Posture Modification for Biped Humanoid Robots Based on Jacobian Method

Dirk Wollherr[0] and Martin Buss
Institute of Automatic Control Engineering
Technische Universität München
München, Germany
D.Wollherr@ieee.org   M.Buss@ieee.org

***Abstract*— An online posture modification method termed Jacobi Compensation is proposed which is suitable to modify precalculated step trajectories for a humanoid robot in certain task coordinate directions. This method can account for modeling errors in trajectory precalculation by shifting e.g. the center of mass (CoM) or certain parts of the humanoid mechanism to increase walking stability and performance. A theoretical analysis of stability properties is given.**

I. Introduction

A major issue in biped locomotion is coping with unforeseen situations. While robots computing their motion trajectories online [1]–[4] generally are able to recalculate the motion pattern based on their intention or on sensor signals, this is more difficult for bipeds disposing of a large set of offline computed trajectories [5]–[8]. The robot selects one of the precalculated trajectories according to the situation [9]. In recent work [5]–[7] it has been shown that suitable step and stride trajectories can be computed using advanced numerical optimal control algorithms, considering a variety of constraints such as dynamic stability, joint actuator torque limitations, contact force constraints at the feet as well as "esthetic" criteria [10] like smoothness and energy efficiency. Such computations generally cannot be executed online with current computational hardware, hence many research groups focus on offline generation of trajectories.

One disadvantage of precalculated trajectories is that the robot is unable to accomplish motions for which trajectories are not available. It is therefore desirable to adapt trajectories making them applicable to slightly different situations than originally intended. One of the key challenges is sensor-based online modification of computed trajectories to improve walking stability and performance.

As a tool to achieve this goal, a method called *Jacobi Compensation* has been proposed in [11], which enables the movement of parts of the body in selected task coordinate directions and thus alters the posture of the robot. The novel point of this paper is an analysis of the stability properties of closed loop control using Jacobi Compensation.

Efforts to modify precalculated trajectories online followed by other research groups focus on ensuring gait stability by controlling the Zero Moment Point (ZMP) [12] or adapting existing trajectories for walking in the plane to ones that are suitable to walking on slopes [13]. Modification of precalculated trajectories involving Jacobians is part of an online obstacle avoidance method presented in [14]. The online use of Jacobians for posture control of biped robots is considered novel.

The paper is structured as follows: Sec. II presents the Jacobi Compensation method, simulation results with this control strategy are shown in Sec. III. In Sec. IV an analysis of the stability properties is given. A solution to increase robustness of the proposed method is discussed in Sec. V

II. Online Compensation

When precalculated optimal control trajectories are applied in practice there usually occur some deviations of stability criteria or constraints due to modeling errors, link flexibilities, gear loss, backlash, joint control errors and external disturbances inflicted on the robot by the environment. These result in a degraded walking performance. Commonly, a sensor-based control strategy has to be applied to cope with such deviations.

In this section a novel method termed *Jacobi Compensation* is proposed, which modifies precalculated trajectories in selected task coordinate directions in order to reduce stability criteria deviation and thereby improve walking performance. Task coordinates can be selected Cartesian directions of e.g. the hip coordinate or others such as the CoM.

The goal of the method is to move a specific set of coordinates $\boldsymbol{x}_c \in \mathbb{R}^m$ of points on the humanoid – e.g. the center of the hip, an ankle, or the CoM – by $\Delta \boldsymbol{x_c}$ in Cartesian space. The joint angles $\boldsymbol{q}_t$, obtained from a precalculated trajectory, are modified by $\Delta \boldsymbol{q} = \boldsymbol{f}(\Delta \boldsymbol{x_c})$, where $\boldsymbol{f}(\cdot)$ transforms the Cartesian motion $\Delta \boldsymbol{x_c}$ into a joint space motion $\Delta \boldsymbol{q}$. As shown in Fig. 1, this correction $\Delta \boldsymbol{q}$ is linearly superimposed with the joint configuration $\boldsymbol{q}_t$ resulting in a new desired posture $\boldsymbol{q}_d = \boldsymbol{q}_t + \Delta \boldsymbol{q}$ of the robot.

The differential relationship $\boldsymbol{f}(\cdot)$ between Cartesian task coordinate motion and joint space motion is described by the Jacobian

$$\boldsymbol{J}(\boldsymbol{q}_a) = \left[\frac{\partial \boldsymbol{x}_c}{\partial q_1} \quad \cdots \quad \frac{\partial \boldsymbol{x}_c}{\partial q_n} \right] \Bigg|_{\boldsymbol{q}=\boldsymbol{q}_a} \in \mathbb{R}^{m \times n},$$

a function of the actual joint angles $\boldsymbol{q}_a \in \mathbb{R}^n$ which maps the velocity $\dot{\boldsymbol{q}}_c$ in joint space to the velocity $\dot{\boldsymbol{x}}_c \in \mathbb{R}^m$ in

[0]Work performed while author was with Control Systems Group, Technical University Berlin

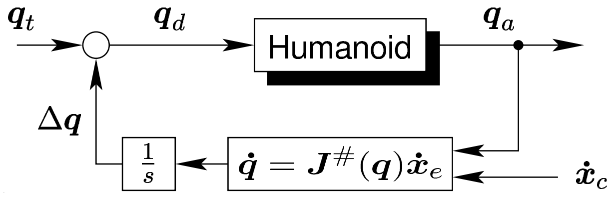

Fig. 1. Jacobi Compensation. The precalculated joint trajectory $\boldsymbol{q}_t$ is modified by addition of an online and task-dependant calculated correction term $\Delta\boldsymbol{q}$. The necessary correction $\dot{\boldsymbol{x}}_c$ is determined in Cartesian space and transformed into joint space via the Jacobian.

Cartesian space according to

$$\dot{\boldsymbol{x}}_c = \boldsymbol{J}(\boldsymbol{q}_a)\,\dot{\boldsymbol{q}}_c. \tag{1}$$

To invert (1) the pseudoinverse $\boldsymbol{J}^{\#}(\boldsymbol{q}_a) := \boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{J}^T)^{-1}$ which minimizes the Euclidian norm $||\dot{\boldsymbol{q}}_c||_2$ is used; here, $m \leq n$ and $\mathrm{rk}\,(\boldsymbol{J}(\boldsymbol{q}_a)) = m$ is assumed for existence of a solution, i.e. the motion is modified along less or equal task coordinates $\boldsymbol{x}_c$ than degrees-of-freedom n of the system. The correction velocity computes to

$$\dot{\boldsymbol{q}}_c = \boldsymbol{J}^{\#}(\boldsymbol{q}_a)\,\dot{\boldsymbol{x}}_c\ , \tag{2}$$

which is integrated to obtain the position modification $\Delta\boldsymbol{q}$ in joint space. Superimposing it with the precalculated trajectory $\boldsymbol{q}_t$ adapts the trajectory to the actual requirements, cf. Fig. 1.

Following [15] it is also possible to account for motion constraints. For each constraint, a constraint Jacobian $\boldsymbol{J}_c$ must be computed and the equation $\boldsymbol{J}_c\dot{\boldsymbol{q}}_c = \boldsymbol{c}$ must be met. Hence the concatenated Jacobian $\boldsymbol{J} = \begin{bmatrix}\boldsymbol{J}_t & \boldsymbol{J}_c\end{bmatrix}^T$ consists of a task Jacobian $\boldsymbol{J}_t$ and a constraint Jacobian $\boldsymbol{J}_c$, i.e. the mapping from joint- to Cartesian space is

$$\begin{bmatrix}\boldsymbol{J}_t\\ \boldsymbol{J}_c\end{bmatrix}\dot{\boldsymbol{q}} = \begin{bmatrix}\dot{\boldsymbol{x}}_c\\ \boldsymbol{c}\end{bmatrix}.$$

Note that the zero moment point (ZMP) is not only a function of joint position and velocities, but also of their accelerations. Therefore a it is impossible to compute a Jacobian for controlling the ZMP directly. Hence, if the Jacobian is required to accommodate for constraints on the ZMP, one solution is to set up a Jacobian manipulating the center of gravity (COG) and control the COG such that it moves along referential COG velocity ${}^{ref}\dot{\boldsymbol{x}}_G$, see [15] for details.

Depending on the control problem there exist a variety of possibilities to compute the correction velocity $\dot{\boldsymbol{x}}_c$. For the experiments described in the following sections, the velocity is chosen as $\dot{\boldsymbol{x}}_c = \boldsymbol{\Gamma}_P\,\Delta\boldsymbol{x}_c + \boldsymbol{\Gamma}_D\Delta\dot{\boldsymbol{x}}_c$, with $\boldsymbol{\Gamma}_P > 0$ and $\boldsymbol{\Gamma}_D > 0$, both being constant.

Applications of this method are plentiful: The method was developed to alter the posture of the robot and thus modify precalculated trajectories to improve walking stability and performance. Other applications include the possibility to adapt precalculated gait trajectories to fit for walking on slopes. If the robot walks up a slope, the center of mass can be slightly shifted to the front by Jacobi Compensation. Apart from that modification the trajectories can remain unchanged, hence less trajectories in a step data base need to be precalculated in advance.

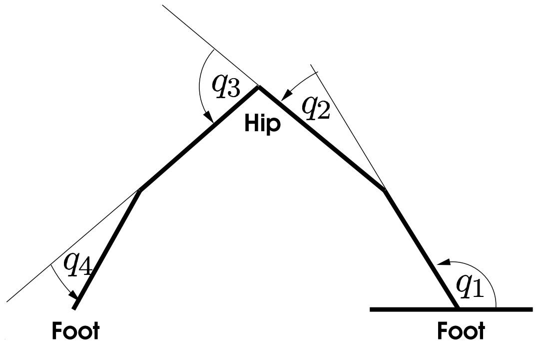

Fig. 2. Scara 4 DoF. Initial Configuration for Simulation.

III. Simulation Results

The Jacobi Compensation described in Sec. II is now demonstrated in a simulation experiment using a mechanically simple, but illustrative example of a planar 4 DoF Scara robot. The Scara robot is a very simple planar approximation of a complete three-dimensional biped robot system, considering only the robot lateral motion. Points on the Scara robot are labeled "hip" and "foot" as sketched in Fig. 2, mimicking the hip and foot positions of the approximated biped robot. Several tasks, that comprise our existing problems in biped stable walking are solved by Jacobi Compensation in simulation. Among those are e.g. "move hip over the supported area" to avoid instability and tilting or "move swing leg outwards" to avoid collisions of the feet.

The dynamics of the Scara robot are described by a differential equation $\dot{\boldsymbol{\xi}} = \boldsymbol{g}(\boldsymbol{\xi}, \boldsymbol{\tau})$ where the motor torques $\boldsymbol{\tau}$ are obtained from a PD position control loop

$$\boldsymbol{\tau} = \boldsymbol{K}_P(\boldsymbol{q}_d - \boldsymbol{q}) + \boldsymbol{K}_D(\dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}}),$$

with $\boldsymbol{K}_P, \boldsymbol{K}_D = \text{const} > 0$, $\boldsymbol{\xi} = (\boldsymbol{q}\ ,\ \dot{\boldsymbol{q}})^T$. The joint angle $\boldsymbol{q}_d(t)$ is the desired trajectory in joint space. The link masses $m = 1$ and the link lengths $l = 1$ are equal for all links. Friction is neglected.

Jacobi Compensation is now used to move the swing foot outwards, given a desired foot position $\boldsymbol{x}_{d,\text{foot}}$. The initial configuration $\boldsymbol{q}_0$ is sketched in Fig. 2 and the reference trajectory $\boldsymbol{q}_t = \boldsymbol{q}_0 = \text{constant}$, $\dot{\boldsymbol{q}}_t = 0$ is assumed, i.e. the robot does not move. Then $\Delta\boldsymbol{x}_{c,\text{foot}}$ is computed as the difference between $\boldsymbol{x}_{d,\text{foot}}$ and the actual Cartesian position $\boldsymbol{x}_{a,\text{foot}}$ of the foot. In this example the Jacobian is

$$\boldsymbol{J}_{\text{foot}} = \begin{bmatrix}\partial\boldsymbol{x}_{c,\text{foot}}/\partial q_1 & \dots & \partial\boldsymbol{x}_{c,\text{foot}}/\partial q_4\end{bmatrix}.$$

The algorithm proceeds as described in Sec. II and the simulation result, concerning initial and calculated final posture, is presented in Fig. 3. The performance of the method for "outward movement of the foot" is satisfying, i.e. settling is fast and stable and the desired position is reached.

Solving the task of moving the hip rightwards in analogy performs well, but an additional practical problem emerges.

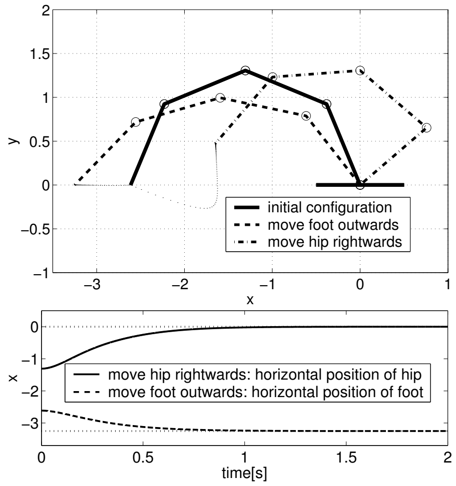

Fig. 3. Scara 4 DoF. Initial and final configuration for "move foot outwards" (dashed plot) and for "move hip rightwards" (dash-dotted plot) and associated hip/foot trajectories over time.

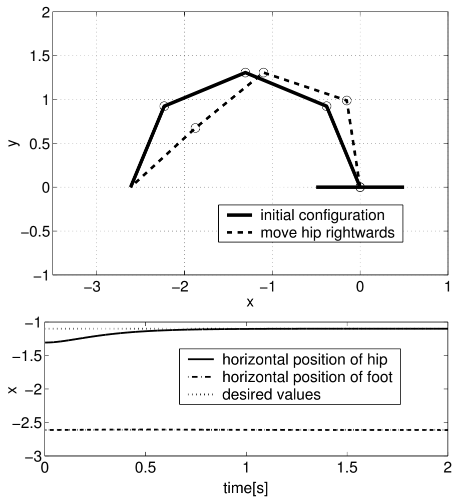

Fig. 4. Scara 4 DoF. Initial and final configuration for "move hip but keep in rest foot" and associated hip/foot trajectories over time.

As it is seen in Fig. 3, the desired change in hip position is involved with an unintentional change in foot position. One possible solution for fixing the foot in Cartesian position is using the concatenated Jacobian

$$\boldsymbol{J}_{\text{foot,hip}} = \begin{bmatrix} \partial \boldsymbol{x}_{c,\text{foot}}/\partial q_1 & \cdots & \partial \boldsymbol{x}_{c,\text{foot}}/\partial q_4 \\ \partial \boldsymbol{x}_{c,\text{hip}}/\partial q_1 & \cdots & \partial \boldsymbol{x}_{c,\text{hip}}/\partial q_4 \end{bmatrix}.$$

There is a geometrical conflict between the two goals hip movement over supported area and foot in rest, e.g. it is never possible to reach both goals at the same time. To avoid singularity of $\boldsymbol{J}_{\text{foot,hip}}$ care has to be taken in choosing the desired Cartesian positions of hip and foot, involving kinematic considerations. The result of simulation is depicted in Fig. 4: the resulting movement of the hip in Cartesian x-direction has to be chosen smaller than before, therefore a nearby constant foot position is maintained now.

These promising results encourage questions about stability properties in closed loop control.

IV. Stability Analysis

In this section the stability of a Jacobi-compensated system will be investigated. For a first analysis, Sec. IV-A gives a qualitative examination of the system dynamics to understand their properties. The stability of the system is then proved based on Lyapunov's theorem in Sec. IV-B.

A. *Qualitative Analysis of System Dynamics*

The analysis shown here refer to a real hardware robot with 17 degrees of freedom, see also [5], [6].

As can be seen from Fig. 5, the Jacobi Compensation closes a feedback loop with the pseudoinverse Jacobian $\boldsymbol{J}^{\#}(\boldsymbol{q}_a)$ being a nonlinear function of the actual joint configuration $\boldsymbol{q}_a$. Without loss of generality $\boldsymbol{J}^{\#}(\boldsymbol{q}_a) \in \mathbb{R}^{n\times m}$, $n > m$, is assumed to have full rank m – otherwise the robot is in a singular configuration which has to be avoided.

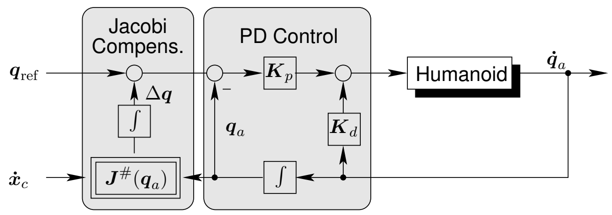

Fig. 5. Control scheme.

For understanding the system dynamics, the Singular Value Decomposition (SVD)

$$\boldsymbol{J}^{\#}(\boldsymbol{q}_{\text{traj}}(t)) = \boldsymbol{U}(t)\boldsymbol{\Sigma}(t)\boldsymbol{V}^*(t) \tag{3}$$

of the pseudoinverse Jacobian $\boldsymbol{J}^{\#}$ along a given trajectory $\boldsymbol{q}_{\text{traj}}(t)$ is investigated. As $\boldsymbol{J}^{\#} \in \mathbb{R}^{n\times m}$, $n > m$, is a non-square matrix, and Σ is required to have the same size as $\boldsymbol{J}^{\#}$, the matrix $\boldsymbol{\Sigma} = \begin{bmatrix}\text{diag}(\sigma_i) & \mathbf{0}\end{bmatrix}^T$ contains the singular values σ_i sorted decreasingly in diagonal form ($\mathbb{R}^{m\times m}$) and is supplemented by a zero-matrix $\mathbf{0} \in \mathbb{R}^{(n-m)\times m}$. The matrices $\boldsymbol{U} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{V} \in \mathbb{R}^{m\times m}$ are orthonormal bases.

By writing (3) element wise as

$$\boldsymbol{J}^{\#}\boldsymbol{v}_i = \sigma_i \boldsymbol{u}_i, \tag{4}$$

where $\boldsymbol{v}_i$ and $\boldsymbol{u}_i$ represent the i-th column vector of $\boldsymbol{V}$ and $\boldsymbol{U}$ respectively, one can see, that the singular value σ_i can be interpreted as the gain of the Jacobian in coordinate direction $\boldsymbol{u}_i$ [16]. As the σ_i vary with the actual working point, the Jacobian can be badly conditioned for certain joint configurations, i.e. have rather different gains in different directions of motion. That means that small changes on the correction term $\dot{\boldsymbol{x}}_c$ lead to large motions in joint space.

Apart from the singular values $\boldsymbol{\Sigma}$ also the principal transformation axes $\boldsymbol{U}$ rotate depending on $\boldsymbol{q}_a$. Therefore the Jacobi output $\dot{\boldsymbol{q}}_c$ can vary even with a constant correction term $\dot{\boldsymbol{x}}_c$ as the joint configuration follows a trajectory.

This is illustrated in Fig. 6, where the minimum singular value σ_m and maximum singular value σ_1 of $\boldsymbol{J}^{\#}(\boldsymbol{q}_{\text{traj}}(t))$

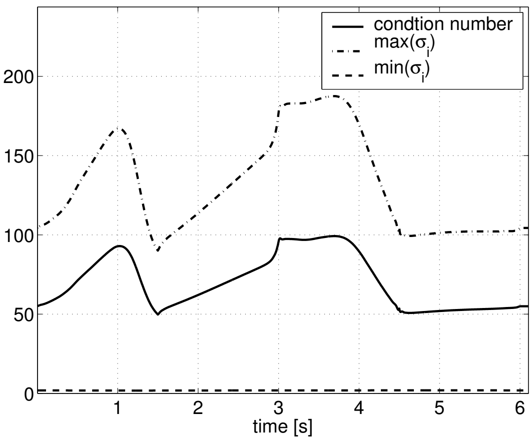

Fig. 6. Minimum and maximum singular value and the corresponding condition number along a trajectory.

along a given walking trajectory $\boldsymbol{q}_{\text{traj}}(t)$ are shown. Furthermore the condition number (see [17])

$$\kappa = \frac{\max(\sigma_i)}{\min(\sigma_i)} = \frac{\sigma_1}{\sigma_m} \tag{5}$$

of $\boldsymbol{J}^{\#}$ at each time step is shown. This condition number κ is commonly used to rate the dexterity in a workspace and should be $\kappa = 1$ for a workspace with equal manipulability in all directions.

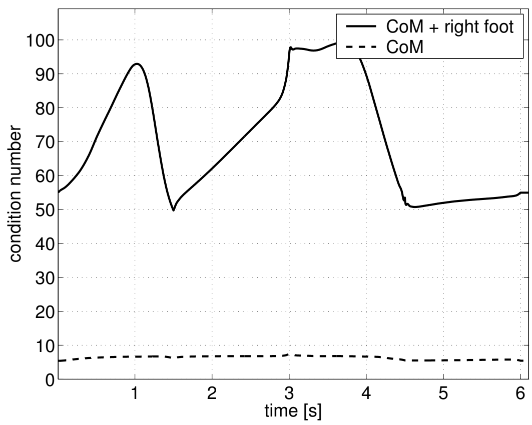

Fig. 7. Condition numbers for Jacobians modifying the CoM and both, CoM and position of the right foot.

The condition number also depends highly on the dimension of the Jacobian. Fig. 7 shows the condition number for a Jacobian adjusting the attitude of the center of mass (CoM) (dashed line) and that of another Jacobian manipulating both, the CoM and the position of the swing foot (solid line).

In order to obtain a measure for the changes in the Jacobians at two consecutive sample times, $\boldsymbol{J}^{\#}$ has been computed along the trajectory $\boldsymbol{q}_{\text{traj}}(t)$ at the constant sample time 4 ms. The changes of the Jacobians at time t^* with respect to the previous sample are expressed in the difference matrix

$$\Delta \boldsymbol{J}(t^*) = \boldsymbol{J}^{\#}(\boldsymbol{q}_{\text{traj}}(t^*)) - \boldsymbol{J}^{\#}(\boldsymbol{q}_{\text{traj}}(t^* - T)),$$

where $T = 4\,\text{ms}$ is the sampling time. The difference measure

$$\Delta(t^*) = \sum_{i=1}^{m} \sum_{j=1}^{n} |\Delta J_{i,j}(t^*)| \tag{6}$$

is the sum of the absolute value of all elements of the difference matrix between two sample times. Fig. 8 shows the evolution of Δ along the trajectory $\boldsymbol{q}_{\text{traj}}(t)$. Comparing this plot with the condition number of the CoM-Jacobian in Fig. 7 confirms the fact, that a badly conditioned Jacobian $\boldsymbol{J}^{\#}$ yields high velocities in joint space.

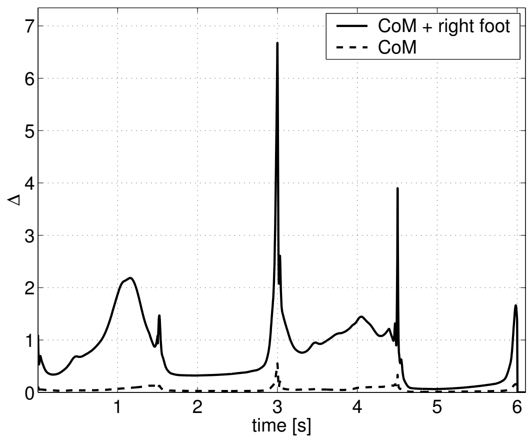

Fig. 8. Evolution of the difference measure Δ defined in (6).

Therefore investigations on the stability of Jacobi compensation are of interest.

B. Lyapunov Stability

The dynamic behavior of the humanoid robot follows the differential equation

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{B}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{k}(\boldsymbol{q}) = \boldsymbol{\tau}, \tag{7}$$

where $\boldsymbol{\tau}$ is the torque applied to the robot.

This stability analysis is based on Lyapunov's theorem, stating that a system is (locally) stable, if a (locally) positive definite function (Lyapunov function) $V(\boldsymbol{x}, t)$ can be found with a time derivative $\dot{V}(\boldsymbol{x}, t) < 0$, $\forall t > 0$ [18].

As suggested in [19], a candidate Lyapunov function

$$V = \frac{1}{2}\dot{\boldsymbol{q}}^T \boldsymbol{M} \dot{\boldsymbol{q}} + \frac{1}{2}\boldsymbol{q}^T \boldsymbol{K}_p \boldsymbol{q} \tag{8}$$

is chosen. As the matrix $\boldsymbol{M}$ and proportional gain $\boldsymbol{K}_p$ of the control law are positive definite, (8) is a valid Lyapunov function where the first term represents the kinetic energy and the second term is a generalized potential energy stored in the system.

The time derivative

$$\dot{V} = \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\dot{\boldsymbol{q}}^T \boldsymbol{M} \dot{\boldsymbol{q}} + \dot{\boldsymbol{q}}^T \boldsymbol{K}_p \boldsymbol{q} \tag{9}$$

of the Lyapunov function contains in its first term the rate of change of kinetic energy which – according to mechanics – is equal to the power provided by external forces. Hence (9) can be written as

$$\dot{V} = \dot{\boldsymbol{q}}^T(\boldsymbol{\tau} - \boldsymbol{g}) + \dot{\boldsymbol{q}}^T \boldsymbol{K}_P \boldsymbol{q}, \tag{10}$$

$\boldsymbol{g}$ is the acceleration of gravity. The torque $\boldsymbol{\tau}$ is obtained from the control law

$$\boldsymbol{\tau} = -\boldsymbol{K}_p \boldsymbol{q}_e - \boldsymbol{K}_d \dot{\boldsymbol{q}}_a + \boldsymbol{g} \tag{11}$$

with

$$\boldsymbol{q}_e = \boldsymbol{q}_{\text{ref}} - \boldsymbol{q}_a + \int \dot{\boldsymbol{q}}_c \, \mathrm{d}t.$$

Inserting (11) into (10) yields

$$\dot{V} = (\dot{\boldsymbol{q}}_{\text{ref}} - 2\dot{\boldsymbol{q}}_a + \dot{\boldsymbol{q}}_c)^T \boldsymbol{K}_p \boldsymbol{q}_e - \dot{\boldsymbol{q}}_a^T \boldsymbol{K}_d \dot{\boldsymbol{q}}_a \tag{12}$$

The first term in (12) can be interpreted as the power necessary for the motion, while the second term is the energy dissipated by the damping gain $\boldsymbol{K}_d$ in the control law.

As Lyapunov's theorem requires the time derivative $\dot{V}$ of the Lyapunov function to be negative, the Jacobi Compensation to be stable, if the dissipated energy is larger than the energy generated by the motion.

As $\boldsymbol{K}_p$ and $\boldsymbol{K}_d$ are the parameters of a PD control loop these matrices are often chosen in diagonal form thus describing separate PD controllers for each robot joint without coupling. With this assumption it is possible to write (12) element wise, where $q_{\text{ref}i}$, q_{ci}, q_{ai} signify the i-th element of the vector $\boldsymbol{q}_{\text{ref}i}$, $\boldsymbol{q}_{ci}$, $\boldsymbol{q}_a$ and k_{pi} and k_{di} represent the i-th diagonal element of $\boldsymbol{K}_p$ or $\boldsymbol{K}_d$ resp., i.e. the element in the i-th row and the i-th column. Hence, using the abbreviation $\Delta \boldsymbol{q} = \int \boldsymbol{J}^{\#}(\boldsymbol{q}_a)\dot{\boldsymbol{x}}_c \, \mathrm{d}t$, (12) is expressed as

$$\dot{V}_i = (\dot{q}_{\text{ref}i} - 2\dot{q}_{ai} + \dot{q}_{ci}) k_{pi} q_{ei} + \dot{q}_{ai} k_{di} \dot{q}_{ai} < 0 \tag{13}$$

The scalar inequality (13) can be solved for

$$\begin{cases} \dot{q}_{ci} & < \frac{k_{di}\dot{q}_{ai}^2}{k_{pi}q_{ei}} - \dot{q}_{\text{ref}i} + 2\dot{q}_{ai} \quad \forall \dot{q}_{ei} > 0 \\ \dot{q}_{ci} & > \frac{k_{di}\dot{q}_{ai}^2}{k_{pi}q_{ei}} - \dot{q}_{\text{ref}i} + 2\dot{q}_{ai} \quad \forall \dot{q}_{ei} < 0. \end{cases} \tag{14}$$

The inequalities (14) represent a boundary for velocity $\dot{\boldsymbol{q}}_c$ of Jacobi compensation. Thus the control loop remains stable, if the posture of the robot is altered sufficiently slowly[1].

This result is in accordance with experimental experiences [11], where Jacobi Control generally showed stable behavior for smooth posture modifications with few constraints. For large Jacobians incorporating many constraints, the pseudoinverse may become badly conditioned, resulting in high joint velocities $\dot{\boldsymbol{q}}_c$. Hence large Jacobians often destabilize the system in certain joint configurations.

A solution to handle this problem of singularities is suggested in the following.

V. Singularity-Robust Inverse

When computing (2), the Moore-Penrose pseudoinverse $\boldsymbol{J}^{\#} = \boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{J}^T)^{-1}$ offers a least-squares solution for $\dot{\boldsymbol{q}}_c$ that fulfills $\min ||\dot{\boldsymbol{x}}_c - \boldsymbol{J}\dot{\boldsymbol{q}}_c||$. As this solution generally is not unique, $\dot{\boldsymbol{q}}_c$ is chosen such that $\min ||\dot{\boldsymbol{q}}_c||$ is satisfied, see [16]. Thus the pseudoinverse minimizes the error rather than considering the feasibility of the motion resulting in badly conditioned pseudoinverses and thus high velocities as observed in Figs. 7 and 8.

To overcome this deficiency, Nakamura [16] proposed to minimize both properties simultaneously by solving

$$\min \left\| \begin{pmatrix} \dot{\boldsymbol{x}}_c - \boldsymbol{J}\dot{\boldsymbol{q}}_c \\ \dot{\boldsymbol{q}}_c \end{pmatrix} \right\|_{\boldsymbol{W}}, \tag{15}$$

where $\boldsymbol{W}$ is a weighting matrix of the norm $\|\boldsymbol{x}\|_{\boldsymbol{W}} = \boldsymbol{x}^T \boldsymbol{W} \boldsymbol{x}$. Minimizing the error and the velocity simultaneously avoids velocity peeks as found by the hierarchical optimization structure of the pseudoinverse solution. Choosing $\boldsymbol{W}$ as the identity matrix $\boldsymbol{I}$, the solution for (15) is found by the *Singualarity Robust (SR-) Inverse*

$$\boldsymbol{J}^*(\boldsymbol{q}_a) = \boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{J}^T + k\boldsymbol{I})^{-1}, \tag{16}$$

see [16] for a detailed derivation. The scalar k is a weighting factor that influences the trade-off between exactness and feasibility of a motion. For $k \to 0$ the SR-Inverse $\boldsymbol{J}^*$ becomes identical to the pseudoinverse $\boldsymbol{J}^{\#}$, while larger k increase the weight of the velocity in the optimization (15) and hence produce inverses that are less sensitive to singularities.

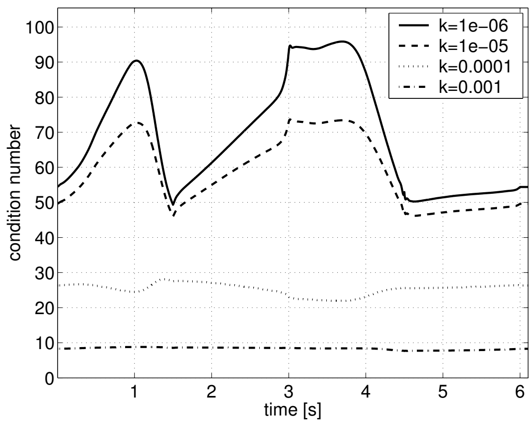

Fig. 9. Condition numbers of SR-Inverses for different weighting factors k.

Fig. 9 shows the influence of the weighting factor k on the condition number of the SR-Inverse along the same trajectory as in the previous experiments: for larger k the negative influence of singularities on the conditioning of the matrix is efficiently reduced.

On the other hand increasing k also leads to a larger deviation of the actual motion $\hat{\dot{\boldsymbol{q}}}_c = \boldsymbol{J}^*(\boldsymbol{q}_a)\,\dot{\boldsymbol{x}}_c$ of the endeffector from the desired motion $\dot{\boldsymbol{q}}_c = \boldsymbol{J}^{\#}(\boldsymbol{q}_a)\,\dot{\boldsymbol{x}}_c$. To ensure small deviation of $\hat{\dot{\boldsymbol{q}}}_c$ from the desired path $\dot{\boldsymbol{q}}_c$, $\boldsymbol{J}\boldsymbol{J}^* \approx \boldsymbol{I}$ should hold. This property is evaluated in the error measure

$$\Gamma = \sum_{i=1}^{m} \sum_{j=1}^{m} |(\boldsymbol{J}\boldsymbol{J}^* - \boldsymbol{I})_{i,j}|, \tag{17}$$

which is the sum of all elements of the matrix $\boldsymbol{J}\boldsymbol{J}^* - \boldsymbol{I} \in \mathbb{R}^{m \times m}$. This error measure Γ should be small to ensure low divergence from the desired motion. The

[1]Unfortunately it is not possible to conclude from this result about general stability of Jacobi Compensation if condition (12) is not met, as Lyapunov's theorem cannot make negative statements on stability.

evolution of Γ for various k along the trajectory is shown in Fig. 10. Comparing Fig. 9 and Fig. 10 shows that by selecting the weighting factor k a tradeoff between accuracy and stability of the control loop is being made; these experiments suggest $k = 0.0001$ to be a reasonable tradeoff.

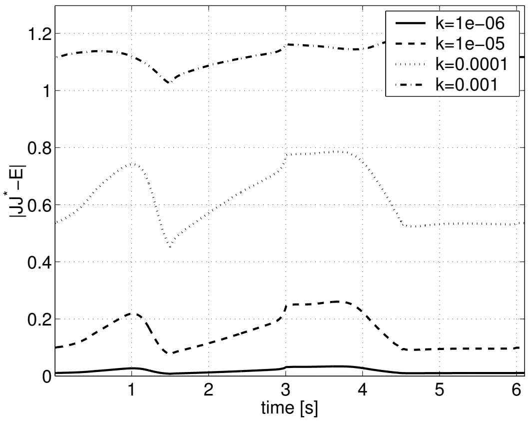

Fig. 10. Error measure Γ defined in (17) along a gait trajectory.

VI. Conclusions

The novel method of *Jacobi Compensation* has been presented which allows modification of precalculated trajectories online during the robot gait. It is possible to shift parts of the robot in given Cartesian directions of selected task coordinates thereby altering the posture of the humanoid to improve e.g. stability. This method can be used to modify precalculated gait trajectories in order to compensate for errors or adapt the trajectories in order to make them applicable to situations other than those they have been computed for.

Dynamics properties of the closed feedback loop containing the pseudoinverse Jacobian have been investigated. Using Lyapunov's theorem it has been shown that this system is stable for moderate posture modifications. To prevent the Jacobi Compensation feedback loop from generating high velocities in the vicinity of singularities, a Singularity-Robust Inverse has been suggested as replacement for the Moore-Penrose pseudoinverse.

Acknowledgment

The authors would like to thank Mrs. Marion Sobotka for her helpful contributions to this work. The authors appreciate the valuable comments by the anonymous reviewers.

References

[1] S. Kajita, T. Yamaura, and A. Kobayashi, "Dynamic walking control of a biped robot along a potential energy conserving orbit," *IEEE Transactions on Robotics and Automation*, vol. 8, pp. 431–438, August 1992. potential energy conserving orbit.

[2] K. Tani, K. Ikeda, T. Yano, S. Kajita, and O. Matsumoto, "The concept of model free robotics for robots to act in uncertain environments," in *Proceedings of the IEEE/Tsukuba International Workshop on Advanced Robotics*, (Tsukuba, Japan), pp. 85–90, November 1993. do not use models as models are specific.

[3] M. Inaba, F. Kanehiro, S. Kagami, and H. Inoue, "Two-armed bipedal robot that can walk, roll over and stand up," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, pp. 297–302, 1995.

[4] F. Yamasaki, K. Endo, M. Asada, and H. Kitano, "A control method for humanoid biped walking with limited torque," in *RoboCup 2001* (A. Birk, S. Coradeschi, and S. Tadokoro, eds.), pp. 60–70, Berlin Heidelberg: Springer Verlag, 2002.

[5] D. Wollherr, M. Hardt, M. Buss, and O. von Stryk, "Actuator selection and hardware realization of a small and fast-moving, autonomous humanoid robot," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, (Lausanne, Switzerland), pp. 2491–2496, 2002.

[6] M. Hardt, D. Wollherr, M. Buss, and O. von Stryk, "Design of an autonomous fast-walking humanoid robot," in *Proceedings of the 5th International Conference on Climbing and Walking Robots*, pp. 391–398, 2002.

[7] J. Denk and G. Schmidt, "Synthesis of a walking primitive database for a humanoid robot using optimal control techniques," in *Proceedings of the IEEE/RAS International Conference on Humanoid Robots*, (Tokyo, Japan), pp. 319–326, 2001.

[8] J.-M. Bourgeot, N. Cislo, and B. Espiau, "Path-planning and tracking in a 3D comlex environment for an anthropomorphic biped robot," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, (Lausanne, Switzerland), pp. 2509–2514, October 2002.

[9] O. Lorch, A. Albert, J. Denk, M. Gerecke, R. Cupec, J. F. Seara, W. Gerth, and G. Schmidt, "Experiments in vision-guided biped walking," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, (Lausanne, Switzerland), pp. 2484–2490, 2002.

[10] Q. Huang, S. Kajita, N. Koyachi, K. Kaneko, K. Yokoi, T. Kotoku, H. Arai, K. Komoriya, and K. Tanie, "Walking pattern and actuator specification for a biped robot," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, (Kyongju, Korea), pp. 1462–1468, 1999.

[11] M. Sobotka, D. Wollherr, and M. Buss, "A jacobian method for online modification of precalculated gait trajectories," in *Proceedings of the 6th International Conference on Climbing and Walking Robots*, (Catania, Italy), pp. 435–442, 2003.

[12] Q. Huang, K. Kaneko, K. Yokoi, S. Kajita, T. Kotoku, N. Koyachi, H. Arai, N. Imamura, K. Komoriya, and K. Tanie, "Balance control of a biped robot combining off-line pattern with real-time modification," in *Proceedings of the IEEE International Conference on Robotics and Automation*, (San Francisco, CA), pp. 3346–3352, April 2000. online adaption of precalculated trajectory.

[13] Y. F. Zheng, J. Shen, and F. R. S. Jr., "A motion control scheme for a biped robot to climb sloping surfaces," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 814–816, 1988.

[14] O. Brock, O. Khatib, and S. Viji, "Task-consistent obstacle avoidance and motion behavior for mobile manipulation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, (Washington D.C.), pp. 388–393, May 2002.

[15] T. Sugihara, Y. Nakamura, and H. Inoue, "Realtime humanoid motion generation through zmp manipulation based on inverted pendulum control," in *Proceedings of the IEEE International Conference on Robotics and Automation*, (Washington, DC), pp. 1404–1409, 2002.

[16] Y. Nakamura, *Advanced Robotics – Redundancy and Optimization*. Reading, Massachusetts: Addison Wesley, 1991.

[17] J. K. Salisbury and J. T. Craig, "Articulated hands: Force control and kinematic issues," *International Journal of Robotics Research*, vol. 1, no. 1, pp. 4–17, 1982.

[18] S. Sastry, *Nonlinear Systems – Analysis, Stability, and Control*. New York: Springer-Verlag, 1999.

[19] J.-J. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, New Jersey: Prentice Hall, 1991.