Literature research notes on Impedance Control

Martin Angerer

Master Student, ITR, TUM

martin.angerer@tum.de

Abstract—**This work gives a brief overview over relevant works in the field of impedance control used in ccoperative manipulation. Different control setups are explained, compared and assesed.**

I. Intrduction

Cooperative manipulation systems are in general a set of *N* manipulators rigidly connected to a rigid object. The rigid connections transmit both force and and are expressed by a number of kinematic constraints, m_C, which reduce the degrees of freedom (DOF) of the constrained system. Motion or the attempt to move in the constrained directions results in internal loading of the object. Therefore trajectories of the manipulators have to be generated to avoid internal stress on the object. In some applications internal forces on the object are desired, so that motion and internal forces are to be controlled independently. Impedance control is a well known strategy to avoid unbounded forces when it comes to inter-action of manipulators with objects/environment. The policy does not control one state variable (e.g. position, velocity, force) but enforces a relation between them. The impedance control scheme can be described as a spring- damper- mass system.

$$M\Delta\ddot{x} + D\Delta\dot{x} + K\Delta x = h \tag{1}$$

Where *M,D,K* can be arbitrarily chosen.

II. Selected work

Selected work which contribute to the problem of coopera-tive manipulation are presented in the following. This will be done chronologically.

A. Schneider, Cannon, "Object Impedance Control for Co-operative Manipulation: Theory and Experimental Results", 1992

An object focussed approach, the motion of the object is controlled and the object itself is being replaced by a "virtual object". The object is now a controlled impedance, which interacts with the environment via a set of damped springs. They are positioned at the virtual centre of mass and they are orthogonal, they connect to the environment in selectable points. The control law is

$$m_d(\ddot{x} - \ddot{x}_d) + k_v(\dot{x} - \dot{x}_d) + k_p(x - x_d) = f_{ext} \tag{2}$$

Where x is the one degree of freedom. x_d is the reference signal, $\ddot{x}_d$ denotes the feed forward desired acceleration. $\ddot{x}$ is the control output. Deviations from the desired trajectory due to the external force/moment f_{ext} are corrected by the impedance relation.

The object's equations of motion are

$$\begin{bmatrix} mI_3 & mR \\ 0_3 & I_3 \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{y}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} + \begin{bmatrix} -m\boldsymbol{g} - m(\boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{r})) \\ \boldsymbol{\omega} \times I\boldsymbol{\omega} \end{bmatrix}$$
$$= \begin{bmatrix} \boldsymbol{f}_{ext} \\ \boldsymbol{\tau}_{ext} \end{bmatrix} + \begin{bmatrix} \sum \boldsymbol{f}_i \\ \sum \boldsymbol{p}_i \times \boldsymbol{f}_i + \sum \boldsymbol{m}_i \end{bmatrix} \tag{3}$$

and written more compactly

$$I_0\ddot{Y} + B_0 = \boldsymbol{F}_{ext} + W\boldsymbol{F} \tag{4}$$

W denotes the grasp matrix.
The controller specifies a desired object behaviour to a external force

$$\begin{bmatrix} M_d & 0_3 \\ 0_3 & I_d \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{y}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} + \begin{bmatrix} 0 \\ \boldsymbol{\omega} \times I_d\boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_{ext} \\ \boldsymbol{\tau}_{ext} - \boldsymbol{r} \times \boldsymbol{f}_{ext} \end{bmatrix} \tag{5}$$

What again can be rewritten more compact

$$I_{0d}\ddot{Y} + B_{0d} = \tilde{\boldsymbol{F}}_{ext} \tag{6}$$

This gives directly the desired acceleration of the virtual object

$$\ddot{Y}_{cmd} = I_{0d}^{-1}(\tilde{\boldsymbol{F}}_{ext} - B_{0d}) \tag{7}$$

This output of the object impedance controller is now rein-serted in the actual object dynamics equation (4) to calculate the required wrench.

$$\boldsymbol{F}_{cmd} = W^{wpi}(B_0 - \boldsymbol{F}_{ext} + I_0 I_{0d}^{-1}(\tilde{\boldsymbol{F}}_{ext} - B_{0d})) + \boldsymbol{F}_{int} \tag{8}$$

W^{wpi} is a weighted pseudoinverse of the grasp matrix. With $\boldsymbol{F}_{int}$ an internal force can be added, but there is no closed force feedback loop. To complete the control loop the desired joint accelerations are computed via the inverse Jacobian. At the point of calculating the joint torques, the manipulator's dynamics are considered

$$\boldsymbol{\tau}_i = M_i\ddot{\boldsymbol{q}}_{i_{cmd}} + C_i(\boldsymbol{q}, \dot{\boldsymbol{q}}) + J_i^T(\boldsymbol{F}_{i_{cmd}}) \tag{9}$$

Finally $\boldsymbol{F}_{ext}$ has to be derived, this is done using (4) with an estimate of $\ddot{Y}$.
Conclusion: This pioneer work introduces a object impedance control scheme in operational space. The object trajectory is corrected w.r.t. to external forces. Dynamics of object and manipulators are compensated. An impedance relation between object and manipulators is not enforced, internal forces on the object can be set but are not regulated. The dynamic properties of the object have to be known.

B. Bonitz, Hsia, "Internal Force Based Impedance Control for Cooperating Manipulators", 1996

This work is going without knowledge of the dynamics of the object. The manipulators are given impedance properties and a relation between velocity and internal force is established. This is done in order to improve tracking and prevent position errors, due to unknown dynamics of the object. If the total forces exerted by the object on the manipulators were incorporated in the impedance control, it would lead to a deviation. Therefore the impedance control law is

$$M_i \Delta \ddot{x} + B_i \Delta \dot{x} + K_i \Delta x = \Delta h_{Ii} \quad (10)$$

M_i, B_i, K_i are the desired inertia, damping and stiffness matrices of the i-th robot.
$\Delta x = x_d - x$ is the position and orientation error of the i-th end-effector in task space.
$\Delta h_I = h_{Id} - h_I$ is the internal wrench error at the i-th robot. Similarly to the previous section the output acceleration can be inserted in the actual dynamic equation to compute the necessary wrench. The joint torque of the i-th manipulator is

$$\tau_i = D_i \{ J_i^{-1}(M_i^{-1}[M_i \ddot{x}_i d + B_i \Delta \dot{x} + K_i \Delta x - \Delta h_{Ii}] - \dot{J}_i \dot{q}_i) \} \\ + E_i + J_i^T h_i \quad (11)$$

D_i is the actual inertia of the i-th robot. E_i is the combined Coriolis, centripetal and gravity vector. h_{Iid} is calculated from the measured wrench using the null space projection of the grasp matrix.
On the input side, desired trajectories are specified for the object. They are mapped to the manipulators, compliant with the grasp geometry.
Conclusion: The interaction of manipulator and object is controlled. With the impedance relation between velocity and internal forces separate control loops for force and position are not necessary. Internal force can be specified and is regulated. The approach is operational space based and knowledge of the object dynamics is not necessary.

C. Caccavale, Villani, "An Impedance Control Strategy for Cooperative Manipulation", 2001

This work promises to combine the previously presented two attempts to control first the interaction forces between environment and object and second the relation between object and manipulator. Object and manipulators are both subject to impedance control. Compliant behaviour is enforced between manipulator and object and the trajectory is modified w.r.t to external forces. The unit quaternion for expressing orientation is introduced. The control strategy consists of a inner motion control loop and a outer position control loop. The inner loop is designed to guarantee tracking of a reference end-effector position and orientation. Therefore an inverse dynamics strategy is adopted for linearising and decoupling of the links. The joint torques are

$$\tau_i = M_i J_i^{-1}(a_i - \dot{J}_i \dot{q}_i) + C_i \dot{q}_i + g_i + J_i^T h_i \quad (12)$$

M_i, C_i are the inertia and Coriolis matrices, g_i, h_i are the gravity and wrench, acting at the i-th end effector, vectors. a_i is a new control input.

$$a_i = \ddot{x}_{i_r} + k_v(\dot{x}_{i_r} - \dot{x}_i) + k_P(x_r - x) \quad (13)$$

x_r, x denoting the reference and actual stacked vectors of position and orientation.
An impedance relation between object and environemnt is used to generate a reference object trajectory from the desired object trajectory:

$$\alpha M_o(\ddot{x}_{o_d} - \ddot{x}_{o_r}) + D_o(\dot{x}_{o_d} - \dot{x}_{o_r}) + K_o(x_{o_d} - x_{o_r}) = h_{env} \quad (14)$$

where M_o denotes the actual object inertia matrix, α is a positive constant to be chosen. D_o, K_o are the tunable damping and stiffness matrices. x_{o_d} along with its derivatives is the control input. The wrench exerted by the object on the environment is calculated:

$$h_{env} = h_e + m_0 g - M_o \ddot{x}_{o_r} \quad (15)$$

h_e is the wrench exerted by the manipulators on the object. Control output is the reference acceleration of the object $\ddot{x}_{o_r}$. With the geometry of the grasp, the inputs $x_{i_d}, \dot{x}_{i_d}, \ddot{x}_{i_d}$ of the manipulator/ object impedance controller can be determined. Similar to the former section the impedance relation is enforced between end effector displacement and internal force

$$M_I(\ddot{x}_{i_d} - \ddot{x}_{i_r}) + D_I(\dot{x}_{i_d} - \dot{x}_{i_r}) + K_I(x_{i_d} - x_{i_r}) = h_{Ii} \quad (16)$$

No desired internal wrench is included in this approach but can easily be added. The reference acceleration $\ddot{x}_{i_r}$ is the control output and along with its integrals the input for the inner motion control loop.
Conclusion: The overall control scheme consists of three levels: an inner motion controller of PD- structure with acceleration feed-forward. With inverse dynamics control it compensates for the manipulator- dynamics. Superimposed is a impedance controller for the manipulator/object relation which corrects the reference trajectories if internal forces are present and therefore allows high gains in the inner motion loop. The highest level is the impedance controller for environment/object behaviour, it ensures object-compliance w.r.t external forces. All controllers are realized in operational space. Inertia of the object has to be known.

D. Caccavale, Chiacchio, et al., "Six-DOF Impedance od Dual-Arm Cooperative Manipulators", 2008

The work is very similar to the former. The overall control scheme is equal, in the inner motion control loop the inverse dynamics controller is replaced by an PID one. This is due to the wide distribution of PID controller in industrial robots, with which the experimental part of the work is carried out. The main contribution of the work is extending the *geometrically consistent stiffness* to the case of finite displacements. Moreover the presented concept allows to specify direction dependent stiffness.
The experimental results show good performance also in the case of non-rigid objects.

E. Erhart, Hirche, "Model and analysis of the interaction dynamics in cooperative manipulation tasks", 2014

This work is confined to task space considerations, only the end-effector dynamics are viewed by assuming perfect linearisation and decoupling of the subjacent manipulator structure. Moreover in this way actual manipulator-dynamics can be replaced by a desired second-order impedance behaviour:

$$M_i[\ddot{x}_i - \ddot{x}_i^d] + D_i[\dot{x}_i - \dot{x}_i^d] + h_i^K(x_i, x_i^d) = h_i - h_i^d \quad (17)$$

x_i denotes the stacked position and orientation vector, h_i the end-effector wrench of the i-th manipulator. The inertia, M_i, and Damping, D_i, matrices are block-diagonal, decoupling translational and rotational behaviour. The term $h_i^K(x_i, x_i^d)$ is the *geometrically consistent stiffness* and is similarly defined to the former. Stiffness is considered to be isotropic, i.e. there is no coupling between the single DOF.
Lagrange formulation of the object dynamics is:

$$M_o \ddot{x}_o + C_o \dot{x}_o + h_g = h_o \quad (18)$$

M_o, C_o, h_g are the object's inertia, centripetal force and gravity parameters.
The kinematic constraints impose additional forces of interaction on the system. Both manipulator and object dynamics can be re-formulated to distinguish between a local or non-interacting wrench and a global or interacting wrench.

$$M_i \ddot{x}_i = h_i^\Sigma + h_i \quad (19)$$

$$M_o \ddot{x}_o = h_o^\Sigma + h_o \quad (20)$$

Where $h_i^\Sigma = M_i \ddot{x}_i^d - D_i[\dot{x}_i - \dot{x}_i^d] - h_i^K(x_i, x_i^d)$ and $h_o^\Sigma = -C_o \dot{x}_i^d - h_g$. The equations of the complete system can be written as

$$M\ddot{x} = h^\Sigma + h \quad (21)$$

with $M = blkdiag(M_o, M_1, ..., M_N)$, $h^\Sigma = (h_o^\Sigma; h_1^\Sigma; ...; h_N^\Sigma)$ and $h = (h_o; h_1; ...; h_N)$ The rigid grasp kinematic constraints can be formulated in matrix- vector notation:

$$A\ddot{x} = b \quad (22)$$

Employing *Gauss principle of the least constraint* an explicit solution for h be found:

$$h = M^{\frac{1}{2}}(AM^{-\frac{1}{2}})^\dagger (b - AM^{-1}h^\Sigma) \quad (23)$$

Where $\dagger$ denotes the *Moore-Penrose generalized inverse* [1]. To address asymmetric load distributions the paper proposes a specific weighted generalized inverse of the grasp matrix for manipulator coordination.
Concluding the system description an overall impedance of the constrained system is formulated, which is effective when it comes to disturbances acting on the object.
The paper proposes to calculate the internal force acting on the object via a reduced system consisting of constrained manipulators without the object. Therefore the reduced kinematic constraints are $A'\ddot{x}' = b'$. $\ddot{x}' = [x_1^T, ..., x_N^T]^T$ is the stacked vector of position and orientation of the N manipulators. Consequently A' is the $6N$ identity matrix. With

$b' = b = [(S(\omega_0)^2 r_1)^T, 0_{3\times 1}, ..., (S(\omega_o)^2 r_N)^T, 0_{3\times 1}]^T$ the accelerations consistent with the constraints are $\ddot{x}' = b$. The proposed calculation of the internal force is

$$h'_{int} = M'^{\frac{1}{2}}(A'M'^{-\frac{1}{2}})^\dagger (b' - A'M'^{-1}h'^\Sigma) \quad (24)$$

with the commanded acceleration vector $\ddot{x}'^x$. Computing this equation one obtains:

$$h'_{int} = [(M_1 S(\omega_0)^2 r_1 - f_1^\Sigma)^T, (-t_1^\Sigma)^T, ..., \\ , (M_N S(\omega_0)^2 r_N - f_N^\Sigma)^T, (-t_N^\Sigma)^T]^T \quad (25)$$

With f_*, t_* being the force and torque vectors exerted by the manipulators. Apart from the centripetal force, the internal forces are just the opposite of the wrench exerted by the manipulators. One gets the same result using Newton's third law. Thus every attempt changing the dynamic state of the object, i.e. accelerating the object, leads to internal loading.
Conclusion: In terms of control architecture, the key idea presented in this work is replacing the actual dynamic properties of the manipulators by a desired mechanical impedance. The rest of the paper is modeling the constrained system: an explicit solution for the constraining wrenches is given. The constrained system response to disturbances is formulated. A (complicated) solution for calculation internal forces is presented. The proposed plant model is exploited for control design in the next section.

F. De Pascali, Erhart, et al., "A Decoupling Scheme for Force Control in Cooperative Multi-Robot Manipulation Tasks", 2015

Although partially written by the same authors than the former, this work comes up with another new calculation method for the internal forces. They are defined as the forces/torques which do not produce virtual work for a virtual displacement compliant with the kinematic constraints. Geometrically spoken this means that internal forces/torques are orthogonal to the kinematic constraints. Internal wrench calculation starts from re-formulating the kinematic constraints for the relative position/orientation of the end-effectors. This is done by subtracting the dynamics equations of the i-th manipulator from the j-th manipulator. This eliminates the dynamics of the object from the formulation. The relative position is $\Delta r_{ji} = r_j - r_i$. The resulting constraint matrix is:

$$\bar{A} = \begin{bmatrix} -I_3 & S(\Delta r_{21}) & I_3 & 0_3 & & \\ 0_3 & -I_3 & 0_3 & I_3 & & \\ \vdots & \vdots & & & \ddots & \\ -I_3 & S(\Delta r_{N1}) & & & I_3 & 0_3 \\ 0_3 & -I_3 & & & 0_3 & I_3 \end{bmatrix} \quad (26)$$

The basis of the nullspace of the grasp matrix G is $\bar{A}^T$. This is equal to $Ker(G) \equiv Im(\bar{A}^T)$, what is shown in the paper. The internal wrench is calculated as

$$h^{int} = \bar{A}^T(\bar{A}M^{-1}\bar{A}^T)^{-1}(\bar{b} - \bar{A}M^{-1}\bar{h}^\Sigma) \quad (27)$$

Where $M = blkdiag(M_1, ..., M_N)$. It is shown that $h^{int} \in ker(G)$, i.e. it does not influence motion. On the other hand it is shown that a commanded object motion mapped to the manipulators using G does not generate internal wrench. Just as in the previous section, the motion control law is:

$$M_i[\ddot{x}_i - \ddot{x}_i^d] + D_i[\dot{x}_i - \dot{x}_i^d] + h_i^K(x_i, x_i^d) = h_i - h_i^d \quad (28)$$

Another re-formulation is introduced:

$$M\ddot{x} = h^x + h^d + h \quad (29)$$

Where h^x accounts for the local end-effector dynamics, h^d can be seen as a feed-forward term of the desired object acceleration, h is again the interaction wrench. Note that in this work the stacked vector $h = [h_1^T, ..., h_N^T]^T$ excludes the wrench acting on the object. h^d is calculated from h_o^d with a non-squeezing pseudo-inverse G_M^+ of the grasp matrix. The desired velocity $\dot{x}^d$ is calculated with the transposed grasp matrix G^T. The control law to render the internal forces is

$$h_w^x = \bar{A}^T(\bar{A}M^{-1}\bar{A}^T)^{-1}(\bar{b} - \bar{A}M^{-1}(h^{int,d} + h^x)) \quad (30)$$

Interestingly, next to the desired internal forces $h^{int,d}$, the impedance control output of the manipulators h^x is incorporated in the internal force allocation. Vice versa this means that h^x is not free of internal force components. $h^{int,d} = \bar{A}^T z$ is calculated to lie in the null space of the grasp matrix. Recall that $\bar{A}^T$ is the basis of the null space of G. If h^x would be free of internal wrench, then $\bar{A}^T h^x = 0_{6N \times 1}$.

$$\bar{b} - \bar{A}M^{-1}h^x = \bar{b} - \bar{A}M^{-1}(M\ddot{x}^d - D[\dot{x} - \dot{x}^d] - h^K(x, x^d)) =$$
$$= \bar{b} - \underbrace{\bar{A}M^{-1}MG^T}_{0_{6(N-1)\times 6}}\ddot{x}_o^d - \underbrace{\bar{A}M^{-1}Mb}_{\bar{b}} + \underbrace{\bar{A}M^{-1}DG^T}_{0_{6(N-1)\times 6}}[\dot{x}_o - \dot{x}_o^d] +$$
$$+ \bar{A}M^{-1}h^K(x, x^d) = \bar{A}M^{-1}h^K(x, x^d) \quad (31)$$

There are two possibilities: 1. the *geometrically consistent stiffness* term produces internal wrench or 2. it does not and I am just incapable of proofing $\bar{A}M^{-1}h^K(x, x^d) = 0_{6(N-1)\times 1}$. If 2. is true then $h_w^x = \bar{A}^T z = h^{int,d}$, in agreement with previous literature.

Finally the resulting object acceleration is derived:

$$\ddot{x}_o = (M_o + GMG^T)^{-1}(h_o^\Sigma + G(h^x + h_w^x + h^d) - GMb) \quad (32)$$

G. Wimböck, Ott, et al.,"Comparison of object-level grasp controllers for dynamic dextrous manipulation", 2011

This paper treating dexterous manipulation goes without the assumption of a rigid grasp. For grasp description the *point contact with friction* model is used. I.e. only forces can be transmitted. It is pointed out that classical impedance relations in form of a *mass-spring-damper* systems require knowledge of the external forces for inertia re-shaping. Therefore the proposed control strategies stick to the natural inertia and and address only stiffness and damping, this is called *compliance control*. In the paper two *Intrinsically Passive Controllers (IPC)* are presented, the first one called "Dynamic IPC" is based on [2].
Herein the actual object is replaced by a virtual one, connected

to the end-effectors and a virtual equilibrium position with spatial springs. The end-effector springs do not connect to the center of the virtual object but to *virtual contact points* on the virtual object. Similar to the actual grasp matrix a virtual equivalent G_v can be defined. This is to calculate the forces exerted by the springs on the virtual object. f_c is the stacked vector of forces generated by the end-effector coupling springs, w_{vo} is the wrench exerted by the desired object position spring. Therefore the total wrench on the object is $w_v = w_{vo} + G_v f_c$. f_c and w_{vo} are the control output from the virtual spring-damper system, which represents a *compliance control* law. End-effector-object stiffness is defined by the distance of end-effector and virtual contact point which describes the equilibrium length of the spring. Since there exists no direct way of specifying internal forces, they can be adjusted by setting the virtual contact points.
Recalling Subsection II-C, here an impedance relation between end-effector and object and environment and object is established. Main difference is the usage of virtual and actual object, this is because of different assumptions on availability of object tracking. One strategy for object pose estimation is the introduction of a virtual object, the dynamics is then calculated by:

$$M_v\ddot{x}_v + C_v\dot{x}_v = w_v \quad (33)$$

M_v, C_v denote Inertia and Coriolis matrices of the virtual object.
The virtual grasp matrix deviates from the actual grasp geometry, wrenches on the object due to end-effector forces will not be calculated correctly. The spatial springs enforce a compliant behaviour even with many unknown parameters. This method goes without tracking of the object and exact knowledge of the grasp geometry (e.g. for non-rigid objects). It is closely related to II-C, but suits for more general tasks.
Another control design is proposed in the paper called *static IPC*, the virtual object is replaced by a virtual-object frame neglecting dynamics of the object. The end-effector coupling springs connect to the frame, i.e. to the center of the virtual object. Stiffness is defined by specifying a equilibrium length for the springs. Object motion is derived by a Jacobian matrix from manipulator motion expressed in generalized coordinates. Object inertia is only considered in the damping design. Over the *dynamic IPC* the *static IPC* controller has some advantages in computational effort and simpler control parameter selection.
Conclusion: Control architecture of the *dynamic IPC* ist similar to II-C, extending the applicability to non object tracking systems and non rigid objects. This comes at the price of introducing an error into object wrench calculation and loosing the ability to re-shape the components' inertia. The *static IPC* is a further simplified control structure which can be of interest of the object inertia is totally unknown.

H. Sieber, Musić, Hirche, "Formation-based robot team controlled by a single human", 2015

In contrast to the previously presented work this control approach is non-object based, instead the formation of the

manipulators is controlled. This *formation control scheme* establishes desired distances d_{ij} between the end-effectors using a *Artificial Potential Fields*.

$$V_{ij,k}(|x_{i,k} - x_{j,k}|) = \frac{1}{2}(|x_{i,k} - x_{j,k}| - d_{ij,k})^2 \quad (34)$$

The subscript k denotes the Cartesian direction. The control signal $u_{i,k}$ is calculated from the quadratic potential field by using the gradient descent:

$$u_{i,k} = -\sum_{j \in N_i} \delta_{ij} \frac{\partial V_{ij,k}(|x_{i,k} - x_{j,k}|)}{\partial x_{i,k}} =$$
$$-\sum_{j \in N_i} \delta_{ij} \frac{x_{i,k} - x_{j,k}}{|x_{i,k} - x_{j,k}|}(|x_{i,k} - x_{j,k}| - d_{ij,k}) \quad (35)$$

Where N_i is described as the "neighbours" of the i-th manipulator, it is assumed that "neighbours" means all the other manipulators. δ_{ij} is a positive gain, which can be defined to adjust convergence rate.

The *Artificial Potential Field* can be seen as a set of springs connecting the end-effectors among each other. δ_{ij} represents the stiffness, d_{ij} the equilibrium length. The control input u_i establishes a constant formation of the manipulators using position measurements and communication, i.e. each manipulator knows the position of the others. This can be seen as the low-level part of the control structure, responsible for maintaining the grasp and depends only on the robot-system.

The high-level part specifies the constrained motion of the structure and incorporates the human operator. In the previous works the presence of an object and the assumption of rigid grasping implicates constraints on the overall system. For this non-object based approach one lacks this restrictions of motion between the end-effectors. To avoid breaking up of the formation the operator should only be able to control only one position, namely the position of the entire formation, which corresponds to the average of the manipulator positions $x_c = \frac{1}{\sqrt{N}}\sum_{i=1}^N x_i$. It is also pointed out that this high-level input has to be feed into all manipulators uniformly. Not doing so entails in a leader-follower situation, this results in internal stress on the object or break-up of formation due to sensing delays of the following manipulators.

On end-effector level an impedance relation is utilized to ensure compliant behaviour for minor deviations.

Conclusion: A simple control scheme, requires only knowledge of the end-effector positions. On the other no dynamics are considered. Clear division in low-level grasp-maintaining control loop and high-level constrained-motion loop.

I. Wimböck, Ott, Hirzinger, "Passivity-based Object-level Impedance Control for a Multifingered Hand", 2006

This paper treats the *static IPC* controller from subsection II-G in further detail. Although the control scheme outputs torques on joint level the key control loops are formulated on object-level. It is an object centred approach, though no actual object is incorporated. Instead the representation is restricted to a virtual object frame lying in the center of the manipulators

$x_o = \frac{1}{N}\sum_{i=1}^N x_i$. Orientation, expressed with the matrix R_o, is also derived dependent on the end-effectors.

The given impedance control law

$$\tau = -D(q)\dot{q} - \frac{\partial V_d(q)}{\partial q} + g(q) \quad (36)$$

computes the joint torques incorporating a damping $D(q)$, a stiffness $\frac{\partial V_d(q)}{\partial q}$ and a gravitational $g(q)$ term. V_d and D are formulated in operational space first, and then differentiated w.r.t to joint angles. First the approach is summarized as in the paper, then the key ideas are re-formulated for pure task space control. The stiffness contribution divides in a object translation/rotation and a object-manipulator connection part.

$$V_d = V_{o,t} + V_{o,r} + V_{conn} \quad (37)$$

The virtual object frame is connected to a desired pose $(p_{o,d}, \epsilon_d)$ via a 6-DOF spring. Translational and rotational stiffness potentials are:

$$V_{o,t} = \frac{1}{2}(p_o - p_{o,d})^T R_o K_{o,t} R_o^T (p_o - p_{o,d}) \quad (38a)$$
$$V_{o,r} = 2\epsilon_d^T K_{o,r} \epsilon_d \quad (38b)$$

To compute power from the potential energy functions, they are differentiated w.r.t to time:

$$\dot{V}_{o,t} = \dot{\theta}^T J_{o,t}^T K_{o,t} R_o^T (p_o - p_{o,d}) \quad (39a)$$
$$\dot{V}_{o,r} = \dot{\theta}^T 4 J_{o,r}^T J_{\omega\epsilon}^T K_{o,r} \epsilon_d \quad (39b)$$

Where $J_{o,t}$ is the Jacobian mapping object velocity to joint velocity: $\dot{p}_o = J_{o,t}\dot{\theta}$. $J_{o,r}$ connects the object angular velocity with the joint velocity: $\omega_o = J_{o,r}\dot{\theta}$. $J_{\omega\epsilon}$ denotes the (reduced) matrix-quaternion product. The gradients of the potentials $\frac{\partial V}{\partial \theta} = \frac{\dot{V}}{\dot{\theta}^T}$ are the object-stiffness wrenches.

The manipulators are connected to the virtual object frame with 1-DOF springs, these have an equilibrium length d_i. This is the desired distance between the manipulator position p_i and the virtual object center p_o: $\Delta p = p_i - p_o$. This ensures a compliant behaviour between object and manipulator even in the case of non-rigid objects. Equilibrium length can be varied to define internal forces on the object. The potential of the springs is:

$$V_{conn} = \frac{1}{2}\sum_{i=1}^N K_{conn,i}\left[|\Delta p_i| - d_i\right]^2 \quad (40)$$

The joint torque contribution of the connecting springs is derived with the gradient of the potential:

$$\frac{\partial V_{conn}}{\partial \theta} = \left(\frac{\partial p}{\partial \theta^T}\right)^T \sum_{i=1}^N \left[\frac{\partial \Delta p_i^T}{\partial p} \frac{K_{conn,i}(|\Delta p_i| - d_i)}{|\Delta p_i|}\Delta p_i\right] \quad (41)$$

This concludes the stiffness contribution to the joint torque control law.

For the derivation w.r.t. time of the potentials [3] is followed:

$$h_o = \begin{bmatrix} \dot{V}_{o,t} \\ \dot{V}_{o,r} \end{bmatrix} = \begin{bmatrix} K_{o,t}\Delta p_o \\ 2(\eta_d I_3 - S(\epsilon_d))K_{o,r}\epsilon_d \end{bmatrix} \quad (42)$$

Virtual object frame and manipulators are connected with 1-DOF springs, the control output is:

$$f_{conn} = \frac{\partial V_{conn}}{\partial p} = \sum_{i=1}^{N} \left[\frac{\partial \Delta p_i^T}{\partial p} \frac{K_{conn,i}(|\Delta p_i| - d_i)}{|\Delta p_i|} \Delta p_i \right]$$

(43)

h_o represents wrenches on object-level, f_conn forces on end-effector-level, the problem of mapping the object-level wrenches to desired end-effector forces arises. As discussed previously this can be done with an inverse grasp matrices. Options are using the actual grasp (if known) or using a virtual grasp.

REFERENCES

[1] Udwadia F. and Kalaba R., "A new perspective on constrained motion", *Proceedings: Mathematical and Physical Sciences*, vol. 439, no. 1906, pp. 407 - 410, 1992

[2] Stramigioli S., "Modeling and IPC Control of Interactive Mechanical Systems: A Coordinate-free Approach", volume 266 of *Lecture Notes in Control and Information Sciences.* Springer, Berlin, 2001

[3] Natale C., "Interaction Control of Robot Manipulators: Six degrees-of-freedom tasks", volume 3 of *Springer Tracts in Advanced Robotics.* Springer, Berlin, 2003