Including virtual constraints in motion planning for anthropomorphic hands

Jan Rosell, Raúl Suárez, Alexander Pérez and Carlos Rosales
Institute of Industrial and Control Engineering (IOC)
Technical University of Catalonia (UPC)
Barcelona, Spain
e-mail: jan.rosell@upc.edu

***Abstract*—This paper copes with the problem of finding a collision-free path for a hand-arm robotic system from an initial unconstrained configuration to a final grasping (or preshape) one. The aim is to obtain a natural motion as a sequence of human-like postures that both capture the coupling that there exist between the fingers of the human hand and also maintain the palm oriented towards the object to be grasped. The proposed method is a sampling-based approach whose efficiency relies in the reduction of the dimensionality obtained by considering, for the finger joints, a subspace determined by the main principal motion directions that capture the coupling and, for the position and orientation of the palm, the submanifold that satisfies the orientation constraint. The approach is illustrated with an example and compared to the case where no virtual constrains are used, validating the proposal.**

I. Introduction

Over the last years there has been an increasing interest in humanoid robots. Besides the aspects related to walking, or those related with expression and human-robot interaction, another key aspect of this type of robots is their manipulation capability, directly linked to the availability of dexterous hands. These hands are complex mechanical devices with a number of degrees of freedom (DOF) ranging from 12 (four fingers with 3 DOF each one) to 25 (five fingers with 4 DOF each one plus some DOF in the palm) [1].

The use of mechanical hands rises the problem of finding force-closure grasps, i.e. grasps that are able to resist any external force applied to the object, as well as the problem of finding collision-free motions of the hand from an initial free and unconstrained configuration to the final grasping (or preshape) one. When the considered hands are anthropomorphic, both the configuration of the hand when grasping the object and its postures through the approaching motions are usually desired to be human-like. This responds to an aesthetic wish of mimicking the human beings, although not being necessary from a practical point of view. The first relevant work in this line is that of Santello et al. [2], that determined a two-dimensional grasp subspace from a set of hand configurations obtained when different subjects where asked to grasp several objects. This subspace captured the coupling between finger joints in grasping postures and was later used for grasp synthesis [3], [4]. In a similar way we used principal component analysis, with a set of unconstrained general hand configurations obtained by demonstration, to model all the real hand workspace (not only potential grasping configurations). A bounded subspace was then sampled to create a roadmap where collision-free motions were planned [5], [6], resulting in sequences of anthropomorphic natural postures. All these methods, besides providing the desired human-like feature of the configurations of the mechanical hands, are very efficient since they result in a substantial reduction of the degrees of freedom of the problem. Similar dimensionality reduction techniques have also been used to synthesize human-like motion in graphic applications [7].

Besides constraining the motion of the fingers, the human-like requirement may also constrain the motion of the palm depending on the shape and size of the object to be grasped. In this sense, for instance, if the object to be grasped is small, the palm may be required to always look towards it (this becomes mandatory if vision-feedback is used with in-hand cameras). Visual constraints have been considered in several works on robotics motion planning following different approaches like, for instance, planning first the path of the camera, determining from it the target trajectory in the image, and using this information to command a control system with visual feedback [8]. Other approaches merge sampling based global path planning and visual servoing with in-hand camera in order to obtain robot trajectories that keep the target in the field of view avoiding any occlusion while avoiding robot collisions and staying away from the robots joint limits, either using rapidly-exploring random trees (RTT) [9] or probabilistic roadmaps (PRM) [10] for this aim. RRTs and PRMs are sampling-based path planners that randomly generate collision-free samples of configuration space and connect them with free paths capturing the connectivity of the free space by forming trees and roadmaps, respectively. Their efficiency is highly dependent on the sample generation and, for difficult path-planning problems like those involving narrow passages or high degrees of freedom robots, the number of samples required might be quite large, therefore, importance sampling or dimension-reduction techniques have been used.

In a hand-arm system, the dimension reduction can be achieved for the finger motions of the mechanical hand as de-

This work was partially supported by the Spanish Government through the projects PI09/90088, DPI2008-02448 and DPI2010-15446.

A. Pérez is also with the Escuela Colombiana de Ingeniería “Julio Garavito”, Bogotá D.C., C. Rosales is also with the Institut de Robòtica i Informàtica Industrial (IRI), CSIC-UPC, Barcelona, Spain.

tailed above; the dimension reduction for the palm movements can be achieved by considering virtual motion constraints. In constrained-based motion planning problems, the robot is constrained to a submanifold with lower dimensionality than its embedding space. If sampling-based methods are used, the planning of collision-free movements of a constrained object does not require the sampling of the whole configuration space, but only the regions of the configuration space where the robot is allowed to move: its configuration submanifold. These submanifolds can be easily described in terms of geometric constraint sets by explicitly stating the relations that must hold between two or more geometric entities. Geometric constraint solvers, like PMF [11], can be used to find the map between constraint sets and configuration submanifolds. PMF was used in [12], where a constraint-based probabilistic roadmap was proposed.

This work proposes a probabilistic roadmap to plan natural motions for a hand-arm system. The proposal pursues efficiency by planning in subspaces of lower dimension. For the hand, a subspace determined by the main principal motion directions that capture the coupling between the finger joints is used, and for the arm, the submanifold that satisfies some given orientation constraints. Planning in this subspaces, moreover, results in sequences of anthropomorphic natural postures. Next section formally presents the problem statement, the solution overview and the organization of the paper.

II. Problem Statement and Solution Overview

Consider the Shunk Antropomorphic Hand (SAH) [13], described in Section III, mounted on a six DOF Stäulbli TX90 robot. Let $\mathcal{C} = \mathcal{C}^{\mathrm{f}} \times \mathcal{C}^{\mathrm{p}}$ be the configuration space of the hand-arm system, with $\mathcal{C}^{\mathrm{f}}$ and $\mathcal{C}^{\mathrm{p}}$ being the configuration spaces of the finger joints and of the position and orientation of the palm, respectively. Let also:

- $c_{ini} = (c^f_{ini}, c^p_{ini})$ and $c_{goal} = (c^f_{goal}, c^p_{goal})$ be the initial and the goal configurations of the hand, with c^f_{ini} and c^f_{goal} being the configurations of the fingers, and c^p_{ini} and c^p_{goal} those of the palm.
- $\mathcal{F}_p$ be the reference frame of the palm with its x-axis, $\hat{x}_p$, forming an angle of 60 degrees with the normal direction of the palm (Fig. 1).
- P be the point located at 14 cm from the origin of $\mathcal{F}_p$ along $\hat{x}_p$. Point P is approximately the center of the hand workspace.
- P_W be the position of P when the hand is at c_{goal}.

Then, the problem to be solved is the search of a collision-free path connecting c_{ini} and c_{goal}, subject to the following constraints:

- The postures of the hand-arm system must have an anthropomorphic appearance.
- The hand must be moved in such a way that $\hat{x}_p$ always points towards P_W (i.e. point P_W must always appear at the center of the image, provided a camera is mounted on the hand with $\hat{x}_p$ as the camera axis).

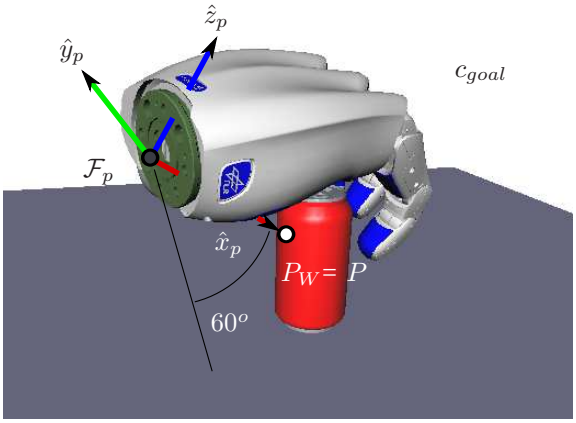


Fig. 1. The location of the hand at c_{goal} is used to define point P_W.

The present paper proposes an approach to solve this problem, that can be easily generalized to any hand-arm system. The solution is based on the construction of a probabilistic roadmap whose nodes are configurations obtained by sampling a lower-dimensional space $\mathcal{SC} = \mathcal{SC}^{\mathrm{f}} \times \mathcal{SC}^{\mathrm{p}}$ with $\mathcal{SC}^{\mathrm{f}} \subset \mathcal{C}^{\mathrm{f}}$ and $\mathcal{SC}^{\mathrm{p}} \subset \mathcal{C}^{\mathrm{p}}$:

- Finger joint values are obtained as a linear combination of vectors that define the main principal motion directions that capture the existing couplings between finger joints (Section III). The coefficients are randomly obtained. The vectors define the subspace $\mathcal{SC}^{\mathrm{f}}$, whose dimension is usually not greater than five, being much lower than the number of finger joints.
- The position and orientation of the palm is obtained by sampling the variables that parameterize the submanifold $\mathcal{SC}^{\mathrm{p}}$ that capture the coupling between position and orientation (Section IV). The number of parameters is 4, less than the six required to fix the position and orientation of a 3D object in the space.

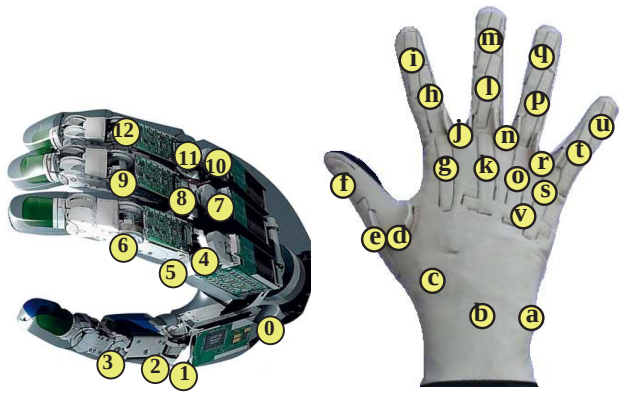
The proposed path planning algorithm is shown in Section V and illustrated and evaluated in Section VI. Section VII presents the conclusions of the work.

III. Sampling Finger Joints

A. Computing PMDs

Principal Motion Directions (PMDs) are coordinated movements of the hand joints. They can be used to reduce the problem of planning the motion of a mechanical hand, considering that the human hand has similar couplings. They are obtained by taking samples of human hand postures using a sensorized glove, mapping them to the mechanical hand and then performing a principal component analysis (PCA) [14] over the set of samples. The samples are taken with the intention of covering the mechanical hand workspace. It is worth to note that this mapping is critical to achieve such goal, thus the more anthropomorphic the mechanical hand is, the easier the mapping results.

As a mechanical hand, we use the Schunk Antropomorphic Hand [13] (Fig. 2a). It has four identical fingers and one is equipped with an additional joint to function as the opposing



a) b)

Fig. 2. a) The anthropomorphic mechanical hand SAH and b) the sensorized glove, both with labelled joints.

thumb. Each finger has four joints, one for abduction and three for flexion, with two of them coupled, having therefore three independent degrees of freedom (DOF) per finger. In total, it has 13 DOF. As a sensorized glove to capture human hand positions we use the CyberGlove© from CyberGlove Systems (shown in Fig. 2b). It is a fully instrumented glove that provides up to 22 joint-angle measurements using resistive bendsensing technology, it includes three flexion sensors per finger, four abduction sensors between the fingers, a palm-arc sensor, and two sensors to measure the flexion and the abduction of the wrist.

The mapping from the glove sensors to the mechanical hand SAH is done according to the relation in Table I (using the joint labels from Fig. 2). This mapping makes the motions of the SAH hand to be defined with 11 independent parameters, although it has 13 DOF (details on this mapping can be found in [5]). With this mapping, the set of samples captured when the operator freely moves the hand are mapped into the mechanical hand workspace, and it is there were a PCA analysis is done. The PCA involves the computation of the eigenvalue decomposition of a data covariance matrix or the singular value decomposition of a data matrix, usually after mean centering the data for each attribute.

The result of the PCA over the samples is the set of vectors (PMDs) that define a new base of $\mathcal{C}^{\mathrm{f}}$ (the corresponding reference frame is located at the mean center value $\mathbf{b}$ as shown in Fig. 3). PMDs can be represented as the columns of a matrix $E = (\hat{\boldsymbol{e}}_1, \hat{\boldsymbol{e}}_2, \ldots, \hat{\boldsymbol{e}}_F)$, where F is the dimension of $\mathcal{C}^{\mathrm{f}}$.

B. The sampling procedure

A subspace $\mathcal{SC}^{\mathrm{f}} \subset \mathcal{C}^{\mathrm{f}}$ of smaller dimension can be obtained by selecting the PMDs with higher associated variances. Choosing the first five, an accumulated total variance of 95% is obtained. This number has been considered enough for this work, and therefore $\mathcal{SC}^{\mathrm{f}}$ will be defined by a subset E' of E: $E' = (\hat{\boldsymbol{e}}_1, \hat{\boldsymbol{e}}_2, \ldots, \hat{\boldsymbol{e}}_5)$.

The sampling procedure obtains samples from an axis-aligned box in $\mathcal{SC}^{\mathrm{f}}$ centered at $\mathbf{b}$ and with the size of each side, $2\lambda_i$, chosen proportional to the standard deviation of the samples along $\hat{\boldsymbol{e}}_i$, such that the box contains around 95% of the samples. This sampling region is then uniformly sampled, and the samples $\boldsymbol{e}$ are directly mapped into the real hand workspace as:

$$\boldsymbol{\theta}^h = E'\boldsymbol{e} + \boldsymbol{b} \tag{1}$$

where $\boldsymbol{e} = (e_1, \ldots e_5)$ with $e_i \in [-\lambda_i, \lambda_i]$. See Fig. 3 for a graphical interpretation on a 2-dimensional space.

TABLE I
CORRESPONDENCE BETWEEN THE CYBERGLOVE SENSORS (FIG. 2B) AND THE JOINTS OF THE SAH HAND (FIG. 2A).

Cyberglove Sensor		SA Hand Joint	
Id.	Name	Id.	Name
c	thumb roll	0	thumb base
c	thumb roll	1	finger base (thumb)
e	thumb inner	2	proximal phalanx (thumb)
f	thumb outer	3	medium phalanx (thumb)
j	index abduction	4	finger base (index)
g	index inner	5	proximal phalanx (index)
h	index middle	6	medium phalanx (index)
-	medium abduction	7	finger base (medium)
k	medium inner	8	proximal phalanx (medium)
l	medium medium	9	medium phalanx (medium)
n	ring abduction	10	finger base (ring)
o	ring inner	11	proximal phalanx (ring)
p	ring medium	12	medium phalanx (ring)

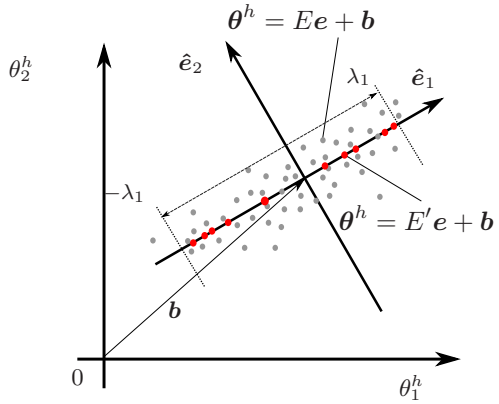


Fig. 3. A 2-dimensional space $\mathcal{C}^{\mathrm{f}}$ modelled with two PMDs, i.e. $E = (\hat{\boldsymbol{e}}_1, \hat{\boldsymbol{e}}_2)$, obtained from the input dataset (gray points). The subspace $\mathcal{SC}^{\mathrm{f}}$ is supposed to be 1-dimensional and defined by $E' = (\hat{\boldsymbol{e}}_1)$. Samples (big red dots on the $\hat{\boldsymbol{e}}_1$-axis) are obtained from the sampling box that in this case is the segment $(-\lambda_1, \lambda_1)$.

IV. SAMPLING THE PALM POSITION AND ORIENTATION

A. Orientation constraint

It is desired that the orientation of the palm be such that vector $\hat{x}_p$ always points towards P_W. The submanifold of $SE(3)$ that satisfies this orientation constraint is 4-dimensional, i.e. the motion of the palm is constrained to four degrees of freedom, three of translation defining the position of the origin of $\mathcal{F}_p$ and one of rotation defining the rotation of $\mathcal{F}_p$ around $\hat{x}_p$, as illustrated in Fig. 4.

Given a position of the origin of $\mathcal{F}_p$, $O_p = (x, y, z)$, the following rotation R_1 must be applied in order to satisfy the orientation constraint. Let:

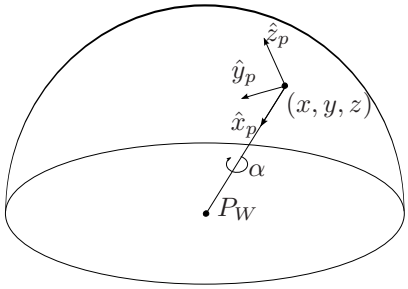


Fig. 4. Parameters x, y, z and α of the 4-dimensional submanifold that satisfies the orientation constraint.

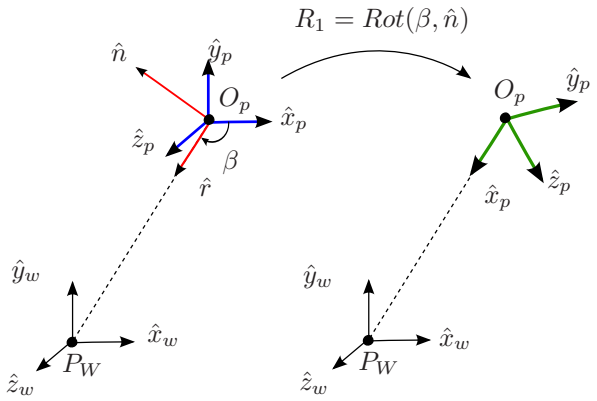


Fig. 5. Rotation $R_1 = Rot(\beta, \hat{n})$ to satisfy the orientation constraint.

- $\hat{r}$: be the unitary vector pointing from O_p to P_W.
- $\hat{n}$: be the unitary vector normal to the plane defined by $\hat{x}_p$ and $\hat{r}$:

$$\hat{n} = \frac{\hat{x}_p \times \hat{r}}{||\hat{x}_p \times \hat{r}||} \tag{2}$$

- β: be the angle between $\hat{x}_p$ and $\hat{r}$.

Then, the rotation R_1 is the rotation of an angle β around the direction $\hat{n}$, as illustrated in Fig. 5:

$$R_1 = Rot(\beta, \hat{n}) \tag{3}$$

Finally, the homogeneous transformation that defines the position and orientation of $\mathcal{F}_p$ satisfying the orientation constraint is:

$$T_p = \left(\begin{array}{c|c} R_2 R_1 & \begin{array}{c} x \\ y \\ z \end{array} \\ \hline 0\ 0\ 0 & 1 \end{array}\right) \tag{4}$$

where R_2 is any given rotation around $\hat{x}_p$:

$$R_2 = Rot(\alpha, \hat{x}_p) \tag{5}$$

B. The sampling procedure

The sampling of the position and orientation of the palm taking into account the orientation constraint is done by sampling the 4-dimensional submanifold $\mathcal{SC}^{\mathrm{p}} \subset \mathcal{C}^{\mathrm{p}}$ that complies with the constraint. The four degrees of freedom are:

- (x, y, z): the position of O_p.
- α: the angle of rotation R_2.

The sampling procedure uses uniform sampling generated with a random sampling source. Parameters are sampled in the range $[0, 1]$. Then, position parameters are scaled to the range determined by the task workspace and the orientation parameter to the range $[-\pi, \pi]$. Finally, the four-tuple (x, y, z, α) obtained is used in Eq. (4) to obtain the homogeneous transformation that defines the position and orientation of the palm, and the inverse kinematics of the robot is used to compute the joint angles of the arm.

V. Proposed approach

A. Sample generation

Let $c = (c^{\mathrm{f}}, c^{\mathrm{p}})$ be a configuration of $\mathcal{C}$, with $c^{\mathrm{f}} \in \mathcal{C}^{\mathrm{f}}$ and $c^{\mathrm{p}} \in \mathcal{C}^{\mathrm{p}}$, and let $sc = (sc^{\mathrm{f}}, sc^{\mathrm{p}})$ be a configuration of $\mathcal{SC}$, with $sc^{\mathrm{f}} \in \mathcal{SC}^{\mathrm{f}}$ and $sc^{\mathrm{p}} \in \mathcal{SC}^{\mathrm{p}}$.

Algorithms 1 and 2, called SampleFingerJoints and SamplePalm, sample configurations of $\mathcal{C}^{\mathrm{f}}$ and $\mathcal{C}^{\mathrm{p}}$, respectively. Algorithm 1 returns a self-collision-free configuration $c^{\mathrm{f}} \in \mathcal{C}^{\mathrm{f}}$, and Algorithm 2 returns a configuration of the palm that satisfies the orientation constraints and that is feasible (i.e. an inverse kinematics solution for the robot arm exists). They use the following functions:

- DIM(S): Returns the dimension of the space S.
- RAND(k,$[a, b]$): Returns a vector of dimension k whose components have random values in the range $[a, b]$.
- MAP1(sc^{f}): Returns the configuration $c^{\mathrm{f}} \in \mathcal{C}^{\mathrm{f}}$ corresponding to the sample $sc^{\mathrm{f}} \in \mathcal{SC}^{\mathrm{f}}$, as detailed in III-B.
- MAP2(sc^{p}): Returns the configuration $c^{\mathrm{p}} \in \mathcal{C}^{\mathrm{p}}$ corresponding to the sample $sc^{\mathrm{p}} \in \mathcal{SC}^{\mathrm{p}}$, as detailed in IV-B.
- SELFCOLLISION(c^{f}): Informs whether the configuration $c^{\mathrm{f}} \in \mathcal{C}^{\mathrm{f}}$ makes the hand to be in self-collision.
- GETPOSITION(c^{p}): Returns the three-dimensional position vector corresponding to configuration c^{p}.
- GETORIENTATION(c^{p}): Returns the α value of rotation R_2 (Eq. 5) corresponding to the orientation of sc^{p} (it is assumed that c^{p} belongs to the manifold that satisfies the orientation constraint).
- INVKIN(c^{p}): Returns true if there exists an inverse kinematics solution of the robot arm that poses the palm at configuration c^{p}, or false otherwise.

In order to ease the connection of the initial and goal configurations to the roadmap, the procedure SamplePalm is used to obtain samples around the segment that joints the positions of c^p_{ini} and c^p_{goal}. Let c^{p}_k be a configuration obtained as a linear interpolation of c^p_{ini} and c^p_{goal}, then SamplePalm obtains a sample of the palm as follows. The position is randomly sampled within a cube of size $2\rho_x$ centered at the position corresponding to c^{p}_k; the orientation α is randomly sampled within a interval of size $2\rho_\alpha$ centered at the value of α corresponding to c^{p}_k. The values of ρ_x and ρ_α take smaller values as c^{p}_k approaches c^p_{goal}, in order to reduce variability and ease the obtention of free samples around the goal configuration (that is a preshape configuration very close to the objects).

Algorithm 1 SampleFingerJoints

repeat
  $sc^{\mathrm{f}} = \mathsf{RAND}(\dim(\mathcal{SC}^{\mathrm{f}}), [0,1])$
  $c^{\mathrm{f}} = \mathsf{MAP1}(sc^{\mathrm{f}})$
until SELFCOLLISION(c^{f}) = false
return c^{f}

Algorithm 2 SamplePalm

Require: c^{p}_k, k
  x^{p}_k= GETPOSITION(c^{p}_k)
  α_k = GETORIENTATION(c^{p}_k)
  repeat
    $\rho_x = (5 + k20)\mathrm{cm}$
    $\rho_\alpha = \pi/4 + k\pi/4$
    $x^{\mathrm{p}} = x^{\mathrm{p}}_k + \mathsf{RAND}(3, [-\rho_{\mathrm{x}}, \rho_{\mathrm{x}}])$
    $\alpha = \alpha_k + \mathsf{RAND}(1, [-\rho_\alpha, \rho_\alpha])$
    sc^{p}= (x^{p},α)
    c^{p}= MAP2(sc^{p})
  until INVKIN(c^{p}) = true
  return c^{p}

B. Sample interconnection

The procedure ConnectSample, shown as Algorithm 3, performs the connection of a configuration c to a roadmap, G. It uses the following functions:

- FINDNEIGHBORS(c): Find the K-nearest neighbors of c from all the nodes of the roadmap G. With a given low probability, adds the goal or the initial configurations as neighbors.
- SAMECOMP(c_i,c_j): Returns true if c_i and c_j belong to the same connected component of G, or false otherwise.
- LOCALPLAN(c_i,c_j): Returns true if a collision-free path between c_i and c_j exists, or false otherwise. This path is computed from a linear interpolation in $\mathcal{SC}$, and therefore all the configurations of the path are such that the orientation of the palm satisfies the constraints. This has a unique exception when connecting c_{ini}, since the initial configuration is not required to satisfy them (c_{goal} satisfies it by construction), therefore neither do the edges connecting c_{ini} with any node in G.
- ADDNODE(c,G): Adds node c to graph G.
- ADDEDGE(c_i,c_j,G): Adds edge (c_i,c_j) to G.
- UPDATE(G): Updates the connected components of G.

C. The general PRM algorithm

Algorithm 4 presents the construction of the roadmap. It uses the following function:

- INTERPOLATE(c^p_i,c^p_j): Returns a configuration of the palm, c^{p}_k, obtained as a linear interpolation between configurations c^p_i and c^p_j, and the interpolation parameter $k \in [0,1]$, such that $c^{\mathrm{p}}_k = c^p_i$ when $k = 0$ and $c^{\mathrm{p}}_k = c^p_j$ when $k = 1$.
- COLLISION(c): Informs whether the configuration $c \in \mathcal{C}$ makes the hand-arm system to be in collision with the environment.

The configurations sampled by Algorithm 4 are such that several finger configurations are associated to a single palm configuration. The rationale behind this decision is that in cluttered environments, like it will be usually the case near the goal configuration, collision-free motions will most likely require finger motions.

Algorithm 3 ConnectSample

Require: c, G
  ADDNODE(c,G)
  N = FINDNEIGHBORS(c)
  for all $g \in N$ **do**
    if SAMECOMP(g,c) = false **and** LOCALPLAN(g,c) **then**
      ADDEDGE(g,c,G)
      UPDATE(G)
    end if
  end for
  return G

Algorithm 4 PRM.

Require: c_{ini}, c_{goal}
  $G \leftarrow \varnothing$
  ConnectSample(c_{ini}, G)
  ConnectSample(c_{goal}, G)
  repeat
    if RAND(1,[0,1])< 0.5 **then**
      c^{p}= SamplePalm(c_{goal})
    else
      (c^{p}_k, k) = INTERPOLATE(c^p_{goal}, c^p_{ini})
      c^{p}= SamplePalm(c^{p}_k, k)
    end if
    for $i = 1$ to cloudsize **do**
      c^{f}= SampleFingerJoints()
      $c = (c^{\mathrm{f}}, c^{\mathrm{p}})$
      if COLLISION(c) = false **then**
        ConnectSample(c, G)
      end if
    end for
    if time>maxtime **then**
      return failure
    end if
  until SAMECOMP(c_{ini}, c_{goal}) = true
  return G

Once the roadmap is built, the A* algorithm is used to connect the initial and the goal configurations. Then, the solution path is smoothed. The smoothing procedure tries to connect all the nodes of the solution path between them, except for the initial node. The nodes of the solution path and all the collision-free edges found are inserted in a new graph together with the edge connecting the initial node. Then the A* is run on this new graph, obtaining a smoothed shorter path. The initial node is treated differently because c_{ini} may not belong to the solution manifold satisfying the orientation constraint, and the path that connects it to another configuration is desired to be as short as possible. If it were included in the smoothing procedure a longer first edge could result.

VI. EXAMPLES

As an example to illustrate the proposed approach, the SAH hand is required to move, among obstacles, from an unconstrained configuration to a preshape configuration to grasp a can. Fig. 6 (Top) shows the snapshots of the solution path encountered when the task has been programmed with the orientation constraint and using the PMDs. Snapshots **1-2**, and **4-5**, are nodes of the PRM along the solution path, being the first and the last one the initial and goal configurations, respectively. Snapshot **3** is an intermediate configuration. The value of variables cloudsize and maxtime in algorithm 4 has been

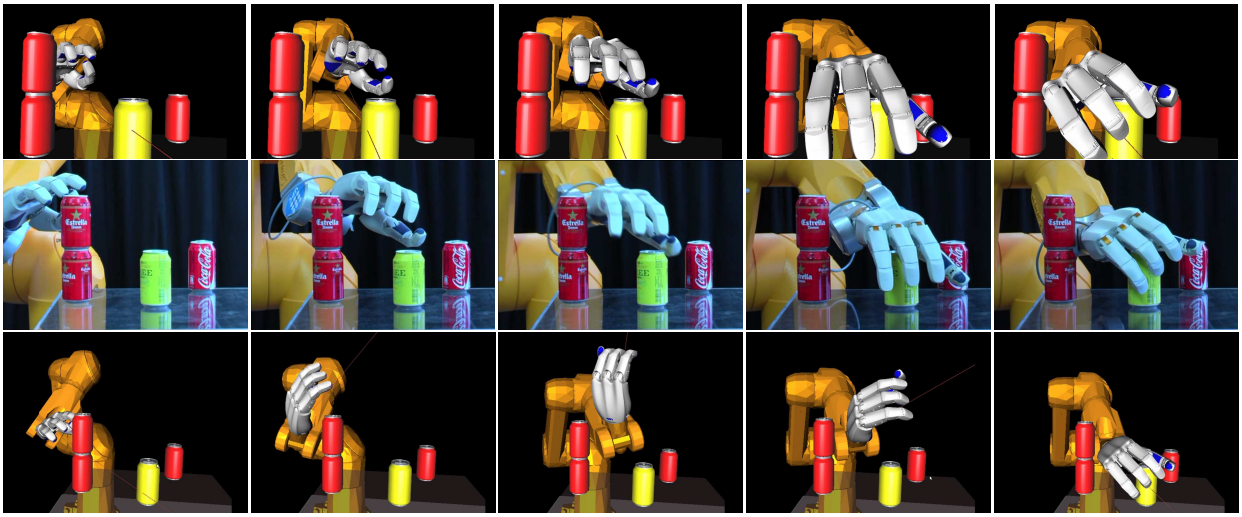


Fig. 6. Snapshots of the results: (Top) task simulation using the orientation constraint; (Middle) task execution using the orientation constraint; (Bottom) task simulation without using the orientation constraint .

TABLE II
COMPARISON BETWEEN ALTERNATIVES

Constraints	Success	# Nodes	Sampling Time (s)	Total Time (s)
With	100%	90 ± 71	8.9 ± 6.7	85 ± 68
Without	62%	37 ± 9	169 ± 85	198 ± 80

set to 3 and 300 s, respectively.

The same task has been programmed without using the orientation constraint. Fig. 6 (Bottom) shows snapshots of this later case corresponding to nodes of the PRM along the solution path (the initial and goal are the same as before). Table II shows the results of the comparison in terms of number of nodes and computational efficiency measured as computational time (running on a PC@3GHz) used by the sampling procedures and by the PRM algorithm (which includes the time spent by the local planner). The sampling time is much shorter when considering constraints because in this case samples are more often collision-free and feasible (i.e. with an existing inverse kinematics solution for the robot).

The benefits of using PMDs for a better efficiency and human-like appearance, was shown in [5]. The present proposal of using virtual constraints further benefits in both senses as shown in the table and in the snapshots. The task has actually been executed in the Lab, as shown in Fig. 6 (Middle), and both simulation and real execution can be seen in the accompanying video, where the real execution goes beyond the preshape configuration and grasps and lifts the can.

VII. CONCLUSIONS

This paper has presented an efficient path planning method to find collision-free paths for a hand-arm system from an initial unconstrained configuration to a final grasping (or preshape) one. Efficiency has been obtained by reducing the dimensionality of the problem by both coupling the finger joints in a similar way as the human hand, and coupling the orientation and position of the palm to meet additional requirements. The resulting planner produces collision-free movements of the hand-arm system as a sequence of anthropomorphic natural postures. Future work is focused on the use of rapidly exploring random trees instead of a roadmap, and in the consideration of a wider range of constraints for the motion of the palm.

REFERENCES

[1] A. Bicchi, "Hands for dexterous manipulation and robust grasping: a difficult road toward simplicity," *IEEE Trans. on Robotics and Automation*, vol. 16, no. 6, pp. 652–662, 2000.

[2] M. Santello, M. Flanders, and J. F. Soechting, "Postural hand synergies for tool use," *Journal of Neuroscience*, vol. 18, no. 23, pp. 10 105–10 115, December 1998.

[3] M. T. Ciocarlie and P. K. Allen, "Hand posture subspaces for dexterous robotic grasping," *The International Journal of Robotics Research*, vol. 28, no. 7, pp. 851–867, July 2009.

[4] A. Tsoli and O. C. Jenkins, "2d subspaces for user-driven robot grasping," in *Proc. of the RSS 2007 Workshop on Robot Manipulation: Sensing and Adapting to the Real World*, 2007.

[5] J. Rosell, R. Suárez, C. Rosales, J. A. García, and A. Pérez, "Motion planning for high DOF anthropomorphic hands," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2009, pp. 4025–4030.

[6] R. Suárez, J. Rosell, A. Pérez, and C. Rosales, "Efficient search of obstacle-free paths for anthropomorphic hands," in *Proc. of the IEEE/RSJ Int. Conf. on Int. Robots and Systems*, 2009, pp. 1773–1778.

[7] A. Safonova, J. K. Hodgins, and N. S. Pollard, "Synthesizing physically realistic human motion in low-dimensional, behavior-specific spaces," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 514–521, 2004.

[8] Y. Mezouar and F. Chaumette, "Path planning for robust image-based control," *IEEE Trans. on Robotics*, vol. 18, no. 4, pp. 534–549, 2002.

[9] M. Kazemi, K. Gupta, and M. Mehrandezh, "Global path planning for robust visual servoing in complex environments," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2009, pp. 326 – 332.

[10] M. Baumann, S. Léonard, E. Croft, and J. Little, "Path planning for improved visibility using a probabilistic road map," *IEEE Trans. on Robotics*, vol. 26, no. 1, pp. 195–200, 2010.

[11] A. Rodríguez, L. Basañez, and E. Celaya, "A relational positioning methodology for robot task specification and execution," *IEEE Trans. Robot.*, vol. 24, no. 3, pp. 600–611, 2008.

[12] A. Rodríguez, A. Pérez, J. Rosell, and L. Basañez, "Sampling-based path planning for geometrically-constrained objects," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2009, pp. 2074–2079.

[13] Schunk GmbH & Co. KG, "Schunk anthropomorphic hand," http://www.schunk.com/, May 2006.

[14] J. I.T., *Principal Component Analysis*. Upper Saddle River, NJ, USA: Springer Series in Statistics, 2002.