingers can be estimated. Now for the one finger shown in Fig. 3.29, the partial hand Jacobian,

$$t_{i,hand} = \begin{pmatrix} a+b+c \\ d+e \\ -L_7\dot{\theta}_6 \\ -s_i\dot{\theta}_6 \\ -c_i\dot{\theta}_6 \\ \dot{\theta}_1+\dot{\theta}_2+\dot{\theta}_3+\dot{\theta}_4+\dot{\theta}_5 \\ \end{pmatrix} \tag{3.15}$$

where

$$a = \dot{\theta}_1(c_iL_4 - L_1s_i) - \dot{r}_7s_i + \dot{\theta}_3(c_iL_9 - L_3s_i)$$

$$b = \dot{\theta}_4(c_iL_2 - L_{10}s_i) + \dot{\theta}_5(c_iL_8 - L_6s_i)$$

$$c = \dot{\theta}_2(c_iL_{11} - L_7s_i)$$

$$d = -c_i\dot{r}_7 + \dot{\theta}_1(c_iL_1 - L_4s_i) - \dot{\theta}_3(c_iL_3 + L_9s_i)$$

$$e = -\dot{\theta}_4(c_iL_{10} + L_2s_i) - \dot{\theta}_5(c_iL_6 + L_8s_i) - \dot{\theta}_2(c_iL_7 + L_{11}s_i)$$

And s_i, c_i represent $sin\theta_i, cos\theta_i$ respectively. Notice that, the required twist stated in (Eq. 3.15) of finger-tip should not be independent from dimensions of target object since the purpose of synthesis is manipulation. Evolution of contact frames through a given trajectory may also required during a manipulation; hence, if the dimension of target object, desired initial and final postures are known, it is possible to estimate required contact movements of finger-tip at a given instance (fig. 3.29b). The choice made from Fig. 3.30, that satisfies the twist using (Eq. 3.15) should be used to synthesize finger.

3.9.4 Mechanism for Manipulation

For manipulation, additional 2 DoF mechanism for finger (1 rotation and 1 translation about y_f) is conceived inspired by well known ball screw mechanism similar to the previous solution. The detail of each finger module is illustrated in Fig. 3.27a-b. Pulley 1 (fig. 3.27b) is coupled with a lead screw, while pulley 2 is coupled with the nut by a special arrangement. A 3D printed part consists of linear guides coupled with pulley 2, and the screw is bypassed through pulley 2. The nut of the screw is mounted with a cylindrical part, that contains three linear bushings; so that the cylindrical part can slide along three linear guides when the pulley 1 (mounted with screw) is being rotated and pulley 2 is stationary. The external part of the cylindrical part is the finger tip. The arrangement of screw and pulley 2 is such that both are co-centric and both can rotate independently. Table 3.2 summarizes all the outcomes of finger tip movements.

3.9.5 Dexclar Prototype Fabrication

According to the chosen mechanisms, Dexclar platform has been built as shown in Fig. 3.29c, with four modular fingers, whose one finger detail is shown in Fig. 3.27c-d, and the CAD arrangement is illustrated in Fig. 3.28c. In this prototype, R is chosen 1.2, which has been derived from actuator dimensions and to meet 150 mm translation (along y_f) of finger tip.

3.9.6 Control Method

This section describes the control architecture of Dexclar. To implement grasping and manipulation, control law is essential. Dexclar consists of two different control laws for

- five bar actuation (2 DoF)
- ball-screw mechanism (2 DoF)

However, due to the proper dimensional synthesis, the five bar linkage of the fingers is very easy to control. For angular and parallel grasp, the two actuated joints of five bar are not necessarily required to move simultaneously. Hence the system become robust for grasping different shaped payload. And a simple PD control is exploited. Also to verify the feasibility at some point, manual control was also introduced, the detail has been illustrated in Appendix A.

However for the finger tip rotation and spiral motion, because both these cases (finger tip rotation and spiral), the pulley 1 and pulley 2 (fig. 3.28b) are required to be rotated very accurately. Since the motion of the system depends on mechanical elements like nut, screw, pulleys and the condition of pure rotation in the current design requires equal speeds of pulley 1 and pulley 2 (fig. 3.28b), difficulties arise to get pure rotation in the finger tip. This is due to the fact that, any difference in speed between the pulleys results a translation along y_f in the finger tip. Hence, a nonlinear technique is needed to nullify translation effect. Moreover, it is often necessary to rotate four finger tips at a same speed. For such cases, a method must be developed. (one such case: a sphere needs to be rotated around vertical axis as shown in fig. 3.33g-h.)

3.9.7 Use of Perturbation observer

Perturbation observer can be very useful to estimate the disturbance of a system. Since it doesn't require an additional sensor, the Dexclar platform will remain robust in case the algorithm can be implemented.

The translation effect can be treated as disturbance and the measured can be used to compensate the effect by using *perturbation observer*. Another advantage of using *perturbation observer* is, it does not need additional sensor except motor feedback [52]. For Dexclar, to get pure rotation, one motor of ball screw mechanism (Finger motor 2, fig. 3.27a) is selected as a driver and the rest one (Finger motor 1, fig. 3.27 a) as follower. The actual state of the driver is given to the follower as a desired state. While the follower is operated with the sliding mode control with sliding perturbation observer (SMCSPO) [52].

3.9.8 Definition of Perturbation & control law

In the governing equation of SMCSPO, the perturbation term is defined as a combination of all uncertainties and external disturbances.

$$\psi(X_1,...,X_m,t) = \Delta f(X_1,...,X_m) + \Delta B(X_1,...,X_m)u + d(t) \tag{3.16}$$

Where, $\mathbf{X}_i \equiv [\mathbf{x}_i, \dot{\mathbf{x}}_i, ..., \mathbf{x}_i^{n_i-1}]^T \in \Re^{n_i}$, $i = 1,..,m$: the state sub vector, which forms the global sate vector $[\mathbf{X}_i^T, ...\mathbf{X}_m^T] \in \Re^r$, $r = \sum_{i=1}^m n_i$, $i = 1,...,m$: independent coordinate. $f = [f_1,...f_m]^T \in \Re^m$ and $\Delta f = [\Delta f_1,...\Delta f_m]^T \in \Re^m$: vector fields corresponding to the uncertainties of nonlinear driving terms and their perturbations, respectively.

$B = [b_{ij}] \in \Re^{m\times m}$ and $\Delta B = [\Delta b_{ij}] \in \Re^{m\times m}, i, j = 1,...,m$: matrices representing control gains and their uncertainties. And,
$d = [d_1,...d_m]^T \in \Re^m$: disturbance vector of the system
$u = [u_1,...u_m]^T \in \Re^m$: Control vector
$x^{(n)} = [x_1^{n_1},...x_m^{n_m}]^T \in \{\Re^m$,
${x_i}^{n_i} \in \Re$ with ${x_i}^k = d^k x_i/dt^k, \dot{x}_i = dx_i/dt$ However in the control cycle, estimated perturbation is calculated by the following equation,

$$\hat{\psi}_j = \alpha_{3j}(-\hat{x}_{3j} + \alpha_{3j}\hat{x}_{2j}) \tag{3.17}$$

$\hat{x}_{2j}, \hat{x}_{3j}$ can be found by integrating observer equations, detail derivations can be found in [52], due to space, only the methodology is stated in this paper.

Where $\hat{x}_{2j}, \hat{x}_{3j}$ can be found by integrating observer equations as follow,

$$\hat{x}_{1j} = \int (\hat{x}_{2j} - k_{1j} sat(\tilde{x}_{1j}) - \alpha_{1j} \tilde{x}_{1j}) dt \tag{3.18}$$

$$\hat{x}_{2j} = \int (\alpha_{3j} \bar{u}_j - k_{2j} sat(\tilde{x}_{1j}) - \alpha_{2j} \tilde{x}_{1j} + \hat{\psi}_j) dt \tag{3.19}$$

$$\hat{x}_{3j} = \int (\alpha_{3j}^2 (-\hat{x}_{3j}) + \alpha_{3j} \tilde{x}_{2j} + \bar{u}_j) dt \tag{3.20}$$

Here, gain α_{3j} selected such that, the variable ψ_j/α_{3j} has a negligible value. $\bar{u}_j$ is the control input and the gains $k_{1j}, k_{2j}, \alpha_{1j}, \alpha_{2j} > 0$. $sat(\tilde{x}_{1j})$ is the saturation function defined as

$$sat(\tilde{x}_{1j}) = \left\{ \begin{array}{ll} \tilde{x}_{1j}/|\tilde{x}_{1j}|, & \text{for } |\tilde{x}_{1j}| \geq \varepsilon_{oj} \\ \tilde{x}_{1j}/\varepsilon_{oj}, & \text{for } |\tilde{x}_{1j}| \leq \varepsilon_{oj} \end{array} \right\}$$

The control law given as follow

$$u_j = \frac{1}{\alpha_{3j}} \{ -K_j sat(\hat{s}_j) + (\frac{k_{2j}}{\varepsilon_{oj}} + c_{1j} \frac{k_{1j}}{\varepsilon_{oj}} - (\frac{k_{1j}}{\varepsilon_{oj}})^2) \tilde{x}_{1j}$$

$$+ \ddot{x}_{1dj} - c_{j1}(\hat{x}_{2j} - \dot{x}_{1dj}) - \hat{\psi}_j \} \tag{3.21}$$

Here, $\hat{s}_j = \dot{\hat{e}}_j + c_{j1} \hat{e}_j$ while estimated error, $\hat{e}_j = \hat{x}_{1j} - x_{1dj}$.
"~"& "^"symbolize the estimation error and estimation quantity respectively. x_{1dj} is the desired state in Equ. 3.16 for the follower, and this is state is given as the actual state of the driver.

To get the values of $K_j, k_{1j}, k_{2j}, c_{1j}, \alpha_{3j}$ poles are needed to have real value in the characteristic equation (43) mentioned in [52].

Table 3.3 Parameter values

Parameter	follower1	follower2	follower3	follower4
ε_{oj}	0.1	0.11	0.1	0.1
K_j	35.6	35.5	33	34
k_{1j}	1200	1200	1200	1200
k_{2j}	0.5	0.5	0.5	0.5
α_{3j}	10.3	10.8	10.8	10.8
c_{1j}	350	350	355	355

Table 3.4 Electronic components in Dexclar Platform

Force Sensor: Loadcell	Burster 8413, range 0-50N, accuracy 0.5%, in line amplifier Typ 9235
Controller	FPGA : Altera, ARM: Sam4s Kit
5 bar actuators	Lynxmotion 12V 10rpm, Torque 5.4Nm
finger tip actuators	Pololu 6V, 1010rpm, Torque 0.5 Nm

3.9.9 Dexclar: Grasp, In-hand Manipulation & Release

As explained in the synthesis process, it is understandable that, the platform should capable to grab several geometric payloads with different grasping techniques by actuating eight DoF (of 4×2 DoF of each five bar, Fig. 3.23c illustrated possible configurations to address grasping of an arbitrary payload) and also the 4 DoF translations of fingertips. Figure 3.32a-g described some grasping examples, thanks to the properties of reconfigurability and DoF, Dexclar can grasp and manipulate small scaled payloads, such as coin, screw (fig. 3.32g-l).

Dexclar is dexterous among three axes for a spherical payload, twist can be obtained in all three planes. Twist is possible even without altering the center position of the spherical payload. Such two examples have been illustrated in Fig. 3.33a-b and g-h. The 6 DoF motions of spherical payload are:

- Rotation 1: equal and opposite motion of *q*-*r* (3.33b)
- Rotation 2: equal and opposite motion of *w*-*s*
- Rotation 3: equal and same directed rotation of two pairs *q*-*r* and *w*-*s* (3.33h)
- Translation 1: equal and same directed translation of two pairs *q*-*r* and *w*-*s*
- Translation 2: actuating θ_1 (of fivebar @θ_5=180) both *q* and *r* with a stable grasp (3.33i-j)
- Translation 3: actuating θ_1 (of fivebar @θ_5=180) both *w* and *s* with a stable grasp

By using the property of reconfigurability, the release issue can be addressed. Considering the posture of sphere shown in Fig. 3.33b is the final desired posture after the rotation, the *q*-*r* can break the contacts and reconfigure themselves as shown in 3.33d. At that point, the sphere (3.33d) can be translated upward by *w*-*s* pair alone for release, while keeping the final posture fixed. For arbitrary payloads, the *q*-*s* may join in the process by re-grasping at proper contact points, if required. Another example of using the reconfigurable property is explained in Fig. 3.33s-z.

Besides, spiral motion of Dexclar-fingertip for a spherical payload can reduce significant manipulation time. Let us assume, a point over a sphere P_i (fig. 3.33k) supposed to be placed at point P_f. The fastest way to change It's posture is to rotate the sphere about the resultant vector (crossproduct of $\vec{oP_i}$ and $\vec{oP_f}$, where o is the center of sphere) with a specific angle. This maneuver is also possible in Dexclar, thanks to the spiral motion of ball screw mechanism.

Few other examples have been illustrated in Fig. 3.33o-r where an egg manipulation is addressed by spiral motion. Let us consider Fig. 3.33m and 3.33n are the given and final postures of an egg. At Fig. 3.33p, ccw upward spiral of w and cw upward spiral of s cause a twist of the egg, where finger r is pushing the egg to keep it grasped. At the state shown in Fig. 3.33q the bidirectional translations of q and r will cause the remain portion of the twist of egg to meet the final posture.

The experimental video [24] can be useful to understand the kinematics of the system.

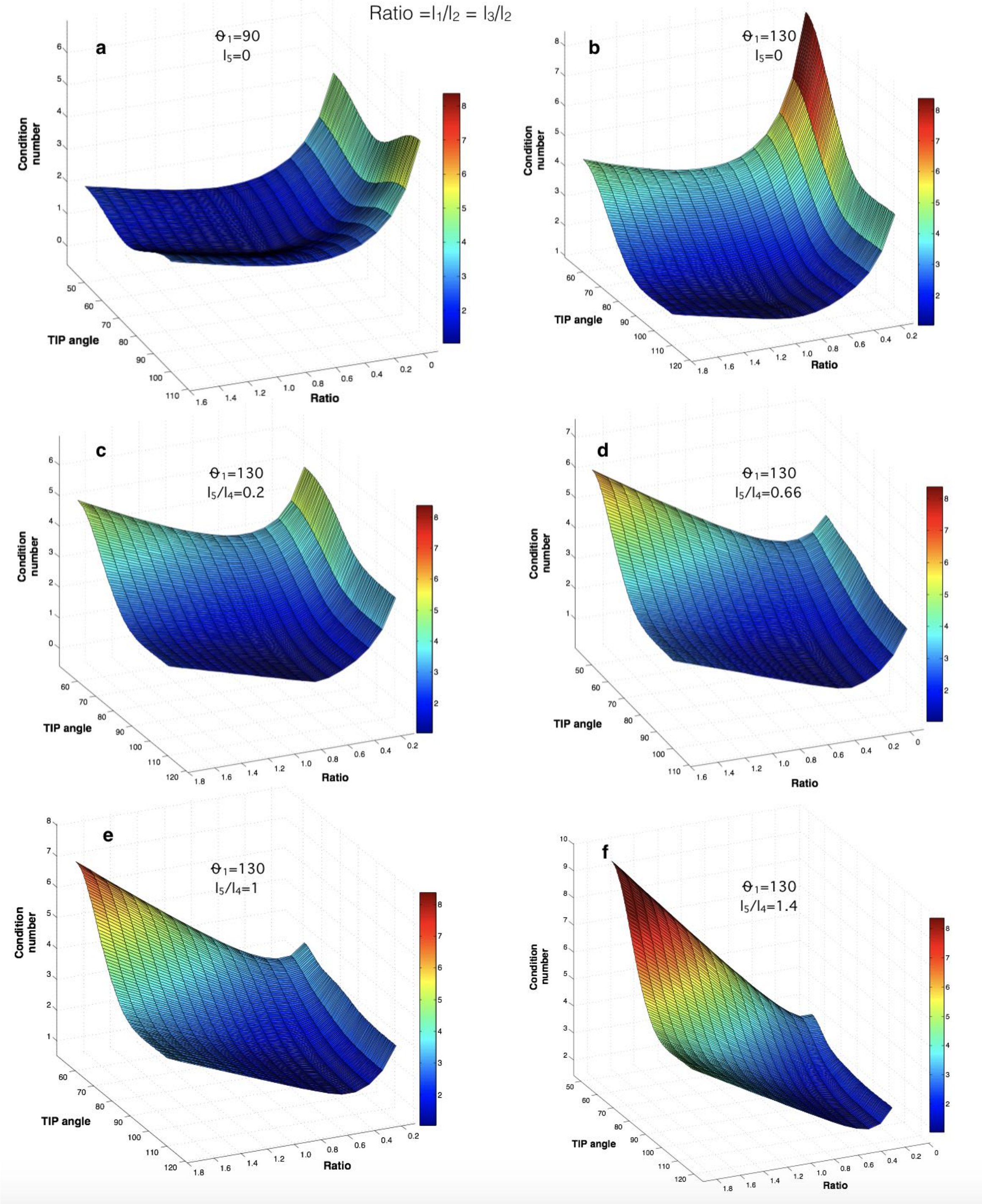


Fig. 3.30 (a) to (f) show mobility/manipulability at various link ratios and conditions for maintaining specific finger-tip angular profile

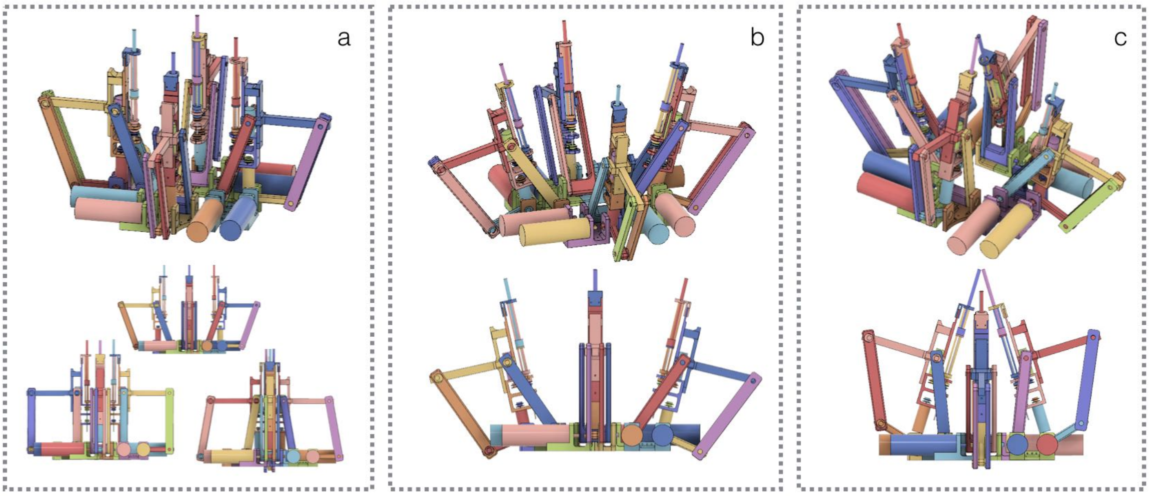


Fig. 3.31 a) workspace of Dexclar, it can grab large (b) and tiny (c) object by exploiting five bar linkages [62, 63]

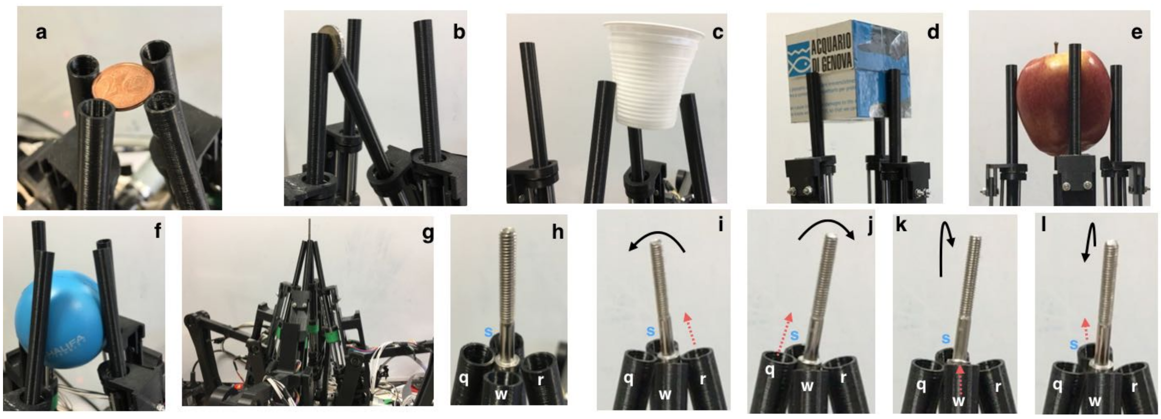


Fig. 3.32 a) pinching grasp of a coin, b) a coin is grasped by 2 fingertips, c) grasping of a cup by 3 fingers. 4 fingered grasping of: d) rectangular solid, e) apple. f) envelope grasping of a sphere. g-h) configuration of platform for grasping a M4 screw, & twisting by: i) translation of r, j) translation of q, k) translation of w, l) translation of s fingertip

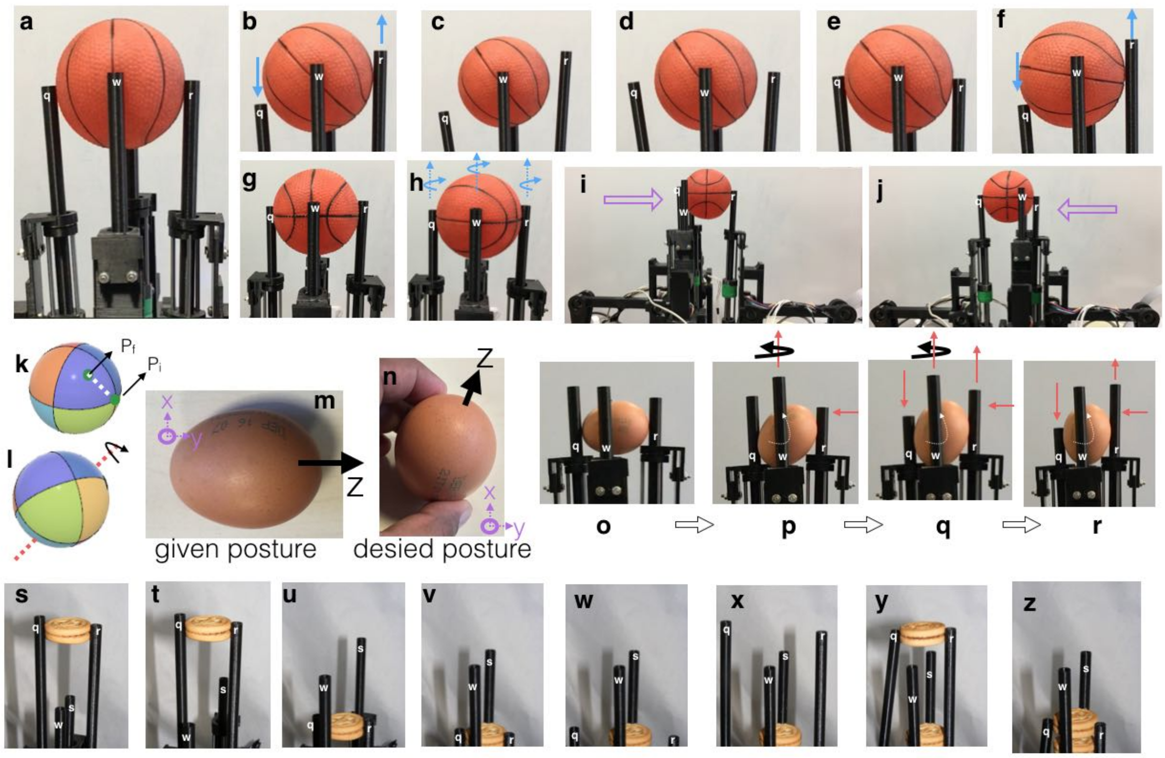


Fig. 3.33 a) a sphere is being grasped, b) bi-directional & equal translations of *q*-*r* pair will cause rotation, c) *q*-*r* pair breaks the contact, d) *q*-*r* pair reconfigures and e) re-grasps, f) re bidirectional translation to rotate the sphere further. g)-h) *q*-*r* and *w*-*s* fingertips rotate CW to create a CCW twist about vertical axis. i) *q*-*r* pair moves rightward with the grasped sphere, j) *q*-*r* pair moves leftward with the grasped sphere. k) A sphere to be rotated from a point to point and it's l) desired posture, m)-n) given and final posture of egg and o)-r) manipulation to achieve the desired posture in Dexclar platform, s) *q*-*r* pair grasps one biscuit, t) *w*-*s* pair opens, u) *q*-*r* pair moves downward with the biscuit, v) *w*-*s* pair grasps the biscuit, w) *q*-*r* pair breaks contact with biscuit, x) *q*-*r* pair moves up, y) *q*-*r* pair grasps second biscuit, z) *q*-*r* pair moves downward with the second biscuit and place it over the previous biscuit

Chapter 4 develops a new gripper named VARO-fi. VARO-fi has been derived from the central 4 DoF finger, also can be stated as an optimal version of Dexclar. The 9 DoF VARO-fi can be used as end-effector and capable of doing grasp, in-hand manipulation and release payload with correct posture.

Chapter 4

End-effector Design

In this Chapter, a highly dexterous but simply constructed reconfigurable platform named VARO-fi (VARiable Orientable fingers with translation) is developed using the concepts formulated in the last Chapter 3. VARO-fi can be used as an industrial end-effector, as well as an alternative of bio-inspired gripper in many robotic applications. The robust four fingered VARO-fi addresses grasp, in-hand manipulation and release (payload with desired configuration) of plurality of payloads, as demonstrated in the following sections.

4.1 Introduction

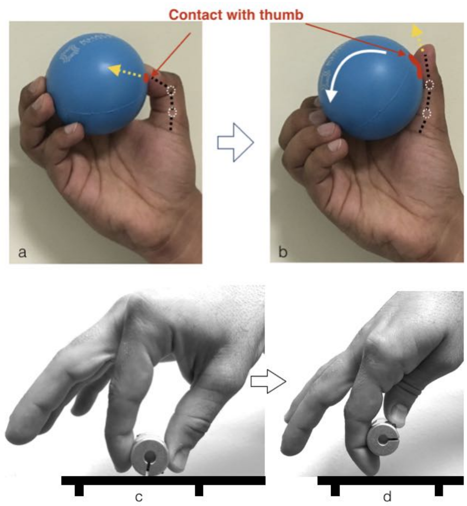


Fig. 4.1 Typical in-hand skills: a)-b) twisting a sphere requires both evolving contact points and direction of thumb, c)-d) twisting a joint by thumb and index finger, difficulties arise at the last state (d) to place it keeping with the final configuration

A gripper or artificial hand is required in many robotic systems to carry out prehensile tasks such as pick-place, manipulation and so on. Figure. 4.1 illustrates few examples of within-hand dexterity of our hands, and from kinematic point of view contact stability [29], grasp synthesis algorithms [67], pre-orientation of fingers such as synergy [26, 11] have been explicitly studied by many researchers. Grasping [34] and manipulation planning [69] in general are also required in many automated systems including social, industrial, space and medical robotics. However, grippers are generally not developed to address payload-centric manipulation, as the capabilities illustrated in Fig. 4.1. And dexterous manipulation (fig. 4.1) is not the primary focus of anthropomorphic and bio-inspired robotic hands, rather they exploit different goals such as adaptability with a plurality of objects, or to mimic a biological maneuver [47]. Much research has been conducted on human like hands [23] on dexterity features. As mentioned, although the capabilities are remarkable because of the use of flexible materials, numerous sensors and higher number of DoF, by those designs [47, 23], however, precision and speed are hardly achievable from them. Hence, in general, industrial grippers exploit simpler mechanisms with least number of fingers and tend to avoid soft materials in the construction primarily to achieve dexterity, reliability, repeatability and speed.

Non-prehensile manipulation [66] and dynamic manipulation [55] are interesting solutions in many robotic applications. However, since they [66, 55] generally exceed the kinematic workspace of the gripper and also if the final payload posture is required to be specific (fig. 4.1d) at release, those techniques are difficult to implement in an industrial gripper or artificial hand. Moreover, theoretical analysis, such as contact kinematics [75], grasp stability [51] cannot predict the nonholonomic behaviors, and therefore uncertainties are always present to restrict a maneuver, even though the gripper is kinematically feasible for performing the task.

The dilemma is, in-hand dexterity of objects such as twisting require certain degrees of flexibility in the gripper design. This is difficult to obtain from a rigid structure and on the other hand, the use of non-rigid materials in the gripper construction reduces speed, accuracy and performance as mentioned. Several mechanism-centric designs such as [79, 27, 80] have been proposed, however the dexterity of these grippers are constrained to fixed planes [79], or the twisting range of the manipulated payload [27] is limited. Also in the designs presented in [79, 27, 80] do not consider the payload release problem (fig. 4.1e). In this paper, a novel gripper named VARO-fi (VARiable Orientable fingers with translation) is proposed (fig. 4.5), which addresses the grasp, manipulation and release problems by a simple and reconfigurable platform. The formulation of VARO-fi has been illustrated explicitly in the following sections. The analysis can be used to develop target specific gripper after evaluating payload centric

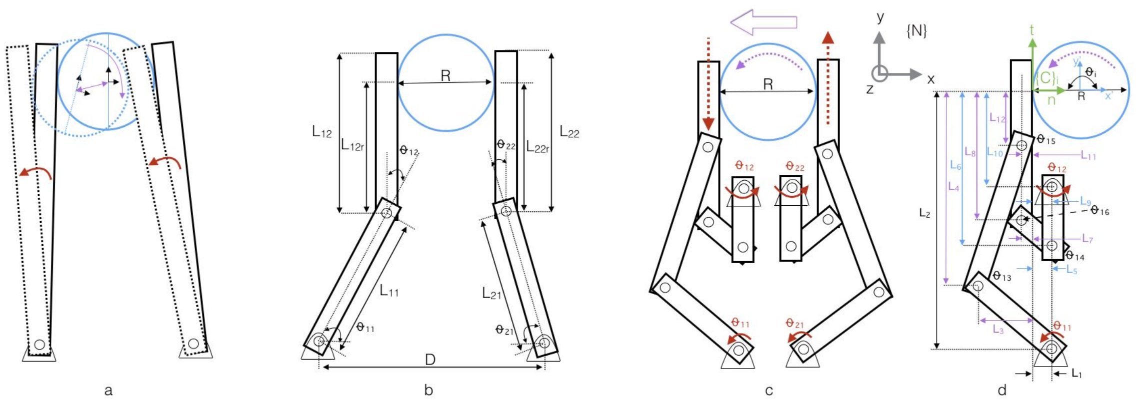


Fig. 4.2 In-hand posture change trials of a spherical payload: a) the payload experiences CW rotation alongside with translation effect due to the CCW motions of the two fingers (each finger consists of 1 DoF) b) two 2 DoF planar fingers and the payload c) two 2 DoF fingers consist of a chosen kinematic chain; if the actuated joints have been rotated by controlled CCW motions, the payload also experiences a transformation (CCW rotation and translation) d) screw parameters of the chosen kinematic chain

requirements as well. Unlike many complicated mechanisms, the four fingered VARO-fi features 9 degrees of freedom with mostly prismatic joints (also can be minimized up to 5 DoF according to target application). VARO-fi is capable of providing 6 DoF motions to a spherical payload. In order to underline the strength and usability of VARO-fi, some comparisons have been made with respect to some existing grippers.

4.2 Concept, Mechanism & Development of Proposed Gripper

In this section, the problem of posture change in a gripper has been illustrated mathematically. The required number of fingers, DoF, types of motion and contact between finger and a generic convex payload (spherical) for manipulation in the proposed gripper have been demonstrated. In theory [75], grasp can be addressed by the following

- force closure approach
- form closure approach

If the payload or object is constrained in form closure grasp shown in fig. 4.3a, it is difficult to apply force or moment in order to change the posture; on the other hand, force closure grasp (fig. 4.3b) allows to manipulate payload in-hand. However, posture change of a

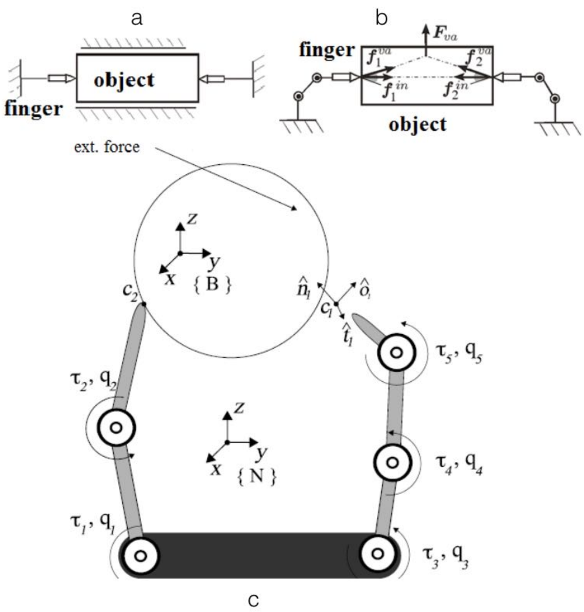


Fig. 4.3 a) form closure grasp b) force closure grasp c) fingertip manipulation

spherical payload by a two multi DoF fingered gripper (as shown in fig. 4.3c) encounters the following constraints

- short payload rotation range
- not dexterous along each plane
- grasp and force analysis required
- joint motion planning and robust control required

Manipulation can also be offered by higher pair of contact (such as line, point and curve) between payload and fingers in an envelope grasping as illustrated in Fig. 4.2; each finger of the gripper is cylindrical in 3D. However, the efforts shown in 4.2a,b also consist of the above mentioned constraints. The problems of joint motion planning and required analysis of grasping force can be optimized by using proper synthesized kinematic chain in the gripper, whose pros and cons have been discussed in the next part.

4.2.1 Posture change in-hand by kinematic chain

A two-fingered (each finger consists of 2 DoF) gripper and a spherical payload have considered for in-hand manipulation, as shown in Fig. 4.2c. Each of the finger is consist of a chosen kinematic chain having 6 joints. Among them θ_{11},θ_{12},θ_{21},θ_{22} are the actuator driven. Notice

that, CCW rotation of the actuated joints (θ_{11},θ_{12},θ_{21},θ_{22}) will shift the fingers towards left with a bidirectional translation-effect at finger-tips (shown as red dotted line in 4.2c).
Lets assume, a proper synthesis is carried for the kinematic chain (4.2c) such as the bidirectional translations can be used to obtain rotation (CCW, fig 4.2c) and translation (along leftwards, fig 4.2c) of a spherical payload. Hence, a desired posture change (of payload) should have addressed by simple motion planning of the driven joints. And so, it can be stated that, a proper synthesized kinematic chain able to

- create a rotation to the payload
- create a translation to the payload (alongside) with the fingers
- maintain a secure grasp by keeping a constant distance (diameter of sphere in this example)

However, to obtain this posture change in-hand, one finger requires to push the payload and the other desired to move away synchronously (fig. 4.2c) at a given instance. Hence, the contact point between payload and finger required to evolve or break completely at a given time during manipulation. For this, a measure of twist of the finger with respect to contact frame is required to develop. (The following approach is also adopted in previous works [62, 65])

Referring to Fig. 4.2c each of the finger is consist of 2 DoF and 6 revolute joints. Considering a specific configuration shown in Fig. 4.2d, where a spherical payload is having a contact (point contact with friction) with one finger. The frame $\{C\}$ is defined at the contact with axes $(\hat{n},\hat{t},\hat{o})$. Let $c \in \Re^3$ be the position of contact defined in a global frame $\{N\}$. While the $L_1,..,L_{12}$ (4.2d) are screw parameters that can describe the distance between joints and contact frame.

Notice that, L_1, L_5 and L_9 are equal in length in the current configuration (4.2d), however they are chosen as independent parameters to represent a generic form. (Same goes for L_7, L_{11}) Also note that, knowing θ_{11}, θ_{12}, the screw parameters (L_1 to L_{12}) can be estimated for a given kinematic chain. According to the illustration of hand Jacobian described in [75], the partial hand Jacobian $\widetilde{J}_i \in \Re^{6\times n}$ defined as (n=6, is the number of joints)

$$\widetilde{J}_i = \bar{R}_i^{\,T} Z_i \tag{4.1}$$

where $\bar{R}_i = \text{Blockdiag}(R_i, R_i) = \begin{pmatrix} R_i & 0 \\ 0 & R_i \end{pmatrix} \in \Re^{6\times 6}$ and $R_i \in \Re^{3\times 3}$ represents the orientation of i-th contact frame $\{C\}_i$ wrt $\{N\}$. While $Z_i \in \Re^{6\times n}$ defined as

$$Z_i = \begin{pmatrix} d_{i,1} & \dots & d_{i,n} \\ l_{i,1} & \dots & l_{i,n} \end{pmatrix} \tag{4.2}$$

where both $d_{i,j}$ and $l_{i,j} \in \Re^{3\times 1}$ such that they are $[0,0,0]^T$ while contact i does not affect joint j. Otherwise,

$$d_{i,j} = \begin{cases} \hat{z}_j & \text{if joint j is prismatic,} \\ S(c_i - \zeta_j)^T \hat{z}_j & \text{if joint j is revolute.} \end{cases}$$

$$l_{i,j} = \begin{cases} 0_{3\times 1} & \text{if joint j is prismatic,} \\ \hat{z}_j & \text{if joint j is revolute.} \end{cases}$$

where ζ_j is the origin of coordinate frame associated with j-th joint and $\hat{z}_j$ is the unit vector of z-axis of the same frame. While $S(r)$ is the skew symmetric matrix defined as

$$S(r) = \begin{pmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{pmatrix}$$

Using (2), estimated Z_i for the finger shown in Fig. 4.2d

$$Z_i = \begin{pmatrix} -L_2 & -L_4 & -L_{12} & -L_{10} & -L_6 & -L_8 \\ -L_1 & L_3 & L_{11} & -L_9 & -L_5 & L_7 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{4.3}$$

And the hand Jacobian of the two fingered planar gripper is $\widetilde{J} \in \Re^{12\times 12}$. In case all dimensions and the configuration of the fingers are same ($\bar{R_1}^T Z_1 \in \Re^{6\times 6}$), the complete Jacobian,

$$\widetilde{J} = \begin{pmatrix} \bar{R_1}^T Z_1 & 0 \\ 0 & \bar{R_2}^T Z_2 \end{pmatrix} \tag{4.4}$$

Now,

$$t_{i,hand} = \widetilde{J}_i \dot{q} \tag{4.5}$$

where $t_{i,hand} \in \Re^6$ is the twist of the hand expressed in $\{C\}_i$ and $q = [q_1....q_n]^T$(here $\theta_{11}, \theta_{12}, \theta_{13}, \theta_{14}, \theta_{15}, \theta_{16}$) is the joint position vector. Using complete Jacobian (4) twist components w.r.t the contact frames constructed by two fingers can be estimated. Now, considering the finger shown in Fig. 4.2d, the partial hand Jacobian,

$$t_{i,hand} = \begin{pmatrix} A+B+C \\ D+E+F \\ 0 \\ 0 \\ 0 \\ \dot{\theta}_{11}+\dot{\theta}_{12}+\dot{\theta}_{13}+\dot{\theta}_{14}+\dot{\theta}_{15}+\dot{\theta}_{16} \end{pmatrix} \tag{4.6}$$

where

$$A = \dot{\theta}_{11}(L_2c_i + L_1s_i) - \dot{\theta}_{13}(L_4c_i - L_3s_i)$$

$$B = \dot{\theta}_{14}(L_6c_i + L_5s_i) - \dot{\theta}_{16}(L_8c_i - L_7s_i)$$

$$C = -\dot{\theta}_{12}(L_{10}c_i + L_9s_i) - \dot{\theta}_{15}(L_{12}c_i - L_{11}s_i)$$

$$D = -\dot{\theta}_{11}(L_1c_i - L_2s_i) + \dot{\theta}_{13}(L_3c_i + L_4s_i)$$

$$E = -\dot{\theta}_{14}(L_5c_i - L_6s_i) + \dot{\theta}_{16}(L_7c_i + L_8s_i)$$

$$F = -\dot{\theta}_{12}(L_9c_i - L_{10}s_i) + \dot{\theta}_{15}(L_{11}c_i + L_{12}s_i)$$

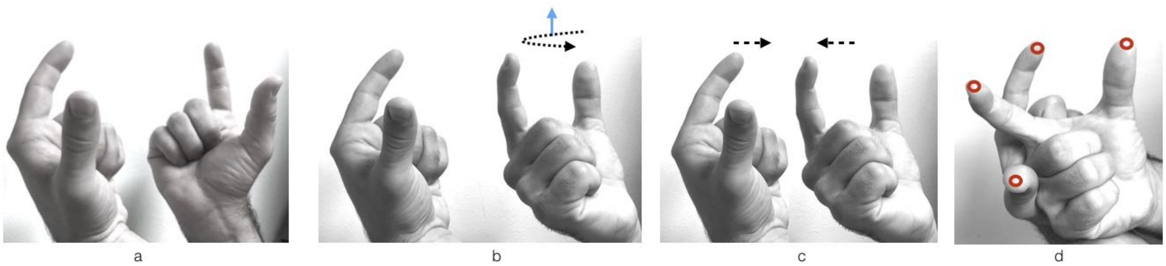


Fig. 4.4 a) Thumbs and index fingers of left and right hand b) the right hand rotated 90 degree ccw vertically, c) two hands are approaching each other with the their configurations and d) the fusion of 4 fingers shown here (consists of 2 pairs of fingers, those act perpendicular to each other

And s_i, c_i represent $sin\theta_i, cos\theta_i$ respectively. It is notable that, the twist stated in (6) of finger is dependent on the dimensions of target object and for a specific manipulation. Hence the example which has been shown in 4.2c requires

- continuous evolution of contact frames between payload and finger
- mobility analysis in order avoid singularity

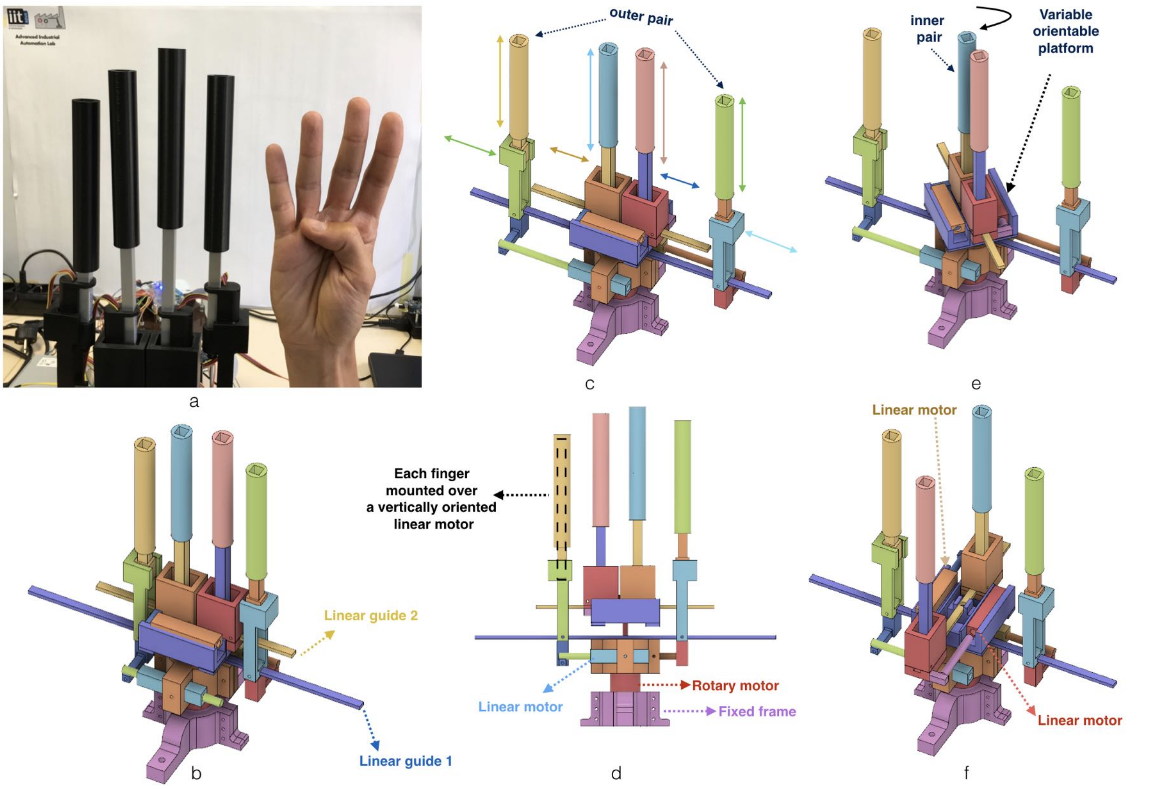


Fig. 4.5 a) Prototype of VARO-fi on human hand appearance b) isometric view c) the eight translations of VARO-fi d) right hand side view e) the variable platform is able to rotate while outer pair moves away f) industrial gripper appearance

- suitable contact set to maintain stable grasp

To such aim, dimensional synthesis of the kinematic chain of the finger (to meet the desired manipulation avoiding singularity conditions, also to obtain smooth motion planning of the actuated joints), (6) can be used. However, although, it is possible to develop gripper in order to fulfill desired manipulation using (6), the solutions will always encounter some limitations, such as

- applicable to specific payloads only
- rotation range is bounded
- gripper end at a new geometric configuration after the posture change (of payload, fig 4.2c). That means, the payload cannot be released with this final posture at the place where it was initially picked from. (limitation of two contact points)

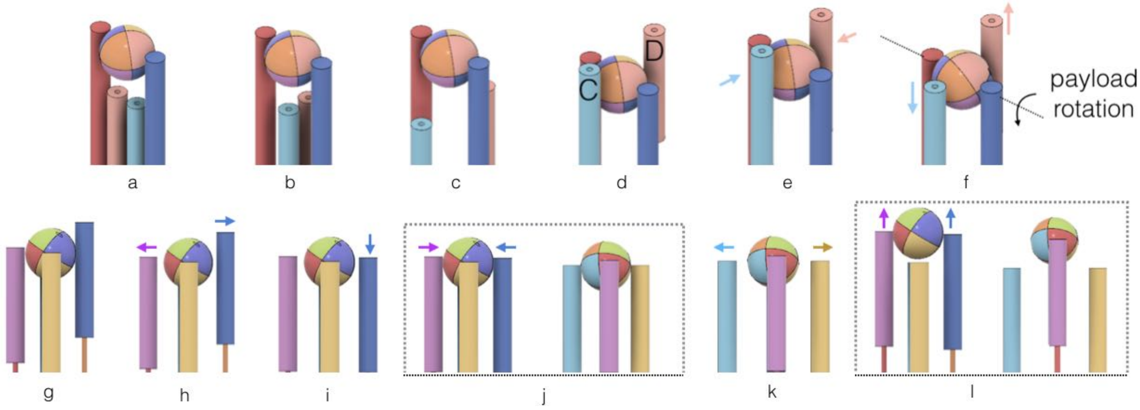


Fig. 4.6 In-hand payload manipulation (top) : a) outer pair maintains a sphere grasped, b) the variable platform rotates 90 degree, c) inner pairs opens up & d) moves up, e) inner pair makes contact with the sphere, f) bi-directional translations of inner pair fingers: create a rotation onto payload about an axis passing through outer pair. (bottom) reconfigurability property for release (or further rotation): g)-i) after a certain degree of rotation of payload one pair can break the contacts and reconfigure themselves either in order to repeat the process (d-f) or j) re-grasp & place the sphere with the desired configuration (k-l)

4.2.2 Requirement of additional contact(s)/finger(s) for release with a desired configuration

In order to release payload (4.1c) with the final configuration at a desired place, additional contact(s) is required. This requirement can be addressed by

- adding one contact point / additional support
- adding more than one contact point / additional support

Again, the idea of increasing one contact point could be conducted in two ways: three fingers might involve in the whole manipulation process (fig. 4.7c) or the additional one may only interfere, once the final posture has been already obtained by exploiting two fingers (fig. 4.2a). However in both cases, there are few constraints. Manipulation carried out by three fingers as a whole is required to be very well motion-planned (fig. 4.7c) and some configurations of payload may not be obtainable in the gripper's kinematic workspace (in order to maintain the grasp stability [29, 51] of the payload throughout the manipulation or at the final state or due to the finger orientation in the gripper platform).

And in the second case, the third finger could be introduced after obtaining the final configuration (of the payload), however, this case is dependent on the reconfigurability of the individual finger chain (such as fig. 4.2d) as well as on the grasp stability; this is mainly because, in order to release the payload at a desired point different from the current state, the third finger is required to create contact with the payload in a configuration, such that, one of the fingers (of previous two, those were involved in the posture change process & currently the payload is being grasped by them) able to break the contact with the payload. Which essentially means, the third finger and one of the previous finger should capable to maintain both the payload grasped and final posture. Since the payload at that point is being grasped by third and one of the previous finger, the free finger should able to reconfigure itself and re-contact with the payload again (with a convenient configuration so that, any of the current two could able to reconfigure) if it is required.

Also, due to different payload geometries, orientations, and singularity constraint of fingers, certain posture changes might not be obtained by the three fingered system.

As explained above, release a payload (with final posture and place at a desired point) by 3 contact points is step oriented, time consuming and not devoid of limitations. Hence, probably the best solution is to exploit more than one additional fingers (in a two fingered system).

4.2.3 VARO-fi formulation

The concept of VARO-fi is conceived from human hand. However, the finger movements and method of manipulation of this proposed hand are very different from our hand. As it is illustrated in fig. 4.1, in-hand manipulation by human hand is primarily governed by thumb and index finger. Hence, the construction of VARO-fi is inspired from this principle and 2 pairs of thumb-index have chosen. The proposed four fingered gripper has been illustrated in fig. 4.5 while the influence of this development has been portrayed in fig. 4.4. It is worth mentioning that, the principle of manipulation is also investigated through simulation in author's previous work [64].

To maintain the desired payload posture fixed & release with this final configuration (of payload), each pair of VARO-fi is capable to break the contacts (with the payload) and reconfigure themselves and

- re-grasp the payload to place at a given point
- re-manipulate the payload to increase the range of rotation for example

while the other pair maintains the payload grasped until release. This phenomenon has been explained in fig. 4.6 later section.

Table 4.1 A comparison between VARO-fi and existing equivalent grippers

	Gripper of [79] (fig. 4.7a)	**Gripper of [27]** (fig. 4.7b)	**Gripper of [80]** (fig. 4.7c)	**Dexclar [61]** (fig. 4.7d)	**VARO-fi** (fig. 4.5)
DoF of gripper/system	9*	10*	15*	16	9 (down to 5)
Number of joints	12*	10*	15*	28	9 (down to 5)
Number of fingers	3	4	3	4	4
Presence of kinematic chain or serial linkage	yes	yes	yes	yes	no
Control difficulties of manipulation	easy	difficult	very difficult	difficult	easy
Dimension Synthesis	required	required	required	required	not mandatory
Grasp Synthesis	required	not required	required	required	not required
Singularity analysis	required	required	required	required	not mandatory
Type of contact	higher pair	higher pair	higher pair	higher pair	higher pair
Manipulation method	form closure	fingertip only	fingertip only	both fingertip & form closure	both fingertip (fig. 4.9i-3) & form closure
Manipulation speed depends on	actuators only	links & actuators	links & actuators	actuators only	actuators only
Manipulation reliability	good	not claimed	not claimed	good	good
Reconfigurable	no	yes	not demonstrated	yes	yes
Payload release with final configuration	not addressed	not addressed	not addressed	addressed	addressed
Twisting range	bounded	bounded	bounded	not bounded	not bounded
DoF motion spherical payload	4	6	6 but 3 fingers required	6	6
Robustness & scalability	not good	medium	average	average	very good
Change of Appearance	no	possible	no	no	possible
* : the number is derived from the figures shown in the paper					

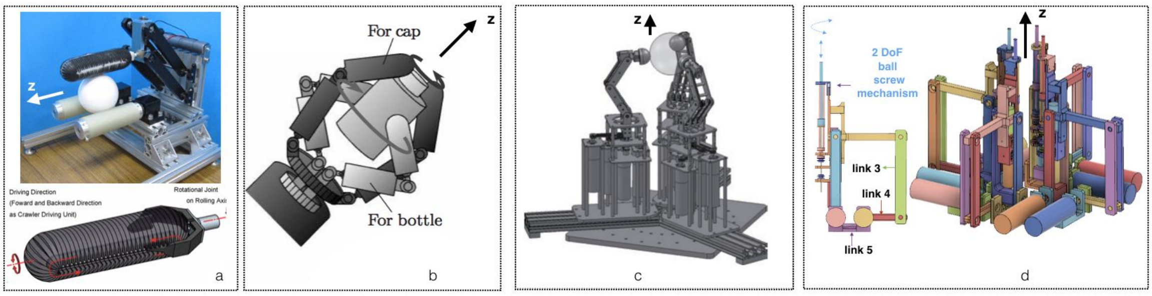


Fig. 4.7 Existing grippers [not presented according to scale] that addressed manipulation and VARO-fi a) gripper proposed in [79] b) gripper proposed in [27] c) gripper proposed in [80] d) Dexclar proposed in [61]

4.2.4 Mechanical description of VARO-fi

The prototype of VARO-fi (shown in fig. 4.5a) consists of 9 DoF, and 8 of them are driven by linear actuators, only the variable platform (shown in fig. 4.5e) is rotary actuator (fig. 4.5d) driven. The variable platform consists of one pair, whose fingers move over *linear guide* 2 (4.5b); while the remaining pair moves along *linear guide* 1. The *linear guide* 1 and rotary motor are mounted with the fixed frame. Each of the 4 fingers is mounted over linear actuator, hence, the finger can move up and down. Once the *outer pair* (yellow and green finger shown in 4.5c) clears the overlap with *linear guide* 2, the variable platform is then capable to rotate freely. By this, VARO-fi

- achieves different appearances
- is able to constrain a payload by the *outer pair* (4.6a), while the variable platform could be exploited to obtain suitable set of contact points, in case the payload is not properly in a stable grasp at pick point, or the outer pair intends to break the contact in the next step
- works as two hands in a single gripper

Figure 4.6 describes manipulation of a sphere and the method of release. As claimed, VARO-fi is capable to provide 6 DoF motions to a spherical payload, these could be obtained:

- Rotation 1: bi-directional translations of *inner* pair fingertips (as shown in 4.6f)
- Rotation 2: bi-directional translations of *outer* pair fingertips at 4.6f, similar to *inner* pair
- Rotation 3: rotating the variable platform 4.9j while keeping the sphere grasped by *inner* pair

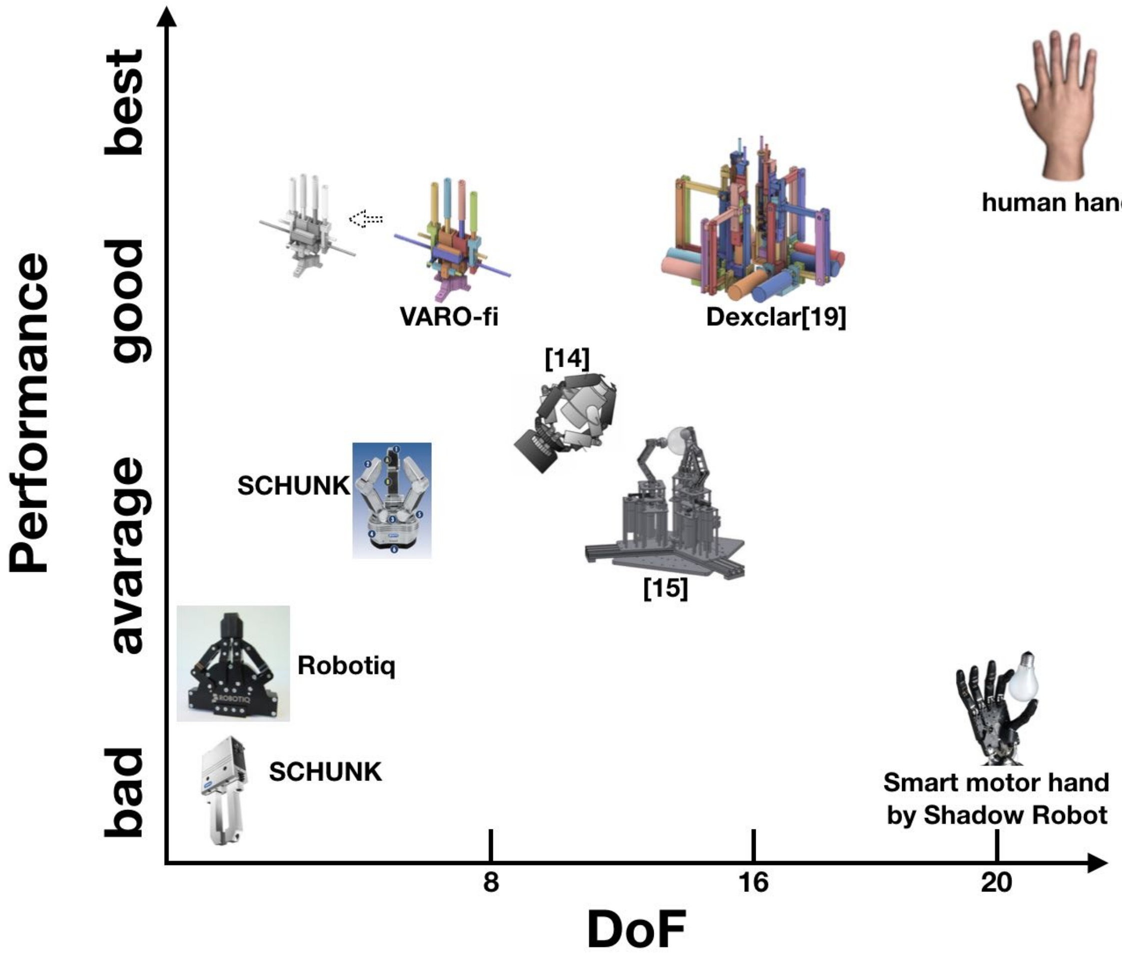


Fig. 4.8 An approximate comparison between some developed grippers and our hand

- Translation 1: equal and same directed translation of *inner* pair (4.5f: the *inner* pair slides along *linear guide* 2 (4.5b))
- Translation 2: equal and same directed translation of *outer* pair (4.5f: the *outer* pair slides along *linear guide* 1 (4.5b))
- Translation 3: translating the fingers upward of inner (4.6l) or outer pair

4.3 Experiments & Comparison with Existing Grippers

Several experiments have been conducted on the developed prototype and the claims have been validated (illustrated in fig. 4.9). The experiments video can be found in [86].

Grippers proposed in [79, 27, 80, 61] are the most relevant contributions compared to VARO-fi, in terms of dexterities and similarities in objective & design; hence, a detail comparison with [79, 27, 80, 61] have been shown in Table 4.1. VARO-fi is dexterous among three planes and the payload rotation on any plane is not bounded; since the variable platform is able to rotate freely, once the outer pair fingers move outwards, hence, a payload grasped by inner pair has no rotation limit, as well as for the remaining two planes (by using the property of reconfigurability, the proposed gripper is able to rotate spherical payload any desired angle: 4.9a-f). On the other hand, payload rotation is bounded (except about respective z axis (fig. 4.7)) in [79, 27]. And [80] requires a thorough motion plan for all of its joint in order to address any possible posture change regardless of any axis. Unlike [27, 80, 61], the VARO-fi does not consist of any kinematic chain or serial linkage; although, grippers of [27, 80, 61] are capable of doing manipulation similar to proposed gripper, however, the difficulties of motion planning cannot be avoided as it is described in (6). Thanks to the simple fingers of VARO-fi, those consist of translation actuations only.

Based on Table 4.1, a comparison has been portrayed in Fig. 4.8. Where, the $performance$ index is considered as the capabilities of in-hand manipulation and release of a spherical payload ($performance$ here is not based on reliability, repeatability, speed). It is notable from the plot that, the $degrees\ of\ freedom$ play a vital role for improving the within hand skills, at the same time, the chosen methodology (whether it is built by flexible materials, presence of complicated kinematic chains) underlines the $performance$.

The prototype which has been built (fig. 4.5), consists of 9 DoF. Although, both the inner & outer pair use 2 actuators individually in order to open and close (4.5f), however, that could be reduced up to 1 for each pair (inner, outer), by exploiting different mechanisms. Also, the vertical translations of the outer pair fingers could possibly avoided. Exploiting the variable platform & vertical translations of inner pair, 6 DoF motions are obtainable from the gripper step by step. And hence, VARO-fi could move leftwards in Fig. 4.8.

Last but not the least, VARO-fi is scalable, which means, it is fully dependent on the opening ranges of inner, outer pair and the required vertical translation of the fingers; and it can be easily built by choosing actuators according to it's target elongations regardless of any further analysis.

4.3.1 Grasp, Manipulation and Release Experiments

Figure 4.10 describes the consequences of grasp, manipulation and release with final configuration of a sphere. The sphere shown in Fig. 4.10a is required to be rotated 70 degree anti-clockwise about a plane similar to one pair's initial alignment (fig. 4.10c) of VARO-fi.

Table 4.2 Instruments used in experiment

Force Sensor: Loadcell	Burster 8413, range 0-50N, accuracy 0.5 in line amplifier Typ 9235
Controller	FPGA : Altera, 0-50N, accuracy 0.5 ARM: Sam4s Kit
1 Rotary actuator	Lynxmotion 12V 10rpm, Torque 5.4Nm
4 finger linear actuators	Actuonix P16, elongation 200mm, 22:1, 12 volt, potentiometer feedback
2 linear actuators over linear guide 1	Actuonix L16, elongation 100mm, 63:1, 12 volt, potentiometer feedback
2 linear actuators over linear guide 2	Actuonix P16, elongation 50mm, 22:1, 12 volt, potentiometer feedback

Instead if the planes are not similar, it is possible for the variable platform of the gripper to align with the required plane by rotating.

4.3.2 Twisting efficiency

Twisting efficiency has been evaluated for VARO-fi prototype on a sphere of 80*mm* diameter. The central idea is to evaluate the precision or accuracy of the manipulation by translation. To that aim, the magnitudes of the required translation are estimated for 20, 30 and 60 degree of rotation. And bi-directional motions with same velocity profile have been applied on the sphere. The results of the trials have been shown in Fig. 4.11. The average twist angles are found as 19.953, 29.517 and 59.0425 degree for desired values of 20, 30 and 60 degree respectively (with the accuracy of 99.76%, 98.39%, 98.40%).

4.3.3 Analysis for Re-grasp and Manipulation

Manipulation using Re-grasp also has been studied. Figure 4.12 illustrates one such attempt. The expected rotation of the sphere is 200 degree in total and 20 degree per cycle, hence re-grasping-manipulation have been applied for 10 times. The sphere has rotated 191 degree with an accuracy of 95.5%. The decrease in efficiency for repeated manipulation could be the result of disorientation of sphere or sliding at contact point in each re-grasping in the process.

4.4 Notes

Grippers/hand or end-effectors are generally built for target specific applications; hence, industrial and bio-inspired contributions in literature are considered as two separate domains. VARO-fi has been proposed in this paper which is a feasible solution for both trends of research lines and it surpassed many limitations of the capabilities of existing grippers of both fields. This paper also demonstrated finger synthesis process in order to obtain desired manipulation, which can be used to develop gripper for payload centric applications. The self reconfigurable VARO-fi is capable of satisfying the three major challenges; grasp, manipulation and release object with desired posture of plurality of payload, also minimizes grasp synthesis problem at some margin. Besides, the key advantages of such device are dexterity at any given plane, robustness such as, scalability and its construction simplicity in particular. In this research, a prototype of VARO-fi has been built and the validity of the claims have conducted through experiments.

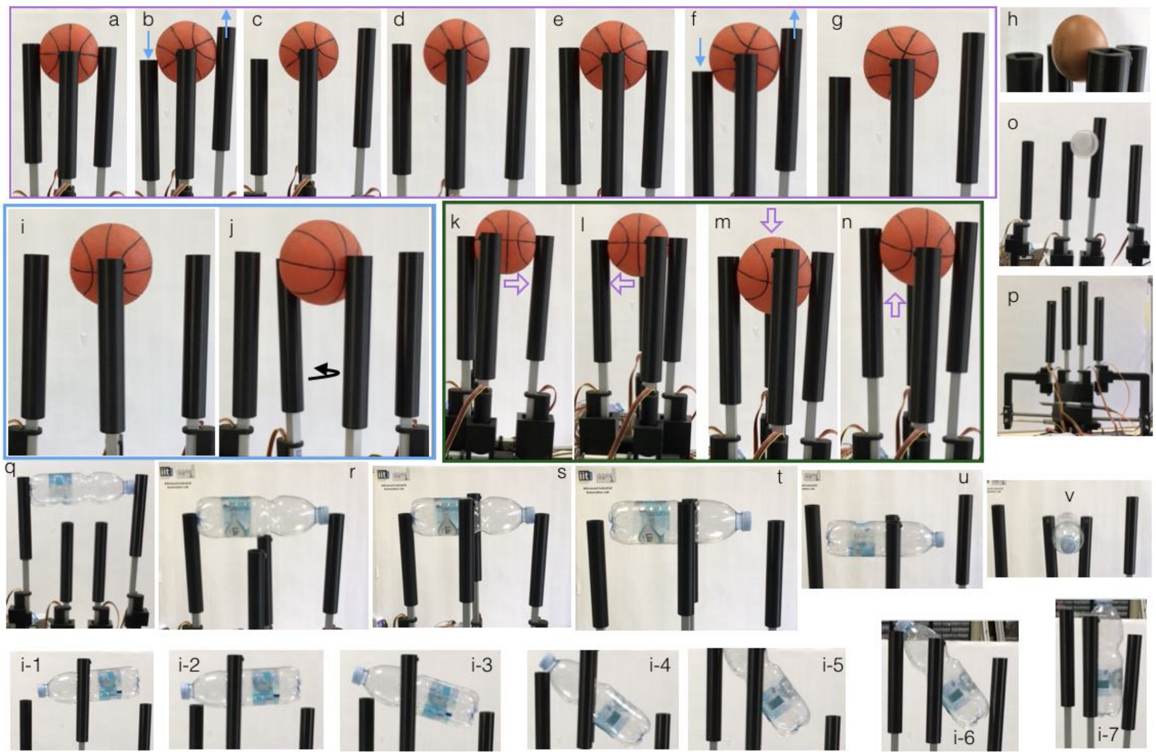


Fig. 4.9 a) a sphere is being grasped, b) bi-directional & equal translations of a pair will cause rotation, c) the pair breaks the contact, & d)reconfigures and e) re-grasps, f) re bidirectional translations to rotate the sphere further. g) the pair breaks the contact, at that point the payload can be released, or repeat (d)-(e), h) a 2-fingered grasp, i)- j) rotating sphere vertically, by actuating variable platform, k)-l) translation towards sideways, m)-n) translation towards upward, o) grasp and manipulation of a cup in human hand appearance, p) a different model of VARO-fi, where the inner pair can rotate without any interferences, q) grasping of a 500ml bottle by outer pair, r) inner pair rotates, opens & s) moves up, t) once inner pair grasped, outer pair breaks the contacts, u)-v) inner pair moves down & rotates 90 degree with the bottle, i-1) inner pair maintains a grasp, i-2)-i-5) outer pair places itself a position from where, the vertical translations of its finger able to twist the bottle. i-6)-i-7) a controlled pushing by outer pair twists the bottle until 90 degree

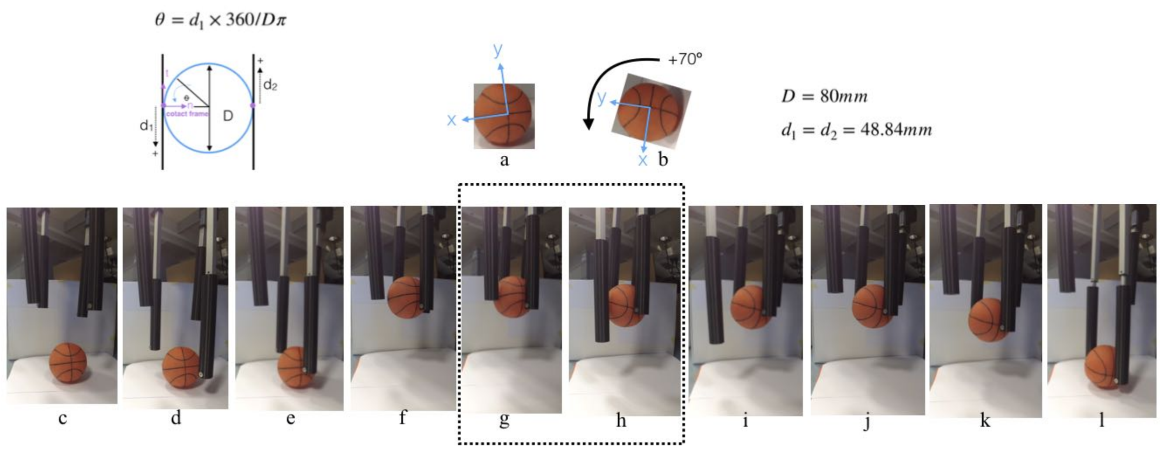
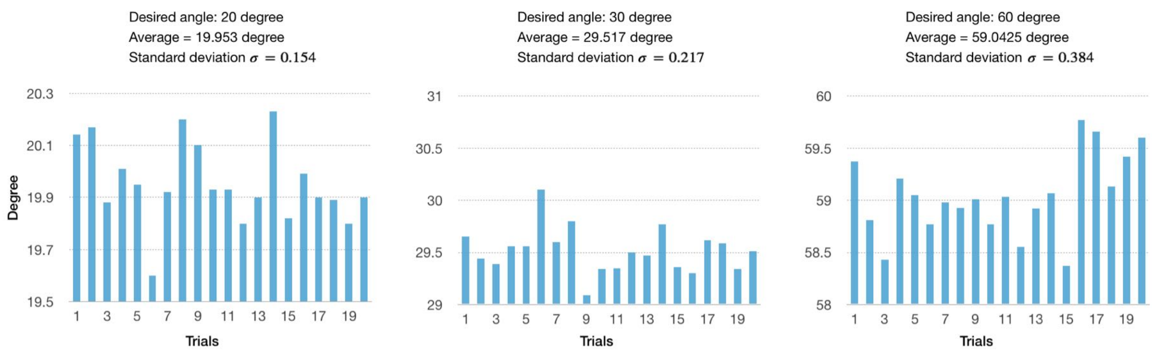
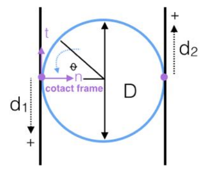


Fig. 4.10 a) the initial and b) desired posture of the sphere c) the sphere is placed with the initial posture under VARO-fi d) one pair of VARO-fi moves downward and e) grasps the sphere, f) the sphere is moved up by the pair and g) the remain pair makes contacts it h) the fingers of later pair translate 48.84mm in order to create the required twist of 70 degree on sphere i) the pair breaks contact and reconfigures k) the initial pair starts move down and l) place the sphere





d_1=d_2

$\theta = d_1 \times 360/D\pi$

Trials	Desired: 20 degree Actual angle twisted	Desired: 30 degree Actual angle twisted	Desired: 30 degree Actual angle twisted
1	20.14	29.65	59.37
2	20.17	29.44	58.81
3	19.88	29.39	58.43
4	20.01	29.56	59.21
5	19.95	29.56	59.05
6	19.6	30.1	58.77
7	19.92	29.6	58.98
8	20.2	29.8	58.93
9	20.1	29.09	59.01
10	19.93	29.34	58.77
11	19.93	29.35	59.03
12	19.8	29.5	58.55
13	19.9	29.47	58.92
14	20.23	29.77	59.07
15	19.82	29.36	58.37
16	19.99	29.3	59.77
17	19.90	29.62	59.66
18	19.89	29.59	59.13
19	19.8	29.34	59.42
20	19.9	29.51	59.60

Fig. 4.11 twisting efficiency analysis on VARO-fi prototype

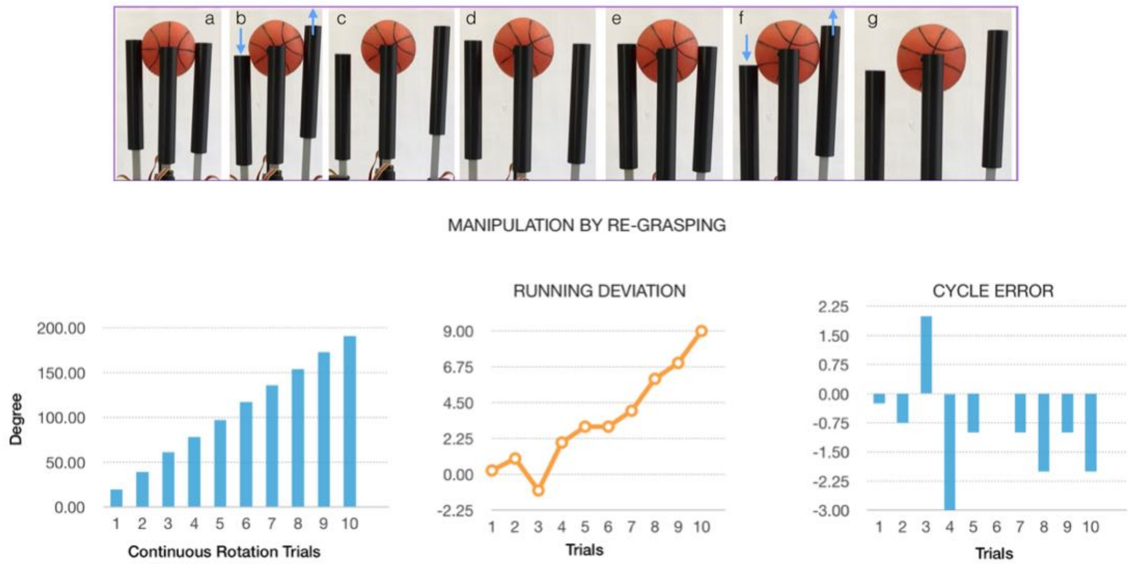


Fig. 4.12 re-grasp and manipulation analysis on VARO-fi prototype

Chapter 5 details a robotic competition called Bayer Robotic Challenge which took place in Berlin, Germany May 8-12, 2017. Team LAIAL participated in this challenge and achieved the top position. The competition tasks, proceedings have been explained in the beginning and then the proposed solutions have shown.

Chapter 5

Bayer Robotic Challenge

5.1 A brief description about Bayer and the competition

Bayer [3] is an innovation company with a more than 150-year history and core competencies in the fields of health care and agriculture. Bayer develops new molecules for use in innovative products and solutions to improve the health of humans, animals and plants. Bayer's research and development activities are based on a profound understanding of the biochemical processes in living organisms. Manufacturing of their products requires cutting edge technology and the highest standards of quality. However for certain tasks, automation of unstructured activities creates a challenge. The competition detail can be found in [4].

The 2017 Grants4Tech competition organized by Bayer aims to strengthen the bond between the Lifescience industry and academic robotic research communities. It shall give insights into the types of unstructured automation challenges those they face while advancing new solutions in the area of robotic automation. The challenge asks participants to build the robotic hardware and develop the software, that will attempt to master a simplified version of taking samples for quality control from a drum in goods receipt. The Robot will be asked to

- open the drum (described in fig. 5.1)
- open the inner bag (described in fig. 5.2)
- collect some sample (described in fig. 5.3a-c)
- close the bag (described in fig. 5.3d-e)
- close the drum (described in fig. 5.4)

The above tasks are being done hundreds of times every day at multiple sites of Bayer worldwide. To involve automation, it requires image recognition, movement planning, precise

Fig. 5.1 a) fiber drum b) drum, lid and clamping ring c) drum with seal (yellow) d) seal breaking by pliers e) the remain part (yellow) f) removing the remaining part g) the clamp is free for opening h)-k) opening the clamp and removing the lid from drum: mix of prehensile and non-prehensile manipulation

handling of different materials and tools and error recognition in a fixed amount of time. The Robots will be scored by fulfilling a sub step successfully. The competition took place at Berlin, Cube Tech fair [21], on May 8-12, 2017.

5.2 Feasibility test

To build the final robot system, it is required to verify the initial idea/concept, in order to obtain a feasible solution at the end. To do so, some analysis with existing robot system (ABB) have been carried. This analysis also helps to develop the final tools and end-effectors for the competition.

5.3 Methodology step by step

For the following description, the two robots are named as Arm1 and Arm2, in order to highlight the role of the two robotic arms. The Arms (Arm1 and Arm2) could be a dual arm or two separate robots. At the beginning of the application, first of all, the robots has to recognize that the drum has been given. Therefore, a vision system is required such as 3D camera, could be mounted on the top such that it able to compute the lid's diameter. Hence, the system can distinguish between the 38x65 and the 48x85 drum (one of two types will be

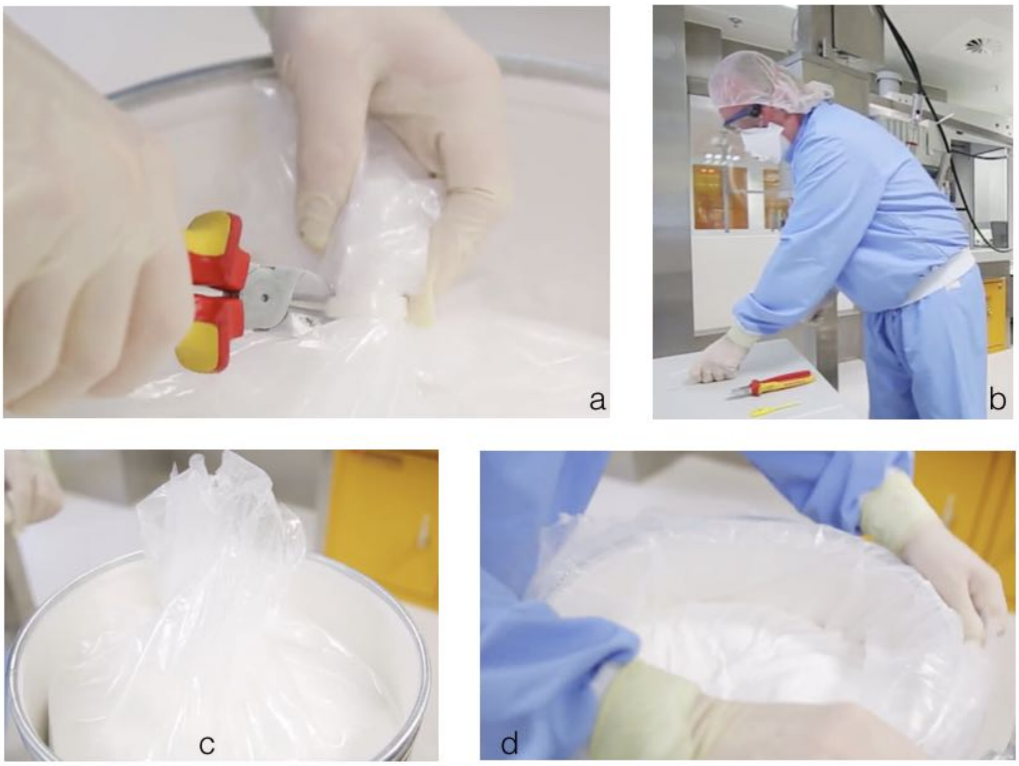


Fig. 5.2 a) cutting the cablestrap with pliers b) removing the cablestrap from the drum and placing it aside c) the bag after removing the cablestrap in the drum d) opening the bag

given in the competition). This determination is very important information for the remaining operations, such as inner bag opening, its closure and also sampling.

5.3.1 Open the drum

Break the seal

1. The position of the seal can be detected either by recognizing the clamp-position from above, or relying on its yellow color of the seal.

2. Arm1 (assume it is the closer to the clamp) grasps the bottom part of the seal.

3. The same Arm1 lifts the seal upward to give the cutting tool on the other arm sufficient space to cut the seal from above.

4. Arm2 picks the cutting tool from tool stand.

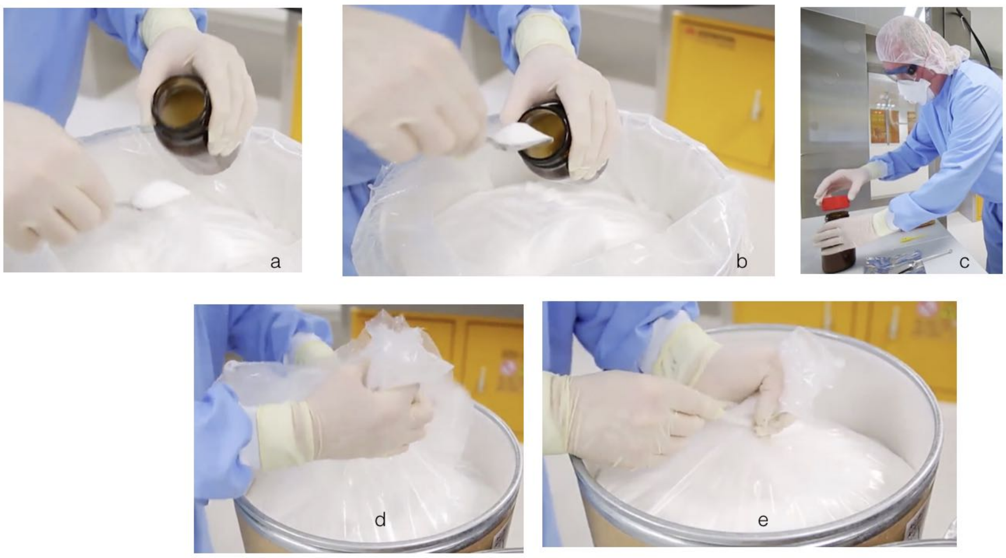


Fig. 5.3 a) - b) powder sampling c) place the bottle with sample aside d) closing the bag e) applying cable strap

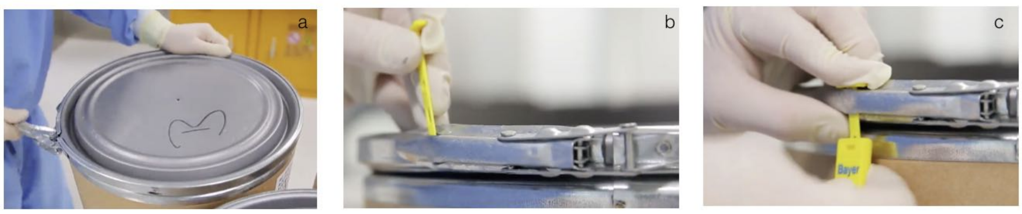


Fig. 5.4 a) closing the lid and locking the clamping ring b) placing the seal top c) locking the seal

5. Arm2 approaches for the cutting. A cutting tool is designed (will be explained later) such a way that, after the cutting operation, it will able to grab the remain-portion of the detached seal.

6. Both the upper and lower part of the seal are placed aside by the two arms.

7. Arm2, which is holding the cutting tool, releases it in the tool-stand.

Open the clamping ring

1. Arm1 holds the lid and the clamping ring in the opposite side of the clamp, to prevent lid and ring from falling down or encountering unexpected situations, once they are released.

2. Arm2 approaches and grabs the clamp.

Fig. 5.5 opening the drum trial

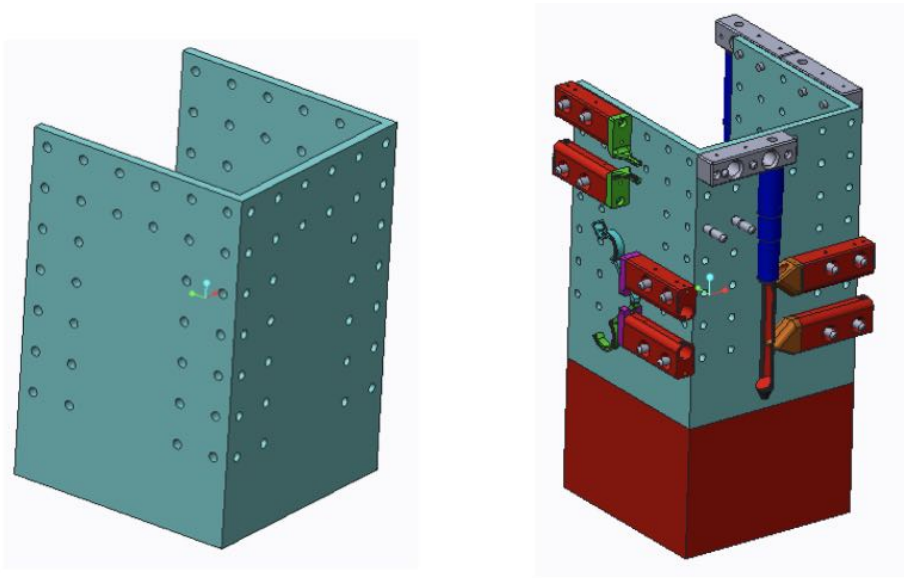
3. Arm2 pulls the clamp and opens the ring.

Securely place the clamping ring aside

1. Arm2 pushes the ring from its side, in order to loosen it on Arm1's side.
2. Arm1 pulls up the ring together with the lid.
3. Arm2 now grabs the ring and the lid from its side.
4. Arm1 and Arm2 both take away lid and ring, placing them aside in their cross-shaped support.

Securely place the lid aside

1. The lid results in being safely placed aside together with the ring in the previous task.



a b

Fig. 5.6 a) a conceptual tool box and mounted tools (b)

5.3.2 Open the inner bag

Cut the cable strap

1. Arm1, with the help of the camera, can locate the upper edge of the inner bag and grasp the neck.
2. Arm1 pulls up the inner bag a little, to create some room for cutting the cable strap.
3. Arm2 grabs an cutting tool from the tool stand.
4. with the help of the 3D camera mounted on the ceiling, Arm2 will go at the right position to cut the cable strap.
5. Arm1 cuts the cable strap.

Place the cable strap aside

1. while Arm1 cuts the cable strap, also grabs it (like done with the seal), so it can easily place it aside.

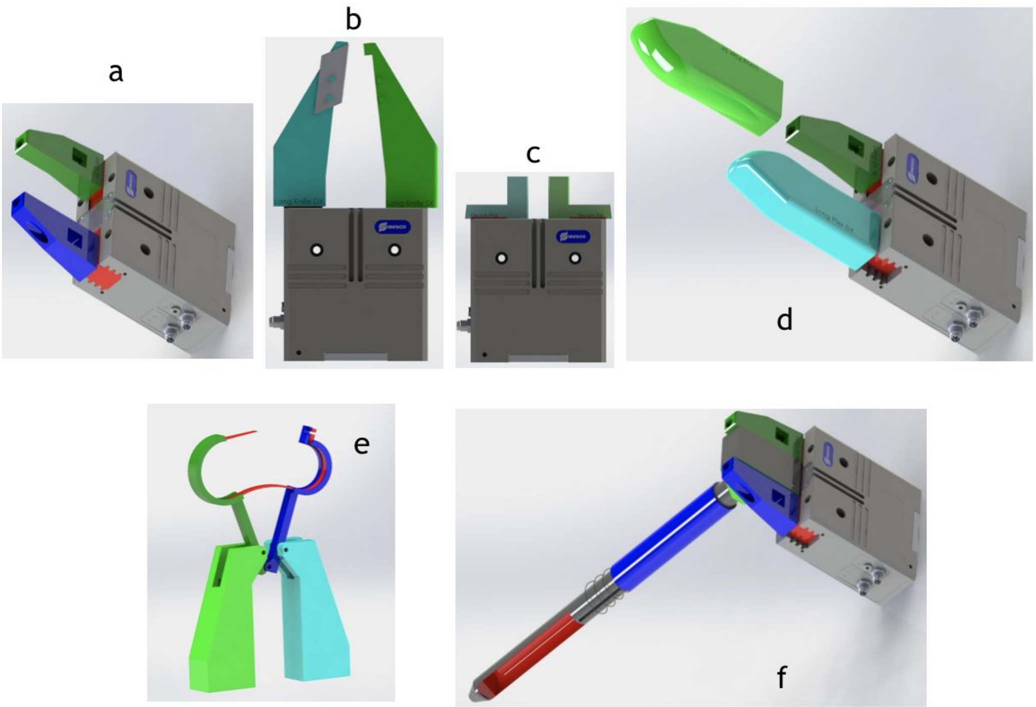


Fig. 5.7 Developed tools for the challenge

Open the bag

1. Arm2 will grab the pusher (will explain later) from tool stand and push along a diameter of the plastic bag from the top. For this maneuever, at the top side of the bag a channel will appear for the entry of the sampling tool. both the diameter and the opening channel will be detected by camera.

2. Arm1 will collect the sampling tool from the tool stand.

3. With the help of the camera, Arm1 inserts the sampling tool in the bag.

5.3.3 Sampling procedure

Take a spoon

1. Already done taking the sampling tool itself. The sampling tool is deigned (explained later) such a way that, the sampling will be done in the entrance process.

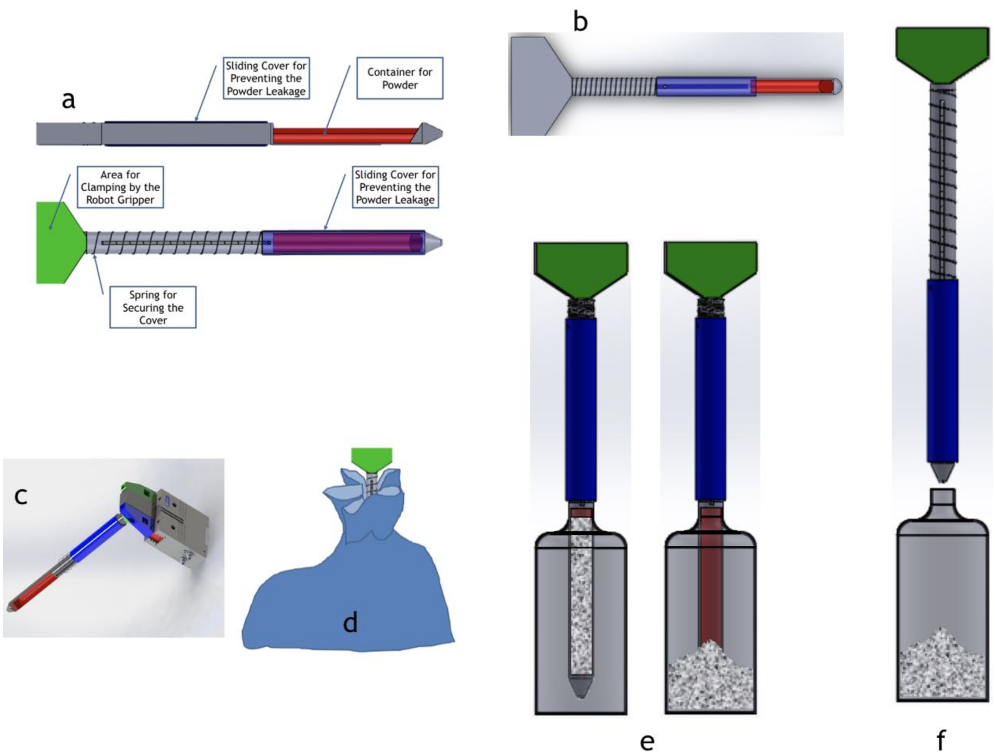


Fig. 5.8 the details of sampling tool

Take a sample

1. Already done taking the sampling tool itself. The sampling tool is deigned (explained later) such a way that, the sampling will be done in the entrance process.
2. Arm1 is pulled out from the powder, and the sampling tool is placed at particular place with sample.

Put the sample in the container

1. The sampling tool is itself designed as sample container. However, an external container can be designed if required.

Securely place the lid on the container

1. according to the freedom given to participants in designing their own sampling tool and container, the designed sampling tool already have done this.

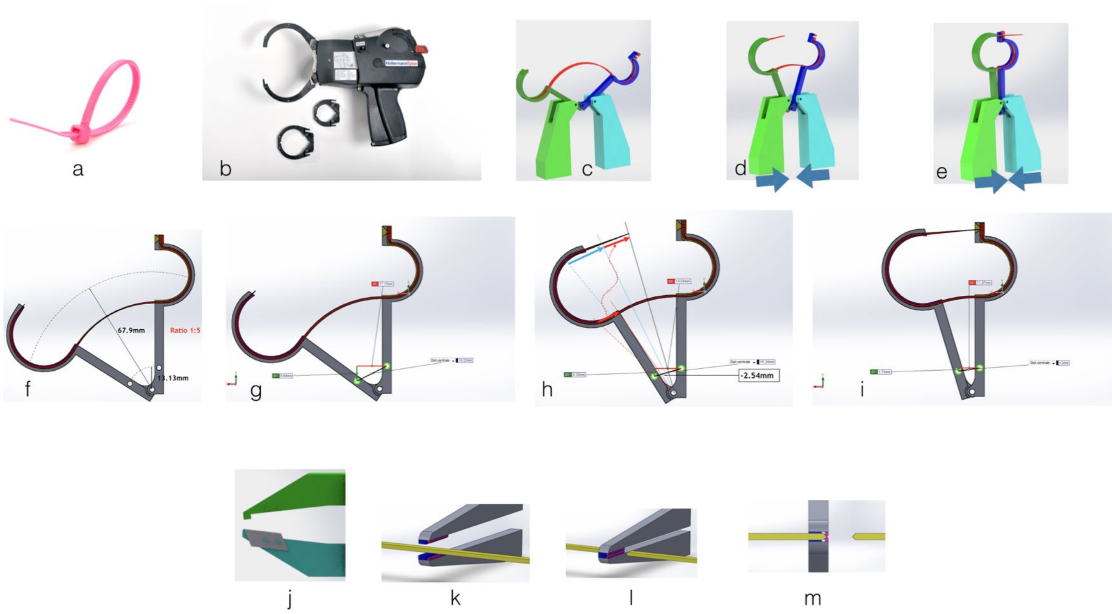


Fig. 5.9 the details of cable strap locker and cutting operation

Place the sample away

1. Arm1 places the sampling tool aside, in the tool stand.

5.3.4 Close the inner bag

Grab and close the bag

1. Arm2 will collect the cable tie tool and it tightens a cable strap to lock the inner bag.
2. Arm2 takes away the cable tie tool and places it aside, in the tool stand.

Attach a new cable strap to the inner bag

1. Done in previous step.

Push down the bag

1. by detecting the cable strap through the camera, Arm1 grabs the inner bag from that position.
2. then Arm1 will push down the inner bag's edge until it finds the resistive force of the powder.

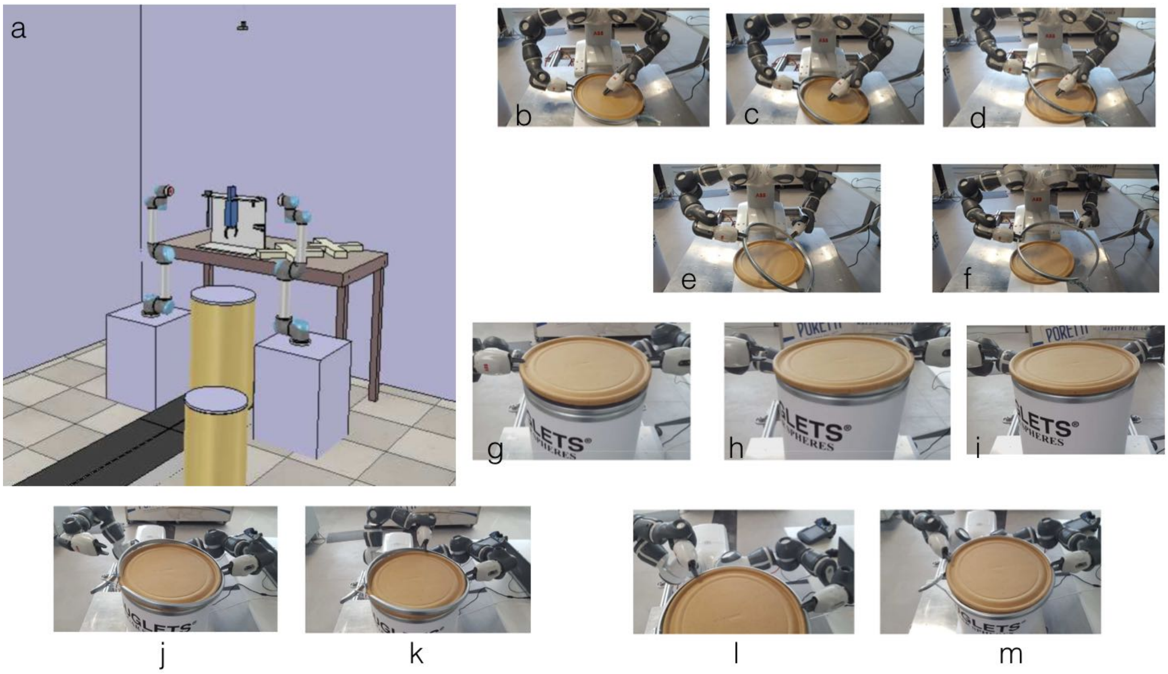


Fig. 5.10 a) typical setup, opening (b-f) and closing procedures (g-m) of drum

5.3.5 Close the drum and attach new seal

Put the lid on the drum

1. Arm1 grabs the lid from its support.
2. Arm1 proceeds with the lid in an angular manner such a way that the edge of the drum grabs the lid in the perfect alignment with the help of camera.
3. Arm1 releases the lid and get fixed with correct alignment.
4. Arm1 pushes the lid at the center from top.

Place the clamping ring on the drum

1. Arm1 grabs the ring (at clamping part) from its support.
2. Arm2 waits at the opposite side of the drum. In order to ensuring the bottom point of the clamping ring up to which it should go.
3. Arm2 slightly pushes the ring from its side, in order to loosen it on Arm1's side.
4. Arm1 slides around the lid's circumference (starting close to Arm1's gripping point) to completely engage the ring on the drum.

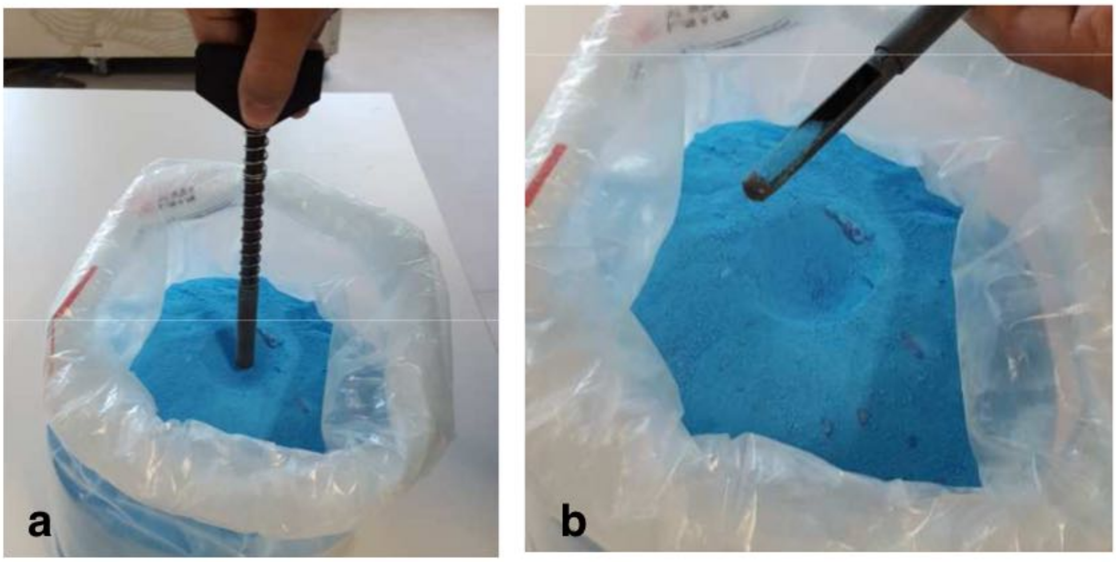


Fig. 5.11 feasibility test of sampling tool

5. Arm1 grabs the clamp.

6. Arm1 pushes and closes it.

5.4 End-effector/ tool design & tool stand

The overall tasks require a high diversity in tool/end-effector design. It is difficult to use multiple robots including multiple type of hands for the given tasks. And exploiting numerous robot and hand required a robust synchronization and parallel control. Cutting plastic seal, opening / closing of plastic bag or drum, sampling are very different demands. Hence, to keep the solution simple and feasible, only two manipulators (each can be mounted with different but one end-effector at a time) are chosen to be exploited.

Therefore, the idea is to develop several passive end-effectors including a tool stand. All the end-effectors will be mounted in the tool stand, and the manipulators will grab the correct tool at a particular time in order to conduct specific operation. Once the operation is done, the manipulator will place the tool in the tool stand again and grab the next required tool for next task. The specific tool will be mounted at $specific\ place$ in the tool stand. Hence, position control of the manipulator will be sufficient enough for the robot to pick the *right* tool and place at *right* place after the operation. The manipulators have been driven by ORI (see appendix B).

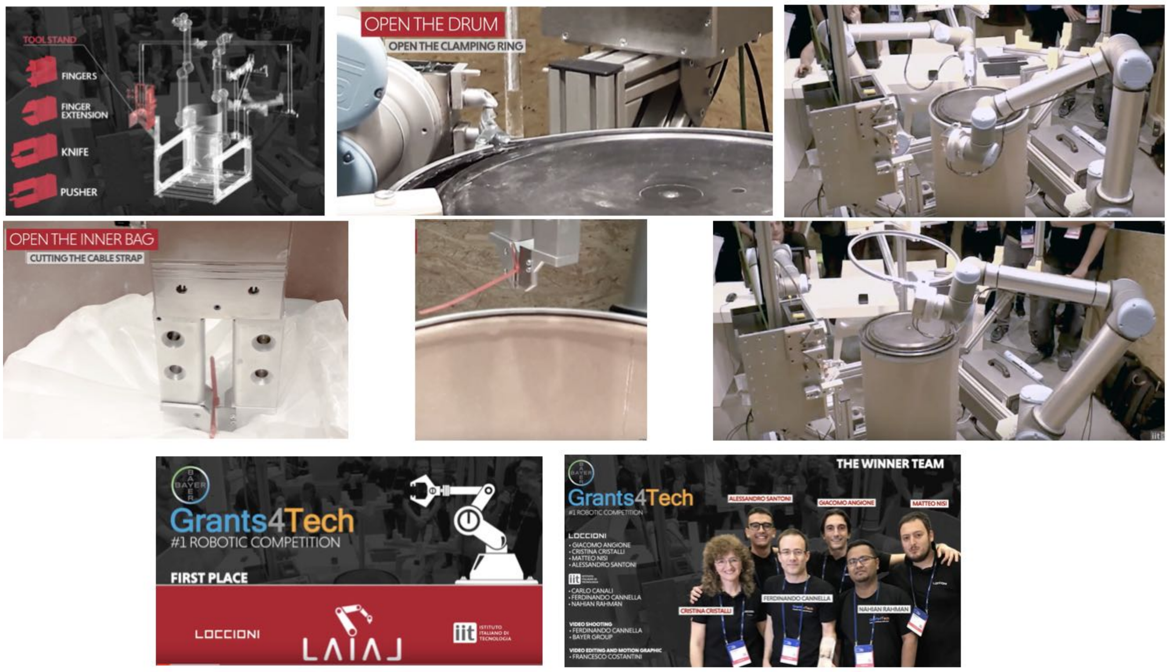


Fig. 5.12 competition pictures

Figure 5.7 shows the developed tools for this challenge, where Fig. 5.7a,c are long and short gripper and 5.7b is the cutting tool. The cutting tool will be mounted in the tool stand, and the gripper can grab it from there. Figure 5.7d is the pusher. Figure 5.7e cable tie tool and in 5.7f the gripper grabs the sampling tool whose detail is portrayed in Fig. 5.8. The feasibility of such sample tool also has been tested (fig. 5.11) during the formulation. The cable tie tool operation detail has been illustrated in Fig. 5.9. One of the major development of such cable tie tool lies on the simplicity and minimum effort to make it, where the automatic cable tie tool (fig. 5.9b) costs approximately 20,000 €.

5.5 Hardware and software used

5.5.1 Hardware: robots, sensors, cameras

Two 6-DoFs manipulators (UR-10 [85]) placed on two basements, mounted one in front of the other. This arrangement resulted in being the best solution to deal with any seal position around the drum's lid. And one 3D Ensenso vision camera form iDS.

5.5.2 SW (libraries, licenses)

- LabVIEW

- ORI

5.5.3 Industry cooperation partners

- Universal Robots
- iDS - Imaging Development Systems

5.6 The final competition & result

Our team LAIAL (Loccioni [36] and Advanced Industrial Automation Lab from IIT) won the first position in the challenge and received a prize money of 40,000 €. The video can be found in [37] (long version) and [74] (short version).

Chapter 6

Conclusion & Future work

Dexterous manipulation has been studied for the last few decades, and as a consequence, numerous grippers came into existence as an outcome. As portrayed in the previous chapters, grippers in literature or, those however the ones have been explicitly used as anthropomorphic, industrial hands, are not targeted the three major challenges (grasping, manipulating and releasing objects) as a whole. Most of the developed grippers either addressed grasping solely or, grasping with some degrees of manipulation. The reasons are plausible: it is really difficult to implement different prehensile capabilities and dexterity in a single device/hand, since it demands a explicit contributions from multiple disciplines such as, kinematics, sensors, dynamics, control, machine learning and so on. This study addressed the problem in one domain only, from kinematic point of view. Since the difficulties also arise due to need of realizing the optimal blend between sensors, control, machine learning and so on, it is essential to evaluate maximum limit achievable by each domain. Therefore, it will be much more convenient for the researchers to exploit particular portion of each domain to develop an efficient gripper for their application.

In this study, initially a gripper has been proposed as a generic platform named Dexclar (chapter 3). Several target specific grippers can be developed using the methodology used during this platform formulation. Also exploiting the manipulation concept of Dexclar, an end-effector named VARO-fi is proposed in chapter 4. Lastly, both the prehensile and non-prehensile manipulation have been used in a competition, thanks to Bayer. Although the explicit study of in-hand manipulation was not required in Bayer competition, however the necessities have been illustrated in chapter 1.

Another contribution of this study is, dexterous application such as, in-hand manipulation requires both the flexibility and dexterity from the gripper design. The major difficulty arises to construct such gripper, since achieving flexibility or adaptability and dexterity urge to use two different type of materials, soft and rigid. Here the problem has been addressed by exploiting primarily mechanisms and their properties such as modularity, reconfigurability.

In-hand manipulation allows posture change of grasped object within hand, hence the gripers having this capability, can recover from a poor initial grasp. How the payload will be grasped at the first place was not considered in this study from this point of view. However, a future study should address ingenious mechanism to conduct a stable grasp at the first place.

Future work

The study can be continued in some different perspectives

- evaluating ideal manipulation and grasping force for task specific application
- addressing arbitrary/irregular shaped payload in order to obtain twist about a given axis
- developing adaptive mechanisms to grasp at the first place automatically
- developing novel manipulators for in-hand manipulation by exploring light weighted flexible links and control to increase speed in industrial system
- developing mechanism centric dexterous prosthetic hands than available
- exploring different *shapes* of finger, *arrangements* and exploiting $attachment - detachment$ could be interesting dimension to proceed

References

[1] Arimoto, S. and Yoshida, M. (2011). A mathematical and numerically integrable modeling of 3d object grasping under rolling contacts between smooth surfaces. *Modelling and Simulation in Engineering*, 2011.

[2] Bai, Y. and Liu, C. K. (2014). Dexterous manipulation using both palm and fingers. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 1560–1565. IEEE.

[3] Bayer (2017). https://www.bayer.com/.

[4] BayerRoboticChallenge: (2017). https://grants4tech.bayer.com/robotic-competition/.

[5] Bicchi, A. (1995). On the closure properties of robotic grasping. *The International Journal of Robotics Research*, 14(4):319–334.

[6] Bicchi, A. (2000). Hands for dexterous manipulation and robust grasping: A difficult road toward simplicity. *IEEE Transactions on robotics and automation*, 16(6):652–662.

[7] Bircher, W. G., Dollar, A. M., and Rojas, N. (2017). A two-fingered robot gripper with large object reorientation range. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 3453–3460. IEEE.

[8] Borràs, J. and Dollar, A. M. (2015). Dimensional synthesis of three-fingered robot hands for maximal precision manipulation workspace. *The International Journal of Robotics Research*, 34(14):1731–1746.

[9] Boubekri, N. and Chakraborty, P. (2002). Robotic grasping: gripper designs, control methods and grasp configurations–a review of research. *Integrated Manufacturing Systems*, 13(7):520–531.

[10] Boughdiri, R., Nasser, H., Bezine, H., M'Sirdi, N. K., Alimi, A. M., and Naamane, A. (2012). Dynamic modeling and control of a multi-fingered robot hand for grasping task. *Procedia Engineering*, 41:923–931.

[11] Brown, C. Y. and Asada, H. H. (2007). Inter-finger coordination and postural synergies in robot hands via mechanical implementation of principal components analysis. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 2877–2882. IEEE.

[12] Bullock, I. M., Feix, T., and Dollar, A. M. (2015). Workspace shape and characteristics for human two-and three-fingered precision manipulation. *IEEE Transactions on Biomedical Engineering*, 62(9):2196–2207.

[13] Butterfaß, J., Grebenstein, M., Liu, H., and Hirzinger, G. (2001). Dlr-hand ii: Next generation of a dextrous robot hand. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 1, pages 109–114. IEEE.

[14] Caffaz, A., Casalino, G., Cannata, G., Panin, G., and Massucco, E. (2000). The dist-hand, an anthropomorphic, fully sensorized dexterous gripper. *IEEE Humanoids*.

[15] Callegari, M., Carbonari, L., Palmieri, G., and Palpacelli, M.-C. (2013). Parallel wrists for enhancing grasping performance. In *Grasping in Robotics*, pages 189–219. Springer.

[16] Canali, C., Cannella, F., Chen, F., Sofia, G., Eytan, A., and Caldwell, D. G. (2014). An automatic assembly parts detection and grasping system for industrial manufacturing. In *Automation Science and Engineering (CASE), 2014 IEEE International Conference on*, pages 215–220. IEEE.

[17] Carbonari, L., Callegari, M., Palmieri, G., and Palpacelli, M.-C. (2014). Analysis of kinematics and reconfigurability of a spherical parallel manipulator. *IEEE Transactions on Robotics*, 30(6):1541–1547.

[18] Chavan-Dafle, N., Rodriguez, A., Paolini, R., Tang, B., Srinivasa, S., Erdmann, M., Mason, M. T., Lundberg, I., Staab, H., and Fuhlbrigge, T. (2014). Extrinsic dexterity: In-hand manipulation with external forces. In *IEEE International Conference on Robotics and Automation (ICRA)*, Pittsburgh, PA.

[19] Chen, F., Cannella, F., Canali, C., Hauptman, T., Sofia, G., and Caldwell, D. (2014). In-hand precise twisting and positioning by a novel dexterous robotic gripper for industrial high-speed assembly. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 270–275. IEEE.

[20] Connolly, C. (2008). Prosthetic hands from touch bionics. *Industrial Robot: An International Journal*, 35(4):290–293.

[21] CubeTechFair (2017). http://tech-fair.cube-global.com/.

[22] Deimel, R. and Brock, O. (2013). A compliant hand based on a novel pneumatic actuator. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 2047–2053. IEEE.

[23] Deimel, R. and Brock, O. (2016). A novel type of compliant and underactuated robotic hand for dexterous grasping. *The International Journal of Robotics Research*, 35(1-3):161–185.

[24] DexclarExperiment: (2017). https://www.youtube.com/watch?v=Fjb2dYJ-by0.

[25] Di Gregorio, R. (2004). The 3-rrs wrist: a new, simple and non-overconstrained spherical parallel manipulator. *Transactions-American society of mechanical engineers journal of mechanical design*, 126:850–855.

[26] Gabiccini, M., Bicchi, A., Prattichizzo, D., and Malvezzi, M. (2011). On the role of hand synergies in the optimal choice of grasping forces. *Autonomous Robots*, 31(2):235–252.

[27] Higashimori, M., Jeong, H., Ishii, I., Kaneko, M., Namiki, A., and Ishikawa, M. (2005). A new four-fingered robot hand with dual turning mechanism. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 2679–2684. IEEE.

[28] Hirose, S. and Umetani, Y. (1978). The development of soft gripper for the versatile robot hand. *Mechanism and machine theory*, 13(3):351–359.

[29] Howard, W. S. and Kumar, V. (1996). On the stability of grasped objects. *IEEE Transactions on Robotics and Automation*, 12(6):904–917.

[30] Hunt, K. H. (1978). *Kinematic geometry of mechanisms*, volume 7. Oxford University Press, USA.

[31] Jones, F. D., Horton, H. L., and Newell, J. A. (1930). *Ingenious mechanisms for designers and inventors...*, volume 1. Industrial Press Inc.

[32] Jung, G.-P., Jeong, U., Koh, J.-S., and Cho, K.-J. (2013). The development of a scalable underactuated gripper based on flexural buckling. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 2077–2077. IEEE.

[33] Khalil, W., Dombre, E., Merlet, J.-P., Pierrot, F., Wenger, P., Haddad, M., Chettibi, T., Lehtihet, H., Dauchez, P., Fraisse, P., et al. (2007). *Modeling, performance analysis and control of robot manipulators*. Wiley Online Library.

[34] Klingbeil, E., Rao, D., Carpenter, B., Ganapathi, V., Ng, A. Y., and Khatib, O. (2011). Grasping with application to an autonomous checkout robot. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 2837–2844. IEEE.

[35] Lee, S., Noh, S., Lee, Y., and Park, J. H. (2009). Development of bio-mimetic robot hand using parallel mechanisms. In *Robotics and Biomimetics (ROBIO), 2009 IEEE International Conference on*, pages 550–555. IEEE.

[36] Loccioni: (2017). http://www.loccioni.com/?lang=en.

[37] LongVersion: (2017). https://www.youtube.com/watch?v=s5lQRyMQ1xY&feature=youtu.be.

[38] Ma, R. R., Odhner, L. U., and Dollar, A. M. (2013). A modular, open-source 3d printed underactuated hand. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 2737–2743. IEEE.

[39] Ma, R. R., Spiers, A., and Dollar, A. M. (2016). M2 gripper: extending the dexterity of a simple, underactuated gripper. In *Advances in reconfigurable mechanisms and robots II*, pages 795–805. Springer.

[40] MacKenzie, C. L. and Iberall, T. (1994). *The grasping hand*, volume 104. Elsevier.

[41] Maeda, S., Tsujiuchi, N., Koizumi, T., Sugiura, M., and Kojima, H. (2012). Development and control of a pneumatic robot arm for industrial fields. *International Journal of Advanced Robotic Systems*, 9(3):59.

[42] Martell, J. S. and Gini, G. (2007). Robotic hands: Design review and proposal of new design process. *World Academy of Science, Engineering and Technology*, 26:85–90.

[43] Mason, M. T. (1982). Manipulator grasping and pushing operations.

[44] Mason, M. T. (1986). Mechanics and planning of manipulator pushing operations. *The International Journal of Robotics Research*, 5(3):53–71.

[45] Mason, M. T. (2001). *Mechanics of robotic manipulation*. MIT press.

[46] Mason, M. T. and Lynch, K. M. (1993). Dynamic manipulation. In *Intelligent Robots and Systems' 93, IROS'93. Proceedings of the 1993 IEEE/RSJ International Conference on*, volume 1, pages 152–159. IEEE.

[47] Mattar, E. (2013). A survey of bio-inspired robotics hands implementation: New directions in dexterous manipulation. *Robotics and Autonomous Systems*, 61(5):517–544.

[48] Merlet, J.-P. (2006). Jacobian, manipulability, condition number, and accuracy of parallel robots. *Journal of Mechanical Design*, 128(1):199–206.

[49] Minguzzi, E. (2013). A geometrical introduction to screw theory. *European Journal of Physics*, 34(3):613–632.

[50] Monkman, G. J., Hesse, S., Steinmann, R., and Schunk, H. (2007). *Robot grippers*. John Wiley & Sons.

[51] Montana, D. J. (1992). Contact stability for two-fingered grasps. *IEEE Transactions on Robotics and Automation*, 8(4):421–430.

[52] Moura, J. T., Elmali, H., and Olgac, N. (1997). Sliding iviode control with sliding perturbation observer. *Journal of Dynamic Systems, Measurement, and Control*, 119:657.

[53] MSCsoftware (2017). http://www.mscsoftware.com/product/adams.

[54] Murray, R. M., Li, Z., Sastry, S. S., and Sastry, S. S. (1994). *A mathematical introduction to robotic manipulation*. CRC press.

[55] Namiki, A., Imai, Y., Ishikawa, M., and Kaneko, M. (2003). Development of a high-speed multifingered hand system and its application to catching. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 3, pages 2666–2671. IEEE.

[56] Napier, J. R. (1956). The prehensile movements of the human hand. *Bone & Joint Journal*, 38(4):902–913.

[57] Odhner, L. U., Jentoft, L. P., Claffee, M. R., Corson, N., Tenzer, Y., Ma, R. R., Buehler, M., Kohout, R., Howe, R. D., and Dollar, A. M. (2014). A compliant, underactuated hand for robust manipulation. *The International Journal of Robotics Research*, 33(5):736–752.

[58] Odhner, L. U., Ma, R. R., and Dollar, A. M. (2013). Open-loop precision grasping with underactuated hands inspired by a human manipulation strategy. *IEEE Transactions on Automation Science and Engineering*, 10(3):625–633.

[59] Palpacelli, M.-C., Carbonari, L., Palmieri, G., and Callegari, M. (2015). Analysis and design of a reconfigurable 3-dof parallel manipulator for multimodal tasks. *IEEE/ASME Transactions on Mechatronics*, 20(4):1975–1985.

[60] Price, A., Jnifene, A., and Naguib, H. (2007). Design and control of a shape memory alloy based dexterous robot hand. *Smart Materials and Structures*, 16(4):1401.

[61] Rahman, N., Carbonari, L., , Canali, C., Caldwell, D., and Cannella, F. (2017a). Dexclar: A gripper platform for payload- centric manipulation and dexterous applications. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pages 1943–1950. IEEE.

[62] Rahman, N., Carbonari, L., Caldwell, D., and Cannella, F. (2017b). Kinematic analysis, prototypation and control of a novel gripper for dexterous applications. *Journal of Intelligent & Robotic Systems*, pages 1–14.

[63] Rahman, N., Carbonari, L., D'Imperio, M., Canali, C., Caldwell, D. G., and Cannella, F. (2016a). A dexterous gripper for in-hand manipulation. In *Advanced Intelligent Mechatronics (AIM), 2016 IEEE International Conference on*, pages 377–382. IEEE.

[64] Rahman, N., D'Imperio, M., Carbonari, L., Chen, F., Canali, C., Caldwell, D. G., and Cannella, F. (2015). A novel bio-inspired modular gripper for in-hand manipulation. In *Robotics and Biomimetics (ROBIO), 2015 IEEE International Conference on*, pages 7–12. IEEE.

[65] Rahman, N., D'Imperio, M., Carbonari, L., Palpacelli, M., Cannella, F., and Caldwell, D. (2016b). Kinematic analysis and synthesis of a novel gripper for dexterous applications. In *Mechatronic and Embedded Systems and Applications (MESA), 2016 12th IEEE/ASME International Conference on*, pages 1–6. IEEE.

[66] Ryu, J.-C., Ruggiero, F., and Lynch, K. M. (2013). Control of nonprehensile rolling manipulation: Balancing a disk on a disk. *IEEE Transactions on Robotics*, 29(5):1152–1161.

[67] Sahbani, A., El-Khoury, S., and Bidaud, P. (2012). An overview of 3d object grasp synthesis algorithms. *Robotics and Autonomous Systems*, 60(3):326–336.

[68] Salisbury, J. K. and Roth, B. (1983). Kinematic and force analysis of articulated mechanical hands. *Journal of Mechanisms, Transmissions, and Automation in Design*, 105(1):35–41.

[69] Saut, J.-P., Sahbani, A., El-Khoury, S., and Perdereau, V. (2007). Dexterous manipulation planning using probabilistic roadmaps in continuous grasp subspaces. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 2907–2912. IEEE.

[70] Schunk, H., Steinmann, R., and Wolf, A. (2005). *Grippers in Motion*. Springer.

[71] SchunkGrippers: (2017). https://schunk.com/us_en/gripping-systems/.

[72] Shahinpoor, M., Bar-Cohen, Y., Simpson, J., and Smith, J. (1998). Ionic polymer-metal composites (ipmcs) as biomimetic sensors, actuators and artificial muscles-a review. *Smart materials and structures*, 7(6):R15.

[73] Shimoga, K. B. (1996). Robot grasp synthesis algorithms: A survey. *The International Journal of Robotics Research*, 15(3):230–266.

[74] ShortVersion: (2017). https://www.youtube.com/watch?v=ExVWy7bIRoQ&t.

[75] Siciliano, B. and Khatib, O. (2016). *Springer handbook of robotics*. Springer.

[76] Smith, C., Karayiannidis, Y., Nalpantidis, L., Gratal, X., Qi, P., Dimarogonas, D. V., and Kragic, D. (2012). Dual arm manipulation?a survey. *Robotics and Autonomous systems*, 60(10):1340–1353.

[77] Soriano, E., Rubio, H., Castejón, C., and García-Prada, J. (2015). Design of a low-cost manipulator arm for industrial fields. In *New Trends in Mechanism and Machine Science*, pages 839–847. Springer.

[78] Sudsang, A., Srinivasa, N., et al. (2000). Grasping and in-hand manipulation: Geometry and algorithms. *Algorithmica*, 26(3-4):466–493.

[79] Tadakuma, K., Tadakuma, R., Higashimori, M., and Kaneko, M. (2012). Robotic finger mechanism equipped omnidirectional driving roller with two active rotational axes. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 3523–3524. IEEE.

[80] Tahara, K., Maruta, K., Kawamura, A., and Yamamoto, M. (2012). Externally sensorless dynamic regrasping and manipulation by a triple-fingered robotic hand with torsional fingertip joints. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 3252–3257. IEEE.

[81] Tincani, V., Catalano, M., Farnioli, E., Garabini, M., Grioli, G., Fantoni, G., and Bicchi, A. (2012). Velvet fingers: A smart gripper with controlled contact surfaces. In *International Conference of Intelligent Robots and Systems-IROS 2012*, pages 7–12.

[82] Tincani, V., Grioli, G., Catalano, M. G., Garabini, M., Grechi, S., Fantoni, G., and Bicchi, A. (2013). Implementation and control of the velvet fingers: a dexterous gripper with active surfaces. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 2744–2750. IEEE.

[83] Tournassoud, P., Lozano-Pérez, T., and Mazer, E. (1987). Regrasping. In *Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, volume 4, pages 1924–1928. IEEE.

[84] Townsend, W. (2000). The barretthand grasper–programmably flexible part handling and assembly. *Industrial Robot: an international journal*, 27(3):181–188.

[85] UniversalRobot: (2017). https://www.universal-robots.com/.

[86] VARO-fiExperiment: (2017). https://youtu.be/b0nZ0TQsiYI.

[87] Zhang, K. and Dai, J. S. (2015). Screw-system-variation enabled reconfiguration of the bennett plano-spherical hybrid linkage and its evolved parallel mechanism. *Journal of Mechanical Design*, 137(6):062303.

Publications

1. Nahian Rahman, Luca Carbonari, Darwin Caldwell and Ferdinando Cannella, "Kinematic Analysis, Prototypation and Control of a Novel Gripper for Dexterous Applications", in Journal of Intelligent & Robotic Systems.

2. Nahian Rahman, Mariapaola D'Imperio, Luca Carbonari, Fei Chen, Carlo Canali and Ferdinando Cannella, "A Novel Bio-inspired Modular Gripper for in-hand Manipulation," Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO) 2015, Zhuhai, China, December 6 - 9, pp. 7-12.

3. Nahian Rahman, Luca Carbonari, Mariapaola D'Imperio, Carlo Canali, Darwin G. Caldwell and Ferdinando Cannella, "A Novel Reconfigurable Modular Gripper for in-hand object manipulation & release with appropriate posture" in the Proceedings of the ASME 2016 IDETC, August 21-24, 2016, North Carolina, USA.

4. Nahian Rahman, Luca Carbonari, Mariapaola D'Imperio, Carlo Canali, Darwin G. Caldwell and Ferdinando Cannella, "A Dexterous Gripper For In-Hand Manipulation," in the Proceedings of IEEE International Conference on Advanced Intelligent Mechatronics (AIM) 2016, Alberta, Canada, July 12-15, pp. 377-382.

5. Nahian Rahman, Luca Carbonari and Ferdinando Cannella, "Kinematic Analysis And Synthesis of a Novel Gripper for Dexterous Applications" accepted in the Proceedings of The 12th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications(MESA), Auckland New Zealand from August 29-31, 2016, pp. 1-6.

6. Nahian Rahman, Luca Carbonari, Darwin Caldwell and Ferdinando Cannella, "Manipulation & Workspace Analysis of Dexclar: A Newly formed Dexterous Gripper" Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO) 2016, Qingdao, China, December 3 - 7.

7. Nahian Rahman, Carlo Canali, Darwin Caldwell and Ferdinando Cannella, "Dexterous Gripper Synthesis from Modular finger Approach", accepted to the Proceedings of

ASME IDETC/MESA 2017, Ohio, USA.

8. Carlo Canali, Nahian Rahman, Fei Chen, Mariapaola D'Imperio, Darwin Caldwell and Ferdinando Cannella, "Theoretical and Kinematic Solution of High Reconfigurable Grasping for Industrial Manufacturing" accepted in 27th International Conference on Flexible Automation and Intelligent Manufacturing, 2017.

9. Nahian Rahman, Luca Carbonari, Carlo Canali, Darwin Caldwell and Ferdinando Cannella, "Dexclar: A Gripper Platform for Payload-Centric Manipulation and Dexterous Applications" accepted to the Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017, Vancouver, Canada, September24 - September28.

10. Mariapaola D'Imperio, Luca Carbonari, Nahian Rahman, Carlo Canali and Ferdinando Cannella, "KARL: A New Bio-Inspired Modular Limb for Robotic Applications," Proceedings of ASME/IEEE International Conference on Advanced Intelligent Mechatronics (AIM2015), 2015, Busan, South Korea, July 7-11, pp. 183-188.

11. Ferdinando Cannella, Mariapaola D'Imperio, Carlo Canali, Nahian Rahman, Fei Chen, Daniele Catelani, Darwin G. Caldwell and Jian S. Dai, "Origami Carton Folding Analysis Using Flexible Panels," Advances in Reconfigurable Mechanisms and Robots (REMAR) 2015, Beijing, China, July 20-22.

12. Mariapaola D'Imperio, Luca Carbonari, Nahian Rahman, Carlo Canali and Ferdinando Cannella, "A Novel Parallely Actuated Bio-Inspired Modular Limb," Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015, Hamburg, Germany, September28 - October 3, pp. 347-352.

13. Mariapaola D'Imperio, Ferdinando Cannella, Luca Carbonari, Nahian Rahman and Darwin G. Caldwell, "Dynamic Modelling and Analysis of an Articulated Robotic Leg," Proceedings of the ASME 2015 International Design Engineering Technical Conferences (IDETC) 2015, August 2-15, 2015, Boston, USA.

14. Ferdinando Cannella, Mariapaola D'Imperio, Carlo Canali, Nahian Rahman, Fei Chen, Daniele Catelani, Darwin G. Caldwell, Jian S. Dai, "Origami carton folding analysis using flexible panels", (2016) Mechanisms and Machine Science, 36, pp. 95-106.

Appendix A

FPGA code

The following VHDL code is used in the experimentation of Dexclar and VARO-fi. (Courtesy: Peter Samarin, peter.samarin@gmail.com for I2c.vhdl)

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.numeric_std.all;
USE ieee.std_logic_unsigned.all;

ENTITY main IS
GENERIC(

sys_clk1 : INTEGER :=50_000_000; –system clock frequency in Hz
pwm_freq1 : INTEGER := 50_000; –PWM switching frequency in Hz
bits_resolution1 : INTEGER := 8; –bits of resolution setting the duty cycle
phases1 : INTEGER := 1;

ADDR : std_logic_vector(6 downto 0) := "0000100");

PORT(
scl_clock : inout std_logic;
sda_data : inout std_logic;
clk_main : in std_logic;
```

```
Rst : in std_logic;

Motor1 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);
Motor2 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);

Motor3 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);
Motor4 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);

Motor5 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);
Motor6 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);

Motor7 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);
Motor8 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);

Motor9 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);
Motor10 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);

Motor11 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);
Motor12 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);

Motor13 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);
Motor14 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);

Motor15 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);
Motor16 : OUT STD_LOGIC_VECTOR(phases1 -1 DOWNTO 0);

Motor1_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor1_p2: out std_LOGIC_VECTOR(0 downto 0);

Motor2_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor2_p2: out std_LOGIC_VECTOR(0 downto 0);

Motor3_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor3_p2: out std_LOGIC_VECTOR(0 downto 0);
```

```
    Motor4_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor4_p2: out std_LOGIC_VECTOR(0 downto 0);

    Motor5_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor5_p2: out std_LOGIC_VECTOR(0 downto 0);

    Motor6_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor6_p2: out std_LOGIC_VECTOR(0 downto 0);

    Motor7_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor7_p2: out std_LOGIC_VECTOR(0 downto 0);

    Motor8_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor8_p2: out std_LOGIC_VECTOR(0 downto 0);

    Motor9_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor9_p2: out std_LOGIC_VECTOR(0 downto 0);

    Motor10_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor10_p2: out std_LOGIC_VECTOR(0 downto 0);

    Motor11_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor11_p2: out std_LOGIC_VECTOR(0 downto 0);

    Motor12_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor12_p2: out std_LOGIC_VECTOR(0 downto 0);

    Motor13_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor13_p2: out std_LOGIC_VECTOR(0 downto 0);

    Motor14_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor14_p2: out std_LOGIC_VECTOR(0 downto 0);

    Motor15_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor15_p2: out std_LOGIC_VECTOR(0 downto 0);
```

Motor16_p1: out std_LOGIC_VECTOR(0 downto 0);
Motor16_p2: out std_LOGIC_VECTOR(0 downto 0);

– User interface
–Read_req : buffer std_logic;
–Data_to_master : in std_logic_vector(7 downto 0);
–Data_valid : buffer std_logic;
–Data_from_master : buffer std_logic_vector(7 downto 0);
–test leds
l5: out std_logic_vector(0 downto 0);
– led2 : out std_logic_vector(0 downto 0);
– led3 : out std_logic_vector(0 downto 0);
– led4 : out std_logic_vector(0 downto 0);

– green_led : std_logic_vector(0 downto 0));
–i2c serial data
green_led : out std_logic);
–i2c serial data
END main;

ARCHITECTURE logic OF main IS;

SIGNAL temp : STD_LOGIC_VECTOR(7 DOWNTO 0);
SIGNAL i2c_data_wr : STD_LOGIC_VECTOR(7 DOWNTO 0);
SIGNAL i2c_data_wr2 : STD_LOGIC_VECTOR(7 DOWNTO 0);

SIGNAL store : STD_LOGIC_VECTOR(7 DOWNTO 0);
SIGNAL store2 : STD_LOGIC_VECTOR(7 DOWNTO 0);

SIGNAL MN : STD_LOGIC_VECTOR(3 DOWNTO 0);

SIGNAL dirM : STD_LOGIC_VECTOR(1 DOWNTO 0);

SIGNAL Allow : STD_LOGIC:='0';
SIGNAL Allow1 : STD_LOGIC:='0';
SIGNAL Allow2 : STD_LOGIC:='0';

```
SIGNAL RoW : integer range 0 to 5:=3;

    signal i2c_data_valid : std_logic;
signal test : integer range 0 to 255;
signal usgn : unsigned(7 downto 0);
signal clk_1hz : std_logic;
signal scaler : integer range 0 to 250:=0;

    signal prevscaler : integer range 0 to 250:=0;

    signal scaler2 : integer range 0 to 250:=0;
signal scalerN : integer range 0 to 25000000:=0;
signal countM : integer range 0 to 25000000:=0;
– signal clk_1hz : std_logic:=’0’;
signal LED : std_logic:=’0’;
signal ind : std_logic:=’0’;

    signal m1 : std_logic:= ’0’;
signal m2 : std_logic:= ’0’;

    signal m3 : std_logic:= ’0’;
signal m4 : std_logic:= ’0’;

    signal m5 : std_logic:= ’0’;
signal m6 : std_logic:= ’0’;

    signal m7 : std_logic:= ’0’;
signal m8 : std_logic:= ’0’;

    signal m9 : std_logic:= ’0’;
signal m10 : std_logic:= ’0’;

    signal m11 : std_logic:= ’0’;
signal m12 : std_logic:= ’0’;
```

```
    signal m13 : std_logic:= '0';
signal m14 : std_logic:= '0';

    signal m15 : std_logic:= '0';
signal m16 : std_logic:= '0';

    signal reset_btn : std_logic:= '1';

    signal reqRead : std_logic;
signal i2c_data : std_LOGIC_VECTOR(7 downto 0);

    constant WORLD_STATE_WIDTH : natural := 38;

    subtype byte_t is std_logic_vector(7 downto 0);

    type world_state_t is array (0 to WORLD_STATE_WIDTH-1) of byte_t;

    signal world_state : world_state_t := (others => X"00");

    signal fpArray : world_state_t := (others => X"00");

    signal received_bytes : integer range 0 to WORLD_STATE_WIDTH-1 := 0;

    signal received_bytes_std : std_logic_vector(15 downto 0) := (others => '0');

    signal Duty11 : std_LOGIC_VECTOR(7 downto 0):="00000000";
signal DutyMotor1 : std_LOGIC_VECTOR(7 downto 0):="00000000";
signal DutyMotor2 : std_LOGIC_VECTOR(7 downto 0):="00000000";

    signal DutyMotor3 : std_LOGIC_VECTOR(7 downto 0):="00000000";
signal DutyMotor4 : std_LOGIC_VECTOR(7 downto 0):="00000000";

    signal DutyMotor5 : std_LOGIC_VECTOR(7 downto 0):="00000000";
signal DutyMotor6 : std_LOGIC_VECTOR(7 downto 0):="00000000";
```

```
    signal DutyMotor7 : std_LOGIC_VECTOR(7 downto 0):="00000000";
signal DutyMotor8 : std_LOGIC_VECTOR(7 downto 0):="00000000";

    signal DutyMotor9 : std_LOGIC_VECTOR(7 downto 0):="00000000";
signal DutyMotor10 : std_LOGIC_VECTOR(7 downto 0):="00000000";

    signal DutyMotor11 : std_LOGIC_VECTOR(7 downto 0):="00000000";
signal DutyMotor12 : std_LOGIC_VECTOR(7 downto 0):="00000000";

    signal DutyMotor13 : std_LOGIC_VECTOR(7 downto 0):="00000000";
signal DutyMotor14 : std_LOGIC_VECTOR(7 downto 0):="00000000";

    signal DutyMotor15 : std_LOGIC_VECTOR(7 downto 0):="00000000";
signal DutyMotor16 : std_LOGIC_VECTOR(7 downto 0):="00000000";

    signal EncComm : std_LOGIC_VECTOR(7 downto 0):="00000000";

    signal valueTobeWritten : std_LOGIC_VECTOR(7 downto 0):="00000000";

    type WTD is (motorNumber, Motor1_write, Motor2_write);

    signal what_to_do : WTD := motorNumber;

    comPONENT pwm IS
GENERIC(
sys_clk : INTEGER := 50_000_000; –system clock frequency in Hz
pwm_freq : INTEGER := 100_000; –PWM switching frequency in Hz
bits_resolution : INTEGER := 8; –bits of resolution setting the duty cycle
phases : INTEGER := 1); –number of output pwms and phases
PORT(
clk : IN STD_LOGIC; –system clock
reset_n : IN STD_LOGIC; –asynchronous reset
ena : IN STD_LOGIC; –latches in new duty cycle
duty : IN STD_LOGIC_VECTOR(bits_resolution-1 DOWNTO 0); –duty cycle
pwm_out : OUT STD_LOGIC_VECTOR(phases-1 DOWNTO 0); –pwm outputs
pwm_n_out : OUT STD_LOGIC_VECTOR(phases-1 DOWNTO 0)); –pwm inverse outputs
```

```
END comPONENT pwm;

    COMPONENT I2c_slave IS
generic (
SLAVE_ADDR : std_logic_vector(6 downto 0):= "0000100");
port (
scl : inout std_logic;
sda : inout std_logic;
clk : in std_logic;
rst : in std_logic;
– User interface
read_req : out std_logic;
data_to_master : in std_logic_vector(7 downto 0);
data_valid : out std_logic;
data_from_master : out std_logic_vector(7 downto 0));
end component I2C_slave;

    BEGIN

    DutyMotor1(7 downto 0)<= world_state(1)(7 downto 0);
– Duty11 (7 downto 0)<= world_state(0)(7 downto 0);
DutyMotor2(7 downto 0)<= world_state(2)(7 downto 0) ;
–DutyMotor3(7 downto 0)<= world_state(2)(7 downto 0) ;
–DutyMotor4(7 downto 0)<= world_state(3)(7 downto 0) ;
–DutyMotor5(7 downto 0)<= world_state(4)(7 downto 0) ;

    – l5(0 downto 0) <= DutyMotor1(0 downto 0);

    –fpArray(0)(7 downto 0) <="00000001";
fpArray(1)(7 downto 0) <="00000010";
fpArray(2)(7 downto 0) <="00000011";
fpArray(3)(7 downto 0) <="00000100";
fpArray(4)(7 downto 0) <="10000101";

    –l5(0 downto 0) <= world_state(0)(0 downto 0);
```

```
–l5(0 downto 0) <= "1";

    task1: I2C_slave
GENERIC MAP(SLAVE_ADDR => ADDR)
PORT MAP(scl => scl_clock, sda => sda_data, clk => clk_main, rst => Rst,
read_req => reqRead,
data_to_master => i2c_data_wr , data_valid =>i2c_data_valid,
data_from_master=> i2c_data);

    Motor1pwm: pwm
GENERIC MAP(sys_clk => sys_clk1,
pwm_freq=> pwm_freq1, bits_resolution =>
bits_resolution1, phases => phases1)
PORT MAP(clk => clk_main, reset_n => ’1’, ena =>
’1’, duty => DutyMotor1(7 downto 0) , pwm_out =>
Motor1 –, pwm_n_out => Motor1_CW );

    Motor2pwm: pwm
GENERIC MAP(sys_clk => sys_clk1, pwm_freq=> pwm_freq1,
bits_resolution => bits_resolution1, phases => phases1)
PORT MAP(clk => clk_main, reset_n => ’1’, ena => ’1’, duty
=>DutyMotor2(7 downto 0) , pwm_out => Motor2 –, pwm_n_out => Motor1_CW
);

    Motor3pwm: pwm
GENERIC MAP(sys_clk => sys_clk1, pwm_freq=> pwm_freq1,
bits_resolution => bits_resolution1, phases => phases1)
PORT MAP(clk => clk_main, reset_n => ’1’, ena => ’1’, duty
=>DutyMotor3(7 downto 0) , pwm_out => Motor3 –, pwm_n_out =>
Motor1_CW );

    Motor4pwm: pwm
GENERIC MAP(sys_clk => sys_clk1, pwm_freq=> pwm_freq1,
bits_resolution => bits_resolution1, phases => phases1)
PORT MAP(clk => clk_main, reset_n => ’1’, ena => ’1’, duty
=>DutyMotor4(7 downto 0) , pwm_out => Motor4 –, pwm_n_out =>
```

```
Motor1_CW );

    Motor5pwm: pwm
GENERIC MAP(sys_clk => sys_clk1, pwm_freq=> pwm_freq1,
bits_resolution => bits_resolution1, phases => phases1)
PORT MAP(clk => clk_main, reset_n => '1', ena => '1', duty
=>DutyMotor5(7 downto 0) , pwm_out => Motor5 –, pwm_n_out =>
Motor1_CW );

    ————————————————————————————————————-
– Receive the world state from the master
————————————————————————————————————-
Receive : process (clk_main) is
begin
if rising_edge(clk_main) then

    case what_to_do is

    when MotorNumber =>
if i2c_data_valid = '1' then
EncComm <= i2c_data;

    if EncComm = "00000001" theN
what_to_do <= Motor1_write;
elsif EncComm = "00000010" theN
what_to_do <= Motor2_write;
else
null;
end if ;

    end if;

    ———————————
when Motor1_Write =>
```

```
    if i2c_data_valid = '1' then

    world_state(1)<= i2c_data;
Motor1_p1(0 downto 0) <= world_state(1)(0 downto 0);
Motor1_p2(0 downto 0) <= world_state(1)(1 downto 1);

    l5(0 downto 0) <= world_state(1)(0 downto 0);
– if l5 = '1' then

    –green_led <= '1';
–end if;
what_to_do <= MotorNumber ;

    end if;
```

```
    when Motor2_Write =>

    if i2c_data_valid = '1' then

    world_state(2)<= i2c_data;

    what_to_do <= MotorNumber ;
end if;
```

```
end case;
– –EncComm <= i2c_data;

–if EncComm = "11001100" then
–scaler <= scaler +1 ;

–else
– temp <= EncComm;
– world_state(scaler) <= EncComm;
–end if;
–end if;
```

```
–scaler <= scaler +1 ;
–Allow1 <= '1';
   – elsif i2c_data_valid = '1' and Allow = '0' then

EncComm <= i2c_data;

   if EncComm = "11001100" then
   Allow <= '1';
end if;
end if;
   if (scaler - prevscaler) = 1 then – 6 is right for 5 data
trasfer or 5 bytes of arrival

   valueTobeWritten <= EncComm ;
world_state(prevscaler) <= EncComm ;
prevscaler <=scaler;

   if scaler = 2 then
prevscaler <= 0;
–scaler <= 0;
Allow1 <= '1';
elsE
Allow1 <= '0';

   – end if;

   end if;
green_led <= Allow1;

   DutyMotor1(7 downto 0)<= temp ;

   world_state(scaler) <= temp ;

   DutyMotor1(7 downto 0)<= world_state(1)(7 downto 0);
Duty11 (7 downto 0)<= world_state(0)(7 downto 0);
```

```
–DutyMotor2(7 downto 0)<= world_state(2)(7 downto 0) ;
– DutyMotor3(7 downto 0)<= world_state(3)(7 downto 0) ;
–DutyMotor4(7 downto 0)<= world_state(4)(7 downto 0) ;
–DutyMotor5(7 downto 0)<= world_state(5)(7 downto 0) ;
green_led <= '1';

    end if;

    end process Receive;

    ————————————————————————————-
————————————————————————————-

    clk_1hz_process : process(clk_main, reset_btn)
begin
if (reset_btn = '0') then
clk_1hz <= '0';
scalerN <= 0;
elsif(rising_edge(clk_main)) then
if (scalerN < 25000000) then
scalerN <= scalerN + 1;
clk_1hz <= '0';
else
scalerN <= 0;
clk_1hz <= '1';
end if;
end if;
end process clk_1hz_process ;
blinking_process : process(clk_1hz, reset_btn)
begin
–if(reset_btn = '0')then
–LED <= '0';
if rising_edge(clk_1hz) then

    –LED <= not LED;
countm <= countm + 1;
```

```
end if;

    –green_led <= LED;
–std_logic_vector(to_unsigned(test,
i2c_data_wr’length))

    fpArray(0)(7 downto 0) <=
std_logic_vector(to_unsigned(countm,
fpArray(0)’length));

    end process blinking_process ;

– Write the data on I2C request from master

process (clk_main) is
begin
if rising_edge(clk_main) then
if reqRead = ’1’ then

    i2c_data_wr <= fpArray(scaler2);
scaler2 <= scaler2 + 1;
–end if;

    if scaler2 = 5 then – 6 I used for 5 data
scaler2 <= 0;
end if;
end if;

    end if;

    end process;
```

```
END logic;

library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
use work.txt_util.all;
```

```
entity I2C_slave is
generic (
SLAVE_ADDR : std_logic_vector(6 downto 0):= "0000100");
port ( scl : inout std_logic;
sda : inout std_logic;
clk : in std_logic;
rst : in std_logic;
– User interface read_req : out std_logic;
data_to_master : in std_logic_vector(7 downto 0);
data_valid : out std_logic;
data_from_master : out std_logic_vector(7 downto 0));
end entity I2C_slave;
```

```
architecture arch of I2C_slave is
type state_t is (i2c_idle, i2c_get_address_and_cmd,
i2c_answer_ack_start, i2c_write,
i2c_read, i2c_read_ack_start,
i2c_read_ack_got_rising, i2c_read_stop);
– I2C state management
signal state_reg : state_t := i2c_idle;
signal cmd_reg : std_logic := ’0’;
signal bits_processed_reg : integer range 0 to 8 := 0;
signal continue_reg : std_logic := ’0’;

– Helpers to figaure out next state
signal start_reg : std_logic := ’0’;
signal stop_reg : std_logic := ’0’;
signal scl_rising_reg : std_logic := ’0’;
```

```
signal scl_falling_reg : std_logic := ’0’;

   – Address and data received from master
signal addr_reg : std_logic_vector(6 downto 0) := (others => ’0’);
signal data_reg : std_logic_vector(7 downto 0) := (others => ’0’);

   – Delayed SCL (by 1 clock cycle, and by 2 clock cycles)
signal scl_reg : std_logic := ’1’;
signal scl_prev_reg : std_logic := ’1’;
– Slave writes on scl
signal scl_wen_reg : std_logic := ’0’;
signal scl_o_reg : std_logic := ’0’;
– Delayed SDA (1 clock cycle, and 2 clock cycles)
signal sda_reg : std_logic := ’1’;
signal sda_prev_reg : std_logic := ’1’;
– Slave writes on sda
signal sda_wen_reg : std_logic := ’0’;
signal sda_o_reg : std_logic := ’0’;

   – User interface
signal data_valid_reg : std_logic := ’0’;
signal read_req_reg : std_logic := ’0’;
signal data_to_master_reg : std_logic_vector(7 downto 0) := (others => ’0’);
begin
process (clk) is
begin
if rising_edge(clk) then
– Delay SCL by 1 and 2 clock cycles
scl_reg <= scl;
scl_prev_reg <= scl_reg;
– Delay SDA by 1 and 2 clock cycles
sda_reg <= sda;
sda_prev_reg <= sda_reg;
– Detect rising and falling SCL
scl_rising_reg <= ’0’;
if scl_prev_reg = ’0’ and scl_reg = ’1’ then
```

```
scl_rising_reg <= '1';
end if;
scl_falling_reg <= '0';
if scl_prev_reg = '1' and scl_reg = '0' then
scl_falling_reg <= '1';
end if;

    – Detect I2C START condition
start_reg <= '0';
stop_reg <= '0';
if scl_reg = '1' and scl_prev_reg = '1' and
sda_prev_reg = '1' and sda_reg = '0' then
start_reg <= '1';
stop_reg <= '0';
end if;

    – Detect I2C STOP condition
if scl_prev_reg = '1' and scl_reg = '1' and
sda_prev_reg = '0' and sda_reg = '1' then
start_reg <= '0';
stop_reg <= '1';
end if;

    end if;
end process;

    ——————————————————————————-
– I2C state machine
——————————————————————————-
process (clk) is
begin
if rising_edge(clk) then
– Default assignments
sda_o_reg <= '0';
sda_wen_reg <= '0';
– User interface
```

```
data_valid_reg <= '0';
read_req_reg <= '0';

    case state_reg is

    when i2c_idle =>
if start_reg = '1' then
state_reg <=
i2c_get_address_and_cmd;
bits_processed_reg <= 0;
end if;

    when i2c_get_address_and_cmd =>
if scl_rising_reg = '1' then
if bits_processed_reg < 7 then
bits_processed_reg <=
bits_processed_reg + 1;
addr_reg(6-bits_processed_reg) <= sda_reg;
elsif bits_processed_reg = 7 then
bits_processed_reg <= bits_processed_reg + 1;
cmd_reg <= sda_reg;
end if;
end if;

    if bits_processed_reg = 8 and scl_falling_reg = '1' then
bits_processed_reg <= 0;
if addr_reg = SLAVE_ADDR then – check req address
state_reg <= i2c_answer_ack_start;
if cmd_reg = '1' then – issue read request read_req_reg <= '1';
data_to_master_reg <= data_to_master;
end if;
else
assert false
report ("I2C: slave address: " &
str(SLAVE_ADDR) &
", requested address: " & str(addr_reg))
```

```
severity note;
state_reg <= i2c_idle;
end if;
end if;

    ————————————————————————-
– I2C acknowledge to master
————————————————————————-
when i2c_answer_ack_start =>
sda_wen_reg <= ’1’;
sda_o_reg <= ’0’;
if scl_falling_reg = ’1’ then
if cmd_reg = ’0’ then
state_reg <= i2c_write;
else
state_reg <= i2c_read;
end if;
end if;

    ————————————————————————-
– WRITE
————————————————————————-
when i2c_write =>
if scl_rising_reg = ’1’ then
if bits_processed_reg <= 7 then
data_reg(7-bits_processed_reg) <= sda_reg;
bits_processed_reg <=
bits_processed_reg + 1;
end if;
if bits_processed_reg = 7 then
data_valid_reg <= ’1’;
end if;
end if;

    if scl_falling_reg = ’1’ and bits_processed_reg = 8 then
state_reg <= i2c_answer_ack_start;
```

```
bits_processed_reg <= 0;
end if;

——————————————————————-
– READ: send data to master
——————————————————————-
when i2c_read =>
sda_wen_reg <= '1';
sda_o_reg <= data_to_master_reg(7-
bits_processed_reg);
if scl_falling_reg = '1' then
if bits_processed_reg < 7 then
bits_processed_reg <= bits_processed_reg + 1;
elsif bits_processed_reg = 7 then
state_reg <= i2c_read_ack_start;
bits_processed_reg <= 0;
end if;
end if;

——————————————————————-
– I2C read master acknowledge
——————————————————————-
when i2c_read_ack_start =>
if scl_rising_reg = '1' then
state_reg <= i2c_read_ack_got_rising;
if sda_reg = '1' then – nack = stop read
continue_reg <= '0';
else – ack = continue read
continue_reg <= '1';
read_req_reg <= '1';
– request reg byte
data_to_master_reg <= data_to_master;
end if;
end if;
```

```
when i2c_read_ack_got_rising =>
if scl_falling_reg = '1' then
if continue_reg = '1' then
if cmd_reg = '0' then
state_reg <= i2c_write;
else
state_reg <= i2c_read;
end if;
else
state_reg <= i2c_read_stop;
end if;
end if;

– Wait for START or STOP to get out of this state
when i2c_read_stop =>
null;

– Wait for START or STOP to get out of this state
when others =>
assert false
report ("I2C: slave address: " &
str(SLAVE_ADDR) &
"ended up in an impossible state.")
severity note;
null;
end case;

————————————————————————————
– Reset counter and state on start/stop
————————————————————————————
if start_reg = '1' then
state_reg <= i2c_get_address_and_cmd;
bits_processed_reg <= 0;
end if;
```

```
if stop_reg = '1' then
state_reg <= i2c_idle;
bits_processed_reg <= 0;
end if;

if rst = '1' then
state_reg <= i2c_idle;
end if;
end if;
end process;

——————————————————————————-
– I2C interface
——————————————————————————-
sda <= sda_o_reg when sda_wen_reg = '1' else 'Z';
scl <= scl_o_reg when scl_wen_reg = '1' else 'Z';
——————————————————————————-
– User interface
——————————————————————————-
– Master writes
data_valid <= data_valid_reg;
data_from_master <= data_reg;
– Master reads
read_req <= read_req_reg;
end architecture arch;

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_unsigned.all;

ENTITY pwm IS
GENERIC(
sys_clk : INTEGER := 50_000_000;
–system clock frequency in Hz pwm_freq : INTEGER := 100_000;
–PWM switching frequency in Hz bits_resolution : INTEGER := 8;
–bits of resolution setting the duty cycle phases : INTEGER := 1);
```

–number of output pwms and phases PORT(clk : IN STD_LOGIC;
–system clock reset_n : IN STD_LOGIC;
–asynchronous reset ena : IN STD_LOGIC;
–latches in new duty cycle duty : IN STD_LOGIC_VECTOR(bits_resolution-1 DOWNTO 0);
–duty cycle pwm_out : OUT STD_LOGIC_VECTOR(phases-1 DOWNTO 0);
–pwm outputs pwm_n_out : OUT STD_LOGIC_VECTOR(phases-1 DOWNTO 0));
–pwm inverse outputs END pwm;

ARCHITECTURE logic OF pwm IS
CONSTANT period : INTEGER := sys_clk/
pwm_freq;
–number of clocks in one pwm period
TYPE counters IS ARRAY (0 TO phases-1) OF
INTEGER RANGE 0 TO period - 1;
–data type for array of period counters
SIGNAL count : counters := (OTHERS => 0);
–array of period counters
SIGNAL half_duty_new : INTEGER RANGE 0 TO period/2 := 0;
–number of clocks in 1/2 duty
cycle TYPE half_duties IS ARRAY (0 TO phases-1) OF
INTEGER RANGE 0 TO period/2;
–data type for array of half duty values
SIGNAL half_duty : half_duties := (OTHERS => 0);
–array of half duty values (for each phase)
BEGIN
PROCESS(clk, reset_n)
BEGIN
IF(reset_n = ’0’) THEN
–asynchronous reset
count <= (OTHERS => 0);
–clear counter
pwm_out <= (OTHERS => ’0’);
–clear pwm outputs
pwm_n_out <= (OTHERS => ’0’);
–clear pwm inverse outputs

```
ELSIF(clk'EVENT AND clk = '1') THEN
–rising system clock edge
IF(ena = '1') THEN
–latch in new duty cycle
half_duty_new <=
conv_integer(duty)*period/
(2**bits_resolution)/2;
–determine clocks in 1/2 duty
cycle END IF;
FOR i IN 0 to phases-1 LOOP
–create a counter for each phase
IF(count(0) = period - 1 - i*period/phases) THEN
–end of period reached
count(i) <= 0;
–reset counter
half_duty(i) <= half_duty_new;
–set most recent duty cycle value
ELSE
–end of period not reached
count(i) <= count(i) + 1;
–increment counter
END IF;
END LOOP;
FOR i IN 0 to phases-1 LOOP
–control outputs for each phase
IF(count(i) = half_duty(i)) THEN
–phase's falling edge reached
pwm_out(i) <= '0';
–deassert the pwm output
pwm_n_out(i) <= '1';
–assert the pwm inverse output
ELSIF(count(i) = period - half_duty(i)) THEN
–phase's rising edge reached
pwm_out(i) <= '1';
–assert the pwm output
pwm_n_out(i) <= '0';
```

```
–deassert the pwm inverse output
END IF;
END LOOP;
END IF;
END PROCESS;
END logic;
```

Appendix B

ORI: Open Robot Interface

ORI is developed by Loccioni Group [36] and had been used in Bayer Robotic challenge.

Loccioni Group, in order to increase the flexibility and modularity of its robotic cells, has developed a control framework for robotic systems, called 'Open Robot Interface' - ORI - that allows to control a generic robot based on external hardware, different from the robot controller itself (e.g. embedded pc connected to the robot controller by Ethernet cable).

The basic idea is not to merely replace the robot controller, as it will manage low-level and safety functions; however, high-level robot functions (e.g. motion control and trajectory generation) will be managed by the external hardware.

The 'Open Robot Interface' framework acts as a middle software layer, encapsulating all the high- level motion and programming functions of the robot, translating them into low-level commands that can be executed by the specific robot controller.

This permits the easy reconfiguration of the robotic cell without being dependent on a particular robot manufacturer, as the application-related robot program (being executed at the external hardware level) will be independent from the individual robot model. As a consequence, it will be possible to change the robot without the need of rewriting the source code (the same software code can be fully re-used). In addition, the reconfiguration of the cell will be eased, also in case of modifications to the layout. For example, in case of multi-robot applications, multiple robots can be easily integrated into the robotic cell, as the framework can efficiently manage the co-activity of the different robots.