GRASP PLANNING & FORCE COMPUTATION FOR DEXTROUS OBJECT MANIPULATION WITH MULTI-FINGER ROBOT HANDS

Günter Wöhlke

INTERNATIONAL COMPUTER SCIENCE INSTITUTE
1947 Center St., Suite 600, Berkeley, California 94704-1198, USA

Technical Report TR-94-018 (revised version)

August 1994

Abstract

This paper deals with the problems of grasp planning and force computation that occur when objects have to be manipulated with dextrous multi-finger robot hands. Assuming that the initial contact points at the object surface are pre-selected, the joint motions to perform a desired operation according to a given object trajectory are determined. This is done under the consideration of the effect of rolling and slipping of the fingertips. Another point is the computation of appropriate grasp forces (internal forces) according to given object forces/moments needed to ensure a stable and secure grip configuration from which the joint torques are derived. This leads to an optimization problem that can be solved with two different approaches based on search procedures for finding the maximum (best fitting value). As major result, the force optimization problem could be generalized for the case of arbitrary robot hands and contact situations that are specified by the following parameters: number of fingers fno, number of joints per finger jno_i, chosen contact models $cmod_i$ and null space dimension ndim. Especially, the resulting object motions and contact forces are demonstrated at a simulated example of a peg-in-hole insertion task with the Karlsruhe Dextrous Hand.

1. Introduction

Compared with a conventional two-jaw gripper, multi-finger hands have inherent advantages: first, they have a higher grip stability due to three and more contact points at the object, and second, it is possible to impart movements onto the gripped object by exerting adequate finger forces. An adequate built-in sensor equipment permits easy data processing and information gathering which enables the use of robot grippers as exploration tool in unstructured environments. Another kind of application for dextrous hands is found in well-structured working areas of industrial manufacturing systems: to perform complex assembly tasks, insertion operations and object manipulations.

Dextrous multi-finger hands represent an interesting multi-disciplinary research area. The development of such a hand is a technological and scientific challenge, and before obtaining a fully operational gripper for the application in manufacturing systems, a lot of problems are to be solved. Up to the present time a number of multi-finger hands have been developed, but there is only little work dealing with the problem of determining efficiently all parameters required for object handling operations with articulated robot hands. A survey of past gripper developments was presented in [1].

Especially, at the Institute for Real-Time Computer Control Systems and Robotics of the University of Karlsruhe a modular three-finger hand with 9 degrees of freedom, the Karlsruhe Dextrous Hand (Fig. 1), has been in development since 1988 [2], [3], [4]. Because the main goals of this project are to investigate dextrous grasping and manipulation, the research efforts are focused on the topics mechanical design, sensor integration, intelligent hand control, programming and grasp planning.

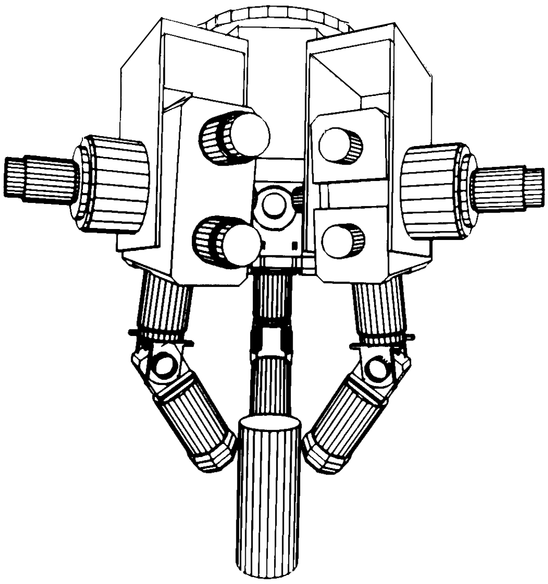

Fig. 1 : The Karlsruhe Dextrous Hand

Therefore, the realized framework consists of a distributed real-time control system in combination with a graphical programming and simulation system [5] that permit reliable operation of the multi-finger gripper. Also, a sophisticated approach to automated grasp planning [6] has been developed. To do this, the task parameterization problem (described here) - the determination of the motion and especially the force parameters required to perform a specific object manipulation - has to be solved.

2. Grasp planning and fine manipulation

One of the first developments was a three-fingered robot hand from Okada [7], [8]. Okada has defined a symbolic notation for the fundamental finger motions. The possible operation modes which can be performed are simple bending or extending, pressing inside or outside, adduction and abduction. Complex motions can be programmed by joining several finger operations together, but this is a difficult and cumbersome task because the exact location of each finger has to be known a-priori.

To avoid the specification of stereotype finger motion sequences with a large number of repeating parameters, the four-fingered human-like Utah/MIT Dextrous Hand [9], [10] is programmed by hand motion primitives like close-hand, grasp-object, spread-fingers. These hand primitives [11] together with the free, guarded and compliant motion types presented later in this section, represent an elegant way to describe a desired hand action schemata with respect to a subset of fingers and joints. A LISP-based programming environment for the three-fingered Salisbury Hand is described in [12].

Attempts have also been made to synthesize grasps to perform specific parts mating operations or assembly tasks. In [13] an algorithm based on geometric-mechanic calculations has been developed to generate stable, force-closure grasps. The authors of [14], [15], [16] have analyzed the human object handling behaviour to derive grasp strategies for articulated hands. Here, the problems arise from the different kinematic hand/finger structure and the difficulties to support force control strategies.

To perform a dextrous manipulation with a robot hand, the object has (1) to be grasped at the contact points with sufficient internal forces and (2) to be moved under the presence of external forces and moments acting onto it. To increase flexibility and to decrease complexity, three basic motion types are introduced consisting of a motion, a force and an optional sensor or stiffness specification:

- The simplest motion is realized as free move by which the execution of an object motion in the working space of the hand is totally unconstrained. Because a collision with other components in the environment is not expected, there is no need of a sensor-guided control strategy. This motion type is normally used as coarse operation for both hand positioning (reaching) and object handling.

  FREE_MOVE ::= <motion_spec> <force_spec>

- The guarded move is a motion in the free space that is executed under sensor supervision and interrupted when a termination condition is fulfilled which depends on the actual sensor configuration and the way the sensor information is interpreted. This motion type is mostly used to control grasp operations, transfer motions and dextrous manipulations after an initial contact has occurred.

  GUARDED_MOVE ::= <motion_spec> <force_spec> <sensor_spec>

- The most complex motion is realized as compliant move by which the execution of the object motion is not pre-computed but depends on an active stiffness control strategy under sensor-guidance to perform a compliant behaviour of the hand/finger system. This motion type is normally used to perform parts mating operations like peg-in-hole insertion or light-bulb/screw turning tasks where the parameters are constrained by environmental constraints and uncertainties of the object location.

  COMPLIANT_MOVE ::= <motion_spec> <force_spec> <stiffness_spec>

Because all motions and forces are specified in terms of the manipulation object, the control system has to perform the transformation from object- into gripper-based commands to provide the trajectory generation and interpolation routines with the parameter values required for command execution.

Starting from a given object trajectory in hand coordinates and considering rolling and slipping of the fingertips, the contact velocities and joint velocities are computed from which the required joint angles are determined. The mathematics of manipulation - the geometry, kinematics and static forces - is described according to the work of [16], [17], [18], [19], [20]. Starting from external object forces and moments, the optimization problem for the computation of internal grip forces can be solved with two different approaches from which the contact forces and the joint torques are derived.

3. Motion computation

Because the motion planner should be well-suited for off-line programming and simulation of dextrous manipulations with multi-fingered robot hands, several assumptions have to be made to compute an object motion: (1) especially the object and fingertip geometry has to be given in analytical form, (2) the hand/finger kinematics has to be known and (3) the initial contact points of the fingertips at the object surface have to be pre-selected. From all this information, its possible to determine the corresponding finger motions which perform the desired operation in form of joint coordinates.

3.1 Description of the object-trajectory

The manipulation object is represented by the frame C_o fixed at the center of mass while the robot hand is represented by the frame C_h fixed at the hand base. The location of the object with respect to the frame C_h is described by the (3×1) translation vector ${}^h\mathbf{r}_{o,h}$ and the (3×3) rotation matrix $\mathbf{A}_{o,h}$. Through an elementary motion of the object, like a translation, a rotation or a screw motion, the actual frame C_o can be transformed into the frame C_o'. The new location of the object with respect to the frame C_h is then described by the translation vector ${}^h\mathbf{r}_{o,h}'$ and the rotation matrix $\mathbf{A}_{o,h}'$.

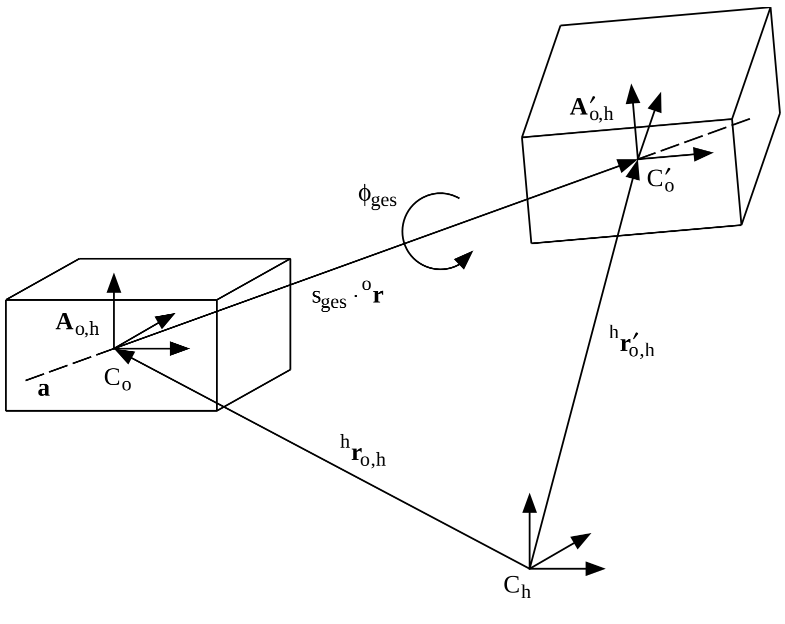

Fig. 2 : Object motion with respect to hand base

A screw motion (Fig. 2) is composed of a rotation around an arbitrary axis $\mathbf{a}$: ${}^o\mathbf{x} = {}^o\mathbf{p} + u \cdot {}^o\mathbf{r}$ with angle ϕ (maximal angle ϕ_{ges}) and a translation along the unit direction-vector ${}^o\mathbf{r}$ of $\mathbf{a}$ with length s (maximal length s_{ges}). Thus, if the temporal relationship of s(t) and ϕ(t) are both known, the exact location of the object can be pre-computed at each timestep t. The translational and rotational part of the motion is given in the following form whereby the parameters are determined with the functions **t** and **R** ($s\phi = \sin\phi$, $c\phi = \cos\phi$, $v\phi = 1-\cos\phi$; the r's and p's are the cartesian components of ${}^{o}\mathbf{r}$, ${}^{o}\mathbf{p}$):

$$ {}^{h}\mathbf{r}'_{o,h} = {}^{h}\mathbf{r}_{o,h} + \mathbf{A}_{o,h} \cdot \mathbf{t}(\mathbf{a},\phi) + \mathbf{A}_{o,h}\left(s \cdot {}^{o}\mathbf{r}\right) \qquad \mathbf{A}'_{o,h} = \mathbf{A}_{o,h} \cdot \mathbf{R}(\mathbf{a},\phi) $$

$$ \mathbf{t}(\mathbf{a},\phi) = \begin{pmatrix} (r_z p_y - r_y p_z)s\phi - \left(r_x r_z p_z - r_z^2 p_x + r_x r_y p_y - r_y^2 p_x\right)v\phi \\ (r_x p_z - r_z p_x)s\phi - \left(r_y r_z p_z - r_z^2 p_y + r_x r_y p_x - r_x^2 p_y\right)v\phi \\ (r_x p_y - r_y p_x)s\phi - \left(-r_y r_z p_y + r_y^2 p_z - r_x r_z p_x + r_x^2 p_z\right)v\phi \end{pmatrix} $$

$$ \mathbf{R}(\mathbf{a},\phi) = \begin{pmatrix} \left(r_y^2 + r_z^2\right)c\phi + r_x^2 & -r_z s\phi + r_x r_y v\phi & r_y s\phi - r_x r_z v\phi \\ r_z s\phi + r_x r_y v\phi & \left(r_x^2 + r_z^2\right)c\phi + r_y^2 & -r_x s\phi + r_y r_z v\phi \\ -r_y s\phi - r_x r_z v\phi & r_x s\phi + r_y r_z v\phi & \left(r_x^2 + r_y^2\right)c\phi + r_z^2 \end{pmatrix} $$

3.2 Geometry of the manipulation system

The hand kinematics is needed to describe a given object trajectory in finger coordinates. Therefore, the relationship between the base frames C_{b_i} of the finger chains and the hand frame C_h are defined by transformation matrices (${}^{h}\mathbf{r}_{b_i,h}, \mathbf{A}_{b_i,h}$) which are constant for a given hand configuration (the index i counts the fingers, the total number of fingers is fno and the number of joints per finger is jno_i). To describe the finger motions, the relationship between the fingertip frames C_{f_i} and the base frames C_{b_i} is defined by transformation matrices (${}^{b_i}\mathbf{r}_{f_i,b_i}, \mathbf{A}_{f_i,b_i}$), the so-called direct kinematics. So, the positions ${}^{h}\mathbf{r}_{f_i,h}$ and orientations $\mathbf{A}_{f_i,h}$ of the fingertips (see Fig. 3 for a finger with $jno_i = 3$ joints) with respect to the hand base can be determined if the actual set of joint angles θ_i is known:

$$ {}^{h}\mathbf{r}_{f_i,h} = {}^{h}\mathbf{r}_{b_i,h} + \mathbf{A}_{b_i,h}\,{}^{b_i}\mathbf{r}_{f_i,b_i}(\theta_i) \qquad \mathbf{A}_{f_i,h} = \mathbf{A}_{b_i,h}\mathbf{A}_{f_i,b_i}(\theta_i)\ , \qquad \theta_i = \left(\theta_1, \ldots, \theta_{jno_i}\right)^T \qquad i = 1,\ldots,fno $$

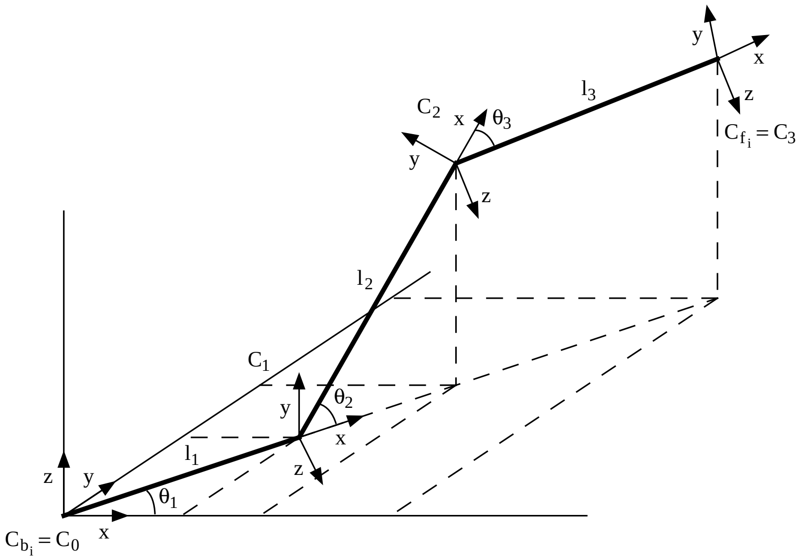

Fig. 3 : Fingertip location with respect to finger base

In the inverse direction, the kinematic problem can be solved directly if the above described relations are invertible. In this case, the joint angles θ_i can be computed from the actual finger positions ${}^{h}\mathbf{r}_{f_i,h}$ which is very fast, but will fail if the effect of rolling or slipping occurs because the exact positions ${}^{h}\mathbf{r}_{t_i,h}$ of the contact points at the fingertips are not known. Nevertheless, for given points of contact, this simple method can be used to determine the initial parameter values of a finger motion.

A more sophisticated computation method, described in [19], [20], starts with the generalized velocity $({}^{h}\mathbf{v}_{o,h}, {}^{h}\mathbf{w}_{o,h})^T$ (linear and angular velocity) of the object to determine the velocity parameters $({}^{h}\mathbf{v}_{o_i,h}, {}^{h}\mathbf{w}_{o_i,h})^T$ of the contact points (the components of the contact velocity $\boldsymbol{v}_i$). According to the contact model, represented by the contact matrix $\mathbf{Q}_i$ (see section 4.1), the respective velocity parameters $({}^{h}\mathbf{v}_{t_i,h}, {}^{h}\mathbf{w}_{t_i,h})^T$ of the fingertips are derived, from which the generalized finger velocities $({}^{h}\mathbf{v}_{f_i,h}, {}^{h}\mathbf{w}_{f_i,h})^T$ are determined. Their relation to the resulting joint velocities is given by a Jacobian:

$$\begin{pmatrix} {}^{h}\mathbf{v}_{t_i,h} \\ {}^{h}\mathbf{w}_{t_i,h} \end{pmatrix} = \begin{pmatrix} {}^{h}\mathbf{v}_{o_i,h} \\ {}^{h}\mathbf{w}_{o_i,h} \end{pmatrix} = \mathbf{Q}_i \cdot \upsilon_i, \quad \begin{pmatrix} {}^{h}\mathbf{v}_{f_i,h} \\ {}^{h}\mathbf{w}_{f_i,h} \end{pmatrix} = \begin{pmatrix} {}^{h}\mathbf{J}_{v_i} \\ {}^{h}\mathbf{J}_{w_i} \end{pmatrix} \dot{\theta}_i \quad \Rightarrow \quad \dot{\theta}_i = \left({}^{h}\mathbf{J}_{v_i}\right)^{-1} \cdot {}^{h}\mathbf{v}_{f_i,h}$$

$$\text{for } \mathbf{Q}_i = \mathbf{Q}_{PF} = (\mathbf{I}_3 \quad \mathbf{0})^T \quad \Rightarrow \quad {}^{h}\mathbf{w}_{t_i,h} = {}^{h}\mathbf{w}_{o_i,h} = 0 \text{ and } \boldsymbol{v}_i = {}^{h}\mathbf{v}_{t_i,h} = {}^{h}\mathbf{v}_{o_i,h}$$

$$\begin{aligned} \begin{pmatrix} {}^{h}\mathbf{v}_{t_i,h} \\ {}^{h}\mathbf{w}_{t_i,h} \end{pmatrix} &= \begin{pmatrix} \mathbf{I}_3 & -\mathbf{A}_{f_i,h}\mathbf{S}\left({}^{f_i}\mathbf{r}_{t_i,f_i}\right) \\ 0 & \mathbf{I}_3 \end{pmatrix} \cdot \begin{pmatrix} {}^{h}\mathbf{v}_{f_i,h} \\ {}^{h}\mathbf{w}_{f_i,h} \end{pmatrix} + \begin{pmatrix} \mathbf{A}_{f_i,h}\,{}^{f_i}\mathbf{v}_{t_i,f_i} \\ \mathbf{A}_{f_i,h}\,{}^{f_i}\mathbf{w}_{t_i,f_i} \end{pmatrix} \\ \begin{pmatrix} {}^{h}\mathbf{v}_{o_i,h} \\ {}^{h}\mathbf{w}_{o_i,h} \end{pmatrix} &= \begin{pmatrix} \mathbf{I}_3 & -\mathbf{A}_{o,h}\mathbf{S}\left({}^{o}\mathbf{r}_{o_i,o}\right) \\ 0 & \mathbf{I}_3 \end{pmatrix} \cdot \begin{pmatrix} {}^{h}\mathbf{v}_{o,h} \\ {}^{h}\mathbf{w}_{o,h} \end{pmatrix} + \begin{pmatrix} \mathbf{A}_{o,h}\,{}^{o}\mathbf{v}_{o_i,o} \\ \mathbf{A}_{o,h}\,{}^{o}\mathbf{w}_{o_i,o} \end{pmatrix} \end{aligned}, \quad \mathbf{S}(\mathbf{r}) = \begin{pmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{pmatrix}$$

The transformations of generalized velocities given in different coordinate systems are performed by above matrix equations (using the skew symmetric matrix **S** as operator for the cross-product) which include the translation vectors **r** and rotation matrices **A** that describe the geometric relationships of the object-finger motion. The computation of rolling finger motions is shown in the following where the contact model is assumed to be point contact with friction (PF), so that only linear and no angular velocity components can be transferred through the fingertip contacts with the object.

3.3 The problem of rolling fingertips

The positions and orientations of the contact points (tip contacts) can be described with respect to the object frame C_o as $({}^{o}\mathbf{r}_{o_i,o}, \mathbf{A}_{o_i,o})$ or with respect to the finger frames C_{f_i} as $({}^{f_i}\mathbf{r}_{t_i,f_i}, \mathbf{A}_{t_i,f_i})$. These vectors and matrices (given as rotations along the directions **d** of the z-axes with rotation angles δ) can be parameterized as functions of two variables (see Fig. 4). Thus, due to rolling of the fingertips, not only the joint angles θ_i but also the contact parameter sets η_i and ξ_i have to be computed.

$${}^{f_i}\mathbf{r}_{t_i,f_i} = \begin{pmatrix} \mathbf{r}_x(\eta_{i_1}, \eta_{i_2}) \\ \mathbf{r}_y(\eta_{i_1}, \eta_{i_2}) \\ \mathbf{r}_z(\eta_{i_1}, \eta_{i_2}) \end{pmatrix} = {}^{f_i}\mathbf{r}_{t_i,f_i}(\eta_i) \text{ with } \eta_i = \begin{pmatrix} \eta_{i_1} \\ \eta_{i_2} \end{pmatrix}, \quad {}^{o}\mathbf{r}_{o_i,o} = \begin{pmatrix} \mathbf{r}_x(\xi_{i_1}, \xi_{i_2}) \\ \mathbf{r}_y(\xi_{i_1}, \xi_{i_2}) \\ \mathbf{r}_z(\xi_{i_1}, \xi_{i_2}) \end{pmatrix} = {}^{o}\mathbf{r}_{o_i,o}(\xi_i) \text{ with } \xi_i = \begin{pmatrix} \xi_{i_1} \\ \xi_{i_2} \end{pmatrix}$$

$$\mathbf{A}_{t_i,f_i} = \mathbf{R}\left(\mathbf{d}_{f_i}, \delta_{f_i}\right) \quad \text{with } \mathbf{d}_{f_i} = 1/\sin\delta_{f_i}\left((0,0,1)^T \times {}^{f_i}\hat{\mathbf{n}}_{t_i,f_i}\right) \text{ and } \delta_{f_i} = \cos^{-1}\left((0,0,1)\cdot {}^{f_i}\hat{\mathbf{n}}_{t_i,f_i}\right)$$
$$\mathbf{A}_{o_i,o} = \mathbf{R}\left(\mathbf{d}_{o_i}, \delta_{o_i}\right) \quad \text{with } \mathbf{d}_{o_i} = 1/\sin\delta_{o_i}\left((0,0,1)^T \times {}^{o}\hat{\mathbf{n}}_{o_i,o}\right) \text{ and } \delta_{o_i} = \cos^{-1}\left((0,0,1)\cdot {}^{o}\hat{\mathbf{n}}_{o_i,o}\right)$$

One of the relations is given by the fact that the generalized velocities of the contact points at time t (constant during manipulation) must be the same with respect to the object and the fingertips. The second relationship is given by the fact that both the object and the fingertips must stay in contact, or equivalent, that the position vectors to the actual points of contact must be the same. Splitted into components, differentiated and subtracted from the first relationship, the following equations arise:

$$ {}^h\mathbf{v}_{t_i,h} = {}^h\mathbf{v}_{o_i,h} \quad \Rightarrow \quad {}^h\mathbf{v}_{f_i,h} - \left(\mathbf{A}_{f_i,h}\mathbf{S}\left({}^{f_i}\mathbf{r}_{t_i,f_i}\right)\right){}^h\mathbf{w}_{f_i,h} = {}^h\mathbf{v}_{o,h} - \left(\mathbf{A}_{o,h}\mathbf{S}\left({}^{o}\mathbf{r}_{o_i,o}\right)\right){}^h\mathbf{w}_{o,h}$$
$$ {}^h\mathbf{r}_{t_i,h} = {}^h\mathbf{r}_{o_i,h} \quad \Rightarrow \quad \mathbf{A}_{f_i,h}\,{}^{f_i}\mathbf{v}_{t_i,f_i} = \mathbf{A}_{o,h}\,{}^{o}\mathbf{v}_{o_i,o}$$

The equations for the contact points at time t together with the continual change of the contact points during manipulation (expressed in form of the two velocity terms ${}^{f_i}\mathbf{v}_{t_i,f_i}$ and ${}^{o}\mathbf{v}_{o_i,o}$) describe the resulting motions of the actual contact points. This yields the so-called *tangential constraint of motion*:

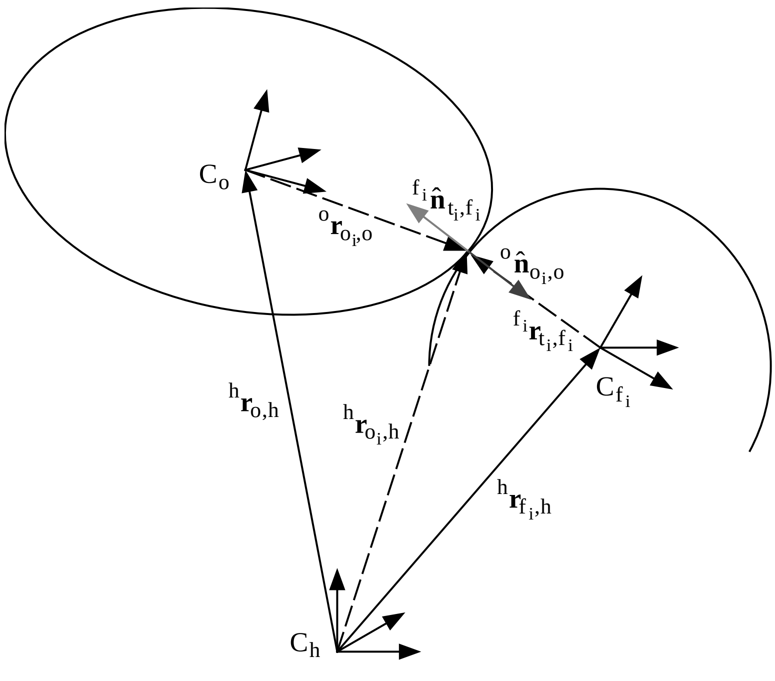

Fig. 4 : Point of contact with respect to object and fingertip

Another constraint for the motion parameters is given by the fact that the tangential planes must coincide, or equivalent, that the normal vectors (${}^h\hat{\mathbf{n}}_{o_i,o} = \mathbf{A}_{o,h}\,{}^{o}\hat{\mathbf{n}}_{o_i,o}$) of the object at the contact points must point diametrically opposite to the corresponding normal vectors (${}^h\hat{\mathbf{n}}_{t_i,f_i} = \mathbf{A}_{f_i,h}\,{}^{f_i}\hat{\mathbf{n}}_{t_i,f_i}$) of the fingertips at the contact points. Differentiating yields then the so-called *normal constraint of motion*:

$$ {}^h\hat{\mathbf{n}}_{t_i,f_i} = -{}^h\hat{\mathbf{n}}_{o_i,o} \;\Rightarrow\; \mathbf{A}_{f_i,h}\,{}^{f_i}\dot{\hat{\mathbf{n}}}_{t_i,f_i} - \left(\mathbf{A}_{f_i,h}\,\mathbf{S}\left({}^{f_i}\hat{\mathbf{n}}_{t_i,f_i}\right)\right){}^h\mathbf{w}_{f_i,h} = -\mathbf{A}_{o,h}\,{}^{o}\dot{\hat{\mathbf{n}}}_{o_i,o} + \left(\mathbf{A}_{o,h}\mathbf{S}\left({}^{o}\hat{\mathbf{n}}_{o_i,o}\right)\right){}^h\mathbf{w}_{o,h}$$

The velocities and all other motion parameters in above equations can be expressed by their corresponding Jacobian matrices that depend on the derivations of the joint angles and contact parameters:

$$\mathbf{v}_i(\eta_i) = \dot{\mathbf{r}}_i(\eta_{i_1}, \eta_{i_2}) = \mathbf{J}_{r_i}\dot{\eta}_i \quad \Rightarrow \quad \begin{matrix} {}^{h}\mathbf{v}_{f_i,h} = {}^{h}\mathbf{J}_{v_i}\dot{\theta}_i & {}^{f_i}\mathbf{v}_{t_i,f_i} = {}^{f_i}\mathbf{J}_{r_i}\dot{\eta}_i & {}^{o}\mathbf{v}_{o_i,o} = {}^{o}\mathbf{J}_{r_i}\dot{\xi}_i \\ {}^{h}\mathbf{w}_{f_i,h} = {}^{h}\mathbf{J}_{w_i}\dot{\theta}_i & {}^{f_i}\dot{\hat{\mathbf{n}}}_{t_i,f_i} = {}^{f_i}\mathbf{J}_{n_i}\dot{\eta}_i & {}^{o}\dot{\hat{\mathbf{n}}}_{o_i,o} = {}^{o}\mathbf{J}_{n_i}\dot{\xi}_i \end{matrix}$$

Substituting these parameters through the Jacobians of $\dot{\theta}_i$, $\dot{\eta}_i$ and $\dot{\xi}_i$ results in the following system:

$$\begin{aligned} \left({}^{h}\mathbf{J}_{v_i} - \left(\mathbf{A}_{f_i,h}\mathbf{S}\left({}^{f_i}\mathbf{r}_{t_i,f_i}\right)\right){}^{h}\mathbf{J}_{w_i}\right)\dot{\theta}_i &= {}^{h}\mathbf{v}_{o,h} - \left(\mathbf{A}_{o,h}\mathbf{S}\left({}^{o}\mathbf{r}_{o_i,o}\right)\right){}^{h}\mathbf{w}_{o,h} \\ \left(\mathbf{A}_{f_i,h}\,{}^{f_i}\mathbf{J}_{r_i}\right)\dot{\eta}_i - \left(\mathbf{A}_{o,h}\,{}^{o}\mathbf{J}_{r_i}\right)\dot{\xi}_i &= (0,\ 0,\ 0)^T \\ -\left(\mathbf{A}_{f_i,h}\mathbf{S}\left({}^{f_i}\hat{\mathbf{n}}_{t_i,f_i}\right)\right){}^{h}\mathbf{J}_{w_i}\dot{\theta}_i + \left(\mathbf{A}_{f_i,h}\,{}^{f_i}\mathbf{J}_{n_i}\right)\dot{\eta}_i + \left(\mathbf{A}_{o,h}\,{}^{o}\mathbf{J}_{n_i}\right)\dot{\xi}_i &= \left(\mathbf{A}_{o,h}\mathbf{S}\left({}^{o}\hat{\mathbf{n}}_{o_i,o}\right)\right){}^{h}\mathbf{w}_{o,h} \end{aligned}$$

Out of this, a matrix equation with the (9×7) matrix $\mathbf{M}_i'$ and the (9×6) matrix $\mathbf{R}_i'$ can be obtained:

$$\mathbf{M}_i' = \begin{pmatrix} {}^{h}\mathbf{J}_{v_i} - \left(\mathbf{A}_{f_i,h}\mathbf{S}\left({}^{f_i}\mathbf{r}_{t_i,f_i}\right)\right){}^{h}\mathbf{J}_{w_i} & 0 & 0 \\ 0 & \mathbf{A}_{f_i,h}\,{}^{f_i}\mathbf{J}_{r_i} & -\mathbf{A}_{o,h}\,{}^{o}\mathbf{J}_{r_i} \\ -\left(\mathbf{A}_{f_i,h}\mathbf{S}\left({}^{f_i}\hat{\mathbf{n}}_{t_i,f_i}\right)\right){}^{h}\mathbf{J}_{w_i} & \mathbf{A}_{f_i,h}\,{}^{f_i}\mathbf{J}_{n_i} & \mathbf{A}_{o,h}\,{}^{o}\mathbf{J}_{n_i} \end{pmatrix} \qquad \mathbf{R}_i' = \begin{pmatrix} \mathbf{I}_3 & -\mathbf{A}_{o,h}\mathbf{S}\left({}^{o}\mathbf{r}_{o_i,o}\right) \\ 0 & 0 \\ 0 & \mathbf{A}_{o,h}\mathbf{S}\left({}^{o}\hat{\mathbf{n}}_{o_i,o}\right) \end{pmatrix}$$

A left hand multiplication of $\mathbf{R}_i'$ with the left generalized inverse ${}^{*}\mathbf{M}_i'$ of $\mathbf{M}_i'$ yields the required velocity vector (three components for the joint velocities, two for each of the contact parameters) as result of the specified object motion. In this case nine equations in seven variables have to be solved:

$$\mathbf{M}_i' \cdot \begin{pmatrix} \dot{\theta}_i \\ \dot{\eta}_i \\ \dot{\xi}_i \end{pmatrix} = \mathbf{R}_i' \cdot \begin{pmatrix} {}^{h}\mathbf{v}_{o,h} \\ {}^{h}\mathbf{w}_{o,h} \end{pmatrix} \quad \Rightarrow \quad \begin{pmatrix} \dot{\theta}_i \\ \dot{\eta}_i \\ \dot{\xi}_i \end{pmatrix} = {}^{*}\mathbf{M}_i'\mathbf{R}_i' \cdot \begin{pmatrix} {}^{h}\mathbf{v}_{o,h} \\ {}^{h}\mathbf{w}_{o,h} \end{pmatrix}$$

The single parameter values for each finger are then calculated by solving the equation system with the Gaußian (elimination) algorithm. To determine the final (integrated) values of θ_i, η_i and ξ_i with the Runge-Kutta-algorithm above calculation has to be performed four times for each motion step.

3.4 The problem of slipping fingertips

Because the relative orientation of the contact frames C_{t_i} and C_{o_i} is changing during object handling due to slipping of the fingertips, the contact angle γ_i between them has to be computed continually. The corresponding frames are fixed at the contact points on the fingertips and the object in such a way, that their x- and y-axes span a tangential plane and their z-axes point into the direction of the surface normal (see Fig. 5). The resulting contact angle γ_i between the x-axes of the two frames C_{t_i} and C_{o_i} is thereafter used to determine the rotation matrix $\mathbf{A}_{o_i,t_i}$ which describes the relative orientation of the two contact coordinate systems and is needed in the next chapter for force computation:

$$\mathbf{A}_{o_i,t_i} = \begin{pmatrix} \cos\gamma_i & \sin\gamma_i & 0 \\ \sin\gamma_i & -\cos\gamma_i & 0 \\ 0 & 0 & -1 \end{pmatrix} \quad \text{with} \quad \gamma_i = \cos^{-1}\left(\frac{{}^h\mathbf{x}_{t_i} \cdot {}^h\mathbf{x}_{o_i}}{\left|{}^h\mathbf{x}_{t_i}\right| \cdot \left|{}^h\mathbf{x}_{o_i}\right|} \right) \quad \begin{matrix} {}^h\mathbf{x}_{t_i} = \mathbf{A}_{f_i,h}\mathbf{A}_{t_i,f_i}(1 \;\; 0 \;\; 0)^T \\ {}^h\mathbf{x}_{o_i} = \mathbf{A}_{o,h}\mathbf{A}_{o_i,o}(1 \;\; 0 \;\; 0)^T \end{matrix}$$

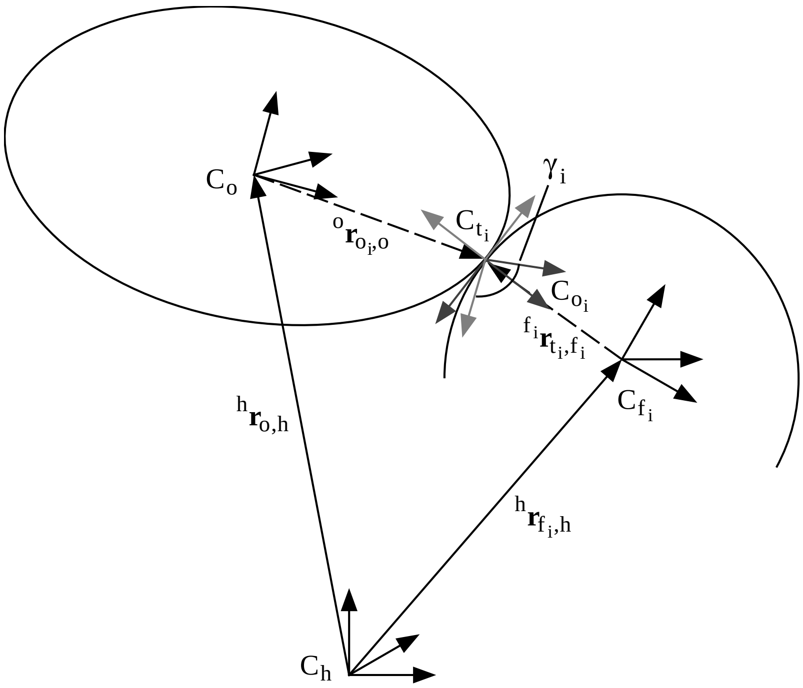

Fig. 5 : Relation between the two contact coordinate systems

4. Computation of forces

Additional to the computation of pure geometric motion parameters (joint angles) needed to manipulate a grasped object with the fingers of a robot hand, appropriate mechanic force parameters (joint torques) to be exerted by the fingertips must be determined to hold the object in a stable grasp configuration. The finger forces that can be applied depend mainly on the contact model which has to be incorporated into the force transformation process from generalized object forces $({}^o\mathbf{f}_o, {}^o\mathbf{m}_o)^T$ over contact specific forces $\boldsymbol{f}_i$ to the required joint torques τ_i. The dimensions of the involved matrices relate to the previously introduced numbers fno, jno_i and the number cmod_i of applicable forces.

4.1 Contact models and contact forces

The components of the generalized forces $({}^{o_i}\mathbf{f}_{o_i}, {}^{o_i}\mathbf{m}_{o_i})^T$ that can be exerted by the fingers as contact forces ${}^{o_i}\boldsymbol{f}_{o_i}$ in the contact points with the object are characterized by the contact matrix $\mathbf{Q}_i$. For this, the matrices and equations that perform the contact specific transformation are described for the case of frictionless point contacts (P), point contacts with friction (PF) and softfinger contacts (SF):

$$\begin{pmatrix} {}^{o_i}\mathbf{f}_{o_i} \\ {}^{o_i}\mathbf{m}_{o_i} \end{pmatrix} = \mathbf{Q}_i \cdot {}^{o_i}\boldsymbol{f}_{o_i} \;,\; \mathbf{Q}_i\!: \; (6 \times \mathrm{cmod}_i) \;,\; {}^{o_i}\boldsymbol{f}_{o_i}\!: \; (\mathrm{cmod}_i \times 1) \;,\; \mathbf{Q}_i \in \{\mathbf{Q}_P, \mathbf{Q}_{PF}, \mathbf{Q}_{SF}\}$$

In the case of a frictionless point contact (P) only single forces can be exerted in the contact points into the normal direction towards the object surface (the force has to be negative) to avoid a loss of contact; also, the force must not exceed a specific maximal value in order to prevent object damage.

$$\mathbf{Q}_{\mathrm{P}} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}^{\mathrm{T}} \text{ , } \mathrm{cmod}_{\mathrm{i}} = 1 \quad \Rightarrow \quad \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right) \le 0 \text{ , } \left| \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right) \right| \le \mathrm{f}_{\mathrm{i_{max}}}$$

In the case of point contact with friction (PF) the fingers can exert any forces pointing into the defined friction cones at the contact points. This means, the tangential forces must stay inside of the friction cones (determined by the tangential coefficients of friction μ_t) to avoid slipping of the fingertips on the tangential plane of the object surface (they are free to rotate about the contact points).

$$\mathbf{Q}_{\mathrm{PF}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \text{ , } \mathrm{cmod}_{\mathrm{i}} = 3 \quad \Rightarrow \quad \begin{array}{c} \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right)_{\mathrm{z}} \le 0 \text{ , } \left| \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right)_{\mathrm{z}} \right| \le \mathrm{f}_{\mathrm{i_{max}}} \\ \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right)_{\mathrm{x}}^2 + \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right)_{\mathrm{y}}^2 \le \mu_{\mathrm{t_i}} \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right)_{\mathrm{z}}^2 \end{array}$$

In the case of softfinger contact (SF) the friction over the area of contact allows the fingers to exert single torques in addition to the pure forces pointing into the defined friction cones. The torques can be exerted in both directions about the normal axis such that the object is constrained with the given contacts (normal coefficients of friction μ_n) and can only break the contacts by sliding downward.

$$\mathbf{Q}_{\mathrm{SF}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \text{ , } \mathrm{cmod}_{\mathrm{i}} = 4 \quad \Rightarrow \quad \begin{array}{c} \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right)_{\mathrm{z}} \le 0 \text{ , } \left| \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right)_{\mathrm{z}} \right| \le \mathrm{f}_{\mathrm{i_{max}}} \\ \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right)_{\mathrm{x}}^2 + \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right)_{\mathrm{y}}^2 \le \mu_{\mathrm{t_i}} \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right)_{\mathrm{z}}^2 \\ \left| \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right)_{\mathrm{r}} \right| \le \mu_{\mathrm{n_i}} \left({}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right)_{\mathrm{z}} \end{array}$$

4.2 Static forces considering rolling and slipping

Starting from generalized forces acting onto the object during manipulation the computation of corresponding finger forces is described in the presence of rolling and slipping. Therefore, the contact forces and moments $({}^{\mathrm{o_i}}\mathbf{f}_{\mathrm{o_i}}, {}^{\mathrm{o_i}}\mathbf{m}_{\mathrm{o_i}})^{\mathrm{T}}$ are converted into $({}^{\mathrm{o}}\mathbf{f}_{\mathrm{o_i}}, {}^{\mathrm{o}}\mathbf{m}_{\mathrm{o_i}})^{\mathrm{T}}$ which is performed by transformation matrices $\mathbf{T}_{\mathrm{o_i,o}}$; the result is then summed up to compensate the object forces and moments $({}^{\mathrm{o}}\mathbf{f}_{\mathrm{o}}, {}^{\mathrm{o}}\mathbf{m}_{\mathrm{o}})^{\mathrm{T}}$. According to the chosen contact models represented by contact matrices $\mathbf{Q}_{\mathrm{i}}$, the respective object contact forces ${}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}}$ are converted into the contact forces and moments $({}^{\mathrm{o}}\mathbf{f}_{\mathrm{o_i}}, {}^{\mathrm{o}}\mathbf{m}_{\mathrm{o_i}})^{\mathrm{T}}$ which is performed by transformations matrices $\tilde{\mathbf{T}}_{\mathrm{o_i,o}}$ and leads to the definition of grip submatrices $\mathbf{G}_{\mathrm{i}}$:

$$\begin{pmatrix} {}^{\mathrm{o}}\mathbf{f}_{\mathrm{o}} \\ {}^{\mathrm{o}}\mathbf{m}_{\mathrm{o}} \end{pmatrix} = \sum_{\mathrm{i=1}}^{\mathrm{fno}} \begin{pmatrix} {}^{\mathrm{o}}\mathbf{f}_{\mathrm{o_i}} \\ {}^{\mathrm{o}}\mathbf{m}_{\mathrm{o_i}} \end{pmatrix} = \sum_{\mathrm{i=1}}^{\mathrm{fno}} \left[\mathbf{T}_{\mathrm{o_i,o}} \begin{pmatrix} {}^{\mathrm{o_i}}\mathbf{f}_{\mathrm{o_i}} \\ {}^{\mathrm{o_i}}\mathbf{m}_{\mathrm{o_i}} \end{pmatrix} \right] = \sum_{\mathrm{i=1}}^{\mathrm{fno}} \left[\mathbf{T}_{\mathrm{o_i,o}} \cdot \mathbf{Q}_{\mathrm{i}} \cdot {}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right] = \sum_{\mathrm{i=1}}^{\mathrm{fno}} \left[\tilde{\mathbf{T}}_{\mathrm{o_i,o}} \cdot {}^{\mathrm{o_i}}\boldsymbol{f}_{\mathrm{o_i}} \right] = \sum_{\mathrm{i=1}}^{\mathrm{fno}} \left[\mathbf{G}_{\mathrm{i}} \cdot \boldsymbol{f}_{\mathrm{i}} \right] = \mathbf{G} \cdot \boldsymbol{f}$$

in general:

$$\mathbf{T}_{a_i,b_i} : (6\times 6) \;\text{ with }\; \begin{pmatrix} {}^{b_i}\mathbf{f}_{c_i} \\ {}^{b_i}\mathbf{m}_{c_i} \end{pmatrix} = \mathbf{T}_{a_i,b_i} \begin{pmatrix} {}^{a_i}\mathbf{f}_{c_i} \\ {}^{a_i}\mathbf{m}_{c_i} \end{pmatrix},\; \mathbf{T}_{a_i,b_i} = \begin{pmatrix} \mathbf{A}_{a_i,b_i} & \mathbf{0} \\ \mathbf{S}\left({}^{b_i}\mathbf{r}_{a_i,b_i}\right)\mathbf{A}_{a_i,b_i} & \mathbf{A}_{a_i,b_i} \end{pmatrix}$$

$$\tilde{\mathbf{T}}_{a_i,b_i} = \mathbf{T}_{a_i,b_i}\cdot \mathbf{Q}_i : (6\times \mathrm{c\,mod}_i) \;\text{ with }\; \begin{pmatrix} {}^{b_i}\mathbf{f}_{c_i} \\ {}^{b_i}\mathbf{m}_{c_i} \end{pmatrix} = \tilde{\mathbf{T}}_{a_i,b_i}\cdot {}^{a_i}\boldsymbol{f}_{c_i} \;,\; {}^{a_i}\boldsymbol{f}_{c_i} : (\mathrm{c\,mod}_i \times 1)$$

For a robot hand with fno (eventually totally different) fingers and chosen contact models $\mathrm{c\,mod}_i$, the grip matrix $\mathbf{G}$ for the transformation of contact forces $\boldsymbol{f}$ that are composed from the force components $\boldsymbol{f}_i$ in each contact point into the object forces and moments $({}^{o}\mathbf{f}_o, {}^{o}\mathbf{m}_o)^T$ is given as follows:

$$\mathbf{G}_i = \tilde{\mathbf{T}}_{o_i,o} = \mathbf{T}_{o_i,o}\cdot \mathbf{Q}_i : (6\times \mathrm{c\,mod}_i) \;\text{ with }\; \begin{pmatrix} {}^{o}\mathbf{f}_{o_i} \\ {}^{o}\mathbf{m}_{o_i} \end{pmatrix} = \mathbf{G}_i \cdot \boldsymbol{f}_i \;\text{ and }\; \boldsymbol{f}_i = {}^{o_i}\boldsymbol{f}_{o_i} : (\mathrm{c\,mod}_i \times 1)$$

$$\mathbf{G} = \left(\mathbf{G}_1 \mid \cdots \mid \mathbf{G}_{fno}\right) : (6\times \sum_{i=1}^{fno} \mathrm{c\,mod}_i) \;\text{ with }\; \begin{pmatrix} {}^{o}\mathbf{f}_{o} \\ {}^{o}\mathbf{m}_{o} \end{pmatrix} = \mathbf{G}\cdot \boldsymbol{f} \,,\; \boldsymbol{f} = \left(\boldsymbol{f}_1,\ldots,\boldsymbol{f}_{fno}\right)^T : (\sum_{i=1}^{fno} \mathrm{c\,mod}_i \times 1)$$

$$\text{and }\; \begin{pmatrix} {}^{o}\mathbf{f}_{o} \\ {}^{o}\mathbf{m}_{o} \end{pmatrix} = \sum_{i=1}^{fno} \begin{pmatrix} {}^{o}\mathbf{f}_{o_i} \\ {}^{o}\mathbf{m}_{o_i} \end{pmatrix} = \sum_{i=1}^{fno} \left[\mathbf{G}_i \cdot \boldsymbol{f}_i\right] = \mathbf{G}_1 \cdot \boldsymbol{f}_1 + \ldots + \mathbf{G}_{fno}\cdot \boldsymbol{f}_{fno}$$

A robot hand with fno = 3 identical fingers and contact models $\mathrm{c\,mod}_i = 3$ (point contact with friction) has following grip matrix $\mathbf{G}$ (only forces and no moments can be exerted at the contact points):

$$\mathbf{G} = \left(\mathbf{G}_1 \mid \mathbf{G}_2 \mid \mathbf{G}_3\right) = \left(\begin{array}{c:c:c} \mathbf{I}_3 & \mathbf{I}_3 & \mathbf{I}_3 \\ \mathbf{S}\left({}^{o}\mathbf{r}_{o_1,o}\right) & \mathbf{S}\left({}^{o}\mathbf{r}_{o_2,o}\right) & \mathbf{S}\left({}^{o}\mathbf{r}_{o_3,o}\right) \end{array}\right) \cdot \begin{pmatrix} \mathbf{A}_{o_1,o} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{o_2,o} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{o_3,o} \end{pmatrix},\; \boldsymbol{f} = \begin{pmatrix} {}^{o_1}\boldsymbol{f}_{o_1} \\ {}^{o_2}\boldsymbol{f}_{o_2} \\ {}^{o_3}\boldsymbol{f}_{o_3} \end{pmatrix}$$

To determine the joint torques τ_i for each finger, that depend on the forces and moments acting at the contact points, the vectors $({}^{f_i}\mathbf{f}_{o_i}, {}^{f_i}\mathbf{m}_{o_i})^T$ are computed out of the contact force components ${}^{o_i}\boldsymbol{f}_{o_i}$ by transformation matrices $\tilde{\mathbf{T}}_{o_i,f_i}$. This is done according to the contact model specific force transformations from the frames C_{o_i} into the frames C_{t_i} (both have the same contact points as origins, so that the position vectors ${}^{t_i}\mathbf{r}_{o_i,t_i}$ are equal to zero) and the force transformations into the frames C_{f_i}:

$$\begin{pmatrix} {}^{f_i}\mathbf{f}_{o_i} \\ {}^{f_i}\mathbf{m}_{o_i} \end{pmatrix} = \mathbf{T}_{t_i,f_i}\cdot \mathbf{T}_{o_i,t_i}\cdot \mathbf{Q}_i \cdot {}^{o_i}\boldsymbol{f}_{o_i} = \begin{cases} \mathbf{T}_{t_i,f_i}\cdot \tilde{\mathbf{T}}_{o_i,t_i}\cdot {}^{o_i}\boldsymbol{f}_{o_i} = \tilde{\mathbf{T}}_{o_i,f_i}\cdot {}^{o_i}\boldsymbol{f}_{o_i} \;,\text{ or} \\ \mathbf{T}_{t_i,f_i}\cdot \mathbf{T}_{o_i,t_i}\cdot \begin{pmatrix} {}^{o_i}\mathbf{f}_{o_i} \\ {}^{o_i}\mathbf{m}_{o_i} \end{pmatrix} = \mathbf{T}_{o_i,f_i}\cdot \begin{pmatrix} {}^{o_i}\mathbf{f}_{o_i} \\ {}^{o_i}\mathbf{m}_{o_i} \end{pmatrix} \end{cases},\; \tilde{\mathbf{T}}_{o_i,f_i} : (6\times \mathrm{c\,mod}_i)$$

Using the principle of virtual work [18], the joint torques τ_i can be computed from the finger force components $\boldsymbol{f}_i$ with the help of the matrices $\mathbf{D}_i$ (see [17]) that are a concatenation of the transformation matrices $\tilde{\mathbf{T}}_{o_i,f_i}$ (depending on $({}^{t_i}\mathbf{r}_{o_i,t_i} = 0, \mathbf{A}_{o_i,t_i})$ and $({}^{t_i}\mathbf{r}_{t_i,f_i}, \mathbf{A}_{t_i,f_i})$) and the finger Jacobians ${}^{f_i}\mathbf{J}^T$. Since the Jacobians ${}^{f_i}\mathbf{J}$ (used in the transposed form) are the derivations of the direct kinematics $({}^{b_i}\mathbf{r}_{f_i,b_i}, \mathbf{A}_{f_i,b_i})$, their dimensions are determined by the jno_i finger parameters (joint angles θ_i):

$$\tau_i = {}^{f_i}\mathbf{J}^T \cdot \tilde{\mathbf{T}}_{o_i,f_i} \cdot {}^{o_i}\boldsymbol{f}_{o_i} = \begin{pmatrix} {}^{f_i}\mathbf{J}_{v_i}^T & {}^{f_i}\mathbf{J}_{w_i}^T \end{pmatrix} \cdot \begin{pmatrix} {}^{f_i}\mathbf{f}_{o_i} \\ {}^{f_i}\mathbf{m}_{o_i} \end{pmatrix} = \mathbf{D}_i \cdot \boldsymbol{f}_i \ , \ {}^{f_i}\mathbf{J}^T = \begin{pmatrix} {}^{f_i}\mathbf{J}_{v_i}^T & {}^{f_i}\mathbf{J}_{w_i}^T \end{pmatrix} , \ \tau_i : (\mathrm{jno}_i \times 1)$$

$$\text{with } \mathbf{D}_i = {}^{f_i}\mathbf{J}^T \cdot \tilde{\mathbf{T}}_{o_i,f_i} : (\mathrm{jno}_i \times 6) * (6 \times \mathrm{c\,mod}_i) = (\mathrm{jno}_i \times \mathrm{c\,mod}_i)$$

Before this relation will be used to deal with the implied force constraints of the chosen contact models $\mathrm{c\,mod}_i$ (see section 6.1), it is presented for the general case of an arbitrary robot hand with fno (not necessary identical) fingers. Thus, the set of joint torques τ can be computed from the total finger forces $\boldsymbol{f}$ with the help of the matrix **D** that describes all required finger force transformations:

$$\tau = {}^{f}\mathbf{J}^T \cdot \tilde{\mathbf{T}}_{o,f} \cdot \boldsymbol{f} = \mathbf{D} \cdot \boldsymbol{f} \text{ and } ({}^{o}\mathbf{f}_o, {}^{o}\mathbf{m}_o)^T = \mathbf{G} \cdot \boldsymbol{f}$$

$${}^{f}\mathbf{J}^T = \mathrm{diag}\left({}^{f_1}\mathbf{J}^T, \ldots, {}^{f_{fno}}\mathbf{J}^T\right) , \ \tilde{\mathbf{T}}_{o,f} = \mathrm{diag}\left(\tilde{\mathbf{T}}_{o_1,f_1}, \ldots, \tilde{\mathbf{T}}_{o_{fno},f_{fno}}\right) , \ \tau = \left(\tau_1, \ldots, \tau_{fno}\right)^T : (\sum_{i=1}^{fno} \mathrm{jno}_i \times 1)$$

$$\text{with } \mathbf{D} = {}^{f}\mathbf{J}^T \cdot \tilde{\mathbf{T}}_{o,f} = \mathrm{diag}\left(\mathbf{D}_1, \ldots, \mathbf{D}_{fno}\right) : (\sum_{i=1}^{fno} \mathrm{jno}_i \times 6) * (6 \times \sum_{i=1}^{fno} \mathrm{c\,mod}_i) = (\sum_{i=1}^{fno} \mathrm{jno}_i \times \sum_{i=1}^{fno} \mathrm{c\,mod}_i)$$

4.3 Decomposition of contact forces

Given the external forces and moments $({}^{o}\mathbf{f}_{ext}, {}^{o}\mathbf{m}_{ext})^T$ acting on a grasped object during manipulation the problem is to compensate the resulting object forces $({}^{o}\mathbf{f}_o, {}^{o}\mathbf{m}_o)^T$ in negative direction with appropriate contact forces $\boldsymbol{f}$; the transformation between them is performed by the grip matrix **G**. Because the space of contact forces (the dimension is $\sum \mathrm{c\,mod}_i$, depending on the contact types) is described by maximal six independent equations, there exists in general (if the dimension is greater than six) more than one solution for the force compensation problem [19]. Thus, all contact forces can be combined of the partial solution $\boldsymbol{f}_p$ and one solution of the homogeneous equation $\mathbf{G} \cdot \boldsymbol{f}_h = \mathbf{0}$:

$$-\begin{pmatrix} {}^{o}\mathbf{f}_{ext} \\ {}^{o}\mathbf{m}_{ext} \end{pmatrix} = \begin{pmatrix} {}^{o}\mathbf{f}_o \\ {}^{o}\mathbf{m}_o \end{pmatrix} = \mathbf{G} \cdot \left(\boldsymbol{f}_p + \boldsymbol{f}_h\right) \text{ with } \boldsymbol{f}_p = \mathbf{G}^* \cdot \begin{pmatrix} {}^{o}\mathbf{f}_o \\ {}^{o}\mathbf{m}_o \end{pmatrix} \text{ and } \mathbf{0} = \mathbf{G} \cdot \boldsymbol{f}_h \ \Rightarrow \ \boldsymbol{f}_h = \mathbf{N} \cdot \lambda$$

The partial solution $\boldsymbol{f}_p$ is computed by the right generalized inverse $\mathbf{G}^*$ of the grip matrix **G**, whereas the homogeneous solution $\boldsymbol{f}_h$ consists of a linear combination of the base vectors of the n-dimensional null space of **G** which is described by the null matrix **N**. Thus, the n-dimensional vector λ represents the internal grip forces (forces acting crosswise between the contact points) that must be exerted in addition to the contact forces $\boldsymbol{f}_p$ (which compensate the external forces and moments):

$$\mathbf{G} \cdot \mathbf{N} \cdot \lambda = \left(\mathbf{G}_1 | \cdots | \mathbf{G}_{fno}\right) \cdot \left(\mathbf{N}_1 | \cdots | \mathbf{N}_{fno}\right)^T \cdot \lambda = \sum_{i=1}^{fno} \left[\mathbf{G}_i \cdot \mathbf{N}_i\right] \cdot \lambda = \left(\mathbf{G}_1 \cdot \mathbf{N}_1 + \ \ldots \ + \mathbf{G}_{fno} \cdot \mathbf{N}_{fno}\right) \cdot \lambda$$

$$\text{with } \mathbf{N} = \left(\mathbf{N}_1 | \cdots | \mathbf{N}_{fno}\right)^T : (\sum_{i=1}^{fno} \mathrm{c\,mod}_i \times \mathrm{ndim}) , \ \lambda = \left(\lambda_1, \ldots, \lambda_{ndim}\right)^T : (\mathrm{ndim} \times 1)$$

Because of the linearity of **G** that defines a dependency between all contact force transformations, the dimension of the null matrix **N** has to be determined with respect to the whole matrix **G** and not independently for the submatrices $\mathbf{G}_i$. Nevertheless, the result of the null space computation can be applied to partition the matrix **N** into fno submatrices $\mathbf{N}_i$ which then depend on the dimension ndim.

5. Grasp force planning

In this chapter, a planning method for the computation of all force parameters required for manipulation of a grasped object with pre-selected contact points is presented [6]. Compared to the general formulation, the special case fno = 3 (jno_i = 3), $cmod_i$ = 3 and ndim = 3 is handled. It is assumed that a range of external forces and moments $({}^{o}\mathbf{f}_{ext}, {}^{o}\mathbf{m}_{ext})^T$ is given or known for a certain type of operation, like a peg-in-hole insertion task, for which appropriate internal forces λ have to be determined. This is realized as knowledge-based approach with rules and criteria (stability and security) that must be satisfied. Here, a major step is the transformation of the problem from the object space with object forces and moments $({}^{o}\mathbf{f}_{o}, {}^{o}\mathbf{m}_{o})^T$ into the contact space with contact forces $\boldsymbol{f}$ (divided into manipulation and grasp forces $\boldsymbol{f}_p$, $\boldsymbol{f}_h$) which is done with the grip matrix **G** and the null matrix **N**.

For the compensation of manipulation forces all applicable grasp forces staying in the friction cones of the ($\sum cmod_i = 9$)-dimensional finger force space must be enumerated. The mapping of the force components onto the contact plane generates a triangle with parameters α, β, γ that are the linear factors of the base vectors of the (n-dim = 3)-dimensional grasp force subspace. The applicable grasp forces are computed from α-parameters determining the β- and γ_1-intervals, β-parameters determining the γ_2-intervals and intersecting γ-parameters that are enumerated from a convex hull algorithm.

Rule 1 : [Applicable grasp forces]
The grasp forces must lie in the intersection of the friction cones with the contact point plane.

Algorithm: [Applicable grasp forces]

Determine $[\alpha_L, \alpha_U]$
FOR $\alpha_0 \in [\alpha_L, \alpha_U]$ DO (* enumerate α - interval *)
    Determine $[\beta_L, \beta_U](\alpha_0)$, $[\gamma_{1_L}, \gamma_{1_U}](\alpha_0)$
    FOR $\beta_0 \in [\beta_L, \beta_U](\alpha_0)$ DO (* enumerate β - interval *)
        Determine $[\gamma_{2_L}, \gamma_{2_U}](\alpha_0, \beta_0)$, $[\gamma_S, \gamma_E] \leftarrow [\gamma_{1_L}, \gamma_{1_U}](\alpha_0) \cap [\gamma_{2_L}, \gamma_{2_U}](\alpha_0, \beta_0)$
        FOR $\gamma_0 \in [\gamma_L, \gamma_U](\alpha_0, \beta_0)$ DO (* enumerate γ - interval *)
            $\boldsymbol{f}_h \leftarrow \alpha_0 \boldsymbol{h}^1 + \beta_0 \boldsymbol{h}^2 + \gamma_0 \boldsymbol{h}^3$
RETURN $\boldsymbol{f}_h$

The determination of suitable grasp forces $\boldsymbol{f}_h$ with respect to the assembly task can be divided into successive steps for the compensation of a single manipulation force through a stability search with minimum security or a security search with minimum stability. Starting from the task area **EA** that describes s range of external forces and moments, the most stable and most secure grasp forces for the corresponding range **MA** of manipulation forces are computed from search procedures, such that the mean value σ of the resulting range **HA** of grasp forces is the desired optimal solution $\boldsymbol{f}_{h_{opt}}$.

To ensure this, two quality measures for the optimization of grasp forces in form of criteria for the object stability Ψ and the grip security Ω (robustness) [22] are introduced; they are based on side and bottom face projections Π of the object contact forces $\boldsymbol{f}$ to the surface of the friction cones FC:

$$\Psi(\boldsymbol{f}) := \underset{i=1}{\overset{3}{\mathrm{Min}}}\ \Psi_i(\boldsymbol{f}_i) \ \text{ with } \ \Psi_i(\boldsymbol{f}_i) = \begin{cases} \mathrm{Min}\left(\left|\boldsymbol{f}_i - \Pi_s(\boldsymbol{f}_i)\right|, \left|\boldsymbol{f}_i - \Pi_b(\boldsymbol{f}_i)\right|\right) & \boldsymbol{f}_i \in \mathrm{FC} \\ 0 & \boldsymbol{f}_i \notin \mathrm{FC} \end{cases}$$

$$\Omega(\boldsymbol{f}_h) := \Omega_{max} - \sum_{i=1}^{3}\left|\boldsymbol{f}_{h_i}\right| \ \text{ and } \ \Omega_{max} := \sum_{i=1}^{3}\sqrt{\left(\mu_{t_i} f_{i_{max}}\right)^2 + \left(f_{i_{max}}\right)^2} \ \text{ maximal length}$$

Rule 2 : [Single compensating grasp force]
Each grasp force must compensate a corresponding manipulation force of a given force area.
Rule 3 : [Range compensating grasp force]
A particular grasp force value must compensate all manipulation forces of a given force area.

Algorithm: [Stability search with minimum security]
Input: grip matrix **G**, null matrix **N**, manipulation force $\boldsymbol{f}_p$, minimum security ω_0
Output: $\boldsymbol{f}_p$ for minimum security ω_0 most stable compensating grasp force $\boldsymbol{f}_{h_{res}}$

$\psi_{max} \leftarrow 0$
$\boldsymbol{f}_h \leftarrow$ **Applicable grasp force**$(\alpha_0, \beta_0, \gamma_0)$
IF $\left[\Psi\left(\boldsymbol{f}_p + \boldsymbol{f}_h\right) > \psi_{max}\right] \wedge \left[\Omega(\boldsymbol{f}_h) > \omega_0\right]$ THEN $\boldsymbol{f}_{h_{res}} \leftarrow \boldsymbol{f}_h$; $\psi_{max} \leftarrow \Psi\left(\boldsymbol{f}_p + \boldsymbol{f}_h\right)$
RETURN $\boldsymbol{f}_{h_{res}}$

Algorithm: [Security search with minimum stability]
Input: grip matrix **G**, null matrix **N**, manipulation force $\boldsymbol{f}_p$, minimum stability ψ_0
Output: $\boldsymbol{f}_p$ for minimum stability ψ_0 most secure compensating grasp force $\boldsymbol{f}_{h_{res}}$

$\omega_{max} \leftarrow 0$
$\boldsymbol{f}_h \leftarrow$ **Applicable grasp force**$(\alpha_0, \beta_0, \gamma_0)$
IF $\left[\Omega(\boldsymbol{f}_h) > \omega_{max}\right] \wedge \left[\Psi\left(\boldsymbol{f}_p + \boldsymbol{f}_h\right) > \psi_0\right]$ THEN $\boldsymbol{f}_{h_{res}} \leftarrow \boldsymbol{f}_h$; $\omega_{max} \leftarrow \Omega(\boldsymbol{f}_h)$
RETURN $\boldsymbol{f}_{h_{res}}$

Definition:
The six-dimensional task force area **EA** modeled as task-polyeder is the convex combination of force vectors which are computed from the range set **ES** of external object forces and moments (describes the specific force spectrum as set of vectors) by the six-dimensional convex hull operator CH:

$$\mathbf{ES} \subset \mathbf{EA} \ \text{ with } \ \mathbf{ES}, \mathbf{EA} \subset \Re^6 \text{: } \mathbf{EA}_{Poly}(\mathbf{ES}) := \mathrm{CH}(\mathbf{ES}) \ \text{ and } \ \mathbf{ES} := \left\{ -\begin{pmatrix} {}^{o}\mathbf{f}_{ext} \\ {}^{o}\mathbf{m}_{ext} \end{pmatrix}^{k} \right\}_{k=1}^{eset}$$

The mean value σ of the range set **HS** of grasp forces compensating the range set **MS** of manipulation forces derived from the force spectrum **ES**, or fully equivalent, the center of gravity of the corresponding grasp force area **HA** computed by the three-dimensional convex hull operator CH, is a best compensating grasp force $\boldsymbol{f}_{\mathrm{h}_{\mathrm{best}}}$ that can be transformed into the range set **IS** of internal forces:

$$\exists \boldsymbol{f}_{\mathrm{h}} \in \mathbf{HS} : \Psi\left(\boldsymbol{f}_{\mathrm{p}}{}^{\mathrm{k}} + \boldsymbol{f}_{\mathrm{h}}\right) > 0 \quad \forall \mathrm{k} \in \{1,\ldots,\mathrm{eset}\} \;\text{ with }\; \mathbf{MA} := \mathrm{CH}(\mathbf{MS}) \;\text{ and }\; \mathbf{MS} = \mathbf{G}^{*}(\mathbf{ES})$$

$$\boldsymbol{f}_{\mathrm{h}_{\mathrm{best}}} = \sigma(\mathbf{HS}) = \sigma\left(\left\{\boldsymbol{f}_{\mathrm{h}}{}^{\mathrm{k}}\right\}_{\mathrm{k}=1}^{\mathrm{eset}}\right) := \frac{1}{\mathrm{eset}} \sum_{\mathrm{k}=1}^{\mathrm{eset}} \boldsymbol{f}_{\mathrm{h}}{}^{\mathrm{k}} \;\text{ with }\; \mathbf{HA} := \mathrm{CH}(\mathbf{HS}) \;\text{ and }\; \mathbf{N}^{*}\big(\sigma(\mathbf{HS})\big) = \sigma(\mathbf{IS})$$

Rule 4 : [Grasp forces for stable grips]
The grasp forces must have a certain stability factor relating to rolling and slipping motions.
Rule 5 : [Grasp forces for secure grips]
The grasp forces must have a certain security factor relating to tolerances of the force areas.

Algorithm: [Most stable grasp force with respect to MA]

Input: grip matrix $\mathbf{G}$, null matrix $\mathbf{N}$, manipulation force area $\mathbf{MA}$, minimum security ω_0
Output: $\mathbf{MA}$ for minimum security ω_0 most stable compensating grasp force $\boldsymbol{f}_{\mathrm{h}_{\mathrm{best}}}$

FOR $\mathrm{k} \in \{1,\ldots,\mathrm{eset}\}$ DO

$\quad\boldsymbol{f}_{\mathrm{h}}{}^{\mathrm{k}} \leftarrow [\textbf{Stability search with minimum security}]\left(\mathbf{G},\mathbf{N},\boldsymbol{f}_{\mathrm{p}}{}^{\mathrm{k}},\ \omega_0\right)$

$\boldsymbol{f}_{\mathrm{h}_{\mathrm{best}}} \leftarrow \sigma\left(\left\{\boldsymbol{f}_{\mathrm{h}}{}^{\mathrm{k}}\right\}_{\mathrm{k}=1}^{\mathrm{eset}}\right)$

IF $\left(\forall \mathrm{k} \in \{1,\ldots,\mathrm{eset}\}: \Psi\left(\boldsymbol{f}_{\mathrm{p}}{}^{\mathrm{k}} + \boldsymbol{f}_{\mathrm{h}_{\mathrm{best}}}\right) > 0\right)$ THEN **Success** $\leftarrow$ TRUE
ELSE **Success** $\leftarrow$ FALSE

RETURN $\left(\boldsymbol{f}_{\mathrm{h}_{\mathrm{best}}}, \textbf{Success}\right)$

Algorithm: [Most secure grasp force with respect to MA]

Input: grip matrix $\mathbf{G}$, null matrix $\mathbf{N}$, manipulation force area $\mathbf{MA}$, minimum stability ψ_0
Output: $\mathbf{MA}$ for minimum stability ψ_0 most secure compensating grasp force $\boldsymbol{f}_{\mathrm{h}_{\mathrm{best}}}$

FOR $\mathrm{k} \in \{1,\ldots,\mathrm{eset}\}$ DO

$\quad\boldsymbol{f}_{\mathrm{h}}{}^{\mathrm{k}} \leftarrow [\textbf{Security search with minimum stability}]\left(\mathbf{G},\mathbf{N},\boldsymbol{f}_{\mathrm{p}}{}^{\mathrm{k}},\psi_0\right)$

$\boldsymbol{f}_{\mathrm{h}_{\mathrm{best}}} \leftarrow \sigma\left(\left\{\boldsymbol{f}_{\mathrm{h}}{}^{\mathrm{k}}\right\}_{\mathrm{k}=1}^{\mathrm{eset}}\right)$

IF $\left(\forall \mathrm{k} \in \{1,\ldots,\mathrm{eset}\}: \Psi\left(\boldsymbol{f}_{\mathrm{p}}{}^{\mathrm{k}} + \boldsymbol{f}_{\mathrm{h}_{\mathrm{best}}}\right) > \psi_0\right)$ THEN **Success** $\leftarrow$ TRUE
ELSE **Success** $\leftarrow$ FALSE

RETURN $\left(\boldsymbol{f}_{\mathrm{h}_{\mathrm{best}}}, \textbf{Success}\right)$

The main difficulty is the determination of the smallest minimum stability parameter ψ_0 for the algorithm searching the most secure grasp force in contact space. The lower bound is normally zero whereas the upper bound can be computed from the algorithm searching the most stable grasp force with minimum security zero (pure stability search). Building a stability interval of them, an interval search procedure based on the security search with minimum stability and termination condition (desired accuracy) computes the optimal grasp force $\boldsymbol{f}_{\mathrm{h}_{\mathrm{opt}}}$ compensating all given manipulation forces.

lower bound: $\psi_{\min} = 0$, upper bound: $\psi_{\max} < \mu_t f_{\max}$ for stability interval $\mathbf{I}_\psi := [\psi_{\min}, \psi_{\max}]$
with $\psi_{\max} := \psi_{\boldsymbol{f}_{h_{best}}}$ and $\boldsymbol{f}_{h_{best}} = [\textbf{Most stable grasp force wrt. MA}](\mathbf{G}, \mathbf{N}, \mathbf{MA}, \omega_0 = 0)$

Algorithm (interval search): [Optimal grasp force with respect to MA]

Input: grip matrix $\mathbf{G}$, null matrix $\mathbf{N}$, manip. force area $\mathbf{MA}$, desired search accuracy ε_0
Output: **MA** for stability interval most stable and secure compensating grasp force $\boldsymbol{f}_{h_{opt}}$

$(\boldsymbol{f}_{h_{opt}}, \textbf{Success}) \leftarrow [\textbf{Most stable grasp force wrt. MA}](\mathbf{G}, \mathbf{N}, \mathbf{MA}, \omega_0 = 0)$
IF **Success** THEN
  $\psi_{\max} \leftarrow \psi_{\boldsymbol{f}_{h_{opt}}}$; $\psi_{\min} \leftarrow 0$; $\mathbf{I}_\psi \leftarrow [\psi_{\min}, \psi_{\max}]$
  REPEAT
    $\psi_0 \leftarrow (\psi_{\max} + \psi_{\min}) / 2$;
    $(\boldsymbol{f}_{h_{best}}, \textbf{Success0}) \leftarrow [\textbf{Most secure grasp force wrt. MA}](\mathbf{G}, \mathbf{N}, \mathbf{MA}, \psi_0)$
    IF **Success0** THEN $\psi_{\max} \leftarrow \psi_0$; $\boldsymbol{f}_{h_{opt}} \leftarrow \boldsymbol{f}_{h_{best}}$
    ELSE $\psi_{\min} \leftarrow \psi_0$
    $\mathbf{I}_\psi \leftarrow [\psi_{\min}, \psi_{\max}]$
  UNTIL $|\psi_{\max} - \psi_{\min}| < \varepsilon_0$
RETURN $(\boldsymbol{f}_{h_{opt}}, \textbf{Success})$

General rule : [Grasp parameter optimization]

The parameters must be chosen, such that the contact point dependent grasp forces $\boldsymbol{f}_h$ in the contact space are optimal and the resulting grip of is both stable and secure with respect to the specific task.

Algorithm: [Optimal internal force with respect to EA]

Input: grip matrix $\mathbf{G}$, null matrix $\mathbf{N}$, task force area $\mathbf{EA}$, desired search accuracy ε_0
Output: **EA** optimal (regarding stability and security) compensating internal force λ_{opt}

FOR $k \in \{1, \ldots, eset\}$ DO
  $\boldsymbol{f}_p{}^k \leftarrow \mathbf{G}^* \left(- \begin{pmatrix} {}^o\mathbf{f}_{ext} \\ {}^o\mathbf{m}_{ext} \end{pmatrix}^k \right)$ (* mapping forward into contact force space *)
$\boldsymbol{f}_{h_{opt}} \leftarrow [\textbf{Optimal grasp force wrt. MA}](\mathbf{G}, \mathbf{N}, \mathbf{MA}, \varepsilon_0)$
$\lambda_{opt} \leftarrow \mathbf{N}^*(\boldsymbol{f}_{h_{opt}})$ (* mapping backward back into object force space *)
RETURN λ_{opt}

The determined optimal grasp force $\boldsymbol{f}_{h_{opt}}$ in the contact space is transformed into the optimal internal force λ_{opt} in the object space which is performed by the right generalized inverse $\mathbf{N}^*$ of the null matrix **N**. Summarizing, Fig. 6 shows the search process for optimal contact forces that starts from external forces and moments mapped to the contact space where all manipulation forces are compensated by one suitable grasp force which is mapped to the object space as the resulting internal force.

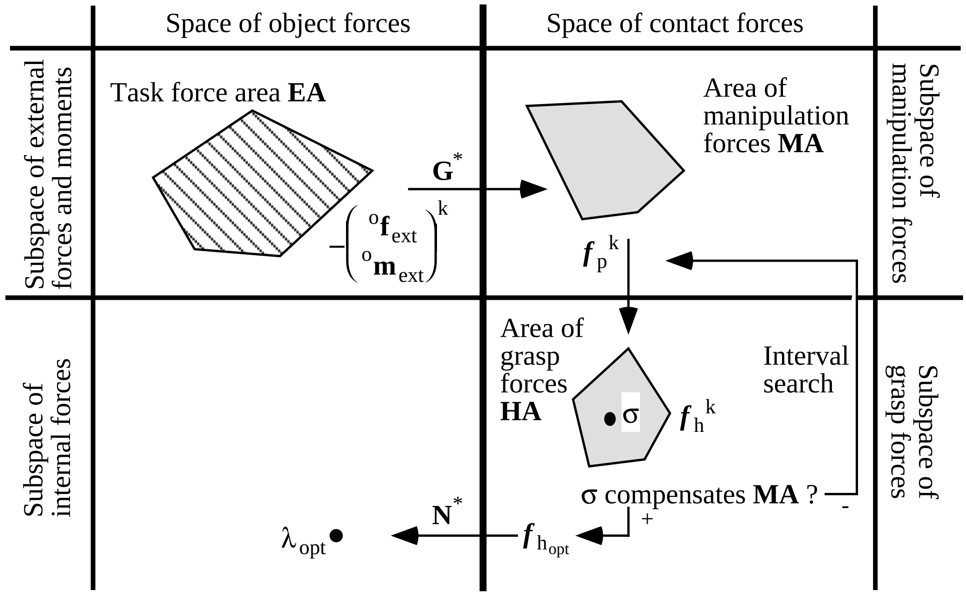

Fig. 6 : Schematic process of contact force searching

6. Grasp force optimization

In this chapter, another planning method for the determination of the required grasp force parameters is presented which is formulated as an optimization problem over given or imposed constraints. Therefore, it incorporates explicitly the contact force constraints (indirectly defined in the previous described method by object stability and grip security criteria) into the process of force computation. The approach chosen here is based on work for the normal case fno = 3 ($\mathrm{jno}_i = 3$), $\mathrm{cmod}_i = 3$ and ndim = 3 (see [17]) which is generalized for the case of an arbitrary robot hand and all kinds of possible contact models. For the final solution of the optimization problem several algorithms can be applied, but preference is given to an algorithm that searches for maxima in n-dimensional spaces.

6.1 Constraints on contact forces

In order to keep the force computation simple and fast, the non-linear equations describing the contact forces inside of the contact model depending friction cones are approximated linearly: the generalized friction cones are replaced by generalized friction pyramids whose sides touch the borders of the cones. By this procedure, additional (not applicable) contact force regions are defined which can be avoided by reducing the friction coefficients proportionally (for example, as shown in Fig. 7 from μ_{t_i} to $\mu'_{t_i} = \mu_{t_i} / \sqrt{2}$). Rewriting of the force constraints mentioned in section 4.1 leads to an inequality system-like formulation of the possible contact forces ${}^{o_i}\boldsymbol{f}_{o_i}$ which are described through friction matrices $\mathbf{F}_i$ and constraint vectors $\mathbf{c}_i$. Again, the respective equations are considered for the case of frictionless point contacts (P), point contacts with friction (PF) and softfinger contacts (SF):

$$\mathbf{F}_i \cdot {}^{o_i}\boldsymbol{f}_{o_i} \geq \mathbf{c}_i \quad \text{with} \quad \mathbf{F}_i : (2 * \mathrm{cmod}_i \times \mathrm{cmod}_i) \ , \ \mathbf{c}_i : (2 * \mathrm{cmod}_i \times 1) \ , \ \mathrm{cmod}_i = \{1, 3, 4\}$$

In the case of frictionless point contact (P) the constraint vector considers only the direction of the contact forces and the maximal possible contact forces that can be applied by the finger of the hand:

$$\left({}^{o_i}\boldsymbol{f}_{o_i}\right) \le 0 \;\Rightarrow\; -\left({}^{o_i}\boldsymbol{f}_{o_i}\right) \ge 0 \;,\; \left|\left({}^{o_i}\boldsymbol{f}_{o_i}\right)\right| \le \mathrm{f}_{i_{max}} \;\Rightarrow\; \left({}^{o_i}\boldsymbol{f}_{o_i}\right) \ge -\mathrm{f}_{i_{max}}$$

$$\Rightarrow\; \mathbf{F}_P \cdot {}^{o_i}\boldsymbol{f}_{o_i} \ge \mathbf{c}_P \;\text{ with }\; \mathbf{F}_P = \begin{pmatrix} -1 \\ 1 \end{pmatrix},\; \mathbf{c}_P = \begin{pmatrix} 0 \\ -\mathrm{f}_{i_{max}} \end{pmatrix},\; \mathrm{cmod}_i = 1$$

In the case of point contact with friction (PF), additionally to the consideration of the maximal contact forces, the contact force components within the tangential plane of the corresponding friction cones are linearized and adapted to the resulting friction pyramid by a reduced friction coefficient μ'_{t_i}:

$$\left|\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_x + \left({}^{o_i}\boldsymbol{f}_{o_i}\right)_y\right| \le \mu_{t_i}\left|\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_z\right| \;\Rightarrow\; \begin{matrix} \left|\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_x\right| \le \mu'_{t_i}\left|\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_z\right| \\ \left|\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_y\right| \le \mu'_{t_i}\left|\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_z\right| \end{matrix} \;\Rightarrow\; \begin{matrix} \pm\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_x - \mu'_{t_i}\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_z \ge \pm 0 \\ \pm\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_y - \mu'_{t_i}\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_z \ge \pm 0 \end{matrix}$$

$$\Rightarrow\; \mathbf{F}_{PF} \cdot {}^{o_i}\boldsymbol{f}_{o_i} \ge \mathbf{c}_{PF} \;\text{ with }\; \mathbf{F}_{PF} = \begin{pmatrix} 0 & 0 & -1 \\ \pm 1 & 0 & -\mu'_{t_i} \\ 0 & \pm 1 & -\mu'_{t_i} \\ 0 & 0 & 1 \end{pmatrix},\; \mathbf{c}_{PF} = \begin{pmatrix} 0 \\ \pm 0 \\ \pm 0 \\ -\mathrm{f}_{i_{max}} \end{pmatrix},\; \mathrm{cmod}_i = 3$$

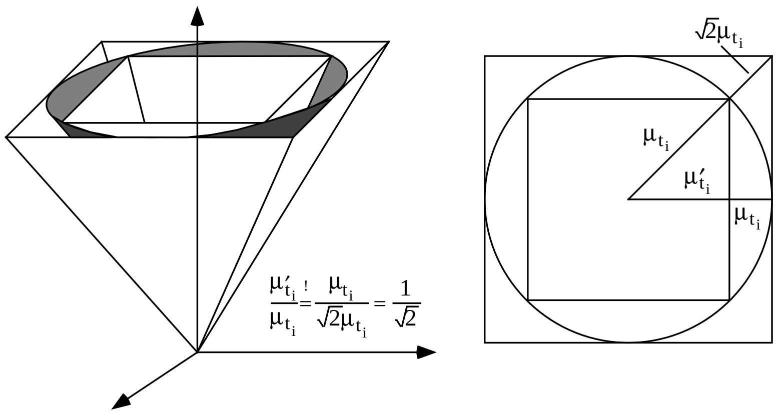

Fig. 7 : Linearization of a friction cone (friction pyramid)

In the case of softfinger contact (SF), additionally to above described restrictions on the contact forces, the contact torque in normal direction is constrained by the friction conditions along the z-axis:

$$\left|\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_r\right| \le \mu_{n_i}\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_z \;\Rightarrow\; \mu_{n_i}\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_z + \left({}^{o_i}\boldsymbol{f}_{o_i}\right)_r \ge 0 \;\text{ and }\; -\mu_{n_i}\left({}^{o_i}\boldsymbol{f}_{o_i}\right)_z + \left({}^{o_i}\boldsymbol{f}_{o_i}\right)_r \ge 0$$

$$\Rightarrow\; \mathbf{F}_{SF} \cdot {}^{o_i}\boldsymbol{f}_{o_i} \ge \mathbf{c}_{SF} \;\text{ with }\; \mathbf{F}_{SF} = \begin{pmatrix} 0 & 0 & -1 & 0 \\ \pm 1 & 0 & -\mu'_{t_i} & 0 \\ 0 & \pm 1 & -\mu'_{t_i} & 0 \\ 0 & 0 & \pm\mu_{n_i} & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix},\; \mathbf{c}_{SF} = \begin{pmatrix} 0 \\ \pm 0 \\ \pm 0 \\ \pm 0 \\ -\mathrm{f}_{i_{max}} \end{pmatrix},\; \mathrm{cmod}_i = 4$$

Another set of constraints is introduced by the fact that the joint torque values τ_i of each finger of a robot hand must not exceed the constructional limits to avoid damage of the motors and/or the gears:

$$\tau_{i_{min}} \le \tau_i \le \tau_{i_{max}} \;\text{ with }\; \tau_i = \mathbf{D}_i \cdot {}^{o_i}\boldsymbol{f}_{o_i} \;\text{ and }\; \tau_{i_{min}}, \tau_{i_{max}} : (\text{jno}_i \times 1)$$
$$\Rightarrow \; \mathbf{D}_i \cdot {}^{o_i}\boldsymbol{f}_{o_i} \ge \tau_{i_{min}} \;\text{ and }\; -\mathbf{D}_i \cdot {}^{o_i}\boldsymbol{f}_{o_i} \ge -\tau_{i_{max}}$$

6.2 Force computation: solving of inequality systems

The restrictions imposed onto the contact forces and joint torques that are described by friction matrices $\mathbf{F}_i$, constraint vectors $\mathbf{c}_i$ and motion dependent matrices $\mathbf{D}_i$ (rolling and slipping of the fingertips at the object surface) can be expressed in matrix form as systems of linear inequalities. These inequality systems are based on the matrices $\mathbf{H}_i$ and vectors $\mathbf{g}_i$ that are composed from above described components and lead to optimization problems for the optimal choice of the contact forces $\boldsymbol{f}_i$.

$$\mathbf{H}_i = \begin{pmatrix} \mathbf{F}_i \\ \mathbf{D}_i \\ -\mathbf{D}_i \end{pmatrix} : ([2 * \text{cmod}_i + 2 * \text{jno}_i] \times \text{cmod}_i) \;,\; \mathbf{g}_i = \begin{pmatrix} \mathbf{c}_i \\ \tau_{i_{min}} \\ -\tau_{i_{max}} \end{pmatrix} : ([2 * \text{cmod}_i + 2 * \text{jno}_i] \times 1)$$
$$\Rightarrow \; \text{inequality system in } \boldsymbol{f}_i \;:\; \mathbf{H}_i \cdot \boldsymbol{f}_i \ge \mathbf{g}_i \;\text{ with }\; \boldsymbol{f}_i = {}^{o_i}\boldsymbol{f}_{o_i} : (\text{cmod}_i \times 1)$$

To compute appropriate solutions, the submatrices $\mathbf{D}_i$ of the matrices $\mathbf{H}_i$ have to be actualized continually in each motion step during object manipulation, because they contain all information about the changing parameter values of the contact points and the joint angles. Generalized for the case of a robot hand with fno (not necessary identical) fingers, the resulting inequality system is described by the matrix $\mathbf{H}$ and the vector $\mathbf{g}$ that are composed from the finger specific matrices $\mathbf{H}_i$ and $\mathbf{g}_i$. The optimization problem is then defined as search for the optimal choice of the total contact force $\boldsymbol{f}$.

$$\mathbf{H} = \text{diag}\left(\mathbf{H}_1, \ldots, \mathbf{H}_{\text{fno}}\right) : \left(\sum_{i=1}^{\text{fno}} [2 * \text{cmod}_i + 2 * \text{jno}_i] \times \sum_{i=1}^{\text{fno}} \text{cmod}_i\right) \;,\; \text{formula for general case}$$
$$\mathbf{g} = \left(\mathbf{g}_1, \ldots, \mathbf{g}_{\text{fno}}\right)^T : \left(\sum_{i=1}^{\text{fno}} [2 * \text{cmod}_i + 2 * \text{jno}_i] \times 1\right) \;,\; \boldsymbol{f} = \left(\boldsymbol{f}_1, \ldots, \boldsymbol{f}_{\text{fno}}\right)^T : \left(\sum_{i=1}^{\text{fno}} \text{cmod}_i \times 1\right)$$
$$\Rightarrow \; \text{inequality system in } \boldsymbol{f} \;:\; \mathbf{H} \cdot \boldsymbol{f} \ge \mathbf{g} \;\Leftrightarrow\; \mathbf{H}_i \cdot \boldsymbol{f}_i \ge \mathbf{g}_i \;\; \forall i \in \{1, \ldots, \text{fno}\}$$

Rearranging above inequality system and applying the subdivision of the forces $\boldsymbol{f}$ in the contact space into manipulation forces $\boldsymbol{f}_p$ and grasp forces $\boldsymbol{f}_h$, the optimization problem can be rewritten as problem of finding the optimal internal forces λ in the object space using the null matrix $\mathbf{N}$. If there is a way to split this matrix into submatrices $\mathbf{N}_i$, the force search process could be done in parallel:

$$\mathbf{H} \cdot \boldsymbol{f} \ge \mathbf{g} \;\Leftrightarrow\; \mathbf{H} \cdot \left(\boldsymbol{f}_p + \boldsymbol{f}_h\right) \ge \mathbf{g} \;\Leftrightarrow\; \mathbf{H} \cdot \boldsymbol{f}_h \ge \mathbf{g} - \mathbf{H} \cdot \boldsymbol{f}_p \;\Leftrightarrow\; \mathbf{H} \cdot \mathbf{N} \cdot \lambda \ge \mathbf{g} - \mathbf{H} \cdot \mathbf{G}^* \cdot ({}^{o}\mathbf{f}_o, {}^{o}\mathbf{m}_o)^T$$
$$\text{with } \; \mathbf{H} \cdot \mathbf{N} = \left(\mathbf{H}_1 \cdot \mathbf{N}_1 \,|\, \ldots \,|\, \mathbf{H}_{\text{fno}} \cdot \mathbf{N}_{\text{fno}}\right) : \left(\sum_{i=1}^{\text{fno}} [2 * \text{cmod}_i + 2 * \text{jno}_i] \times \text{ndim}\right)$$
$$\Rightarrow \; \mathbf{H}_i \cdot \mathbf{N}_i \cdot \lambda \ge \mathbf{g}_i - \mathbf{H}_i \cdot \mathbf{G}_i^* \cdot ({}^{o}\mathbf{f}_{o_i}, {}^{o}\mathbf{m}_{o_i})^T \quad \forall i \in \{1, \ldots, \text{fno}\}$$

6.3 Optimization of internal (grip) forces

To compute the best internal force λ, a distance measure d_k for evaluating the distances between all defined force constraints (rows of the matrices and vectors) and the resulting hyper-planes (generalized volumes) that are splitted into "allowed" and "forbidden" half-spaces is introduced (see [17]). The optimal selection of λ is the one which has the greatest value of the minimal distance to all given constraints (hyper-planes). In the three-dimensional case this is the center of the biggest hemisphere fitting into the "allowed" polyhedra. Therefore, the following optimization problem has to be solved:

$$(\mathbf{H}\cdot\mathbf{N})\cdot\lambda \ge \mathbf{g}+\mathbf{H}\cdot\mathbf{G}^{*}\cdot\begin{pmatrix}{}^{o}\mathbf{f}_{ext}\\ {}^{o}\mathbf{m}_{ext}\end{pmatrix} \;\Rightarrow\; d_k = (\mathbf{H}\cdot\mathbf{N})\big|_{row=k}\cdot\lambda-\left(\mathbf{g}+\mathbf{H}\cdot\mathbf{G}^{*}\cdot\begin{pmatrix}{}^{o}\mathbf{f}_{ext}\\ {}^{o}\mathbf{m}_{ext}\end{pmatrix}\right)\Bigg|_{row=k} \ge 0$$

$$\text{maximize the function}: \; d(\lambda) = \min\,(d_k) \text{ with}: \; d_k \ge 0 \quad \forall k = 1,\ldots,\sum_{i=1}^{fno}[2 * cmod_i + 2 * jno_i]$$

The goal function $d(\lambda)$ which has to be maximized, is a continual, non-linear and non-differentiable function in $\Re^{ndim}$ that is composed of linear functions d_k. Hence, for finding the optimal value of λ both the simplex-method (goal function not linear) and the gradient-descend method (goal function not differentiable) will fail. This means, that in order to compute the optimal force parameters, an algorithm considering the non-linearity and non-differentiability of the goal function is required.

To solve the optimization problem, the Hooke-Jeeves algorithm [17], [23], [24] for finding maxima in $\Re^{n}$ can be used which steps from an arbitrary point (start point must not lay within the allowed space) to the point with the greatest distance to each constraint. After an optimal solution is found, the internal forces λ together with the external object forces and moments $({}^{o}\mathbf{f}_{ext}, {}^{o}\mathbf{m}_{ext})^{T}$ are used to compute the total contact force $\boldsymbol{f}$ and finally the corresponding joint torques τ for the motor drives:

$$\tau = \mathbf{D}\cdot\boldsymbol{f} = \mathbf{D}\cdot\left(\boldsymbol{f}_p+\boldsymbol{f}_h\right) = \mathbf{D}\cdot\left(-\mathbf{G}^{*}\cdot\begin{pmatrix}{}^{o}\mathbf{f}_{ext}\\ {}^{o}\mathbf{m}_{ext}\end{pmatrix}+\mathbf{N}\cdot\lambda\right)$$

Thus, the optimization problem for internal forces could be generalized for the case of a robot hand with fno (arbitrary) fingers with jno_i joints per finger, applying forces according to the contact models $cmod_i$ and the n-dimensional null space of the grip matrix. Summarizing, the force computation process can be fully characterized by the parameters (dimensions) fno, jno_i, $cmod_i$ and ndim.

7. Example manipulation

Here, the result of a concrete object manipulation (a peg-in-hole insertion task) with the Karlsruhe Dextrous Hand [2], [3], [4] is presented in graphical visualized form. This robot hand consists of three identical fingers whose bases are fixed at the edges of an equal-sided triangle. The hand frame C_h is fixed at the center of this triangle, so that the transformations between the finger base frames C_{b_i} and this coordinate frame are given by following translation vectors $\mathbf{r}$ and rotation matrices $\mathbf{A}$:

$${}^{h}\mathbf{r}_{b_1,h} = \begin{pmatrix}0\\ 50\\ -80\end{pmatrix},\; {}^{h}\mathbf{r}_{b_2,h} = \begin{pmatrix}-50\cos 30^\circ\\ -50\sin 30^\circ\\ -80\end{pmatrix},\; {}^{h}\mathbf{r}_{b_3,h} = \begin{pmatrix}50\cos 30^\circ\\ -50\sin 30^\circ\\ -80\end{pmatrix}$$

$$ {}_{b_1,h} = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \end{pmatrix}, \ \mathbf{A}_{b_2,h} = \begin{pmatrix} 0 & \sin 30° & -\cos 30° \\ 0 & -\cos 30° & -\sin 30° \\ -1 & 0 & 0 \end{pmatrix}, \ \mathbf{A}_{b_3,h} = \begin{pmatrix} 0 & \sin 30° & \cos 30° \\ 0 & \cos 30° & -\sin 30° \\ -1 & 0 & 0 \end{pmatrix} $$

Each of the three finger modules of the hand consists of a two-link structure mounted on a revolute base and connected via rotational joints which is the minimal configuration needed to perform a desired object manipulation in space. The length of the links are (link one is closest to the finger base):

$$ l_1 = 20\,\text{mm} \quad l_2 = 55\,\text{mm} \quad l_3 = 54\,\text{mm} $$

At the end of link three, a half-sphere formed fingertip with radius r = 10 mm is fastened. By defining frames at each joint in a way that the x-axes point in direction of the respective link and the z-axes point in direction of the rotation axis (Fig. 3), the transformations from the finger frames C_{f_i} fixed at the end of link three into the finger base frames are obtained by ($c_j = \cos\theta_{i_j}$ and $s_j = \sin\theta_{i_j}$):

$$ {}^{b_i}\mathbf{r}_{f_i,b_i} = \begin{pmatrix} c_1\big(l_3(c_2c_3 - s_2s_3) + l_2c_2 + l_1\big) \\ s_1\big(l_3(c_2c_3 - s_2s_3) + l_2c_2 + l_1\big) \\ l_3(c_2s_3 + s_2c_3) + l_2s_2 \end{pmatrix} \qquad \mathbf{A}_{f_i,b_i} = \begin{pmatrix} c_1(c_2c_3 - s_2s_3) & -c_1(c_2s_3 + s_2c_3) & s_1 \\ s_1(c_2c_3 - s_2s_3) & -s_1(c_2s_3 + s_2c_3) & -c_1 \\ (c_2s_3 + s_2c_3) & c_2c_3 - s_2s_3 & 0 \end{pmatrix} $$

From them, all other transformations, especially the corresponding Jacobians ${}^{b_i}\mathbf{J}$ can be obtained.

Now, the computation process of all force parameters required for dextrous manipulation of a grasped object with pre-selected contact points is shown on the base of the grasp planning approach presented in chapter 5 that deals with the standard case of fno = 3 (jno_i = 3), cmod_i = 3 and ndim = 3.

The planning process of parameter determination for a peg-in-hole insertion task [25] uses a specific force area that describes the external forces and moments $({}^{o}\mathbf{f}_{ext}, {}^{o}\mathbf{m}_{ext})^T$ acting onto the object during manipulation to determine appropriate contact forces $\boldsymbol{f}$ for the fingers of the hand. According to positional and orientational tolerances between peg and hole, several simulation runs are performed to compute the external forces in x- and z-direction and the external moment in y-direction. The resulting force area **EA** is represented as task polyeder formed by 2^6 = 64 (or 2^3 = 8) force vectors.

$$ \text{Tolerances of peg location [mm]}: \begin{array}{l} \left(({}^{o}\mathbf{x}_{peg})_x, ({}^{o}\mathbf{x}_{peg})_y, ({}^{o}\mathbf{x}_{peg})_z\right) = \{[-5,5],[0,0],[-3,3]\} \\ \left(({}^{o}\mathbf{x}_{peg})_{\phi x}, ({}^{o}\mathbf{x}_{peg})_{\phi y}, ({}^{o}\mathbf{x}_{peg})_{\phi z}\right) = \{[0,0],[-7,7],[-0,0]\} \end{array} $$

$$ \text{External forces and moments [mN]}: \begin{array}{l} \left(({}^{o}\mathbf{f}_{ext})_x, ({}^{o}\mathbf{f}_{ext})_y, ({}^{o}\mathbf{f}_{ext})_z\right) = \{[0,550],[0,0],[0,320]\} \\ \left(({}^{o}\mathbf{m}_{ext})_x, ({}^{o}\mathbf{m}_{ext})_y, ({}^{o}\mathbf{m}_{ext})_z\right) = \{[0,0],[0,-33000],[0,0]\} \end{array} $$

The determination of a stability interval is done with the help of a search procedure that computes in a first run the most stable grasp force with minimum security (= 0) and the maximum stability parameter (= 1100). Because this step fails on the strength of the defined friction conditions, an interval search (most secure grasp force with minimum stability) is performed over the stability interval [0, 1100] to compute the best compensating grasp force for the manipulation force area MA. This leads after an appropriate number of iterations (desired accuracy = 0.01) to the optimal internal force λ.

$$\text{Resulting internal (grip) force [mN]} : (\lambda_1, \lambda_2, \lambda_3) = (5400, 6300, 5200)$$

Each compensating grasp force in the contact space that is determined with respect to a single manipulation force or a range of manipulation forces can be graphically visualized by the simulation system. For that purpose, perpendicular projections of the friction cones in the contact points are displayed together with the tangential forces in the yz-plane and the normal forces along the x-direction.

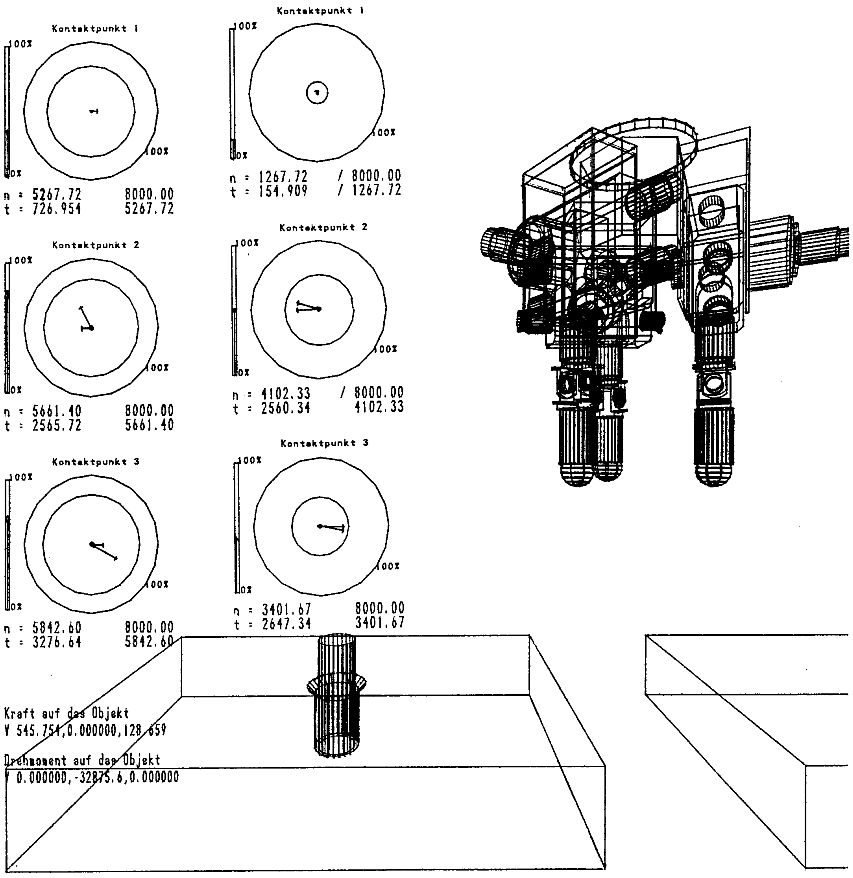

Fig. 8 : Computation of stable and secure grasp forces

In Fig. 8, the exterior circles represent the borders of the friction cones at the height of the maximal contact force whereas the interior circles lie on the borders of the friction cones at the height of the superposition of manipulation and compensating grasp forces. The diagrams of the computed contact forces point out very clearly that the stability search shown on the left hand side maximizes the length of the grasp forces. Consequently, the distances between the interior and exterior circles are relatively small. In contrary, the security search shown on the right hand side minimizes the length of the grasp forces, so that the distances between the interior and exterior circles are relatively large.

The simulation of reaction forces and moments occurring during part mating operations according to positional and/or rotational uncertainties of object motions and their effect on the world model are also considered at the example of the peg-in-hole insertion problem. The force simulator is a program package based on a stiffness control model of the insertion process [26] which has the ability to modify the compliance parameters and to analyze the finger-object interactions at the defined contact points. This permits testing of manipulation strategies for specific assembly tasks by taking into account the object forces and moments, contact forces and stability conditions of the grasped object.

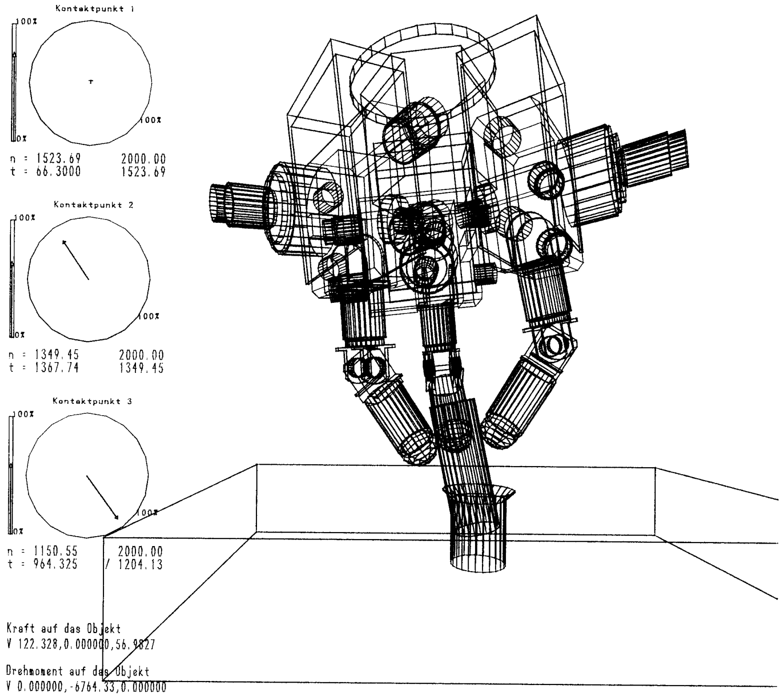

Fig. 9 : Visualization of a peg-in-hole insertion process

Fig. 9 shows the graphical visualization of a peg-in-hole insertion operation with the three-fingered Karlsruhe Dextrous Hand. This is basis for the comparison of different grasp configurations according to their robustness against external influences (disturbances) and allows also the verification of the planned force parameters. Therefore, the grasp planning process must ensure that the computed contact forces stay inside of the friction cones what results in a stable and secure grip of the object.

8. Conclusions und future work

This paper describes the fundamentals of object manipulation and grasp planning for dextrous robot hands considering both rolling and slipping of the fingertips on the object surface. The process of force computation and evaluation of suitable grasp configurations is shown for two different approaches. As major result, the force optimization problem could be generalized for the case of arbitrary robot hands and contact models which are specified by the parameters fno, jno_i, $cmod_i$ and ndim.

The computation methods with their combination of matrix multiplications, matrix inversions, solution of equation systems and search for optimal force parameters are well-suited for the application and execution on parallel machines. Since the presented algorithms, like the maximum search algorithm, are formulated in a way that supports rather their realization on a serial or distributed computer, their transformation and implementation on a parallel computer is an interesting new approach. Especially, because the parallelization of algorithms and their adaptation to massive parallel machines with lots of processor nodes is actually an underdeveloped area in the field of robotics research.

One major problem arises hereby from the nature of the contact forces and therefore for the internal grip forces λ itself: they are crosswise depending on each other, so that they cannot be treated independently. This is expressed in the additive nature of the grip submatrices $\mathbf{G}_i$ which form altogether the grip matrix $\mathbf{G}$ that describes both the contact points and contact models, and characterizes the resulting contact forces that can be applied with respect to given external forces and moments. This means, that the computational effort required for solving the optimization problem cannot easily be subdivided upon independent processes which would improve the search procedure tremendously.

Nevertheless, it is possible to use parallelism to speed up the search procedure that tries to optimize the force parameters by fitting them into the biggest volume in $\Re^{ndim}$ that has the greatest distance to all given constraints. This could be done by splitting the whole search space into several regular shaped areas from which independent working processes could start their parameter search; because there are no restrictions on the choice of the start points, the convergence of the algorithm to the resulting end points which are possible (optimal) solutions is always ensured. Thus, research will in the future be more concentrated on questions of parallelizing the search and optimization algorithm.

Acknowledgment

Part of this research work was performed at the Institute for Real-Time Computer Control Systems and Robotics, Prof. Dr.-Ing. Ulrich Rembold and Prof. Dr.-Ing. Rüdiger Dillmann, Faculty of Informatics, University of Karlsruhe, 76128 Karlsruhe, Federal Republic of Germany. The continuation of the work and extension of the theory was performed during my post-doctoral research stay at the International Computer Science Institute, Prof. Jerome Feldman, Berkeley 94704 CA, USA.

References

[1] Hollerbach, J.M.: "Robot Hands and Tactile Sensing", AI in the 1980s and Beyond, Grimson, W.E.L.; Patil, R.S.; MIT Press, 1987.

[2] Wöhlke, G.: "Development of the Karlsruhe Dextrous Hand", Proc. of the 3rd Int. Conf. on New Actuators (ACTUATOR), 1992.

[3] Doll, Th.; Schneebeli, H.J.: "Framework of the Karlsruhe Dextrous Hand", Proc. of the Symp. on Robot Control (SYROCO 88), 1988.

[4] Doll, Th.: "Development and Programming of a Robot Hand", PhD thesis (in German), University of Karlsruhe, 1989.

[5] Wöhlke, G.: "A Programming and Simulation Environment for the Karlsruhe Dextrous Hand", Jour. of Robotics and Autonomous Systems, North-Holland, vol. 6(3), pp. 243-263, 1990.

[6] Wöhlke, G.: "Automatic Grasp Planning for Multifingered Robot Hands", Jour. of Intelligent Manufacturing, vol. 3(5), pp. 297-316, 1992.
[7] Okada, T.; Tsuchiya, S. : "On a Versatile Finger System", Proc. of the 7th Int. Symp. of Industrial Robots, 1977.
[8] Okada, T.: "Computer Control of Multijoined Finger System for Precise Object Handling", IEEE Trans. on Systems, Man and Cybernetics, vol. 12 (3), May/June 1982.
[9] Jacobson, S.C.; Wood, J.E.; Knutti, D.F.; Biggers, K.B.: "The Utah/MIT Dextrous Hand: Work in Progress", First Int. Conf. on Robotics Research, MIT Press, 1984.
[10] Jacobson, S.C. et.al.: "Design of the Utah/MIT Dextrous Hand", Proc. of the IEEE Int. Conf. on Robotics and Automation, 1986.
[11] Narasiham, S.: "Dextrous Robotic Hands: Kinematics and Control", Master thesis, Department of Electrical Engineering and Computer Science, MIT, 1988.
[12] Salisbury, K.; Brock, D.; Chiu, S.: "Integrated Language, Sensing and Control for a Robot Hand", Proc. of Int. Symp. on Robotics Research, MIT Press, 1986.
[13] Nguyen, V.: "Constructing Stable, Force-Closure Grasps", Master Thesis, Department of Electrical Engineering and Computer Science, MIT, 1986.
[14] Arbib, M.A.; Iberall, T.; Lyons, D.: "Coordinated Control Programs for Movements of the Hand", Center of Systems Neuroscience and Laboratory for Perceptual Robotics, COINS Technical Report 83-25, Massachusetts, 1983.
[15] Lyons, D.M.: "A simple Set of Grasps for a Dextrous Hand", Proc. of the IEEE Int. Conf. on Robotics and Automation, 1985.
[16] Iberall, T.: "Grasp Planning for Human Prehension", Proc. of the Int. Joint Conf. on Artificial Intelligence, 1987.
[17] Härtl, H.: "Analysis of Manipulation Processes with Multifinger Gripper Systems under the Consideration of Rolling and Slipping of the Fingertip", Master thesis (in German), University of Karlsruhe, 1992.
[18] Craig, J.J.: "Introduction to Robotics: Mechanic and Control". Addison-Wesley, 1986.
[19] Kerr, J.; Roth, B.: "Analysis of Multifingered Hands", Int. Jour. of Robotics Research, vol. 4(4), pp. 3-17, Winter 1986.
[20] Cole, A.; Hauser, J.; Sastry, S.: "Kinematics and Control of Multifingered Hands with Rolling Contact". Proc. of the IEEE Int. Conf. on Robotics and Automation, 1988.
[21] Hsu, P.; Li, Z.; Sastry, S.: "On Grasping and Coordinated Manipulation by a Multifingered Robot Hand". Proc. of the IEEE Int. Conf. on Robotics and Automation, 1988.
[22] Vogelgesang, V.: "Development of a planning component for the determination of stable grip configurations to do parts assembly with multi-finger grippers", Master thesis (in German), University of Karlsruhe, 1990.
[23] Bronshtein, I.N.; Semendyayev, K.A.: "Taschenbuch der Mathematik", Harri Deutsch Verlag, 1987.
[24] Bronshtein, I.N.; Semendyayev, K.A.: "Ergänzende Kapitel zum Taschenbuch der Mathematik", Harri Deutsch Verlag, 1991.
[25] Wöhlke, G.: "Knowledge-Based Grasp Planning for Multi-Finger Robot Hands", PhD these (in German), University of Karlsruhe, 1991.
[26] Whitney, D.E.: "Quasi-static Assembly of Compliantly Supported Rigid Parts", ASME Transactions on Dynamic Systems, Measurement, and Control, vol. 104(1), 1982.