Graphical visualization of contact forces and hand movements during in-hand manipulation

Raúl Suárez
Institute of Industrial and Control Engineering (IOC)
Universitat Politècnica de Catalunya (UPC)
Barcelona, Spain (raul.suarez@upc.edu)

Andrés Montaño
Industry and Mobility, TECNALIA
Basque Research and Technology Alliance (BRTA)
San Sebastián, Spain (andres.montano@tecnalia.com)

***Abstract*—The paper presents a tool to graphically display the contact forces applied by the fingers of a robotic hand when doing grasping and in-hand manipulation. The forces are computed in two ways, on one side, directly using the measurements of tactile forces in the fingertips and, on the other, using the torques applied by the motors in the finger joints. The implemented tool also allows to command and move the real robotic hand by specifying the complete hand configuration or any single joint, and see graphically the hand simulation. Real results are shown using the Allegro hand with tactile sensors WTS-FT.**

***Index Terms*—Robotics, Grasping, Manipulation, Tactile sensors.**

I. Introduction

Currently, in robotics, there are multiple software tools with different functionalities, types of licenses, and complexity. For instance, for motion planning there are libraries such as Open Motion Planning Library (OMPL) [1], Task-Motion Kit [2], or the Kautham project [3]; and for grasp planning and simulation libraries such as GraspIt [4], OpenGrasp [5] or Simox [6]. An updated and complete review about robot simulation tools can be found in [7].

In this work we introduce a software tool that allows the graphic visualization of a robotic hand and the forces that the fingertips are applying when grasping an object, either obtained from the torques applied to the finger joints or obtained from tactile sensors at the fingertips. The developed tool also allows the real hand to be moved in synchronization with the graphical simulation, either by specifying the complete hand configuration or by individual instructions for each of its joints.

The problem of computing the forces exerted on a manipulated object using the torques applied to the joint motors has already been previously addressed, e.g. using a robotic hand without tactile sensors [8], or using a bi-manual robot [9] looking for the optimization of the forces applied on the manipulated object, but graphical representations of the real applied forces to help the user understanding the grasp during the development of grasping and manipulation strategies are not available.

After this introduction, Section II presents the description of the proposed tool and Section III shows some application examples. Finally, some conclusions and future work are presented in Section IV.

This work was partially supported by the Spanish Government through the project PID2020-114819GB-I00, and by EC research project REMODEL through the grant number: 870133.

II. Description of the tool

A. General Schema

The tool is implemented in C++ and based on *Robot Operating System* (ROS) [10], which works as communication layer to interconnect software modules. Figure 1 shows a block diagram with the main components of the tool:

- *Configuration files:* Include three files with information to set up the application (to be provided by the user):
  - A launch file with the paths to the communication interfaces connecting the software tool with the real hardware (hand and tactile sensors).
  - A YAML (yaml.org) configuration file with specific information of the hardware, including: number of degrees of freedom, specific controller parameters, and a path to a file with the description of the hand and the sensors.
  - A Unified Robot Description File (*URDF*, wiki.ros.org/urdf) with the description of the elements in the scene for their graphical representation (including the hand and the sensor models).
- *The hardware:* Robotic hands and tactile sensors.
- *The drivers (1):* Software modules including basic functionalities to open and close the communication ports and send and receive data packets following a protocol to interact with the hardware. The drivers are encapsulated into a library allowing other applications to use the hardware as an instance of an object belonging to the library. The hand driver allows to read the values of the finger joints and, depending on the type of hand controller, to command either the torques to be applied by the joint motors or the positions of the joints. The tactile sensor driver allows to obtain the contact point location and the measured force.
- *The visualization module:* Application with five nodes, two of them standard ROS nodes:
  - *"Joint State" Publisher:* This node publishes the joint values of a robot described by a given *URDF*. The node reads the robot_description parameter from the

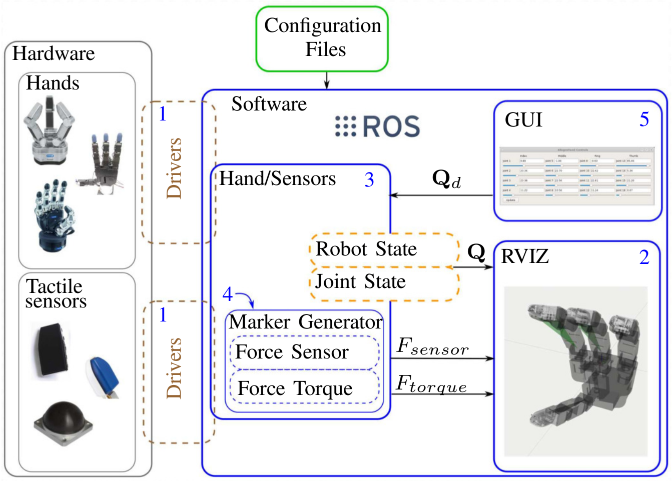


Fig. 1. Block diagram with the components (hardware and software) of the developed system.

ROS parameter server, finds all the non-fixed joints and publishes a Joint State message with values for the joints.

- *"Robot State" Publisher:* This node takes the joint values as input and publishes the 3D poses (transformation tf2 (wiki.ros.org/tf2) of the robot links following the kinematic tree model of the robot defined in the *URDF*.

and another three ROS nodes developed specifically for this work:

- *A visualization node (2):* This node, based on *RVIZ* (wiki.ros.org/rviz) deals with the graphic representation of the hand (using CAD models with triangular meshes to represent the robot links and the sensors) and the measured and computed contact forces, represented using *arrows*, a kind of marker that easily shows the location of the contact point and the force magnitude.
- *The hand/sensor node (3):* This module manages the hand and the sensors using their specific drivers; provides the services and topics to exchange information with the others nodes; computes the grasping forces and generates the data for the graphical visualization. This node has also a marker generator (4) with two submodules, one to generate the markers to visualize the force measured by the tactile sensors and another to generate the markers to visualize the force computed using the torques applied by the joint motors.
- *A GUI node (5):* This node, based on QT (www.qt.io), allows to command the hand position by individual sliders for each hand joint. This node takes the information of the Hand/Sensor node using a service to determine the limits of the joints and publish the information of the desired configuration using a Joint State message.

B. Contact force computation

The grasping force can be measured/estimated using the tactile sensors at the fingertips. In this specific implementation the tactile sensors provide a tactile sensing matrix, and the drivers to manage, configure and read information from the sensors were developed as part of this work. The representation of the grasping forces is done considering a frictional point-contact model [11], and therefore, the resultant contact force at each fingertip is considered to be the summation of the forces detected at each texel of the sensor and the application point is considered to be the barycenter of contact region detected by the tactile sensor.

On the other hand, the force F_i applied by finger i on the grasped object can be computed from the joint torques as follows. The vector $\boldsymbol{\tau}_i$, containing the torques τ_{i_j} applied to the j-th joint of finger i, has a component $\mathbf{g}(\mathbf{q}_i)$ compensating the gravity effect acting on the finger links, and another component with the torques $\boldsymbol{\tau}_{i_{\text{grasp}}}$ that produce the desired grasping force F_i on the object, i.e.

$$\boldsymbol{\tau}_i = \boldsymbol{\tau}_{i_{\text{grasp}}} + \mathbf{g}(\mathbf{q}_i) \qquad (1)$$

$\boldsymbol{\tau}_i$ is measured at the joint actuators and $\mathbf{g}(\mathbf{q}_i)$ is computed using the dynamic model of the hand, therefore $\boldsymbol{\tau}_{i_{\text{grasp}}}$ can be directly obtained from (1) and the force F_i applied by the finger f_i on the manipulated object is computed as [12],

$$F_i = J(\mathbf{q}_i)^{T^*} \boldsymbol{\tau}_{i_{\text{grasp}}} \qquad (2)$$

where $J(\mathbf{q}_i)^{T^*}$ is the inverse of the transposed Jacobian of the finger, considering as punctual contact point on the fingertip the barycenter of contact region detected by the tactile sensor. Both measurements of the contact forces can be done either with the hand in a static configuration or while doing the object manipulation.

III. Illustrative Validation Examples

The tool has been implemented as a general package and instances were developed for two robotic hands: Allegro Hand from Wonik Robotics with modified fingertips to include Weiss WTS-FT tactile sensors (set used for the examples below), and the Schunk SDH2 that already has its own Weiss tactile sensors. The Allegro hand has four fingers, with four degrees of freedom each one. The WTS-FT tactile sensors have a resistive tactile matrix of 4×8 square texels with a side of 3.8 mm that detect only the normal component of the applied force (i.e. tangential forces are not detected). The measurements of the hand sensors is performed at 20Hz.

Figure 2 shows a screenshot of the visualization tool, including the following windows, whose numbers are in correspondence with those in Figure 1: 1) corresponds to the nodes managing the drivers of the hand and the sensors using CAN and USB ports, respectively; 2) displays the graphical representation of the hand and the applied forces; 3) corresponds to the ROS node linking the graphical interface with the controllers of each joint fo the hand; 4) corresponds to the ROS node that computes the grasping forces and allows

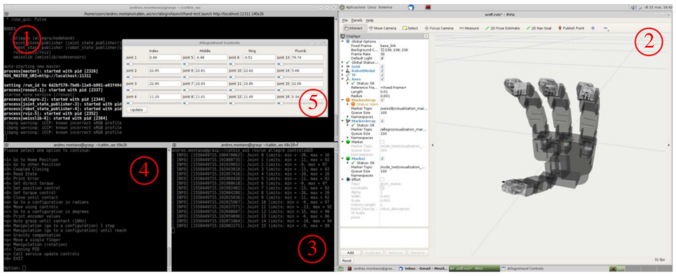


Fig. 2. Screenshot of the visualization tool when using the Allegro Hand.

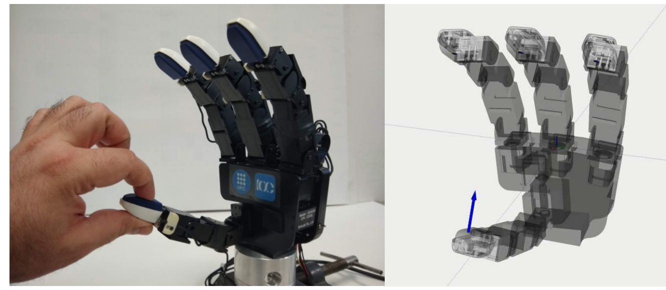
their visualization; and, 5) displays graphically the sliders to move each joint of the hand.

Figure 3 shows three examples in which external forces are applied on a finger in different conditions (with and without pressure on the tactile sensor) while the hand controller try to keep the current hand configuration.

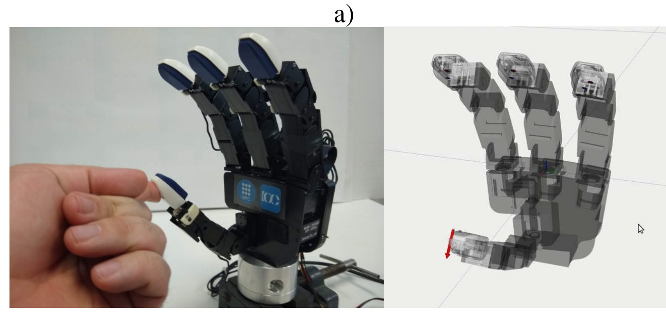
Figure 4 shows four examples, grasping different objects. Each example has a snapshot of the grasp and two *RVIZ* screenshots with the representation of the forces computed using the joint torques and the forces measured using the tactile sensors. Note that in the last two examples there is a notable difference between the forces produced by the torques in the finger joints and the forces measured by the sensors, this is due to the fact that the sensor cannot detect tangential forces applied on its pad surface as it happens in these cases. This effect shows the usefulness of the proposed tool that quickly allows the graphical visualization of this difference, and therefore helps in understanding what is really happening with the grasping forces during the manipulation.

IV. Summary and Future Work

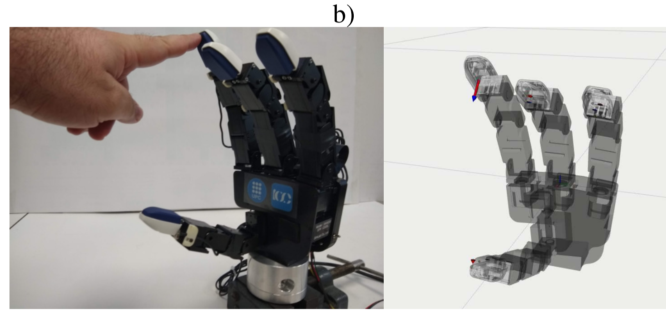
The paper has presented a tool for the graphical representation of the grasping forces obtained from the torques applied at the actuators of robotic hand and also from the tactile sensors. The tool helps in the process of development, supervision and evaluation of grasping and manipulation strategies, and will be fully integrated in an open package developed with this aim.



a)



b)



c)

Fig. 3. Examples of interaction with the fingers and the tactile sensors. a) A force is applied on the tactile sensor and another equivalent force is applied on the back of the sensor (i.e. the net force on the finger is null), thus the sensor detects a force (blue arrow) while there is no resultant force computed from the joint torques; b) A force is applied on the finger without pressing on the tactile sensor, thus a force is detected from the joint torques (red arrow) but not by the sensor; c) A force is applied on the tactile sensor pushing the finger, thus the sensor detects a force (blue arrow) and there is also a resultant force from the joint torques (red arrow), note that red and blue arrows are similar and almost overlapped.

References

[1] I. A. Şucan, M. Moll, and L. E. Kavraki, “The Open Motion Planning Library,” *IEEE Robot Autom Mag*, vol. 19, no. 4, pp. 72–82, 2012.

[2] N. T. Dantam, S. Chaudhuri, and L. E. Kavraki, “The task-motion kit: An open source, general-purpose task and motion-planning framework,” *IEEE Robot Autom Mag*, vol. 25, no. 3, pp. 61–70, Sep. 2018.

[3] J. Rosell, A. Pérez, A. Aliakbar, Muhayyuddin, L. Palomo, and N. García, “The kautham project: A teaching and research tool for robot motion planning,” in *IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA)*, 2014.

[4] A. T. Miller and P. K. Allen, “Graspit! a versatile simulator for robotic grasping,” *IEEE Robot Autom Mag*, vol. 11, no. 4, pp. 110–122, 2004.

[5] B. León, S. Ulbrich, R. Diankov, G. Puche, M. Przybylski, A. Morales, T. Asfour, S. Moisio, J. Bohg, J. Kuffner, and R. Dillmann, “Opengrasp: A toolkit for robot grasping simulation,” in *Int. Conf. on Simulation, Modeling, and Programming for Auton. Robots*, 2010, pp. 109–120.

[6] N. Vahrenkamp, M. Krohnert, S. Ulbrich, T. Asfour, G. Metta, R. Dillmann, and G. Sandini, “Simox: A robotics toolbox for simulation, motion and grasp planning,” *Advances in Intelligent Systems and Computing*, vol. 1, pp. 585–594, 2013.

[7] J. Collins, S. Chand, A. Vanderkop, and D. Howard, “A Review of Physics Simulators for Robotic Applications,” *IEEE Access*, p. In Press.

[8] Zhixing Xue, M. Schmidt, J. M. Zoellner, and R. Dillmann, “Internal force computation of grasped object using joint torques,” *2008 SICE Annual Conference*, pp. 2795–2800.

[9] A. Rojas-de Silva and R. Suarez, “Contact force computation for bimanual grasps,” in *2017 22nd IEEE Int. Conf. on Emerging Technologies and Factory Automation (ETFA)*. IEEE, sep, pp. 1–6.

[10] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, “Ros: an open-source robot operating system,” in *Proc. of IEEE International Conference on Robotics and Automation – Workshop on Open Source Software*, 2009, pp. 1–6.

[11] J. K. Salisbury and B. Roth, “Kinematic and Force Analysis of Articulated Mechanical Hands,” *Journal of Mechanisms, Transmissions, and Automation in Design*, vol. 105, pp. 35–41, 1983.

[12] R. M. Murray, Z. Li, and S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. CRC Press LLC, 1994.

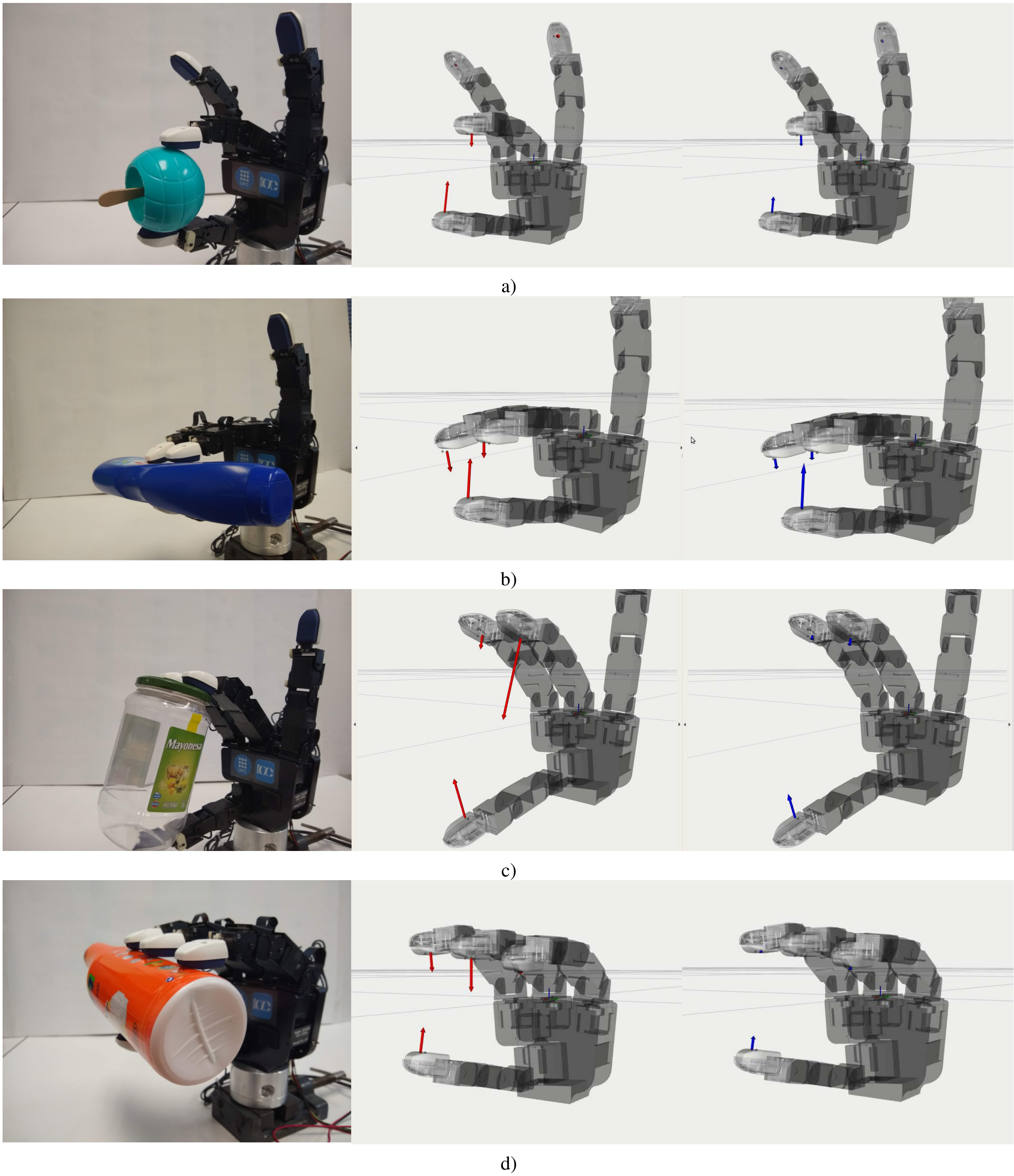


Fig. 4. Examples of grasping different objects. a) Object with a spherical shape held with two fingers; b) Bottle with flat faces held with three fingers; c) Cylindrical object held with three fingers at the upper and lower ends; d) Cylindrical bottle held using four fingers.