Proceedings of the 2002 IEEE/RSJ
Intl. Conference on Intelligent Robots and Systems
EPFL, Lausanne, Switzerland • October 2002

Calculating hand configurations for precision and pinch grasps

Ch. Borst, M. Fischer and G. Hirzinger

Institute of Robotics and Mechatronics
German Aerospace Research Center (DLR), Wessling, Germany
Christoph.Borst@dlr.de, Max.Fischer@dlr.de

Abstract

Usually, grasp planning can be split up into two phases: In the first phase one tries to find a set of contacts that allow for stable grasping of an object. This phase has been of major research interest, which is also reflected in the (reasonable) definition of a grasp as a set of contact points. In the second phase a feasible hand pose that realizes the grasp with a given hand is calculated. While this point is important for a practical grasp planning system, it has either been considered trivial or been solved by crude heuristics in most cases.

Here we present an approach for calculating the hand and finger pose for a given grasp. The problem is formulated as a constraint satisfaction problem and then solved using optimization techniques. The method is applied to two different grasp types: To the well known precision grasp and to the pinch grasp which is the grasp type preferred by men when grasping small objects.

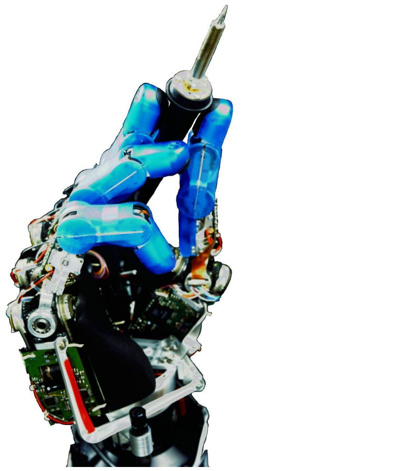

Figure 1: *The DLR Hand II executing a pinch grasp on a solder pen*

1 Introduction

In recent years a couple of new artificial hands have been designed and developed by different research groups [3], [4], [5], [8], [12], [17]. In teleoperation settings the extended features of these hands can already be used and demonstrated well. Systems for autonomous grasping and manipulation in contrast, are still a topic of research.

For such systems the grasp analysis and planning part has been of major interest. Most approaches concentrated on precision grasps, where only the fingertip is in contact with the object to grasp [7], [14], [15], [16].

Assuming one point contact for each fingertip the grasp analysis problem can be solved independently of any manipulator. A grasp in this context can be defined as the contact points on the object and their surface normals. Therefore algorithms for generating optimal grasps (eg. in terms of force or form closure) considering only the object boundary can be developed (see [1] and [13] for an overview). This makes the grasp theory applicable to any dextrous hand, but it obviously leaves the problem of finding a suitable hand configuration to execute a generated grasp open. Only a few approaches consider this problem and take hand configurations into account (eg. [11]).

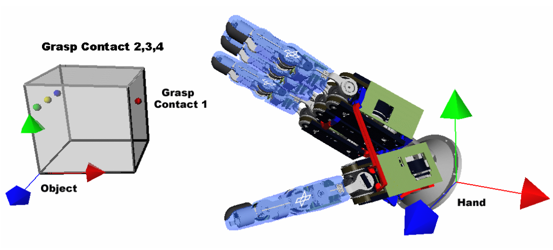

Figure 2: *How to find a hand configuration (wrist frame, finger joints) for a given grasp contact set?*

In this paper we assume a set of grasp contacts that meet certain quality criteria to be given. Such contact sets can be generated by the grasp planning system presented in [2] or any other grasp quality based precision grasp planner. We adress the problem of finding a kinematic feasible wrist position and finger joint configuration for a given contact set (see fig. 2). Our target is to develop a practicable method for an online grasp planning system. So contrary to [10] we ignore possible hand-object-collisions. The integration of these constraints for real world objects with far more than 1000 faces seemed computationally too expensive for an online grasp planner. To ensure collision constraints fast collision detection algorithms are used instead [9]. This decision also allows us to solve the kinematic problem with general methods that are efficient and easy to implement.

We now first take a short look at the geometry of each finger contact point. Then we give a problem formulation for the whole manipulator and finally show an efficient way to solve these problems.

2 Finger Contact Geometry and Pinch Grasp

The easiest and most commonly used fingertip model is a sphere. The fingertip's position is then determined by the grasp contact point and the radius of the sphere whereas the orientation of the fingertip's link is unrestricted (see fig. 3).

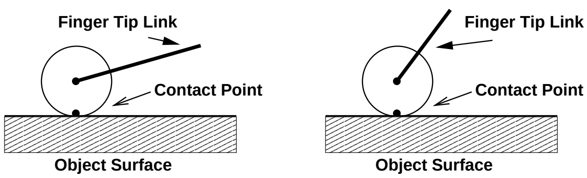

Figure 3: *Simple fingertip model: Same contact reached with two different distal link orientations.*

Robot hands are often built to meet this model (DLR Hand I [3], DIST-Hand [5]) as it makes it possible to calculate finger joint configurations by inverse kinematics - fixed wrist frames assumed. Let's take the DLR Hands finger as an example (fig. 4).

Each finger has four joints. As joint 3 and 4 are coupled there remain three degrees of freedom (3 DOF). The grasp contact for the finger tip restrains 3 DOF (fig. 3) so there is a single solution for the joint angles given a fixed hand wrist position or none if joint limits or link lengths are exceeded.

In our earlier work [2] we used a simple heuristic to determine a wrist position for a given set of grasp contacts and then got the appropriate joint angles with inverse kinematics. This method posed

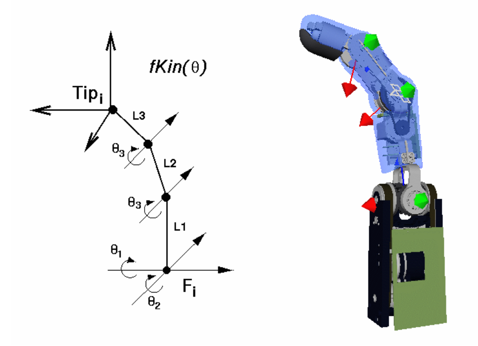

Figure 4: *DLR Hand II kinematic model of finger.*

the problem that some grasps could theoretically be reached by the hand but due to a bad hand pose decision the grasp execution failed. Also this method produces typical "robot shaped" grasps (fig. 5) that are not very human like.

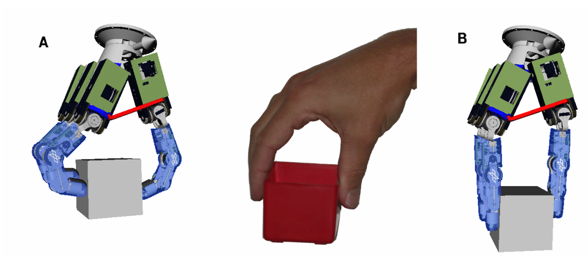

Figure 5: *Typical robot (A) and human (B) precision grasp shape*

This was also the feedback we got from discussions with a neurologist. She mentioned that the normal way humans pick up and manipulate small and light objects is, they use the pad of their fingertips to increase stability. Only people with neurologic deseases would grasp things like our robot hand does. The grasp shape she proposed is is called *pinch grasp* in the grasp taxonomy of Cutkosky and Howe [6] for a two fingered grasp. In the following we want to show the changes on the fingertip model to kinematically describe pinch grasps.

2.1 Pinch grasp fingertip model

Again we take a look how humans usually grasp small objects: They use their finger pads for contacting the object surface (fig. 5). The distal finger link is applied almost parallel to the surface of the object around the contact point.
Transferred to a robot hand that means: The distal finger link (L3) of the hand is modeled as a cylinder

(fig. 6). For a pinch grasp the cylinder axis ($\mathbf{z}_{\mathrm{Tip}_i}$) of the distal finger link should be perpendicular to the surface normal ($\mathbf{z}_{\mathrm{GP}_i}$) in the grasp contact point (GP_i). The intersection of the normal vector in the grasp contact with the distal cylinder axis (Tip_i) is allowed to lie anywhere inside the cylinder. Therefore the length of the distal link in kinematics (l_i) is between 0 and the length of the real distal link of the hand (L3). (Figure 7A shows the kinematic description of the finger)

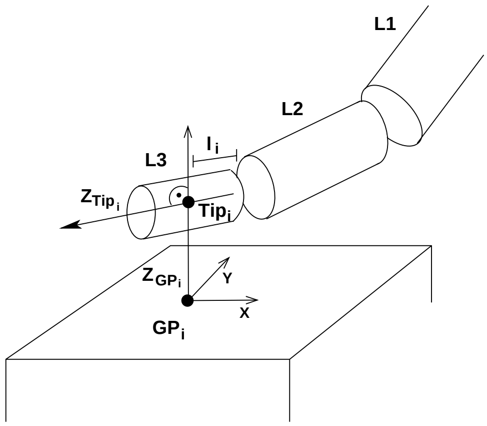

Figure 6: *Contact situation for a pinch grasp*

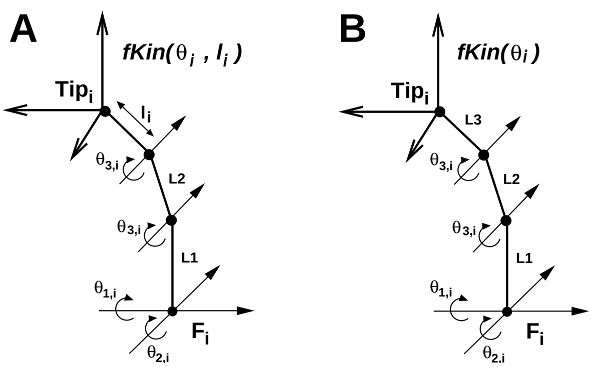

Figure 7: *Finger kinematics with (A) and without virtual prismatic joint (B)*

It's obvious that for this fingertip model a smarter strategy to determine the wrist position and orientation is needed to get feasible grasp configurations.

3 Calculation of pinch grasps

Now we want to calculate a hand pose and a finger joint configuration for a given set of stable grasp contacts. The model of the hand kinematics and an initial hand position is given. First we describe the kinematics of the whole setting, then give a mathematical description of the problems constraints to hold. As there is no direct way to find a solution for the stated problem we last show a method to transform the problem to an unconstrained optimization problem which provides us one of the possible pinch grasp configurations for the given grasp contacts.

3.1 Kinematic Description

With $\mathbf{T}_{A,B}$ we denote the homogeneous matrix representation of a transformation which moves the coordinate system A to the coordinate system B and $\mathbf{T}_{A,B}(x)$ is the homogeneous matrix representing a transformation, parameterized by a 6-dimensional vector describing translation and rotation (e.g. Euler or RPY representation).

$$\mathbf{T}_{A,B} = \begin{pmatrix} \mathbf{x}_{A,B} & \mathbf{y}_{A,B} & \mathbf{z}_{A,B} & \mathbf{t}_{A,B} \\ 0 & 0 & 0 & 1 \end{pmatrix}, \qquad (1)$$

$$\mathbf{T}_{A,B}(\tau) \quad \text{with} \quad \tau = (t_x, t_y, t_z, r_x, r_y, r_z)^T. \qquad (2)$$

The finger kinematics of the DLR Hand II is described in (fig. 7 B). There are four rotational joints and three links in between. Joint 3 and 4 are coupled $1:1$ and described by $\theta_{3,i}$. In order to allow any point on the cylinder of the distal finger link for contact placement, a virtual prismatic joint, described by l_i (for finger i), is introduced (fig. 7 A). The transformation of the i-th fingers base frame to the fingertip frame is then specified by the forward kinematics $f_{\mathrm{Kin}}(\theta_i, l_i)$, where θ_i is the joint vector of finger i.

We model the fingertip as a cylinder of radius r (fig. 6). Assuming the grasp contact frames not to lie directly on the object surface, but in a distance r perpendicular to it, the grasp contact constraints can be simplified. For a pinch grasp that means the grasp contact point lies on the fingertip cylinder axis and for traditional precision grasp the grasp contact coincide with the fingertip frame (sphere center fig. 3).

Now we can describe the setting of an object to be grasped by a n-fingered hand and n given grasp points (see fig. 8):

$$\begin{aligned} \mathbf{T}_{\mathrm{World,Tip}_i} &= \mathbf{T}_{\mathrm{World,Hand}}(\tau) \cdot \mathbf{T}_{\mathrm{Hand},f_i} \cdot f_{\mathrm{Kin}}(\boldsymbol{\theta}_i, l_i), \\ \mathbf{T}_{\mathrm{World,GP}_i} &= \mathbf{T}_{\mathrm{World,Obj}} \cdot \mathbf{T}_{\mathrm{Obj,GP}_i}. \end{aligned}$$

So far this is not describing a grasp as the constraints to ensure contact are missing. But with our previous definitions we can easily give these constraints.

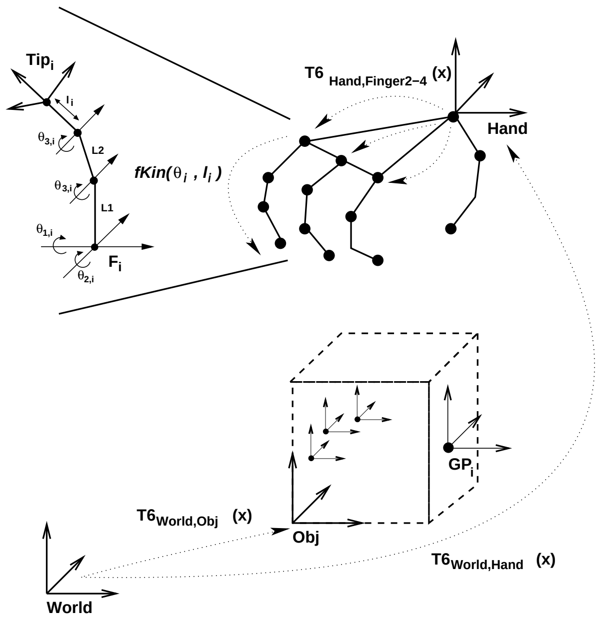

Figure 8: *Kinematic description of a hand grasping an object*

Contact is guaranteed if the *position* of the fingertip frame $\mathbf{t}_{\mathrm{Tip}_i}$is the same as the *position* of the grasp point frame $\mathbf{t}_{\mathrm{GP}_i}$. For a pinch grasp we also want to control the fingertips orientation. The z-axis of the distal finger link should be perpendicular to the z-axis of the grasp contact frame. This leads to the following two constraints:

$$\mathbf{z}_{\mathrm{Tip}_i} \cdot \mathbf{z}_{\mathrm{GP}_i} = 0, \tag{3}$$
$$\mathbf{t}_{\mathrm{Tip}_i} = \mathbf{t}_{\mathrm{GP}_i}. \tag{4}$$

3.2 Problem Formulation

With equation 3 and 4 we have set up the grasp contact constraints for the problem. We also want to guarantee that the grasp can be executed with the real hand. So we have to set the joint angles and the virtual prismatic joint within appropriate intervals.

This introduces a set of inequality constraints:

$$\Theta_j^{\mathrm{low}} \leq \quad \theta_{j,i} \quad \leq \Theta_j^{\mathrm{up}}, \tag{5}$$
$$l^{\mathrm{row}} \leq \quad l_i \quad \leq l^{\mathrm{up}}. \tag{6}$$

Where i is the number of the finger and Θ_j^{low} , Θ_j^{up} the lower and upper limits of joint j. The virtual joint length is restricted by l^{low} and l^{up} (usually 0 .. length of the real link).

We want to point out that the parameters for the hand wrist transform are totally free in our setting. Assuming a robot that carries the hand, the wrist position and orientation may also be a subject to constraints.

The problem we have to solve is:

> Find parameters $\{\tau, \theta_1, l_1, \theta_2, l_2, .., \theta_4, l_4\}$ that fulfill equations 3 to 6.

Let's take a closer look at the contact constraints. We have four equality constraints and four free parameters $(\theta_{1..3}, l)$ for each fingertip and six free parameters for the hand wrist position and orientation. So we expect a solution space rather than a single parameter vector for the problem. Although equations 3 and 4 may look very simple, they are nonlinear and rather complex. Therefore symbolic calculation of the solution space is not possible.
To get a numerical algorithm to solve the problem we transferred it to a pseudo optimization problem.

The principal idea is to formulate an optimization problem without constraints, where the objective function is constructed in such a way that its minimization converges into the solution space of the initial problem. The so found optimal parameter set should then fulfill all the constraints if the solution space was not empty. The value of the objective function for the optimized parameters allows a decision whether a solution was found or not.

We explicitly want to state that for the original problem there is nothing to optimize although optimization criteria (eg. maximal exertion of finger force) can be added to our approach. We just convert the problem to get an algorithm for finding any solution for our "pinch grasp problem".

The next section describes the construction of an objective function for the above problem.

3.3 The Optimization Algorithm

We decided to formulate the problem with penalty terms in the objective function to ensure constraints. First we shortly present the generic method and the requirements the constraint terms have to meet. Then we show our solution for this particular problem.

Generally one has an objective function $f(x)$ which is to be minimized and a set of equality $h(\mathbf{x})$ and unequality $g(\mathbf{x})$ constraints:

$$\mathrm{argmin}_x(f(\mathbf{x})) \quad \text{with} \quad h_i(\mathbf{x}) = 0, \quad g_j(\mathbf{x}) \leq 0, \tag{7}$$

with $\quad i = 1, 2, .., n \quad j = 1, 2, .., m$ and $\mathbf{x} \in \Re^n$.

This can be transformed to a sequence of unconstrained optimization problems

$\min(H(\mathbf{x}, p_1, .., p_{m+n}))$ with

$$H(\mathbf{x}, p_1, p_2, .., p_{m+n}) = f(\mathbf{x}) + \sum_{k=1}^{m+n} S_k(\mathbf{x}, p_k) \quad (8)$$

with $p_k > 0, \quad p_k \to \infty \quad k = 1, 2, ...$

Let M be the solution space for the parameter set $\mathbf{x}$ of the initial problem. Then S is chosen so that for the additional terms $S_k(\mathbf{x}, p_k)$ holds

$$\lim_{p_k \to \infty} S_k(\mathbf{x}, p_k) \begin{cases} \to C & \text{if} \quad \mathbf{x} \in M, \\ \to \infty & \text{if} \quad \mathbf{x} \notin M. \end{cases} \quad (9)$$

For the algorithm that means: The p_k parameters are increased in each step. So if the sequence of unconstrained problems converges to a minimum, the found parameter set is in the solution space of the initial problem. Otherwise it would violate any of the constraints and no minimum of the sequence could be found (as p_k increases). Note that the single optimization steps always converge to a local minimum but the sequence of optimization does only if there is a $\mathbf{x}$ that fulfills all constraints.

The only thing left to be done is to pick suitable penalty functions and parameterize them for the constraints.

In our particular case there is no initial objective function given. So $f(\mathbf{x}) := 0$. On choosing appropriate penalty function types we considered the following:

We have two different types of terms, equality constraints for the contact terms and inequality constraints for the joint limits. Ideally we would like to have a continuous objective function where all terms derived from the constraints are active in the whole domain. As an alternative one could construct a partial objective function where terms are switched on and off. Last but not least we wanted to use an off the shelve nonlinear least squares minimizer so we tried to get a quadratic form. We decided to form a continuous objective function with different terms for equality and inequality constraints.

For the equality constraints we use a simple quadratic penalty term:
Let $h(\mathbf{x}) = 0$ be an equality constraint, we add

$$S_k(\mathbf{x}, p_k^e) = p_k^e h^{\mathbf{2}}(\mathbf{x}) \quad (10)$$

to the objective function. Figure 9 shows the function type for different p_k. If equality is fulfilled the resulting penalty value is zero.

For the joint limit constraints we use an exponential penalty term that is nearly zero if the joint is in its limits and rises rapidly on the joints boundary:
Let $\Theta^{\text{low}} \leq \theta \leq \Theta^{\text{up}}$ be a bound constraint, we add

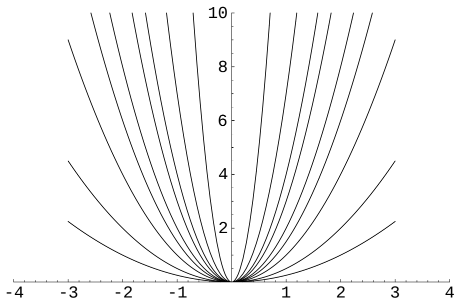

Figure 9: *Quadratic functions for modeling equality constraints ($p_k^e \in \{0.25, 0.5, 1, 1.5, 2, 3, 4, 7, 20\}$)*

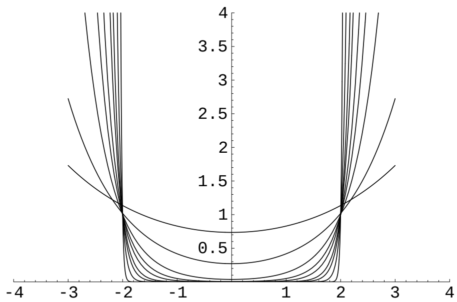

Figure 10: *Plot of terms for modeling bound constraints ($p_k^b \in \{0.25, 0.5, .., 20\}$, $\Theta^{\text{low}} = -2, \Theta^{\text{up}} = 2$)*

$$S(\mathbf{x}, p_k^b) = \exp^{p_k^b(-\theta + \Theta^{low})} + \exp^{p_k^b(\theta - \Theta^{up})} \quad (11)$$

to the objective function. Figure 10 shows the function type for bound constraints with different p_k^b. One can easily see that these terms meet requirement (eq. 9).

With these types of functions we can reformulate the initial problem (equations 3 to 6). In the following, $\mathbf{x}$ is a shortcut for the variable parameters (hand wrist frame, joint angles and length of prismatic joint):

$$\mathbf{x} = (\tau, \boldsymbol{\theta}_1, l_1, \boldsymbol{\theta}_2, l_2, \boldsymbol{\theta}_3, l_3, \boldsymbol{\theta}_4, l_4) \quad (12)$$

The whole objective function for the stated problem with four fingers, starting with the two equality constraints (eq. 3, 4) then adding the bound constraints for the 3 joints (eq. 5) and the virtual prismatic joint (eq. 6) is then:

$$H(\mathbf{x}, p_{1,1}^e, .., p_{4,2}^e, p_{1,1}^b, .., p_{4,4}^b) = \quad (13)$$
$$\sum_{i=1}^{4} \Big[\quad p_{i,1}^e \| t_{\text{Tip}_i}(\mathbf{x}) - t_{\text{GP}_i}(\mathbf{x}) \|^2 +$$
$$p_{i,2}^e (z_{\text{Tip}_i}(\mathbf{x}) \cdot z_{\text{GP}_i}(\mathbf{x}))^2 +$$

$$\sum_{j=1}^{3}\left(\exp^{p^b_{i,j}(-\theta_{j,i}+\Theta_j^{\mathrm{low}})} + \exp^{p^b_{i,j}(\theta_{j,i}-\Theta_j^{\mathrm{up}})}\right) + \exp^{p^b_{i,4}(-l_i+l^{\mathrm{low}})} + \exp^{p^b_{i,4}(l_i-l^{\mathrm{up}})}\Bigg].$$

With this objective function implemented in Mathematica and the builtin Levenberg-Marquardt minimizer we received the results shown in Figure 11 to 12.

3.4 Calculating other grasp shapes

This method can easily be applied to other grasp shapes. One has only to find proper constraints or optimization criteria that describe the preferred shape. We will show this for the traditional precision grasp. What is different to the pinch grasp setting?

- The virtual prismatic joint in the distal finger link needs to be removed. Instead, the contact frame gets fixed to the finger tip.
- The orientation constraint for the finger tip may be omitted or relaxed in such a way that the distal finger link does not collide with the contact tangent plane. That means the angle between finger tip z-axis and contact normal has more than 90 degrees.

This leads to the following constraints for traditional precision grasps:

$$\begin{aligned} \mathbf{z}_{\mathrm{Tip}_i} \cdot \mathbf{z}_{\mathrm{GP}_i} &\leq 0, && (14)\\ \mathbf{t}_{\mathrm{Tip}_i} &= \mathbf{t}_{\mathrm{GP}_i}, && (15)\\ \Theta_j^{\mathrm{low}} \leq \theta_{j,i} &\leq \Theta_j^{\mathrm{up}}. \end{aligned}$$

Again we can transform this to an optimization problem using the same kind of penalty functions.

$$H(\mathbf{x}, p^e_1, .., p^e_4, p^b_{1,1}, .., p^b_{4,4}) = \quad (16)$$
$$\sum_{i=1}^4 \Bigg[\exp^{p^b_{i,4}\left(z_{\mathrm{Tip}_i}(\mathbf{x})\cdot z_{\mathrm{GP}_i}(\mathbf{x})\right)} + p^e_i \|t_{\mathrm{Tip}_i}(\mathbf{x}) - t_{\mathrm{GP}_i}(\mathbf{x})\|^2 + \sum_{j=1}^{3}\left(\exp^{p^b_{i,j}(-\theta_{j,i}+\Theta_j^{\mathrm{low}})} + \exp^{p^b_{i,j}(\theta_{j,i}-\Theta_j^{\mathrm{up}})}\right)\Bigg].$$

4 Results

We have implemented the optimization problem in Mathematica and tested the proposed method with some generated grasp contact sets on different objects. Some of the resulting hand configurations are presented in figure 11 to 12. We noticed that the minimization of the objective function as stated in equation 13 converged into slightly different local minima, dependent on the initial hand configuration. The residual errors of the minima were almost equal. This is what we expected, recalling that we have a solution space with many equal solutions. We added a simple optimization criterion, maximize the distance to the limits for each finger joint, then the different minima converged to a single solution vector. So it should be possible to search for special configurations that for example optimize the force the fingers can exert on the object boundary or that allow better manipulation of the object for certain tasks.

5 Future Work

One topic, as mentioned above, is the optimization in the solution space. One can try to optimize the joint angles of the fingers in such away that the force the finger can exert at the contact is a maximum. For given tasks one can try to optimize the forces that can be exerted on the object best suits the task.
Another big subject is the stability analysis and models for contact geometry of the pinch grasp. Humans prefer this grasp type as it gives more stability to the grasp and the contact region on the objects grows. It has to be studied how one can model this contact type and also if a pinch grasp can be controlled more robust than a fore fingertip grasp. At last we have to integrate that method in our grasp planner in a way that the flexibility of the method can be used to plan individual grasp shapes for different tasks.

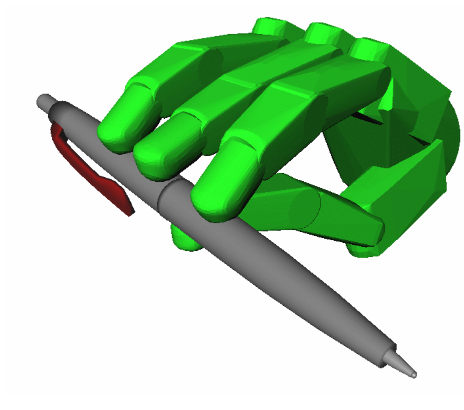

Figure 11: *Planned pinch grasp on a pen*

References

[1] Antonio Bicchi and Vijay Kumar. Robotic grasping and contact: A review. In *Proc. IEEE Conf. on Robotics and Automation*, pages 348 – 353, San Francisco, California, April 2000.

[2] Ch. Borst, M. Fischer, and G. Hirzinger. A fast and robust grasp planner for arbitrary 3d objects. In

Figure 12: *Planned pinch grasp on a cocktail glass*

Figure 13: *Planned pinch grasp on a cellular phone*

Proc. IEEE Conf. on Robotics and Automation, pages 1890–1896, Detroit, Michigan, May 1999. .

[3] J. Butterfass, G.Hirzinger, S. Knoch, and H. Liu. DLR's Multisensory Articulated Hand. part I: Hard- and software architecture. In *Proc. IEEE Conf. on Robotics and Automation*, pages 2081 – 2086, Leuven, 1998. .

[4] J. Butterfass, M. Grebenstein, H. Liu, and G. Hirzinger. DLR-Hand II: Next Generation of Dextrous Robot Hand. In *Proc. IEEE Conf. on Robotics and Automation*, pages 109 – 114, Seoul, Korea, May 2001. .

[5] Andrea Caffaz and Giorgio Cannata. The Design and Development of the DIST-Hand Dextrous Gripper. In *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 2075 – 2080, Leuven, Belgium, May 1998.

[6] M. Cutkosky and R. D. Howe. Human grasp choice and robotic grasp analysis. In Subramanian T. Venkataraman and Thea Iberall, editors, *Dextrous Robot Hands*, chapter 1. Springer Verlag, 1990.

[7] C. Ferrari and J. Canny. Planning Optimal Grasps. In *Proc. IEEE Conf. on Robotics and Automation*, pages 2290–2295, Nice, France, May 1992.

[8] Th. Fischer and H. Woern. Structure of a robot system: Karlsruhe dextrous hand II. In *Proc. of Mediteranean Conf. on Control and Systems*, 1998. .

[9] Stefan Gottschalk, Ming C. Lin, and Dinesh Manocha. OBB-Tree: A Hierarchical Structure for Rapid Interference Detection. In *Proc. of ACM Siggraph*, New Orleans, Louisiana, USA, August 1996. .

[10] Yisheng Guan and Hong Zhang. Kinematic Feasibility Analysis of 3D Grasps. In *Proc. IEEE Conf. on Robotics and Automation*, pages 2197 – 2202, Seoul, Korea, May 2001.

[11] R. D. Hester, M. Cetin, C. Kapoor, and D. Tesar. A criteria-based approach to grasp synthesis. In *Proc. IEEE Conf. on Robotics and Automation*, pages 1255 – 1260, Detroit, Michigan USA, May 1999.

[12] C. S. Lovchik and M. A. Diftler. The robonaut hand: A dextrous robot hand for space. In *Proc. IEEE Conf. on Robotics and Automation*, pages 907 – 912, Detroit, Michigan, USA, May 1999.

[13] Xanthippi Markenscoff, Luqun Ni, and Christos H. Papadimitriou. The geometry of grasping. *The Intl. Journal of Robotics Research*, 9(1):61 – 74, February 1990.

[14] Brian Mirtich and John Canny. Easily Computable Optimum Grasps in 2-D and 3-D. In *Proc. IEEE Conf. on Robotics and Automation*, pages 739 – 747, San Diego, CA USA, May 1994.

[15] B. Mishra, J.T. Schwartz, and M. Sharir. On the existence and synthesis of multifinger positive grips. *Algorithmica, Special Issue: Robotics*, 2(4):541 – 558, November 1987.

[16] Van-Duc Nguyen. Constructing Force-Closure Grasps in 3D. In *Proc. IEEE Conf. on Robotics and Automation*, pages 240 – 245, Raleigh, North Carolina, USA, April 1987.

[17] S. Schulz, Ch. Pylatiuk, and G. Bretthauer. A new ultralight anthropomorphic hand. In *Proc. IEEE Conf. on Robotics and Automation*, pages 2437–2441, Seoul, Corea, May 2001.