

Finger Readjustment Algorithm for Object Manipulation Based on Tactile Information

Juan Antonio Corrales Ramón, Fernando Torres Medina, Véronique Perdereau

▶ **To cite this version:**

Juan Antonio Corrales Ramón, Fernando Torres Medina, Véronique Perdereau. Finger Readjustment Algorithm for Object Manipulation Based on Tactile Information. International Journal of Advanced Robotic Systems, 2013, 10 (1), pp.9. 10.5772/53561 . hal-01583304

HAL Id: hal-01583304

https://hal.sorbonne-universite.fr/hal-01583304

Submitted on 7 Sep 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

INTECH
open science | open minds

OPEN ACCESS
ARTICLE

International Journal of Advanced Robotic Systems

Finger Readjustment Algorithm for Object Manipulation Based on Tactile Information

Regular Paper

Juan Antonio Corrales Ramón[1,*], Fernando Torres Medina[2] and Véronique Perdereau[1]

1 Institut des Systèmes Intelligents et de Robotique, Université Pierre et Marie Curie, Paris, France.
2 Departamento de Física, Ingeniería de Sistemas y Teoría de la Señal, Universidad de Alicante, Alicante, Spain
* Corresponding author E-mail: corrales_ramon@isir.upmc.fr

Received 9 Apr 2012; Accepted 19 Sep 2012

DOI: 10.5772/53561

© 2013 Ramón et al.; licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract This paper presents a novel algorithm which registers pressure information from tactile sensors installed over the fingers of a robotic hand in order to perform manipulation tasks with objects. This algorithm receives as an input the joint trajectories of the fingers which have to be executed and adapts it to the real contact pressure of each finger in order to guarantee that undesired slippage or contact-breaking is avoided during the execution of the manipulation task. This algorithm has been applied not only for the manipulation of normal rigid bodies but also for bodies whose centre of mass can be changed during the execution of the manipulation task.

Keywords Robotic Manipulation, Tactile Sensors, Contact Breaking, In-hand Object Reconfiguration

1. Introduction

Robotic manipulators are becoming increasingly widespread nowadays [1] due to their positional precision, repeatability and durability. They are usually applied for repetitive tasks where all of the objects to be handled by the robot are in the same position and only the position control of the robot is needed. Nevertheless, the application of robotic manipulators which can react to changes in their environment may improve their productivity and the flexibility of their tasks [2]. In addition, this context-awareness will permit the inclusion of new agents in the robot's workspace, such as human operators. The first step that must be completed in order to make humans and robotic manipulators cooperate in the same environment is the guaranteeing of the safety of the human operators. This requirement has been completed by previous research with the development of a safe human-robot interaction system [3][4] which is able to localize precisely the human during the development of the task. The second step that should be completed in order to develop real collaboration between human operators and robots is the development of tasks where they can handle and share objects.

Most previous robotic manipulators which handle objects are based on the use of simple parallel grippers which can only perform pick-and-place operations. Nevertheless, if

a more flexible collaboration between humans and robots is to be implemented, more complex robotic hands with multiple fingers should be used [5]. In fact, multi-fingered robotic hands have two important features for the replication of the manipulation skills of human beings: dexterity and anthropomorphism [6]. These features enable multi-fingered hands to develop more complex manipulation tasks which not only involve restraining the object with a firm grasp - as with parallel grippers - but which also involve generating controlled trajectories of the object with the movements of the fingers. This form of the controlled movement of the grasped object is usually known as 'dexterous manipulation'[7] and it involves achieving a goal configuration of the object and its contacts with the fingers of the hand.

The main aim of this paper is the implementation of dexterous manipulation tasks of objects with a multi-fingered hand installed at the end-effector of a robotic manipulator. In this paper, the robotic manipulator is able to change the configuration of a grasped object so that it can be easily given to a human operator for the development of human-robot collaboration tasks. This paper is organized in the following sections: section 2 presents a summary of previous systems which try to solve dexterous manipulation tasks with multi-fingered hands; section 3 describes the model of the hand which is used in this paper; section 4 describes the finger readjustment algorithm which has been developed in order to implement manipulation tasks; section 5 enumerates the real experiments which have been performed and section 6 draws the main conclusions of this paper.

2. Previous Research on Robotic Manipulation

Previous research works on robotic manipulation have proposed different representations for solving the motion planning associated with the dexterous manipulation problem: graph representations, hybrid control systems and probabilistic trees.

In [8] and[9], contact configurations are represented as nodes of a graph and the transitions between these configurations generated by fingers' movements are represented as edges in the graph. In[10], a similar approach is presented but, in this case, each node only represents the object configuration and it does not include the configuration of the hand, which is only considered in the edges of the graph. All these planners based on graph representations solve the dexterous manipulation problem as the search process of a path between two specific nodes of the graph.

In [11] and[12], two manipulation planners based on stratified control systems are developed. Each stratum of these systems contains the equations of the contact evolution depending on the number of fingers which are in contact with the object. In this case, the dexterous manipulation problem is represented as transitions between these strata (when there are changes in the number of contacting fingers) and as continuous evolutions inside each stratum (when motion is applied to the object by movements of the fingers).

In [13], a random sampling process of the configuration space of the object is performed in order to generate a RRT (*Rapidly-exploring Random Tree*) with all its possible configurations. In[14], a random sampling process of the system object-hand is developed in order to obtain a graph which represents the topology of all the possible object-hand configurations. In both systems, the dexterous manipulation problem is solved by an A* search of a path in the RRT or the graph.

Some of the previous planners (such as [8] and[9]) do not take into account the contact forces which are applied to the object and they are only based on the maintenance of the geometric contact between the surfaces of the fingers and the object. Other planners use point contact models with friction in order to define force closure restrictions which guarantee - theoretically - that each grasp performed during the manipulation task is able to counteract any external force applied to the object. These restrictions are usually implemented as inequalities between the components of the frictional forces which are solved by linear/quadratic programming (such as [10] and [13]). The force closure condition is also verified theoretically for each intermediate configuration of the object to be manipulated in [11] and [14].

Nevertheless, all these manipulation planners use approximations which simplify the surfaces of the object and/or the fingers so that only contact points are considered. These simplifications involve the fact that these planners are only applied in simple simulations and they are not applied in manipulation tasks with real robotic hands. This paper proposes a new algorithm which readjusts the finger trajectories computed by a high level geometric planner in order to apply them in real manipulation tasks. The main goal of these finger readjustments is to guarantee that the contact between the fingers and the object is not broken during the manipulation tasks. In order to do this, the proposed algorithm uses real pressure contact values which are registered by tactile sensors installed over the surface of the robotic hand. Therefore, this algorithm is not limited by the approximations of the object and fingers surfaces of previous planners. In addition, the use of real tactile information permits the extension of the application of the algorithm to more complex objects, even with moving parts which modify their centre of mass.

3. Modelling and Control of the Robotic Hand

In this paper, a *Barrett* hand [15] has been used to develop manipulation tasks. The *Barrett* hand (see Figure 1) is a

robotic hand with three fingers and 4 independent degrees of freedom, each one driven by an electric motor. Each finger contains two joints which are able to close the finger towards the palm, but their movements are coupled since they are controlled by a unique motor for each finger. In addition, fingers 1 and 2 can rotate around the palm due to the movement of the fourth motor. This fourth degree of freedom permits the establishment of different relative configurations between the fingers: by opposing fingers 1 and 2 with finger 3 (for precision grasps with their fingertips) and by placing all the fingers at the same side of the palm (for power grasps of cylindrical objects).

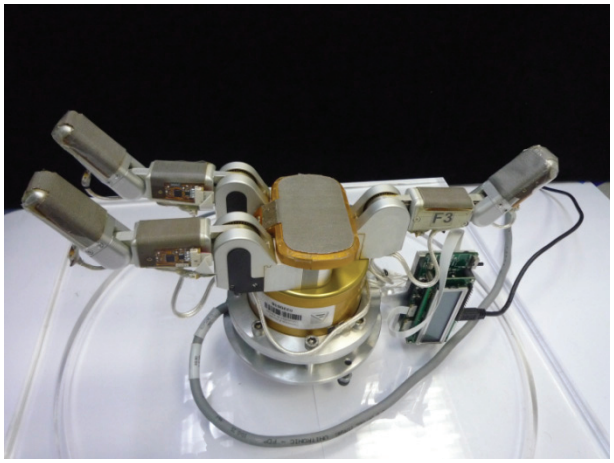


Figure 1. Barrett Hand with tactile sensors for manipulation tasks.

In order to control these degrees of freedom and thus develop manipulation tasks, this paper uses the direct and inverse kinematic models of this hand which have been developed in the previous research [16]. In particular, and from a specific contact configuration between the fingers and the object to be manipulated, the inverse kinematic model can be applied in order to calculate the number of motor counts which have to be applied in order to place the fingers over the desired contact points.

A higher level manipulation planner can be used in order to determine the evolution of these contact points. This manipulation planner should be able to link the initial and final configurations of the object by taking into account that contacts cannot be broken (i.e., contact maintenance restrictions) and that kinematic limits of the joints of the fingers cannot be exceeded (i.e., kinematic restrictions). This paper has used the geometric planner presented in [17] to solve these restrictions and thus obtain a trajectory for the fingers' joints that drive the object from the initial configuration to the final configuration.

This geometric planner represents the surfaces of the object and the fingers as triangular meshes in order to calculate the evolution of their contacting primitives (i.e., vertices, edges and faces) along the intermediate configurations of the object. As such, this planner is able to guarantee - theoretically - that contact points are kept and that the fingers' joints are able to perform the required movements during the execution of the manipulation task. Although the computed fingers' joints' trajectories are executed successfully in software simulations (such as in [16]), they usually cannot be completed in real experiments when long manipulation movements are required. This limitation, which is common in previous manipulation planners, is caused by the accumulation of small errors during the execution of the joint trajectory and also by the successive loss of contact pressure between the fingers and the object. In fact, this loss of contact pressure usually provokes the breaking of contacts and, thus, the instability of the grasp. This paper proposes a novel finger readjustment algorithm which monitors the real contact pressure of each contacting finger in order to guarantee that contacts are not broken and, thus, previously-computed object-hand trajectories can be applied in real manipulation tasks.

Since the finger readjustment algorithm that is proposed in this paper needs the real contact pressure which is applied by the fingers at each step of the manipulation task, the robotic hand has to include tactile sensors over its fingers. In particular, the *Barrett* hand used in this paper has 7 *PPS Robotouch* tactile sensors [18], as shown in Figure 1. Two sensors are installed over each finger, one over the outer phalanx and one for the inner phalanx. Furthermore, there is also a sensor over the palm of the hand. The outer phalanx sensors are composed by an array of 22 tactels with an area of 36 mm^2 per tactel. The inner phalanx sensors are composed by an array of 24 tactels with and area of 25 mm^2 per tactel. The palm sensor contains an array of 24 tactels with an area of 100 mm^2 per tactel. All these tactels can register pressure values in the range 0-140KPa with a frequency of 30 Hz and a sensitivity of 0.7KPa. Figure 2 shows a 3D representation of the pressure values registered by these tactile sensors during a manipulation task.

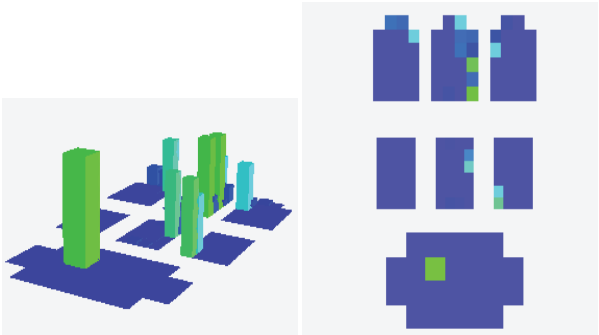


Figure 2. 3D representation of the pressure values registered by tactile sensors.

The information from these tactile sensors is stored in a vector of tactel pressure values. This maximum pressure value between all the tactels of each finger will be considered by the algorithm presented in the following section in order to determine whether the contact is sufficiently strong in order to continue the manipulation task. The use of the maximum value to represent the contact state of each finger reduces the influence of random pressure errors which are usually registered by tactels which are not in contact with the object.

4. Finger Readjustment Algorithm

The geometric planner presented in [17] solves the dexterous manipulation problem by taking into consideration contact maintenance and finger kinematics constraints. The contact maintenance constraint is guaranteed by computing rolling movements between the primitives (i.e., vertices, edges and faces) of the triangles' meshes which represent the surfaces of the fingers and the object. The finger kinematics constraint is verified by the application of the inverse kinematic model of the hand. Nevertheless, as stated before, one of the main limitations of this planner and other previous planners is that it does not consider the real contact forces which are applied from the fingers to the manipulated object. As such, this geometric planner can be applied for small in-hand manipulation tasks where the initial configuration of the object and the desired final configuration are very near to each other. When both configurations are far away from each other, the forces applied by the fingers over the manipulated object tend to decrease after each execution step of the planner. This reduction of the contact forces makes the grasp unstable, where by the object eventually slips and the contacts between the fingers and the object are lost when they are applied in a real experiment rather than in a software simulation. The finger readjustment algorithm proposed in this paper tries to solve this problem so that the finger joint trajectories computed by a manipulation planner (such as[17]) can be applied in a real manipulation task.

The main goal of this finger readjustment algorithm is to guarantee that each contacting finger always applies contact force above a pre-established safety threshold. The proposed finger readjustment algorithm is executed following the geometric planner described in[17]. In particular, this algorithm receives as input the finger joint values which are generated by the geometric planner and which are required in order to move the object to the following intermediate configuration of its desired trajectory. Figure 3 depicts a general scheme of the main steps of this finger readjustment algorithm and its relation with the geometric planner.

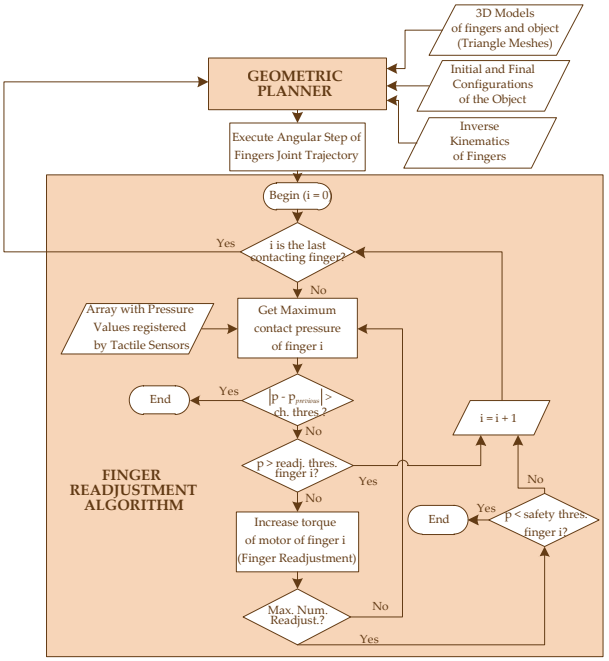


Figure 3. General scheme of the finger readjustment algorithm executed after each step of the finger joint trajectory obtained from the geometric planner.

First of all, the finger joint values obtained from the geometric planner are executed following which the finger readjustment algorithm has to determine the contact force that the fingers are applying over the object. As stated in the previous section, this value is proportional to the pressure values registered by the tactile sensors that are installed over the hand. Therefore, the proposed algorithm obtains an array of pressure values for each finger. Next, it looks for the maximum pressure value of the tactels of each finger in order to obtain the maximum pressure which is being applied to the object by the corresponding finger.

In each execution step of the finger readjustment algorithm, the maximum finger pressure is firstly compared with a readjustment threshold, which represents an upper bound for the pressure. Accordingly, if the maximum pressure of the finger is above this threshold, no finger readjustment is needed since the applied pressure is sufficient.

However, if the maximum pressure of the finger is below this threshold, the algorithm slightly increases the torque of the motor corresponding to the finger. This torque increase does not provoke any movement - either of the object or of the fingers - because they are blocked by the current grasp and, thus, an increase of the contact pressure applied from this finger to the object is obtained. This readjustment procedure is repeated iteratively until the pressure exceeds the readjustment pressure threshold of the finger or until a maximum number of readjustments are executed. This maximum number of readjustments has been added to this algorithm in order

to guarantee that the algorithm is not blocked in manipulation configurations where the pressure cannot be increased but where it is above the safety pressure threshold.

If the contact pressure is above the readjustment threshold, the algorithm continues to verify the contact pressure of the following contacting finger. Otherwise, the contact pressure cannot be increased over the readjustment threshold and the new contact pressure obtained after the maximum number of readjustments is compared with a safety threshold. If the pressure is below this safety threshold, the manipulation algorithm stops, since the contact between the object and the finger is very weak and further finger movements may make the grasp unstable. Otherwise, the algorithm continues and the pressure of the following contacting finger is verified. Thus, the safety threshold represents a lower bound for the pressure. In fact, the interval of the pressure values between the readjustment and the safety thresholds represents a range of possible contact pressure values which are admissible for the development of manipulation tasks. The values of the readjustment and the safety thresholds are obtained during a previous learning process which is executed for each object to be manipulated.

This learning algorithm consists of repeating several manipulation tasks in relation to the object while the values of the safety thresholds are changed. First of all, the safety threshold is established to the minimum contact force for each finger which keeps the object in equilibrium (i.e., the contact forces of the fingers are able to counteract the weight of the object) at the initial grasp configuration while the readjustment threshold is established to a percentage increase with respect to this safety threshold. For these two initial values of the thresholds, several manipulation tasks of the object (rotations and/or translations) are executed. If, in this first set of trials, there is contact breaking or slippage, the safety threshold is increased as is, consequently, the readjustment threshold. This process is repeated until there is neither contact breaking nor slippage. After the safety threshold is established, the same set of trials is repeated while progressively reducing the readjustment threshold. This reduction is performed until contact breaking or slippage occurs. As such, both thresholds are determined for each object.

After the pressure values of all the contacting fingers have been verified, the finger readjustment algorithm returns the control to the geometric planner. This process is repeated until the object reaches the desired final configuration or until the finger readjustment detects weak contacts identified by pressure values smaller than the safety threshold.

An additional end condition has been added to the finger readjustment algorithm for the manipulation of objects with variable centres of mass. When the finger readjustment algorithm detects that the difference between the current contact pressure of a finger and its previous pressure is higher than a change threshold, the algorithm also ends. In this case, the manipulation cannot be continued because this change represents the fact that the centre of mass of the object has changed dramatically and so the pressure thresholds of the finger (i.e., readjustment and safety thresholds) are not valid anymore. This condition permits the extension of the finger readjustment algorithm to complex objects composed of several moving parts, such as those which are usually found in industrial environments where robotic manipulators are used.

5. Experimental Results

Several manipulation tasks have been performed in order to verify the correct behaviour of the proposed finger readjustment algorithm. The experimental setup employed to develop these tasks is composed of two main components: a *Mitsubishi PA-10* robotic manipulator and a *Barrett* hand with 7 tactile sensors (as described at Section 3). As shown in Figure 4, the *Barrett* hand is installed at the end-effector of the robotic manipulator and is controlled with integer values which represent the steps/counts of each motor of the hand (one for each finger and one for the spread joint).A detailed description of the kinematic and dynamic models of this hand and the correspondence between motor counts and joint angles can be found in [16].

This robotic system can be used to develop human-robot interaction tasks where human operators can cooperate in the development of assembly and disassembly tasks. Nevertheless, this paper is only focused on the robotic manipulation of objects and its application in human-robot tasks can be verified in previous works ([19] and [20]).Three different manipulation tasks with three different objects have been executed: translation of a cylinder (i.e., a tin), translation of a sphere (i.e., a plastic ball) and translation of a parallelepiped with a variable centre of mass (i.e., a wooden box with a lead ball inside).

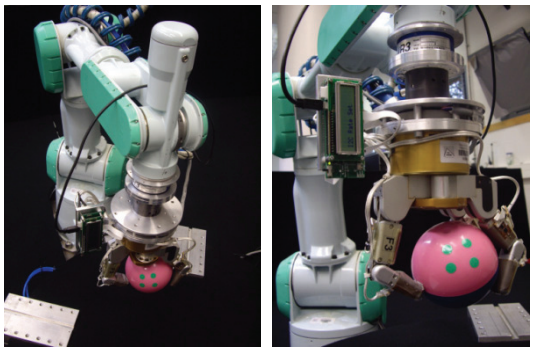


Figure 4. Top (left) and front (right) views of the experimental setup.

5.1 Manipulation of a Cylinder

The first object to be manipulated is a cylinder (i.e. a tin). The cylinder is initially placed over a table. Firstly, the robotic hand is closed automatically (Figure 5a) in order to grasp the cylinder from t=0s to t=10s, where the motor counts of the three fingers rise during this period (Figure 6a). These motor counts are proportional to the joint angles of the fingers since the joints of each finger are driven by the same motor, as described in Section3. After this closing process of the fingers (Figure 5b), they apply different pressure values to the object: 12KPa in the case of finger 1, 6KPa in the case of finger 2 and 39KPa in the case of finger 3 (as shown at t=10s in Figure 6b). Next, the robotic manipulator moves the hand and the object up in order to separate it from the table from t=10s to t=18s while the geometric planner of [17] in conjunction with the finger readjustment algorithm presented in this paper begin to be executed from this initial grasp (Figure 5c). The execution of the steps of the joint trajectory computed by the geometric planner can be identified in Figure 6a as linear slopes of the plot from t=18s to t=46s (Figure 5d-Figure 5f).

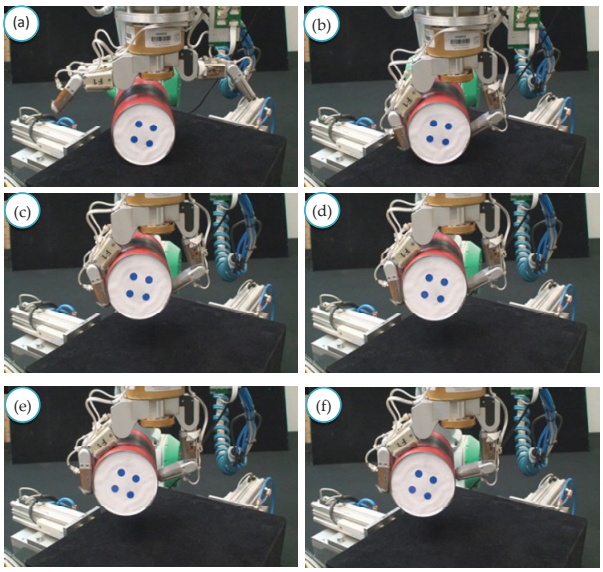
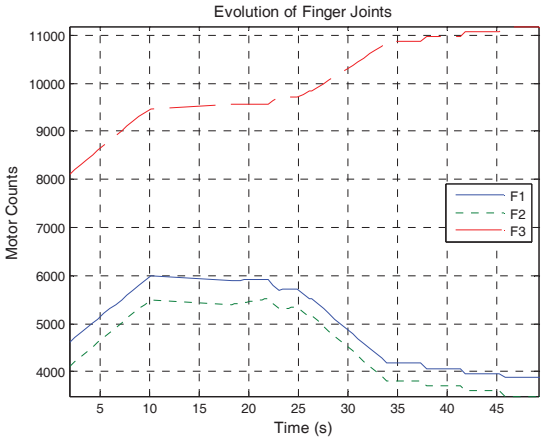
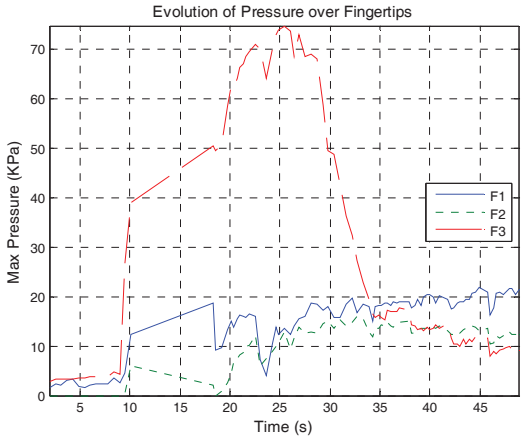


Figure 5. Sequence of frames of a manipulation task for a cylinder applying the finger readjustment algorithm.

After these variations of the joint angles, the finger readjustment algorithm is executed and tries to keep the contact pressure of each finger above the readjustment thresholds: 10KPa for fingers 1 and 2 and 20KPa for finger 3. For instance, several readjustments of fingers 1 and 2 are executed between t=18s and t=26s. They can be identified in Figure 6a by horizontal lines where the motor counts do not change (i.e., the fingers push the object but their joint angles do not change since they are blocked by the object). In Figure 6b, they can be identified by downwards peaks of pressure which go below the pressure value of 10KPa and subsequent upward peaks of pressure which go above this pressure value.



(a)



(b)

Figure 6. Evolution of the manipulation task for a cylinder: (a) evolution of the motors' counts; (b) evolution of the maximum contact pressure.

The correspondence between the evolution of these pressure values and the movements of the fingers can be verified in the video [21]. After executing some readjustments of fingers 1 and 2 between t=18s and t=26s, the planner executes several increments of the finger joint angles from t=26s to t=34 s without finger readjustments since all the pressure values are above the thresholds. At t=34s, a new readjustment of finger 3 is executed, since its pressure is below 20KPa. This readjustment is executed again after each increment of the local planner, but the pressure of this finger decreases unyieldingly until it is below the safety pressure threshold (10KPa for finger 3 and 5.0KPa for fingers 1 and 2) and the manipulation task is definitively stopped. This safety requirement is identified in Figure 5f because finger 3 is touching the object near the end of its fingertip and the grasp is near instability.

5.2 Manipulation of a Sphere

Another manipulation task based on translating a sphere (i.e. a plastic ball) has been performed. The readjustment

pressure thresholds are established to the following values: 15KPa for fingers 1 and 2 and 30KPa for finger 3). The safety thresholds are established to the following values: 10KPa for fingers 1 and 2 and 20KPa for finger 3. Figure 7 shows three frames of the linear trajectory of the object which is executed during the manipulation task. Figure 7a shows the initial configuration, Figure 7b shows an intermediate configuration and Figure 7c shows the final configuration. Firstly, as in the previous task, the fingers are closed in order to form the initial grasp of Figure 7a. This can be identified in Figure 8a by a slope in the evolution of the finger motor counts between t=0s and t=28s. Then, the geometric planner and the finger readjustment algorithms are executed. Between t=28s and t=50s, several readjustments of fingers 1 and 2 are executed, since their maximum contact pressure values are below 15KPa. Between t=50s and t=73s, several readjustments of finger 3 are executed since its contact pressure is below 30KPa. These readjustments are able to keep the pressure of finger 3 near the readjustment threshold of 30KPa and above the safety threshold of the object without any risk of grasp instability. As such, this manipulation task can be completed and the object is driven to its final desired configuration (Figure 7c). The video of this manipulation task can be viewed in[22].

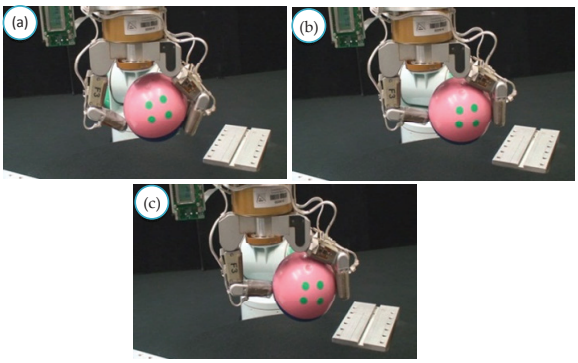
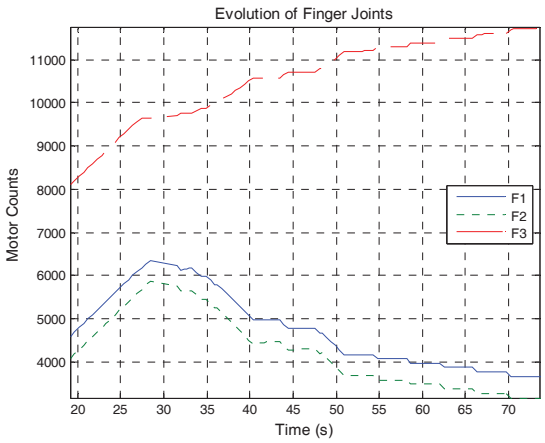
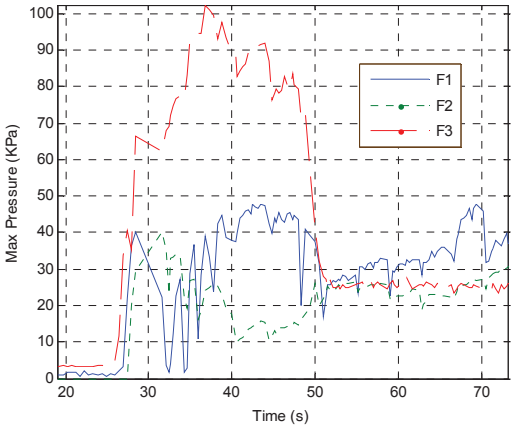


Figure 7. Sequence of frames of a manipulation task for a sphere applying the finger readjustment algorithm.



(a)



(b)

Figure 8. Sequence of frames of a manipulation task for a sphere applying the finger readjustment algorithm.

5.3 Manipulation of an Object with a Variable Centre of Mass

The last manipulation task is based on the translation of a box with a variable centre of mass (i.e., a wooden box with a lead ball inside). A video of this task can be viewed in [23]. The readjustment pressure thresholds are established to the following values: 30KPa for finger 3 and 20KPa for fingers 1 and 2. The safety pressure thresholds are established at the following values: 20KPa for finger 3 and 5KPa for fingers 1 and 2. At the beginning of the manipulation task, the readjustment algorithm presents its normal behaviour and several successive readjustments are executed for fingers 1 and 2 from t=0s to t=27s because their contact pressure values are sometimes under the readjustment thresholds (see Figure 10b). These readjustments can be identified by horizontal lines in Figure 10a. The finger readjustment algorithm is able to keep the contact pressure of all of the fingers above the safety thresholds and, thus, the desired trajectory of the object is achieved while the inside ball does not change its position (Figure 9a andFigure 9b).

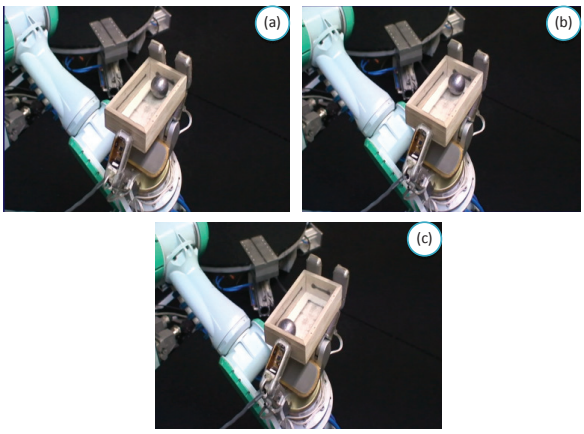
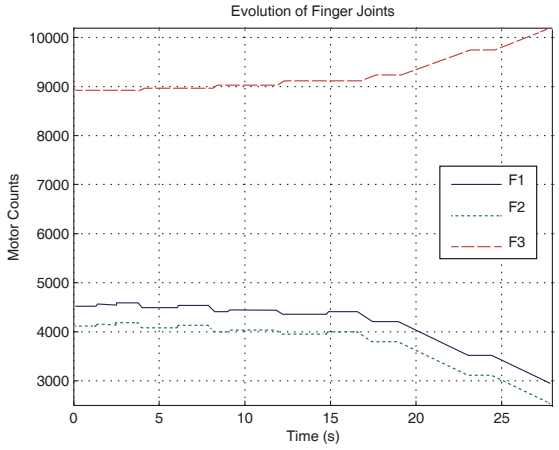
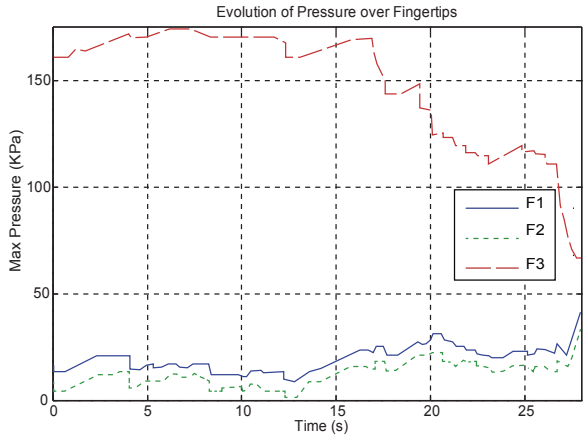


Figure 9. Sequence of frames of a manipulation task for a box with a moving ball applying the finger readjustment algorithm.

Nevertheless, the successive translation movements of the box finally cause the ball to move and relocate to the other side of the box (Figure 9c). This fact provokes a sudden change in the contact pressure of all of the fingers. This change can be verified as an important increase in the contact pressure of fingers 1 and 2 and an important decrease in the contact pressure of finger 3, as shown between t=27s to t=28s in Figure 10b. This sudden pressure change makes the finger readjustment algorithm finish the manipulation task, as described in Section4. In fact, this end condition is necessary because the movement of the ball changes the location of the centre of mass of the box completely and this change involves the fact that the contact pressure values between the fingers and the hand will also be different, as is demonstrated by their sudden change in Figure 10b. Therefore, the readjustment algorithm cannot be executed for any further steps because the contact thresholds which it is using are now obsolete. This solution permits the application of the readjustment algorithm to more complex objects and the stability of the manipulation task is guaranteed because its execution is stopped as soon as a new contact condition might provoke an undesired loss of contact.



(a)



(b)

Figure 10. Evolution of the manipulation task for a box: (a) evolution of the motors' counts; (b) evolution of the maximum contact pressure.

6. Conclusions

This paper proposes a novel finger readjustment algorithm which receives as input a joint trajectory for a robotic hand computed by an upper level manipulation planner which is able to guarantee the execution of this trajectory in a real hand. As such, this algorithm is able to drive the manipulated object to its desired configuration. It verifies that the contact pressure between the fingers of the hand and the object is always above a specific threshold so that contact-breaking and undesired sliding are always avoided. In addition, when the algorithm detects that the manipulation task might be unstable due to the loss of contact pressure or a sudden change of contact pressure, it stops the manipulation task as a safety measure in order to avoid the failure of the grasp.

This algorithm has been successfully verified in three manipulation tasks for three different objects: a cylinder, a sphere and a box with a variable centre of mass. The experiment with the cylinder is finished when one of the fingers reaches the end of its surface. This fact demonstrates the good behaviour of the algorithm in predicting undesired contact situations which can culminate in undesired grasp instabilities. The experiment with the sphere demonstrates that the proposed algorithm is able to complete a desired object trajectory until its end, when no undesired contact situations are detected. Finally, the manipulation task of the box also demonstrates the ability of the algorithm to stop the manipulation when the centre of mass of the object changes unexpectedly.

In future work, the proposed algorithm will be improved by adding to possibility of finger gaiting so that the manipulation task will not be stopped when an undesired contact situation is detected. In this case, the position of the fingers over the surface of the object will be changed in order to continue the manipulation task and avoid weak contact situations. Nevertheless, this extension of the algorithm will also involve an improvement of the geometric planner which calculates the finger joint trajectories because the possibility of controlled contact breaking should be added. In addition, a new surface model for the objects and the fingers could be developed in order to extend this algorithm to soft and deformable objects.

7. Acknowledgements

The research leading to the results presented in this paper has been funded by the research projects DPI2008-02647and DPI2011-22766 of the Spanish Ministry of Science and Innovation and by the research project GRE10-16 of the University of Alicante. This research has also received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under

Grant Agreement no. 231640 and the research project HANDLE.

8. References

[1] International Federation of Robotics (2008) World Robotics Industrial Robots. Frankfurt, Germany.

[2] Kulic, D., Croft, E. (2004)Safe Planning for Human-Robot Interaction. In: Proc. International Conference on Robotics and Automation, pp. 1882-188. New Orleans, LA, USA.

[3] Corrales, J. A., Candelas, F. A., Torres, F. (2010)Sensor data integration for indoor human tracking. Robotics and Autonomous Systems, 58(8), 931-939.

[4] Corrales, J. A., Candelas, F. A., Torres, F. (2011)Safe human-robot interaction based on dynamic sphere-swept line bounding volumes. Robotics and Computer-Integrated Manufacturing, 27(1), 177-185.

[5] Yoshikawa, T.: (2010) Multifingered robot hands: Control for grasping and manipulation. Annual Reviews in Control, 34(2), 199-208.

[6] Melchiorri, C., Kaneko, M. (2008) Robot Hands.In: Siciliano, B., Khatib, O. (eds.) Handbook of Robotics.Springer, Heidelberg.

[7] Han, L., Trinkle, J. C. (1998) Dexterous Manipulation by Rolling and Finger Gaiting.In: Proc. IEEE International Conference on Robotics and Automation, pp. 730-735. Leuven, Belgium.

[8] Trinkle, J., Hunter, J. J. (1991)A Framework for Planning Dexterous Manipulation. In: Proc. IEEE International Conference on Robotics and Automation, pp. 1245-1251. Sacramento, USA.

[9] Zhang, H., Tanie, K., Maekawa, H. (1996)Dexterous Manipulation Planning by Grasp Transformation. In: Proc. IEEE International Conference on Robotics and Automation, pp. 3055-3060. Minneapolis, USA.

[10] Cherif, M., Gupta, K. K. (1999) Planning Quasi-Static Fingertip Manipulations for Reconfiguring Objects. IEEE Transactions on Robotics and Automation, 15(5), 837-848.

[11] Goodwine, B. (1999)Stratified Motion Planning with Application to Robotic Finger Gaiting.In: Proc. 14th IFAC World Congress. Beijing, China.

[12] Harmati, I., Lantos, B., Payandeh, S. (2002)On Fitted Stratified and Semi-Stratified Geometric Manipulation Planning with Fingertip Relocations. The International Journal of Robotics Research, 21(5-6), 489-510.

[13] Yashima, M. (2004)Manipulation Planning for Object Re-orientation Based on Randomized Techniques.In: Proc. IEEE International Conference on Robotics and Automation, pp. 1245-1251.New Orleans, USA.

[14] Saut, J. P., Sahbani, A., El-Khoury, S., Perdereau, V. (2007)Dexterous Manipulation Planning Using Probabilistic Roadmaps in Continuous Grasp Subspaces. In: Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2907-2912. San Diego, USA.

[15] Barrett Technology (2012)The Barrett Hand. http://www.barrett.com/ (2012). Accessed March 2012.

[16] Corrales, J. A., Jara, C. A., Torres, F. (2010) Modelling and simulation of a multi-fingered robotic hand for grasping tasks. In: Proc. 11th International Conference on Control Automation, Robotics and Vision, pp. 1577-1582. Singapore, Singapore.

[17] Corrales, J. A. (2011)Safe Human-Robot Interaction based on Multi-Sensor Fusion and Dexterous Manipulation Planning. Ph.D. dissertation, Physics, Systems Engineering and Signal Theory Department, University of Alicante, Spain.

[18]Pressure Profile Systems (2012) RoboTouch. http://www.pressureprofile.com/. Accessed March 2012.

[19] Pomares, J., Candelas, F. A., Torres, F., Corrales, J. A., García, G. J. (2010)Safe Human-Robot Cooperation based on an Adaptive Time-Independent Image Path Tracker. International Journal of Innovative Computing, Information and Control, 6(9), 3819-3842.

[20] Gil, P., Pomares, J., Puente, S., Candelas, F. A., García, G. J., Corrales, J. A., Torres, F. (2009)A cooperative robotic system based on multiple sensors to construct metallic structures. The International Journal of Advanced Manufacturing Technology, 45(5-6), 616-630.

[21] Corrales, J. A., García, G. J., Torres, F. (2012)Video of the Manipulation of a Cylinder. http://www.aurova.ua.es/DPI/videos/manipulacion%20experimento%201.flv. Accessed March 2012.

[22] Corrales, J. A., García, G. J., Torres, F. (2012) Video of the Manipulation of a Sphere, http://www.aurova.ua.es/DPI/videos/manipulacionesfera2.flv.Accessed March 2012.

[23] Corrales, J. A., García, G. J., Torres, F. (2012)Video of the Manipulation of a Box with Variable Center of Mass.http://www.aurova.ua.es/DPI/videos/centromasasvariable1.flv. Accessed March 2012.