

Towards a Generic Grasp Planning Pipeline using End-Effector Specific Primitive Grasping Actions

Liana Bertoni , Davide Torielli , Yifang Zhang , Nikos G. Tsagarakis , and Luca Muratore

Humanoids and Human Centered Mechatronics (HHCM), Istituto Italiano di Tecnologia, Genova, Italy
{liana.bertoni, davide.torielli, yifang.zhang, nikos.tsagarakis, luca.muratore}@iit.it

Abstract—In the past few years, several robotic end-effectors based on diverse kinematics and actuation principles have been developed to provide grasping and manipulation functionalities. To ease the control and application of these wide-ranging end-effectors, the development of effective reusable tools that can facilitate the end-effector motion planning and control is necessary. In this work, we introduce a generic grasp planner that leverages on the concept of the primitive grasping actions. Given the specific characteristics of an end-effector, including its kinematic and actuation arrangements, a number of primitive grasping actions are extracted and employed by the proposed grasp planner to autonomously plan and synthesize more complex grasping behaviours. The grasp planner is validated through experimental trials involving the HERI II robotic hand, a four-fingers tendon-driven under-actuated hand. The results of these experiments demonstrate the efficacy of the proposed method to generate appropriate planning actions enabling to grasp objects of different shapes.

I. INTRODUCTION

To enhance the manipulation and grasping capabilities of robotics systems, several robotics end-effectors, ranging from simple grippers to sophisticated bio-inspired robotic hands [1], have been realized in the past. In general, the grasping and manipulation motion generation and control of these robotic end-effectors relies on methodologies based on analytical [2], or data-driven solutions [3]. Quite often, analytical solutions does not exploit end-effector motion capabilities and feasibility to reach a desired pose due to an object-centered policy. On the other side, data-driven approaches build solutions for a specific robotic hand or gripper lacking in flexibility and portability when a new end-effector has to be controlled.

The lack of a unified approach that can generically address the grasp planning on different end-effectors eventually represents a barrier for the use of new end-effectors in existing robotic and automation systems requiring extensive effort to provide a full-fledged grasping system. As a result, current industrial solutions make use of a limited number of end-effectors systems, potentially missing the opportunity to explore the richer capabilities of more sophisticated robotic hands.

The development of more generic grasp planning and control tools, which can address different robotic end-effectors is required to permit the exploration of these end-effectors in industrial applications.

In [4], the ROS End-Effector software framework is presented: it introduces the concept of primitive grasping



Fig. 1. Three fingers grasp performed during the experiments of the proposed grasp planner. The experiments have been conducted with HERI II Hand mounted on a robotic arm.

actions needed to abstract the hardware and the kinematics of different end-effectors and allowing the user to manually command a grasp pose. Herein, to turn it into a fully automatic control system, we propose a new generic grasp planner able to compute grasp poses for different robotics end-effectors leveraged on the concept of primitive grasping actions. Thanks to the above, the proposed grasp planner is capable to abstract the end-effector hardware in use, not looking at the particular end-effector instructions but rather focusing on the grasp capabilities of the specific end-effector involved.

In fact, given an arbitrary end-effector, depending on its motion capabilities, a series of primitive grasping actions are available and suitably combined by the proposed planner to grasp an object.

The main contributions of this work are:

- The development of a new generic grasp planner leveraging on the primitive grasping actions concept, capable to abstract the specific end-effector hardware and plan at a more high-level.
- The proposed generic grasp planner can generate grasps for a given object, by composing more complex grasping poses starting from the primitive grasping actions available for the specific end-effector in use without to consider the embodiment instructions. The planning process is addressed in two phases. In the first phase, the problem of finding and reaching the contact points

locations is addressed. Then, in the second phase the necessary primitive grasping actions required to grasp the object are synthesized and the final composed action determined.

- The grasp planner enables to achieve a transparent and generic grasp planning procedure, which can automatically plan grasping actions in an agnostic way with respect to the specific end-effector kinematic structure.

The proposed grasp planner is validated in a number of experiments conducted on the HERI II end-effector, a four fingers tendon-driven under-actuated robotic hand [5], shown in Figure 1. The rest of the paper is organized as follows. Section II presents the related work, Section III introduces the primitive grasping action principle and with more details the primitive grasping actions model is presented in Section IV. Then, in Section V the hand-object model to plan primitives grasping actions is described. The primitive grasping actions planner is described in the Section VI. Lastly, Section VII introduces the experiments while the conclusions are drawn in Section VIII.

II. RELATED WORK

A grasp is commonly defined as a set of contacts on the object surface, targeting to constrain the potential movements of the object in the presence of external disturbances or loads. The literature dealing with the grasping problem is extensive and well known [6],[7], and it can be divided in grasp analysis and grasp synthesis.

An exhaustive discussion on existing algorithms is reported in [8],[9] from more analytical approaches guided by objects shape to data-driven solutions. Grasp taxonomy [11] has also been investigated and used in the online phase to select the more proper grasp pose driven by object and task descriptions. Since an object can be grasped assuming different poses, grasp quality measures were employed to enable robots to select the best grasp pose. Main qualities are reviewed in [10].

Some approaches make use of human expertise building data-sets to teach the robot how to grasp objects [12]. From human hand motion studies [13], it has also been demonstrated that only two motion synergy components could account for 80% of the variance of the grasps performed, implying a substantial reduction of the grasp complexity. Dimensionality reduction was adopted in several works to model and control multi-fingers robotics hands as in [14] and [15].

Contrary to the existing solutions, in our work we aim to provide a grasp planner able to synthesize grasp pose, which are composed by an a number of grasping primitive actions that are available on a robotic end-effector given its specific kinematics and hardware implementation. Therefore, these primitive grasping actions are extracted and used by the proposed planner to compose desired complex grasps.

III. PRIMITIVE GRASPING ACTIONS

Primitive grasping actions encode essential fingers movements. In particular, a primitive grasping action is a funda-

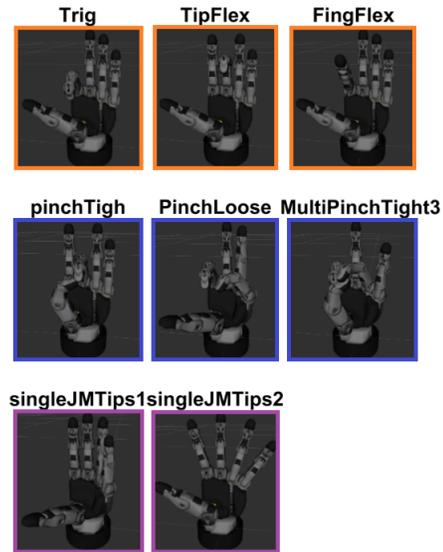


Fig. 2. An example of primitive grasping actions extracted from the Schunk SVH Hand. From top to bottom: trig-type(orange), pinch-type (blue) and singleJointMultipleTips(purple) primitive grasping actions.

mental motion of the end-effector’s elements, that can not be decomposed into smaller primitives. These movements, powered by the actuators, are the particular end-effector’s mean to grasp an object. We recognize three categories of primitive grasping actions: *trig-type*, *pinch-type* and *singleJointMultiTips*.

- The trig-type category includes: *Trig*, *TipFlex*, and *FingFlex*, which move a single finger or a phalanx toward the palm.
- The pinch-type category includes: *PinchTight*, *PinchLoose*, and *MultiPinchTightN*. They move the fingers towards each other to form a grasp pose.
- The *singleJointMultiTips* category is related to grasping primitives generated when a single actuator moves $N(\geq 2)$ fingertips.

An illustration of the primitive grasping actions for the Schunk SVH hand¹ is shown in figure 2.

As humans learn how to grasp through trails, primitive grasping actions are extracted performing an iterative procedure. Fundamental movements are identified and encapsulated as a collection of actuators involved in a specific primitive grasping action while exploring end-effector kinematics.

Consequently, given an arbitrary end-effector, a set of primitive grasping actions is available for it and any grasping configuration can be expressed using primitive grasping actions. For example, a power grasp can be seen as a composition of primitive grasping actions. In such a way, a general posture of the end-effector is mapped into primitive grasping actions without considering the particular mechanical implementation and actuation involved which can generate that posture.

¹https://schunk.com/it_en/gripping-systems/highlights/svh/

IV. PRIMITIVE GRASPING ACTIONS MODEL

Based on the specific end-effector embodiment, a set of available primitives can be extracted [4]. Hereafter, primitives will replace primitive grasping actions mentioned in the previous sections.

Each primitive $v_i \in \mathbb{R}^{n_{v_i}}$ with $i = 1, 2, \dots, N_v$, where N_v is the number of primitives associated to a generic end-effector and n_{v_i} is the dimension of the i -th primitive, can be collected as below,

$$v = [v_1, v_2, \dots, v_{N_v}]^T \in \mathbb{R}^{n_v}. \quad (1)$$

n_v is the sum of the dimensions of each primitive vector. The primitives generate essential movements of an end-effector, and this can lead to involved a number of actuators fewer than available, therefore $n_{v_i} \leq n_a$. Assuming a linear relationship between the primitives and a generic posture of the end-effector, defined through its actuated joints, the following relationship is derived

$$q_a = P\eta, \quad (2)$$

where

$$P = \begin{bmatrix} | & | & & | \\ v_1 & v_2 & \dots & v_{N_v} \\ | & | & & | \end{bmatrix} \in \mathbb{R}^{n_a \times N_v}$$

is the matrix formed by the primitive vectors, called Primitive matrix, and

$$\eta = \begin{bmatrix} \alpha_1 \beta_1 \\ \alpha_2 \beta_2 \\ \vdots \\ \alpha_{N_v} \beta_{N_v} \end{bmatrix} \in \mathbb{R}^{N_v}$$

represents the *intention* and *intensity* of a primitive. The coefficient α_i , the intention, provides the scale of how much a singular primitive contributes to generate a particular end-effector posture. The coefficient β_i , the intensity, indicates the percentage of the primitive activated. A primitive encodes finger movements, and can be regulated from an initial position, fully opened (corresponding to 0%), to final position (100%). Therefore, given an intention-intensity vector η , a general posture of the hand is completely determined through the eq (2).

V. HAND-OBJECT SYSTEM MODELS

This section briefly introduces the modeling of hand, object and equilibrium equations involved to model the hand-object system in the proposed grasping planner.

A. Hand Model

To demonstrate the proposed generic grasp planner tool, in this work we make use of the HERI II end-effector, a four fingers robotic hand. HERI II is an under-actuated end-effector with each finger powered by a single actuator through a uni-directional acting tendon transmission combined with elastic return elements. Contact force exerted by or on a finger can be related through the transpose of the Jacobian matrix

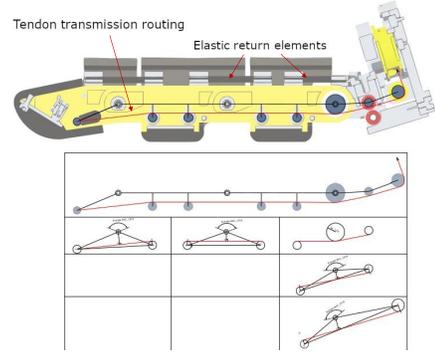


Fig. 3. Heri-II under-actuated finger. The under-actuated mechanism of the finger is shown. The mechanism is a modified Da Vinci's Mechanism. A finger is powered by a single actuator through a uni-directional acting tendon transmission combined with elastic return elements.

$J^T \in \mathbb{R}^{n_q \times n_c n_l}$ (n_q is the number of finger joints) with the torques generated by these contact forces $\tau_c \in \mathbb{R}^{n_q}$ as follows

$$\tau_c = J^T f_c \quad (3)$$

with $f_c \in \mathbb{R}^{n_c n_l}$ and n_l is the dimension of the force vector, and n_c is the number of contact points. In order to balance the torque generated by the contact forces, we adopt the model used in [16] from which we obtain the following equation

$$\tau_{in} = T^T \tau_c. \quad (4)$$

$\tau_{in} \in \mathbb{R}^{n_q}$ is the torque generated by the actuators and eventual passive element present in the finger (n_q is the number of the finger joint) and $T \in \mathbb{R}^{n_q \times n_q}$ is the Transmission matrix. For a fully-actuated finger, the Transmission matrix is equal to Identity matrix. Therefore, given an under-actuated finger, as shown in figure 3, we use the adopted model (equation 4) to generate the required torques to balance the external contact forces.

Considering a proportional controller at the actuated joints

$$\tau_a = K_a(q_a^t - q_a) \quad (5)$$

where $K_a \in \mathbb{R}^{n_a \times n_a}$ is the actuators controller stiffness matrix, the amount of contact forces can be related with the actuator joint position using the equations (3), (4) and (5).

B. Grasp Model

In order to grasp an object, the contact forces vector f_c must satisfy the equilibrium equation

$$Gf_c = w \quad (6)$$

that relates the contact forces with an external load $w \in \mathbb{R}^6$ acting on the object at its center of mass (COM), through the Grasp matrix $G \in \mathbb{R}^{6 \times n_c n_l}$ [17]. For each contact, a contact model is assumed. In our work, we consider a contact point with friction (CPWF) model, involving a selection matrix $B_i \in \mathbb{R}^{n_l \times 6}$ properly defined.

To avoid slippage at the contact, each contact force f_c^i must lie inside the friction cone generated, so the following constraint need to be satisfied

$$f_c^i \leq \mu f_c^{n_i}. \quad (7)$$

f_c^i and f_n^i are the friction and normal forces of the i -th contact force, respectively and μ is the friction coefficient of the contacting materials. For computational purposes, the friction cone is approximated by an inscribed regular polyhedral cone with m -faces. The wrenches generated by forces along the edge of the discretized friction cone are referred to as primitive wrenches.

C. Object Model

An object can be represented with different kind of approximated models. In our work, we represent objects as superquadrics. First investigated in [18], the superquadric surfaces can represent a wide range of smooth objects, with a smooth transition between them, leading in a versatile representation of the object.

Given the spherical product of two two-dimensional curves, each point of superquadric surface can be computed as

$$x(v, \omega) = \begin{pmatrix} a_1 \cos^{\varepsilon_1} v \cos^{\varepsilon_2} \omega \\ a_2 \cos^{\varepsilon_1} v \sin^{\varepsilon_2} \omega \\ a_3 \sin^{\varepsilon_1} v \end{pmatrix} \quad (8)$$

with $\pi/2 \leq v \leq \pi/2$ and $\pi \leq \omega \leq \pi$. The parameters a_1 , a_2 and a_3 determines the superquadric size and ε_1 and ε_2 determines its particular shape. A superquadric can be expressed through its implicit function ("inside-outside" function)

$$F(x, y, z) = \left(\left(\frac{x}{a_1} \right)^{\frac{2}{\varepsilon_2}} + \left(\frac{y}{a_2} \right)^{\frac{2}{\varepsilon_2}} \right)^{\frac{\varepsilon_2}{\varepsilon_1}} + \left(\frac{z}{a_3} \right)^{\frac{2}{\varepsilon_1}}. \quad (9)$$

A point lying on the surface of the superquadric returns: $F(x, y, z) = 1$, if lies on the surface, $F(x, y, z) > 1$ if outside and $F(x, y, z) < 1$ if inside.

A superquadric is uniquely determined given the following vector

$$\Lambda = [a_1, a_2, a_3, \varepsilon_1, \varepsilon_2, p_x, p_y, p_z, \phi, \theta, \psi]^T \in \mathbb{R}^{11},$$

where (p_x, p_y, p_z) and (ϕ, θ, ψ) are position and orientation (Euler angles) of the superquadric, respectively.

This representation enables our planner to fit well with real applications where a vision or sensor system provides object input data.

VI. PRIMITIVE GRASPING ACTIONS PLANNER

In this section we present the proposed generic primitive grasping action planner. The planning process is divided in two main phases: pre-grasp and grasp phase. In the pre-grasp phase (VI-A), the levels of intention and intensity of each primitive are computed in order to reach desired contact points. Contact points are established in according with Force-Closure (FC) property of a grasp given object representation and pose of the end-effector exploring fingers workspaces. In the grasp phase (VI-B), the intention and intensity levels of each primitive are further regulated to maintain and grasp the object by solving force distribution problem [19]. Summing both, pre-grasp and grasp contributions, the final action is sent to the end-effector. In figure 4, planner pipeline is shown. Given all possible poses of the

end-effector, the procedure can be easily iterated and the best grasp selected by a grasp quality metric. The entire process is explained for a particular pose of the end-effector.

A. Pre-grasp Phase

1) Superquadric and Fingers Workspaces Intersection:

To compute the contact points locations around the object, we perform an intersection check between the surface of the object, represented by a superquadric model, and the fingers workspaces.

Given a robotic hand or gripper with a number of fingers equals to n_f , we determine a set of fingers workspaces as follows

$$\Psi = \{\Psi_1, \Psi_2, \dots, \Psi_{n_f}\}. \quad (10)$$

Then, given a known pose of the hand, through the transformation matrix ${}^{\mathcal{W}}T_{\mathcal{P}} \in \mathbb{R}^{4 \times 4}$ from a frame attached in a known point on the palm \mathcal{P} to a fixed frame \mathcal{W} attached somewhere, the fingers workspaces can be defined through a collection of fingers poses

$$\Psi_i = \{T_1, T_2, \dots, T_{n_{\Psi_i}}\} \quad (11)$$

where $T_i = ({}^{\mathcal{P}}T_{\mathcal{F}})_i$ gives the transformation matrix from a frame \mathcal{F} attached in a known point of the finger to the frame attached to the palm \mathcal{P} . This set contains n_{Ψ_i} finger poses with i indicating the i -th finger involved. The number of finger poses is equal to the number of samples used to discretize the finger workspace. The discrete fingers workspaces are computed sampling the actuated joint space and the fingers poses obtained using transmission matrix and forward kinematics. In the figure 5, the fingertips workspaces for HERI II hand are shown.

Given an object described with a set of parameters Λ and a frame \mathcal{O} attached in its center with a certain position and orientation with respect to a fixed frame \mathcal{W} , we determine a set of Independent Contact Regions (ICRs)

$$\mathcal{R} = \{\mathcal{R}_1, \mathcal{R}_2, \dots, \mathcal{R}_{n_r}\} \quad (12)$$

along the surface of the object. ICRs are determined through an intersection between fingers workspaces Ψ_i and the surface of the object S . The intersection is addressed via classification algorithm. Each point of the fingers workspaces $p_i \in \Psi$ is evaluated through the inside-outside function, eq. (9). Using this function, each point p_j of the fingers workspaces is classified as inside p_j^i , outside p_j^o or lying on the surface of the object S p_j^s . Then, an ICR \mathcal{R}_i is formed by the points lying on the surface of the object

$$\mathcal{R}_i = \{p_1^s, p_2^s, \dots, p_{n_s}^s\}. \quad (13)$$

If no abduction is present in the fingers, the ICRs is diminished to a single point. In figure 6, ICRs formed by only a point are shown. Since we are not focusing in finding workspaces, we assume that the fingers workspaces have no intersections between them, i.e., there are no points on the object surface which could belong to more than one finger workspace. If fingers workspaces overlap, opportune strategies have to be adopted in order to avoid it. In addition, we consider fingertip grasps.

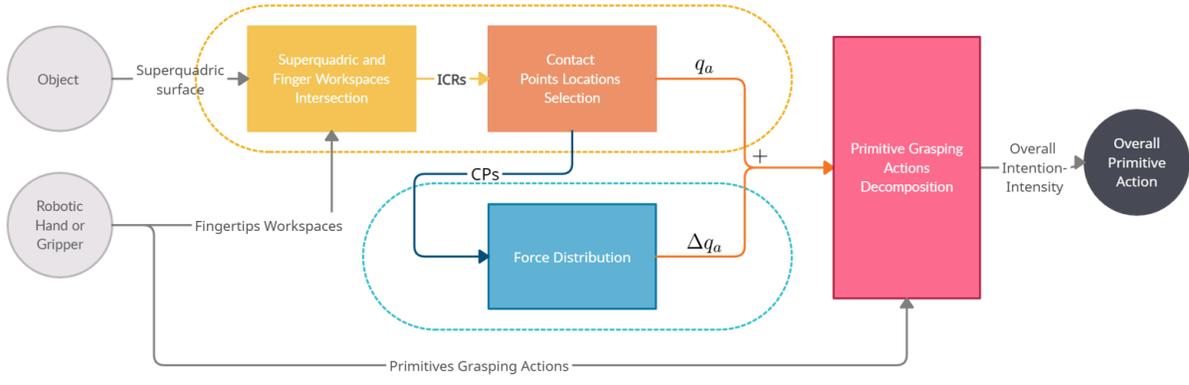


Fig. 4. Grasp Planner Pipeline showing the pre-grasp(top) and grasp(bottom) phases. The pre-grasp phase consists of superquadric and independent contact regions intersection and contact points locations selection. Once given as an input the superquadric parameters, the pre-grasp phase computes the contact points locations and motors positions as outputs. The grasp phase consists in solving a force distribution problem and generate the necessary motors positions to generate such a forces. Once both phases are completed, the motors positions obtained are decomposed in intention and intensity levels for the grasping primitives building up the overall primitive action.

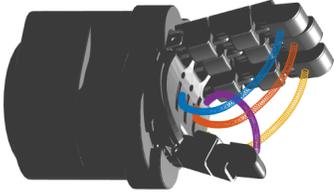


Fig. 5. Fingertips workspaces. For each finger, the fingertip workspace Ψ_i is shown colored in blue, red, yellow and violet.

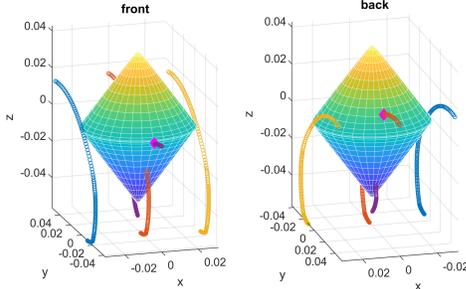


Fig. 6. Independent Contact Regions. All points, which belong to fingertips workspaces, are categorized with eq. (9).

2) *Contact Points Locations Selection*: Given a set of ICRs \mathcal{R} , the contact points (CPs) locations will be selected inside the set. Since we select CPs in order to determine poses which are grasps, further constraints on ICRs must be imposed. If $n_r \leq 1$, i.e. there is only one finger touching the object, and/or the ICR related to the opposing finger (thumb) do not belong to the set of ICRs, i.e. the thumb is not touching, the set is not considered valid for generating grasp poses. With only one finger, the object is not restrained as well as if the set of ICRs does not contain ICR of the opposing finger.

Once these constraints are satisfied, CPs selection is achieved via optimization problem (14). Within the optimization problem, contact points are validated through FC

condition.

In this work, FC condition is verified as follows: the origin of the primitive wrench space has to strictly lie inside the Convex Hull (CH) formed by the primitive contact wrenches [20]. This condition represents a necessary and sufficient condition for existence of a FC grasp.

$$\begin{aligned} \min_{\lambda_i, z_i} \quad & \|q - \sum_i^{n_c} (\lambda_i w_i) z_i\| \\ \text{s.t.} \quad & \sum_i^{n_c} \lambda_i = 1, \lambda_i > 0 \\ & \sum_{i=1}^{n_{R_i}} z_i = 1, z_i \in \{0, 1\}. \end{aligned} \quad (14)$$

The optimization problem determines the best combination of contact points locations available in the set \mathcal{R} which can generate a FC grasp. Considering one contact point for each contact region, we search for a set

$$\Omega_c = \{p_c^1, p_c^2, \dots, p_c^{n_c}\} \quad (15)$$

of CPs with $n_c = n_r$. Considering a contact wrench w_i applied in each point $p_i^s \in \mathcal{R}$ of the ICRs, the best combination of contact points selected by the optimization problem, is the combination of contact wrenches with the centered largest ball inscribed in the Convex Hull $CH(W)$ generated. In the optimization problem, the distance of the convex hull $CH(W)$ from the origin is evaluated through the objective function and λ_i represents the coefficient of the CH [21]. If there are not combinations where the origin strictly lies inside the $CH(W)$, the optimization problem returns the combination with the convex hull nearest to the origin, which is the closer combination to obtain the FC property. A contact wrench is written as a linear combination of primitive contact wrenches w_{ij}

$$w_i = \sum_{j=1}^{n_m} a_{ij} w_{ij}, \quad a_{ij} > 0 \quad (16)$$

which linearly approximates the friction cone with a number of n_m primitive contact wrenches. In the optimization

problem, the combination is determined with the use of binary variables, $z \in \mathbb{Z}^{n_c}$, leading the problem to be a non linear mixed-integer optimization problem. Using the big M-method similar to [22], the problem has been cast in a linear optimization problem and used in the computation.

3) *Primitive Grasping Actions Optimal Decomposition:* Given a general posture of the hand, we want to decompose it into primitives. Considering the equation (2), where given a vector of motor positions q_a which defines a generic posture of the hand, a vector of intention and intensity η is determined in according with the Primitive matrix P . Therefore, we define the following optimization problem

$$\begin{aligned} \min_{\eta} \quad & \|P\eta - q_a\|_2^2 \\ \text{s.t.} \quad & \eta^L \leq \eta \leq \eta^U. \end{aligned} \quad (17)$$

The solution vector is minimized following the 2-norm, and the constraint represents the feasibility of the solution within a lower and upper bound. Equivalently, the minimization can follow the 1-norm

$$\begin{aligned} \min_{\eta} \quad & \|\eta\|_1 \\ \text{s.t.} \quad & q_a \leq P\eta \leq q_a \\ & \eta^L \leq \eta \leq \eta^U. \end{aligned} \quad (18)$$

leading to a more sparse solution. We use the 1-norm solution to compute the solution in order to involve a less number of primitive to decompose a general posture.

B. Grasp Phase

1) *Force Decomposition:* Given a set of reachable contact points Ω_c around the surface of the object, the corresponding posture of the hand can be determined. In our planning, the contact points of the set Ω_c are reachable points for the hand, in fact they belong to fingers workspaces. Then, given a set of contact points reachable by the hand Ω_c^R , the contact forces $f_c \in \mathbb{R}^{n_c}$ generated to hold the object can effectively be performed by the hand. Completed the pre-grasp phase, the desired contact forces required to grasp the object are given by the force decomposition problem. The problem of the contact forces decomposition is accomplished using the static force equilibrium equation (6) as below

$$\begin{aligned} \min_{f_c} \quad & \|Gf_c - w\|^2 \\ \text{s.t.} \quad & q_c^L \leq Q_c f_c \leq q_c^U \\ & \tau^L \leq J^T f_c \leq \tau^U \\ & f_c^L \leq f_c \leq f_c^U \end{aligned} \quad (19)$$

The optimization problem above solves the force distribution considering contact constraints. Contact forces f_c must satisfy friction cone constraint, eq (7), for avoiding slippage at the contact. Feasibility conditions are also taken into account by the problem. The constraint of friction cone is written considering its linear approximation and the normal force can only exert positive forces $f_c^{t_i} > 0$, i.e., only push is admitted. Once the contact forces are determined, the motors positions can be computed using the equations (4), (3) and (5). The

intention and intensity of the primitives are then computed via primitive grasping actions decomposition solving (18).

VII. EXPERIMENTS

To demonstrate the effectiveness of our approach, the algorithm has been implemented and tested by grasping three different objects with three types of grasps using the HERI II robotic hand.

A. Experiments Setup

To perform our experiments, we use the HERI II Hand mounted on a robotic arm controlled using the XBot [23] software architecture. The objects involved in the experiments are shown in figure 7. We will refer to them as object 01, object 02 and object 03. Their

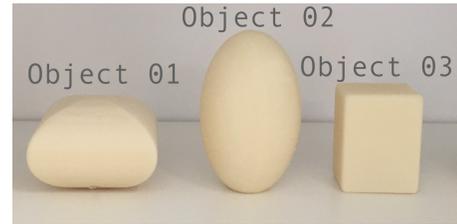


Fig. 7. Objects used for experiments. In order we have: object 01, object 02 and object 03. The superquadric parameters related to the objects are: $\Lambda_{SQ}^{01} = [3.25, 3.75, 2.7, 1.0, 0.2]$, $\Lambda_{SQ}^{02} = [2.89, 3.0, 4.825, 1.0, 1.0]$, $\Lambda_{SQ}^{03} = [1.75, 2.25, 3.0, 0.1, 0.1]$.

poses are expressed with respect to the frame attached to the palm of the hand ${}^{\mathcal{P}}T_{SQ}$ defined through its minimal representation (positions and Euler-Angles), given by $\Lambda_{\mathcal{P}T_{SQ}}^{01} = [0.01, 0.0, 0.22, 90.0, -90.0, 0.0]$, $\Lambda_{\mathcal{P}T_{SQ}}^{02} = [0.0, 0.0, 0.223, 0.0, -90.0, 0.0]$ and $\Lambda_{\mathcal{P}T_{SQ}}^{03} = [0.0, 0.0, 0.22, 0.0, 0.0, 0.0]$. The angles are expressed in degrees and positions in meters. For the HERI II hand, we have available five primitive grasping actions,; *pinchTight* and *trigs* for each fingers.

B. Experiments Results

The set of experiments carried out involves the entire primitive grasping actions planning procedure. The setup is shown in figure 8.



Fig. 8. Experiments scenario: HERI II hand mounted on robotic arm with the object to grasp in front.

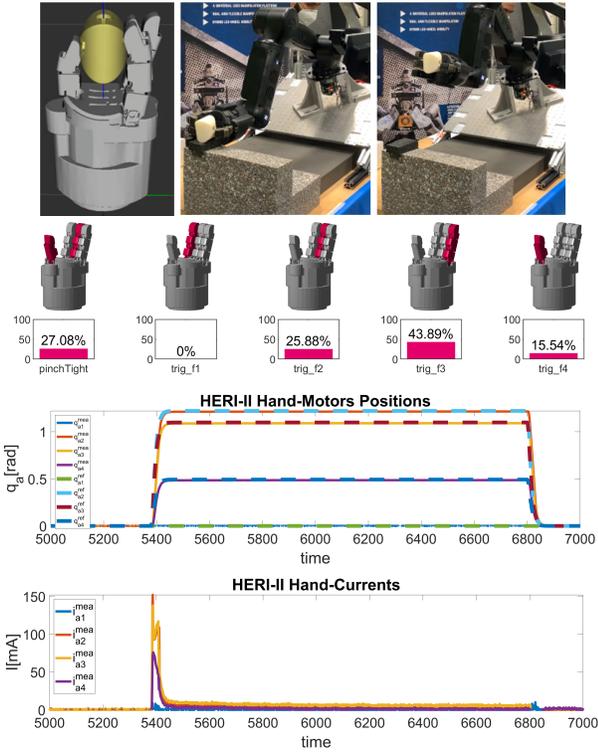


Fig. 9. Experiment with object 01. Images from the experiment performed and the grasp pose of the hand in the simulation environment RViz are shown. The primitives involved in the grasp pose and their product of intention and intensity are presented. The plots depict the motors positions references and measurements occurred and their currents measurements during the execution of the grasping action.

The experiments start with the robot in an initial pose. Then the robot starts to move to reach a known pose of the hand with respect to object. After that, fingertips contact points and desired contact forces are computed by the planner which allows grasping and lifting up the object until bringing the object to the goal location releasing it inside the white box. The grasp poses are those given by $\Lambda_{\mathcal{P}T_{SQ}}^{01}$, $\Lambda_{\mathcal{P}T_{SQ}}^{02}$ and $\Lambda_{\mathcal{P}T_{SQ}}^{03}$. In the computation, we admit a threshold on the value obtained from eq (9) due to the numerical precision. Therefore, to make the algorithm more robust, we use a virtual internal (virtual) superquadric. The gap between the real and virtual superquadric can freely be tuned. In our experiments, we fix it equal to 0.003m. The fingertips workspaces have been sampled with 120 samples. The overall primitives grasping actions have been performed on three objects as shown in the figures 9, 10, and 11, respectively. For each object, the planned intention and intensity levels for each primitive were applied for grasping the objects. As it can also be seen in the video² accompanying this work, the proposed grasp planner enabled to generate effective plans based on the composition of the primitive grasping actions, resulting in the successful grasping of the the objects involved in the experiments.

²<https://youtu.be/CbwgZALC5kc>

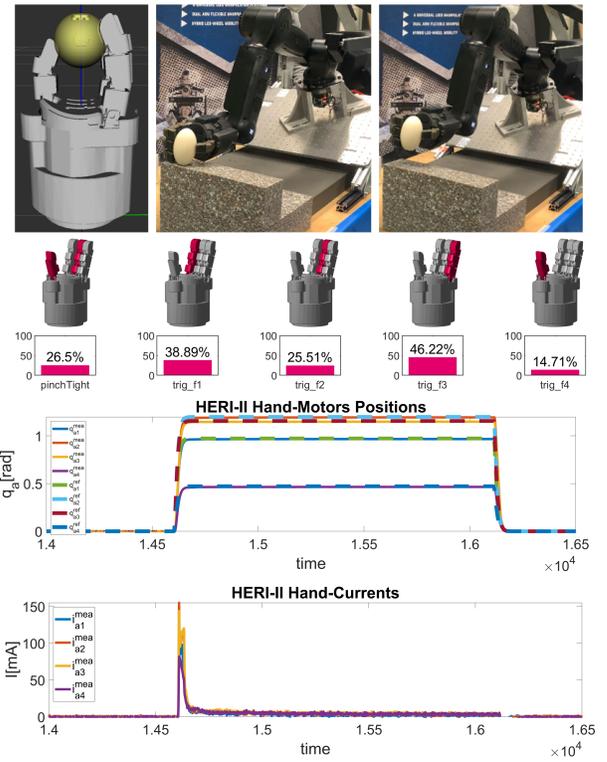


Fig. 10. Experiment with object 02. Images from the experiment performed and the grasp pose of the hand in the simulation environment RViz are shown. The primitives involved in the grasp pose and their product of intention and intensity are presented. The plots depict the motors positions references and measurements occurred and their currents measurements during the execution of the grasping action.

VIII. CONCLUSIONS

In this paper we presented a generic grasp planner that leverages on the composition of grasping postures using a number of primitive grasping actions extracted from a robotic end-effector. The grasp planner determines the intention and intensity levels required for each primitives to grasp an object ensuring FC condition. A formal model of the primitives grasping actions was introduced as well as the complete and generic procedures for finding ICRs and CPs. The proposed grasping planner was implemented on the HERI II robotic hand and experimentally verified illustrating its effectiveness to produce plans for grasping objects of different shapes. Direction for future works will consider the incorporation of perception to provide the pose of the objects with respect to the hand and the integration of task requirements in the present formulation. Again, tests on the robustness due to inaccurate models of the proposed planner have to be conducted. Again, tests on the robustness due to inaccurate models of the proposed planner have to be conducted.

ACKNOWLEDGMENT

This work was supported by the European Union's Horizon 2020 research and innovation programme [grant numbers 732287 (ROS-Industrial) and 101016007 (CONCERT)] and the Italian Fondo per la Crescita Sostenibile - Sportello

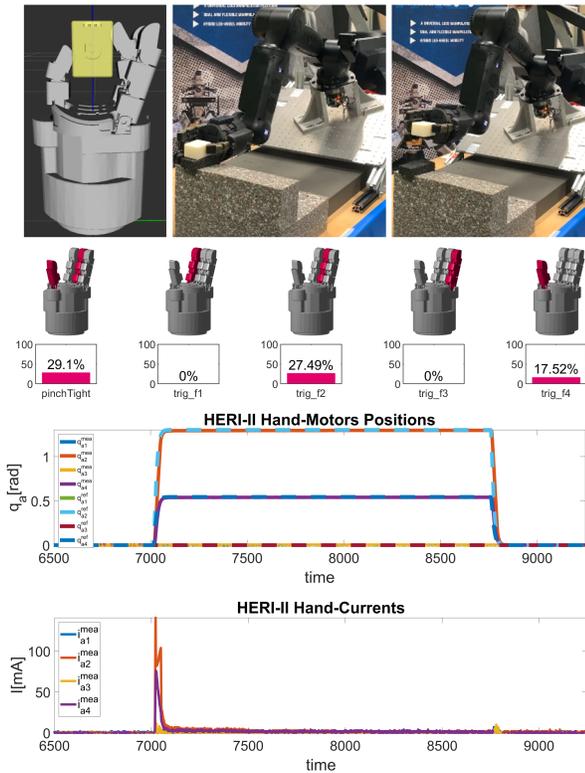


Fig. 11. Experiment with object 03. Images from the experiment performed and the grasp pose of the hand in the simulation environment RViz are shown. The primitives involved in the grasp pose and their product of intention and intensity are presented. The plots depict the motors positions and measurements occurred and their currents measurements during the execution of the grasping action.

Fabbrica intelligente, PON I&C 2014 - 2020, project number F/190042/01-03/X44 RELAX. The authors want to thank Stefano Carrozzo, Diego Vedelago and Phil Hudson for their support with the experiments and Enrico Mingo Hoffmann and Arturo Laurenzi for their guidance in the implementation of the planner.

REFERENCES

- [1] Piazza, C., et al. "A century of robotic hands." *Annual Review of Control, Robotics, and Autonomous Systems* 2 (2019): 1-32..
- [2] Shimoga, Karun B. "Robot grasp synthesis algorithms: A survey", *IJRR*, 15.3 (1996): 230-266.
- [3] Bohg, Jeannette, et al. "Data-driven grasp synthesis—a survey." *IEEE TRO* 30.2 (2013): 289-309.
- [4] D. Torielli, et al. "Towards an Open-Source Hardware Agnostic Framework for Robotic End-Effectors Control", 2021 20th International Conference on Advanced Robotics (ICAR)
- [5] REN, Zeyu, et al. HERI II: A robust and flexible robotic hand based on modular finger design and under actuation principles (IROS 2018), p. 1449-1455.
- [6] Bicchi, Antonio, and Vijay Kumar. "Robotic grasping and contact: A review." *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*. Vol. 1. IEEE, 2000.
- [7] Bicchi, Antonio. "On the closure properties of robotic grasping." *The International Journal of Robotics Research* 14.4 (1995): 319-334.
- [8] Garzón, Máximo Roa, and Raúl Suárez. "Grasp synthesis for 3D objects." *Institut d'Organització i Control de Sistemes Industrials*, 2006.

- [9] Sahbani, Anis, Sahar El-Khoury, and Philippe Bidaud. "An overview of 3D object grasp synthesis algorithms." *Robotics and Autonomous Systems* 60.3 (2012): 326-336.
- [10] Roa, Máximo A., and Raúl Suárez. "Grasp quality measures: review and performance." *Autonomous robots* 38.1 (2015): 65-88.
- [11] Cutkosky, Mark R., and Robert D. Howe. "Human grasp choice and robotic grasp analysis." *Dexterous robot hands*. Springer, New York, NY, 1990. 5-31.
- [12] Geng, Tao, Mark Lee, and Martin Hülse. "Transferring human grasping synergies to a robot." *Mechatronics* 21.1 (2011): 272-284.
- [13] Santello, Marco, Martha Flanders, and John F. Soechting. "Postural hand synergies for tool use." *J. of Neuroscience* 18.23 (1998): 10105-10115.
- [14] Gabbicini, Marco, et al. "On the role of hand synergies in the optimal choice of grasping forces." *Autonomous Robots* 31.2 (2011): 235-252.
- [15] Ciocarlie, Matei, Corey Goldfeder, and Peter Allen. "Dimensionality reduction for hand-independent dexterous robotic grasping." 2007 IEEE/RISJ International Conference on Intelligent Robots and Systems. IEEE, 2007.
- [16] Birglen, Lionel, and Clément M. Gosselin. "Kinetostatic analysis of underactuated fingers." *IEEE TRO*, 20.2 (2004): 211-221.
- [17] Prattichizzo, Domenico, and Jeffrey C. Trinkle. "Grasping." *Springer handbook of robotics*. Springer, Cham, 2016. 955-988.
- [18] Barr, Alan H. "Superquadrics and angle-preserving transformations." *IEEE Computer graphics and Applications* 1.1 (1981): 11-23.
- [19] Bicchi, Antonio. "On the problem of decomposing grasp and manipulation forces in multiple whole-limb manipulation." *Robotics and Autonomous Systems* 13.2 (1994): 127-147.
- [20] Roa, Máximo A., and Raúl Suárez. "Computation of independent contact regions for grasping 3-d objects." *IEEE Transactions on Robotics* 25.4 (2009): 839-850.
- [21] Sartipizadeh, Hossein, and Tyrone L. Vincent. "Computing the approximate convex hull in high dimensions." *arXiv preprint arXiv:1603.04422* (2016).
- [22] E. M. Hoffman, A. Rocchi, A. Laurenzi and N. G. Tsagarakis, "Robot control for dummies: Insights and examples using OpenSoT," (Humanoids 2017), p. 736-741.
- [23] L. Muratore, A. Laurenzi, E. Mingo Hoffman and N. G. Tsagarakis, "The XBot Real-Time Software Framework for Robotics: From the Developer to the User Perspective," in *IEEE Robotics and Automation Magazine*, vol. 27, no. 3, p. 133-143, Sept. 2020.