Multi-Contact Whole Body Force Control for Position-Controlled Robots

Quentin Rouxel, Serena Ivaldi, and Jean-Baptiste Mouret

***Abstract*—Many humanoid and multi-legged robots are controlled in positions rather than in torques, preventing direct control of contact forces, and hampering their ability to create multiple contacts to enhance their balance, such as placing a hand on a wall or a handrail. This paper introduces the SEIKO (Sequential Equilibrium Inverse Kinematic Optimization) pipeline, drawing inspiration from flexibility models used in serial elastic actuators to indirectly control contact forces on traditional position-controlled robots. SEIKO formulates whole-body retargeting from Cartesian commands and admittance control using two quadratic programs solved in real time. We validated our pipeline with experiments on the real, full-scale humanoid robot Talos in various multi-contact scenarios, including pushing tasks, far-reaching tasks, stair climbing, and stepping on sloped surfaces. This work opens the possibility of stable, contact-rich behaviors while getting around many of the challenges of torque-controlled robots. Code and videos are available at https://hucebot.github.io/seiko_controller_website/.**

***Index Terms*—Whole Body Admittance Control, Multi-Contact, Teleoperation, Joint Flexibility, Humanoid Robot.**

I. Introduction

Humans often use additional contact points to enhance their stability, for instance, using a handrail or a wall when walking, or to extend their reach, for instance, to grasp an object that is too far forward. While humanoid robots would benefit from a similar strategy, current robots minimize the number of contacts and use them only for feet and required interactions with the environment, such as pushing a button [1].

The primary challenge in controlling multi-contact lies in the redundancy of force distribution resulting from closed kinematic chains [2]. For a given posture with several contacts, there exists an infinity of ways to distribute force among them. For instance, a humanoid with both hands on a table can apply more or less force to the hands without any visible change in joint position.

To explicitly regulate forces, most work on multi-contact whole-body control relies on torque-controlled robots with inverse dynamics controllers [3]–[5]. Unfortunately, inverse dynamics is highly sensitive to model and calibration errors, and identifying models for humanoids is particularly challenging [6]. Perfect identification of environment's properties is generally not possible. This is why most deployed robots use position control, which is simpler and more reliable [7], but lacks direct control authority over contact force, thus hindering the exploitation of multi-contact strategies.

In this paper, we present a control pipeline (Fig. 1) designed to regulate contact forces using position-controlled robots.

The authors are with Inria, CNRS, Université de Lorraine, France.
This research is supported by the EU Horizon project euROBIN (101070596).

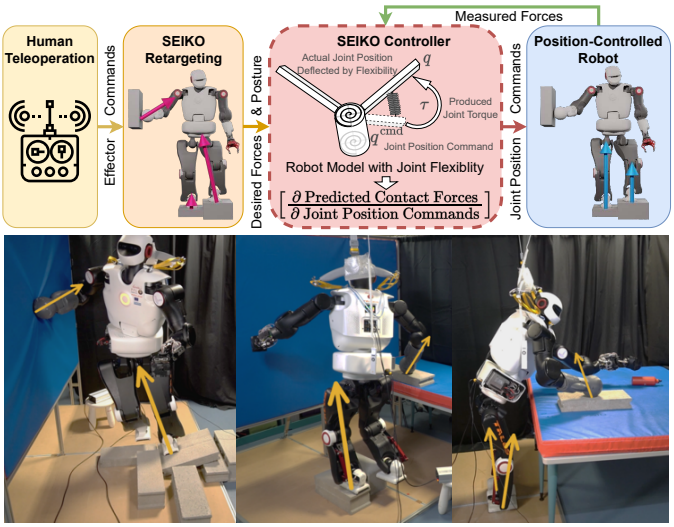

Figure 1. Overview of our control pipeline (top), and illustrations of teleoperated multi-contact experiments on Talos humanoid robot (bottom).

Drawing inspiration from series elastic actuators [8], our main idea is to leverage the "flexibility" that stems from mechanical parts that bend with high load but also from the non-perfect joint position tracking with PID controllers, which act as "elastic elements". While it is not possible to compute the contact force from the posture alone [2], the flexibility will make the forces converge to a unique equilibrium. We model this phenomenon to link position commands to force distribution. To invert this relation, that links force distributions to joint positions, we formulate it as a Quadratic Programming (QP) problem to leverage fast QP solvers.

We conducted experiments on the Talos humanoid robot [9], equipped with powerful arms but known for significant hip mechanical flexibility [10]. Our control pipeline is compatible with commands from autonomous planners and teleoperation, with a focus on the latter in this study. Well-suited for teleoperation, our method is robustness against operator errors related to awareness and embodiment challenges. Unlike most existing methods, our approach enables motions close to feasibility boundaries (both in term of kinematic, balance and torque limits), allowing to fully exploit the capabilities of the hardware.

Our work named SEIKO for Sequential Equilibrium Inverse Kinematic Optimization provides the following contributions:

- An SQP formulation that computes posture deflection and joint command correction, accounting for joint flexibility in multi-contact quasi-static conditions.
- A multi-contact retargeting and control architecture for

position-controlled robots with contact switch and pushing capabilities, designed to be robust against model errors.
- Validation on the hardware Talos humanoid robot with several multi-contact tasks, including the validation of our prior retargeting work, which was previously tested only in simulation for humanoid robots.

II. Related Work

Our previous works [11]–[13] explored teleoperation and retargeting for feasible multi-contact tasks on simulated humanoids and hardware bimanual manipulators. While torque-controlled robots and inverse dynamic controllers were used for regulating contact wrenches, this new work introduces a whole-body admittance controller to address position-controlled robots, enhancing robustness to model errors.

Many works have studied teleoperation of complex robots with floating bases [14], but few have explicitly addressed multi-contact scenarios. [15] demonstrated multi-contact teleoperation with operator whole-body tracking on HRP-4, including retargeting and position control. [16] introduced a dedicated human-robot interface for multi-contact teleoperation on Valkyrie, employing joint impedance control. However, both approaches lack explicit regulation of contact forces.

On torque-controlled robots, multi-contact tasks have been explored both in simulation [3] and with real humanoids [4], [5]. Whole-body inverse dynamic controllers are used for direct regulation of contact wrenches and internal forces.

Joint impedance control provides a robust alternative to pure torque control, accommodating model errors while enabling force commands. [17], [18] showcased multi-contact setups on CENTAURO and COMAN+ robots, employing a QP-based whole-body inverse kinematics for postural control. A second QP calculates contact force references from quasi-static assumption, integrating these references as torque feedforward term in the low-level joint impedance scheme.

On position-controlled robots, contact force regulation is frequently ignored. [19] demonstrated balance stabilization through feedback laws applied prior to inverse kinematics, and [20] showcased ladder climbing on HRP-2 using an inverse dynamic controller optimizing joint accelerations, integrated twice to obtain position commands. While purely kinematic control is effective when the robot is far from feasibility limits, it inherently lacks control over the robot's full configuration.

Foot force difference control from [21] influenced a common approach for regulating contact forces: applying an admittance scheme on the effector's Cartesian position normal to the contact surface [22]–[25]. This scheme implicitly relies on joint impedance or flexibility, often requiring ad-hoc feedback laws on foot height, ankle joints, and CoM.

The idea of explicitly modeling torques produced from position-controlled actuators, proposed by [26] and studied in [27], has been applied to multi-contact tasks on the Walk-Man robot [28], [29]. Similar to our approach, they differentiate the quasi-static equilibrium but their method uses pseudo-inverses and does not consider retargeting nor constraints.

The closest related work is [30], showcasing multi-contact tasks on the position-controlled HRP-2 humanoid. They differentiate the quasi-static equilibrium and use elastic joint models, but their method solves a cascade of QP problems, while our formulation is unified. Their purely reactive control architecture, lacking feedforward terms and retargeted references, is more sensitive to noise and violations of the quasi-static assumption. In contrast, our method allows faster motions and assumes actual joint positions cannot be measured but are estimated by the model, accommodating robots with mechanical flexibility like the Talos robot.

Table I
Mathematical notations

Notation	Description
$n \in \mathbb{N}$	Number of joints
$m_{\text{plane}} \in \mathbb{N}$	Number of enabled plane contacts
$m_{\text{point}} \in \mathbb{N}$	Number of enabled point contacts
$m = 6m_{\text{plane}} + 3m_{\text{point}}$	Dimension of stacked wrench
$(\bullet)^{\text{read}}$	Estimated measured quantities
$(\bullet)^{\text{op}}$	Operator's raw Cartesian commands
$(\bullet)^{\text{adm}}$	Effectors admittance scheme quantities
$(\bullet)^{\text{target}}$	Processed commands for retargeting input
$(\bullet)^{\text{d}}$	Desired state computed by retargeting
$(\bullet)^{\text{flex}}$	Flexible state computed by controller
$(\bullet)_{\text{on}}$	Enabled contact quantities
$(\bullet)_{\text{off}}$	Disabled contact (free effector) quantities
$\boldsymbol{X} \in SE(3)$	Cartesian pose
$\boldsymbol{\nu} \in \mathbb{R}^6$	Cartesian spatial velocity
$\boldsymbol{q} \in \mathbb{R}^{7+n}$	Posture position (floating base and joints)
$\dot{\boldsymbol{q}} \in \mathbb{R}^{6+n}$	Posture velocity
$\boldsymbol{\theta}, \dot{\boldsymbol{\theta}} \in \mathbb{R}^n$	Joint position and velocity
$\boldsymbol{\theta}^{\text{cmd}} \in \mathbb{R}^n$	Joint position command sent to robot
$\boldsymbol{\theta}^{\min}, \boldsymbol{\theta}^{\max} \in \mathbb{R}^n$	Joint position min/max bounds
$\Delta\boldsymbol{\lambda}^{\text{effort}} \in \mathbb{R}^m$	Wrench effort (input to controller)
$\boldsymbol{\tau} \in \mathbb{R}^n$	Joint torque
$\boldsymbol{\tau}^{\max} \in \mathbb{R}^n$	Absolute maximum joint torque
$\tilde{\boldsymbol{\tau}}^{\max} \in \mathbb{R}^n$	Joint torque limits used in Controller
$\boldsymbol{\lambda} \in \mathbb{R}^m$	Stacked contact wrench
$\boldsymbol{k} \in \mathbb{R}^n$	Joint stiffness vector
$\boldsymbol{K} = \mathsf{diag}(\boldsymbol{k}) \in \mathbb{R}^{n \times n}$	Joint stiffness matrix
$\boldsymbol{S} \in \mathbb{R}^{(6+n) \times n}$	Selection matrix joint to full dimension
$\boldsymbol{S'} \in \mathbb{R}^{n \times (6+n)}$	Selection matrix full to joint dimension
$\boldsymbol{g}(\boldsymbol{q}) \in \mathbb{R}^{6+n}$	Gravity vector
$\boldsymbol{J}_{\text{on/off}}(\boldsymbol{q}) \in \mathbb{R}^{(6+n) \times m}$	Stacked effectors Jacobian matrix
$K_p, K_d \in \mathbb{R}$	Proportional and derivative control gains
$K_{\text{adm}} \in \mathbb{R}$	Effectors admittance gain
$\Delta t \in \mathbb{R}$	Time step
$\mathsf{FK}_{\text{on/off}}(\boldsymbol{q})$	Effector poses (forward kinematic)
$\oplus, \ominus$	Operations on $SE(3)$ Lie algebra

III. Problem Definition

Quasi-static robot configurations are defined by postural positions, joint torques, and contact wrenches $\boldsymbol{q}, \boldsymbol{\tau}, \boldsymbol{\lambda}$. For position-controlled robots, control inputs only consist of joint position commands $\boldsymbol{\theta}^{\text{cmd}}$. The whole-body retargeting stage (Fig. 1, [11]) provides a stream of desired quasi-static configurations $\boldsymbol{q}^{\text{d}}, \boldsymbol{\tau}^{\text{d}}, \boldsymbol{\lambda}^{\text{d}}$ expected to be feasible.

Achieving desired contact wrenches $\boldsymbol{\lambda}^{\text{d}}$ is essential for multi-contact tasks, but contact wrenches can not be directly commanded on position-controlled robots. Our approach aims to indirectly control contact wrenches through joint position commands $\boldsymbol{\theta}^{\text{cmd}}$ optimized to take into account the flexibility of the robot. Table I lists the notations and quantities used throughout this letter.

Addressing the problem involves overcoming the following challenges:

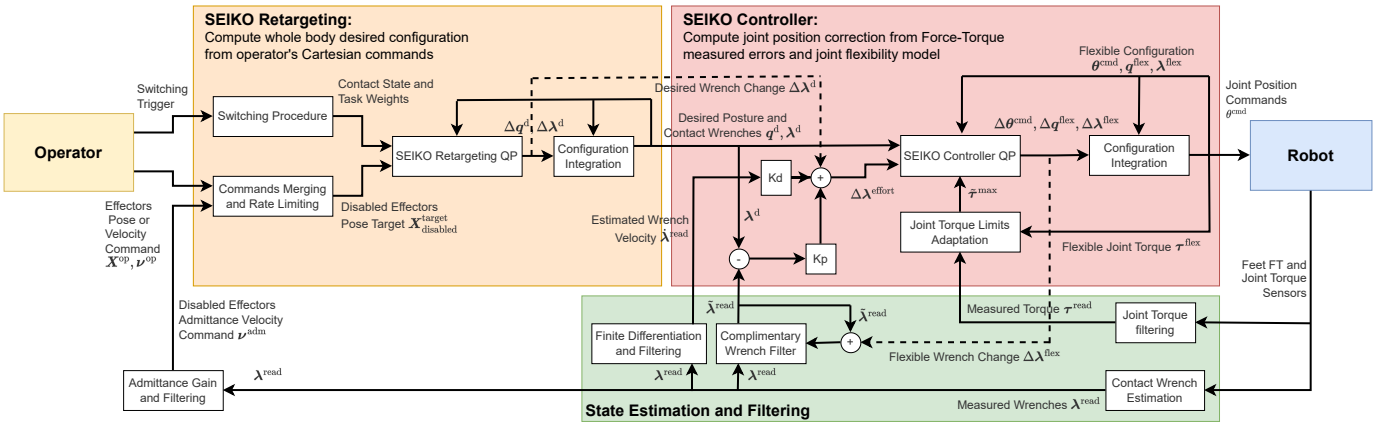

Figure 2. Control architecture for position-controlled robots: Operator's Cartesian commands are retargeted into a feasible whole-body configuration. The controller uses a joint flexibility model to adjust actuator position commands for contact wrench control and prevent exceeding joint torque limits.

- Multi-contact tasks exhibit redundancy in both kinematics and contact wrench distribution, akin to the Grasp matrix's nullspace in manipulation [31]).
- While adding contacts is generally feasible, removing contacts challenge the robot's balance and can be infeasible.
- Transitioning between contact states (enabled or disabled) involves discrete changes in problem formulation. Ensuring continuity in contact wrenches (from non-zero to zero and vice versa) and posture is essential for smooth transitions.
- To ensure safety, physical limits must be enforced such as balance, joint kinematics, actuator torque limits, and contact stability conditions [32] prohibiting pulling, sliding, tilting.
- To apply the controller to hardware, it must be robust to model errors and violations of simplifying assumptions.

IV. Method

A. Main Idea

According to rigid body theory in multi-contact [2], [31], the contact wrenches of an ideal infinitely stiff mechanical system are non-unique and lie in a redundant nullspace. Real systems, however, always exhibit inherent flexibility: the structure slightly bends, and both the deflected posture and contact wrenches uniquely evolve towards the configuration minimizing overall elastic energy. Therefore, given constant joint position commands, the mapping taking into account flexibility $\boldsymbol{\theta}^{\text{cmd}} \mapsto (\boldsymbol{q}^{\text{flex}}, \boldsymbol{\lambda}^{\text{flex}})$ is unique and well-defined. Our approach models and predicts this whole-body non-linear deflection effect, utilizing it for the control of contact wrenches.

Specifically, we differentiate and linearize the deflection effect to consider how contact wrenches change with variations in joint position commands through the Jacobian matrix $\frac{\partial \boldsymbol{\lambda}^{\text{flex}}}{\partial \boldsymbol{\theta}^{\text{cmd}}}(\boldsymbol{q}^{\text{flex}}, \boldsymbol{\lambda}^{\text{flex}}, \boldsymbol{\theta}^{\text{cmd}})$. Instead of directly inverting this Jacobian matrix, we formulate the control problem as a Quadratic Programming (QP) which solves for position command changes and optimizes multiple objectives, similar to task space inverse dynamic approaches. We explicitly model the system's flexibility by treating each robot joint as a spring, encompassing both internal actuator impedance and mechanical flexibilities.

B. Overall Architecture

Our proposed control architecture depicted in Fig. 2 consists of a two-stage pipeline. Firstly, *SEIKO Retargeting*, previously introduced in [11], optimizes a desired whole-body configuration $\boldsymbol{q}^{\text{d}}$,$\boldsymbol{\lambda}^{\text{d}}$,$\boldsymbol{\tau}^{\text{d}}$ within feasibility limits. Subsequently, our novel *SEIKO Controller* computes corrected joint position commands $\boldsymbol{\theta}^{\text{cmd}}$ for tracking $\boldsymbol{\lambda}^{\text{d}}$. These joint commands are then sent to the robot's low-level servomotors and tracked by stiff internal position controllers.

The controller has three goals: (i) achieve the desired contact wrenches $\boldsymbol{\lambda}^{\text{d}}$, (ii) avoid violations of joint torque limits $\boldsymbol{\tau}^{\max}$, and (iii) enhance robustness against model inaccuracies. The Retargeting step is crucial as it enforces feasibility limits a priori, and generates a desired configuration to be tracked. The controller indeed exhibits reduced stability when tracking a highly infeasible non-retargeted reference.

The set of effectors that may come into contact with the environment is pre-defined. Each effector's state is either: "enabled", standing for fixed and in contact transmitting forces and torques to the environment, or "disabled", indicating that it is free to move and is commanded by the operator. Our formulation handles both plane contacts (6 DoFs, e.g., feet) and point contacts (3 DoFs, e.g., hands).

An external planner or human operator provides commands as input to the Retargeting stage: (i) Cartesian pose $\boldsymbol{X}^{\text{op}}$ or velocity $\boldsymbol{\nu}^{\text{op}}$ commands for each free (disabled) effector, (ii) a Boolean signal that manually triggers the transition between contact states, and (iii) an optional "pushing mode" enabling explicit control of the normal force of a specific enabled contact. Our method does not plan contact sequencing, relying on external decisions for contact stances and sequence.

The proposed method operates instantaneously without considering the future of unknown intention, and relies on the quasi-static assumption. The nonlinear whole-body optimizations are solved using SQP schemes with only one QP iteration per time step. This allows for quick convergence at high frequency (500 Hz) and responsiveness to input changes.

C. Equilibrium Equation and Flexibility Model

Motions of mobile robots with a floating base are governed by the equation of motion in joint space [2]. Under the quasi-static assumption, where $\ddot{\boldsymbol{q}} \approx \dot{\boldsymbol{q}} \approx \mathbf{0}$, this equation simplifies to represent the equilibrium, i.e. system's balance, between contact wrenches, gravity effects, and applied torques:

$$\boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{S}\boldsymbol{\tau} + \boldsymbol{J}(\boldsymbol{q})^{\mathsf{T}}\boldsymbol{\lambda}, \tag{1}$$

which is non-linear in $\boldsymbol{q}$. The equilibrium equation is linearized by considering small variations of the configuration $(\boldsymbol{q}+\Delta\boldsymbol{q}, \boldsymbol{\lambda}+\Delta\boldsymbol{\lambda}, \boldsymbol{\tau}+\Delta\boldsymbol{\tau})$. The differentiation is written:

$$\begin{aligned}\boldsymbol{g}(\boldsymbol{q}) + \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{q}}\Delta\boldsymbol{q} = \boldsymbol{S}\boldsymbol{\tau} + \boldsymbol{S}\Delta\boldsymbol{\tau} \\ + \boldsymbol{J}(\boldsymbol{q})^{\mathsf{T}}\boldsymbol{\lambda} + \boldsymbol{J}(\boldsymbol{q})^{\mathsf{T}}\Delta\boldsymbol{\lambda} + \left(\frac{\partial \boldsymbol{J}^{\mathsf{T}}}{\partial \boldsymbol{q}}\boldsymbol{\lambda}\right)\Delta\boldsymbol{q},\end{aligned} \tag{2}$$

while neglecting second order terms.

Stiff position-controlled robots deviate from the rigid assumption due to inherent hardware flexibility arising from factors like Series Elastic Actuators [8], deformations in links or transmissions [10], impedance of non-ideal position control [26], or the inclusion of soft damper elements within the structure [33]. In this work, we model this flexibility as joint elastic flexibility, where the relation between joint position and generated torque is expressed as follows:

$$\boldsymbol{\tau}^{\text{flex}} = \boldsymbol{K}(\boldsymbol{\theta}^{\text{cmd}} - \boldsymbol{\theta}^{\text{flex}}). \tag{3}$$

Note that link flexibility can also be modeled in a similar manner by introducing passive joints without actuation. Its differentiated expression is written:

$$\Delta\boldsymbol{\tau}^{\text{flex}} = \boldsymbol{K}(\Delta\boldsymbol{\theta}^{\text{cmd}} - \Delta\boldsymbol{\theta}^{\text{flex}}) = \boldsymbol{K}\left(\Delta\boldsymbol{\theta}^{\text{cmd}} - \boldsymbol{S}'\Delta\boldsymbol{q}^{\text{flex}}\right), \tag{4}$$

where $\boldsymbol{q}^{\text{flex}}$ is the deflected posture under joint flexibility and $\boldsymbol{\theta}^{\text{cmd}}$ is the joint position command of actuators.

The differentiated equilibrium equation (2) combined with flexibility model (4) is linear w.r.t. configuration changes:

$$\begin{aligned}&\boldsymbol{S}\boldsymbol{K}\Delta\boldsymbol{\theta}^{\text{cmd}} = \boldsymbol{T}(\boldsymbol{q}^{\text{flex}}, \boldsymbol{\lambda}^{\text{flex}})\begin{bmatrix}\Delta\boldsymbol{q}^{\text{flex}} \\ \Delta\boldsymbol{\lambda}^{\text{flex}}\end{bmatrix} + \boldsymbol{t}(\boldsymbol{q}^{\text{flex}}, \boldsymbol{\lambda}^{\text{flex}}, \boldsymbol{\theta}^{\text{cmd}}) \\ &\text{where } \boldsymbol{T}(\boldsymbol{q}^{\text{flex}}, \boldsymbol{\lambda}^{\text{flex}}) = \\ &\left[\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{q}}(\boldsymbol{q}^{\text{flex}}) - \left(\frac{\partial \boldsymbol{J}}{\partial \boldsymbol{q}}^{\mathsf{T}}(\boldsymbol{q}^{\text{flex}})\boldsymbol{\lambda}^{\text{flex}}\right) - \boldsymbol{S}\boldsymbol{K}\boldsymbol{S}' \quad \middle| \quad -\boldsymbol{J}(\boldsymbol{q}^{\text{flex}})^{\mathsf{T}}\right], \\ &\boldsymbol{t}(\boldsymbol{q}^{\text{flex}}, \boldsymbol{\lambda}^{\text{flex}}, \boldsymbol{\theta}^{\text{cmd}}) = \boldsymbol{g}(\boldsymbol{q}^{\text{flex}}) - \boldsymbol{S}\boldsymbol{\tau}^{\text{flex}} - \boldsymbol{J}(\boldsymbol{q}^{\text{flex}})^{\mathsf{T}}\boldsymbol{\lambda}^{\text{flex}}.\end{aligned} \tag{5}$$

Therefore $\Delta\boldsymbol{\theta}^{\text{cmd}}$ can also be linearly expressed from $\Delta\boldsymbol{q}^{\text{flex}}$ and $\Delta\boldsymbol{\lambda}^{\text{flex}}$ using the following row decomposition:

$$\begin{bmatrix}\mathbf{0} \\ \boldsymbol{K}\Delta\boldsymbol{\theta}^{\text{cmd}}\end{bmatrix} = \begin{bmatrix}\boldsymbol{T}_{\text{B}} \\ \boldsymbol{T}_{\text{J}}\end{bmatrix}\begin{bmatrix}\Delta\boldsymbol{q}^{\text{flex}} \\ \Delta\boldsymbol{\lambda}^{\text{flex}}\end{bmatrix} + \begin{bmatrix}\boldsymbol{t}_{\text{B}} \\ \boldsymbol{t}_{\text{J}}\end{bmatrix} \tag{6}$$

$$\Delta\boldsymbol{\theta}^{\text{cmd}}(\Delta\boldsymbol{q}^{\text{flex}}, \Delta\boldsymbol{\lambda}^{\text{flex}}) = \boldsymbol{K}^{-1}\left(\boldsymbol{T}_{\text{J}}\begin{bmatrix}\Delta\boldsymbol{q}^{\text{flex}} \\ \Delta\boldsymbol{\lambda}^{\text{flex}}\end{bmatrix} + \boldsymbol{t}_{\text{J}}\right), \tag{7}$$

where $\boldsymbol{T}_{\text{B}}, \boldsymbol{t}_{\text{B}}$ stands for the floating base rows and $\boldsymbol{T}_{\text{J}}, \boldsymbol{t}_{\text{J}}$ for the joint rows.

D. SEIKO Retargeting

This section summarizes the SEIKO Retargeting method developed in [11], [12].

The Retargeting preprocesses inputs for each disabled effector, which includes the commanded motion from the operator (comprising both pose $\boldsymbol{X}^{\text{op}}$ and velocity $\boldsymbol{\nu}^{\text{op}}$) and the admittance velocity command $\boldsymbol{\nu}^{\text{adm}}$ (see Section IV-F). Processing includes filtering and merging these commands:

$$\begin{aligned}&\boldsymbol{X}_{\text{off}}^{\text{target}} = \mathsf{filtering}(\boldsymbol{X}_{\text{ref}}(t) \oplus \boldsymbol{X}^{\text{op}}) \\ &\boldsymbol{X}_{\text{ref}}(t+\Delta t) = \\ &\mathsf{boundDistance}\big(\boldsymbol{X}_{\text{ref}}(t) \oplus \Delta t(\boldsymbol{\nu}^{\text{op}} + \boldsymbol{\nu}^{\text{adm}}),\ \boldsymbol{X}_{\text{off}}^{\text{d}}\big),\end{aligned} \tag{8}$$

where $\boldsymbol{X}_{\text{ref}} \in SE(3)$ is a reference pose that integrates velocity commands at each time step. It allows the Cartesian pose command to be expressed relative to this reference. The filtering process incorporates a smoothing low-pass filter and enforces signal's velocity and acceleration limits through time-optimal bang-bang trajectory planning. We also constrain $\boldsymbol{X}_{\text{ref}}$ within a radius of $\boldsymbol{X}_{\text{off}}^{\text{d}}$ to prevents the reference pose to windup when the retargeted motion is saturated by the feasibility constraints.

At each time step, SEIKO Retargeting solve the QP:

$$\begin{aligned}&\underset{\Delta\boldsymbol{q}^{\text{d}}, \Delta\boldsymbol{\lambda}^{\text{d}}, \Delta\boldsymbol{\tau}^{\text{d}}}{\text{argmin}} && \text{(9a)} \\ &\qquad \left\|\mathsf{FK}_{\text{off}}(\boldsymbol{q}^{\text{d}}) \oplus \boldsymbol{J}_{\text{off}}(\boldsymbol{q}^{\text{d}})\Delta\boldsymbol{q}^{\text{d}} \ominus \boldsymbol{X}_{\text{off}}^{\text{target}}\right\|^2 + && \text{(9b)} \\ &\qquad \left\|\boldsymbol{\theta}^{\text{d}} + \Delta\boldsymbol{\theta}^{\text{d}} - \boldsymbol{\theta}^{\text{target}}\right\|^2 + && \text{(9c)} \\ &\qquad \left\|\boldsymbol{\tau}^{\text{d}} + \Delta\boldsymbol{\tau}^{\text{d}}\right\|^2 + && \text{(9d)} \\ &\qquad \left\|\boldsymbol{\lambda}^{\text{d}} + \Delta\boldsymbol{\lambda}^{\text{d}}\right\|^2 + && \text{(9e)} \\ &\qquad \left\|\Delta\boldsymbol{q}^{\text{d}}\right\|^2 + \left\|\Delta\boldsymbol{\lambda}^{\text{d}}\right\|^2 && \text{(9f)} \\ &\text{such that} \\ &\qquad \text{differentiated equilibrium equation (2)} && \text{(9g)} \\ &\qquad \mathsf{FK}_{\text{on}}(\boldsymbol{q}^{\text{d}}) \oplus \boldsymbol{J}_{\text{on}}(\boldsymbol{q}^{\text{d}})\Delta\boldsymbol{q}^{\text{d}} \ominus \boldsymbol{X}_{\text{on}}^{\text{target}} = \mathbf{0} && \text{(9h)} \\ &\qquad \boldsymbol{\theta}^{\min} \le \boldsymbol{\theta}^{\text{d}} + \Delta\boldsymbol{\theta}^{\text{d}} \le \boldsymbol{\theta}^{\max} && \text{(9i)} \\ &\qquad -\boldsymbol{\tau}^{\max} \le \boldsymbol{\tau}^{\text{d}} + \Delta\boldsymbol{\tau}^{\text{d}} \le \boldsymbol{\tau}^{\max} && \text{(9j)} \\ &\qquad C_{\text{contact}}(\boldsymbol{\lambda}^{\text{d}})\Delta\boldsymbol{\lambda}^{\text{d}} + c_{\text{contact}}(\boldsymbol{\lambda}^{\text{d}}) \ge \mathbf{0} && \text{(9k)} \\ &\qquad -\Delta t\dot{\boldsymbol{\theta}}^{\max} \le \Delta\boldsymbol{\theta}^{\text{d}} \le \Delta t\dot{\boldsymbol{\theta}}^{\max} && \text{(9l)} \\ &\qquad -\Delta t\dot{\boldsymbol{\lambda}}^{\max} \le \Delta\boldsymbol{\lambda}^{\text{d}} \le \Delta t\dot{\boldsymbol{\lambda}}^{\max}. && \text{(9m)}\end{aligned}$$

The QP solves for the configuration change (9a), integrating it to update the desired configuration, e.g., $\boldsymbol{\lambda}^{\text{d}}(t+\Delta t) = \boldsymbol{\lambda}^{\text{d}}(t) + \Delta\boldsymbol{\lambda}^{\text{d}}$. The optimization minimizes tasks weighted by manually tuned parameters for stability and desired trade-off. The cost function includes disabled effector pose targets (9b), default joint position targets (9c) for regularization and mitigating kinematic local minima, joint torque minimization (9d) for human-like postures, contact wrench penalization (9e), and decision variable regularization (9f).

Equality constraints enforce the equilibrium equation (9g) and ensure enabled contacts are fixed (9h). Inequality constraints include joint position limits (9i), joint torque limits (9j), and contact stability conditions (9k) considering unilaterality, friction pyramid, and center of pressure (see [32]). Additional constraints involve limits on joint changes (9l) and contact wrench changes (9m).

We enhanced the contact switching procedure compared to prior work. To remove a contact, we instantly increase the weight of the wrench penalty task to a very high value and use joint velocity $\dot{\boldsymbol{\theta}}^{\max}$ and wrench velocity $\dot{\boldsymbol{\lambda}}^{\max}$ limits to ensure a smooth transition. When the integrated desired wrench falls below a small threshold, the contact is removed. Enabling a contact is straightforward, as it doesn't require any special considerations, thanks to these limits.

E. SEIKO Controller

We assume that actual joint positions under flexibility cannot be directly measured but can be estimated from the model. Despite model errors, our approach relies on the model's derivatives direction to provide sufficient information about system evolution. The controller uses differentiation of the equilibrium equation with flexibility (5) to model how contact wrench distribution changes with joint command changes $\Delta\boldsymbol{\theta}^{\text{cmd}}$. This approach generalizes previously used admittance control laws such as "foot difference control" [21] which implicitly depends on flexibility without considering it.

A unique feedback law is applied from measured wrenches:

$$\Delta\boldsymbol{\lambda}^{\text{effort}} = \Delta\boldsymbol{\lambda}^{\text{d}} + K_p(\boldsymbol{\lambda}^{\text{d}} - \tilde{\boldsymbol{\lambda}}^{\text{read}}) - K_d\dot{\boldsymbol{\lambda}}^{\text{read}}, \tag{10}$$

where $\Delta\boldsymbol{\lambda}^{\text{effort}}$ is the desired effort in the controller optimization, and $\Delta\boldsymbol{\lambda}^{\text{d}}$ acts as a feedforward term. SEIKO Controller solves the following QP at each time step::

$$\underset{\Delta\boldsymbol{q}^{\text{flex}},\Delta\boldsymbol{\lambda}^{\text{flex}}}{\operatorname{argmin}} \tag{11a}$$

$$\left\|\Delta\boldsymbol{\lambda}^{\text{effort}} - \Delta\boldsymbol{\lambda}^{\text{flex}}\right\|^2 + \tag{11b}$$

$$\left\|\mathsf{FK}_{\text{off}}(\boldsymbol{q}^{\text{flex}}) \oplus \boldsymbol{J}_{\text{off}}(\boldsymbol{q}^{\text{flex}})\Delta\boldsymbol{q}^{\text{flex}} \ominus \boldsymbol{X}^{\text{d}}_{\text{off}}\right\|^2 \tag{11c}$$

$$\left\|\boldsymbol{\theta}^{\text{cmd}} + \Delta\boldsymbol{\theta}^{\text{cmd}} - \boldsymbol{\theta}^{\text{d}}\right\|^2 + \tag{11d}$$

$$\left\|\Delta\boldsymbol{\theta}^{\text{cmd}}\right\|^2 + \tag{11e}$$

such that

$$\boldsymbol{T}_{\text{B}}\begin{bmatrix}\Delta\boldsymbol{q}^{\text{flex}}\\ \Delta\boldsymbol{\lambda}^{\text{flex}}\end{bmatrix} + \boldsymbol{t}_{\text{B}} = \boldsymbol{0} \tag{11f}$$

$$\mathsf{FK}_{\text{on}}(\boldsymbol{q}^{\text{flex}}) \oplus \boldsymbol{J}_{\text{on}}(\boldsymbol{q}^{\text{flex}})\Delta\boldsymbol{q}^{\text{flex}} \ominus \boldsymbol{X}^{\text{target}}_{\text{on}} = \boldsymbol{0} \tag{11g}$$

$$\boldsymbol{\theta}^{\min} \le \boldsymbol{\theta}^{\text{cmd}} + \Delta\boldsymbol{\theta}^{\text{cmd}} \le \boldsymbol{\theta}^{\max} \tag{11h}$$

$$-\tilde{\boldsymbol{\tau}}^{\max} \le \boldsymbol{\tau}^{\text{flex}} + \Delta\boldsymbol{\tau}^{\text{flex}} \le \tilde{\boldsymbol{\tau}}^{\max}. \tag{11i}$$

The QP solves for flexible configuration changes $\Delta\boldsymbol{q}^{\text{flex}}$,$\Delta\boldsymbol{\lambda}^{\text{flex}}$ (11a). Joint command changes $\Delta\boldsymbol{\theta}^{\text{cmd}}$ are obtained from the decision variables using (7) and $\boldsymbol{q}^{\text{flex}}$,$\boldsymbol{\lambda}^{\text{flex}}$,$\boldsymbol{\theta}^{\text{cmd}}$ are then obtained by integration.

The cost function primarily computes joint position correction $\Delta\boldsymbol{\theta}^{\text{cmd}}$ and resulting posture deflection $\Delta\boldsymbol{q}^{\text{flex}}$ to achieve the control effort on contact wrench changes $\Delta\boldsymbol{\lambda}^{\text{effort}}$ (11b). It also adjusts disabled effector poses influenced by flexibility toward Retargeting's desired poses (11c). As secondary objectives, the optimization penalizes the discrepancy between corrected and desired joint positions (11d) and regularizes changes in joint commands (11e).

Equality constraints enforce differentiated equilibrium equation with flexibility (11f) through upper floating base rows decomposition (6) and ensure no Cartesian motion for enabled contacts (11g). Inequality constraints ensure kinematic limits of joint position commands $\boldsymbol{\theta}^{\text{cmd}}$ (11h) and restrict maximum joint torques (11i).

Joint torque limits $\tilde{\boldsymbol{\tau}}^{\max}$ used as constraints are dynamically updated to prevent the integrated state $|\boldsymbol{\tau}^{\text{flex}}|$ from continuously increasing when the measured joint torque $|\boldsymbol{\tau}^{\text{read}}|$ reaches the defined torque limit $\boldsymbol{\tau}^{\max}$. For each joint at each time step:

$$\tilde{\tau}^{\max}(t+\Delta t) = \begin{cases} \tau^{\text{flex}} + \epsilon_1 & \text{if } |\tau^{\text{read}}| > \tau^{\max} \wedge \\ & \tilde{\tau}^{\max}(t) > \tau^{\text{flex}} + \epsilon_1, \\ \tilde{\tau}^{\max}(t) + \epsilon_2 & \text{else if } |\tau^{\text{read}}| < \tau^{\max} - \epsilon_3 \wedge \\ & \tilde{\tau}^{\max}(t) < \tau^{\max}, \\ \tau^{\max} & \text{else if } \tilde{\tau}^{\max}(t) > \tau^{\max}, \\ \tilde{\tau}^{\max}(t) & \text{else,} \end{cases} \tag{12}$$

where $\epsilon_1, \epsilon_2, \epsilon_3 \in \mathbb{R}$ are small positive margin parameters implementing a hysteresis effect to improve stability.

F. State Estimation and Effectors Admittance

The estimated measured wrench $\tilde{\boldsymbol{\lambda}}^{\text{read}}$ in feedback law (10) is computed using a complementary filter:

$$\tilde{\boldsymbol{\lambda}}^{\text{read}}(t+\Delta t) = \alpha\left(\tilde{\boldsymbol{\lambda}}^{\text{read}}(t) + \Delta\boldsymbol{\lambda}^{\text{flex}}\right) + (1-\alpha)\boldsymbol{\lambda}^{\text{read}}. \tag{13}$$

This filter enhances closed-loop stability by mitigating dynamical effects affecting $\boldsymbol{\lambda}^{\text{read}}$ neglected by the quasi-static assumption. It introduces a trade-off between the reactive measurement and the term estimated through the integration of the predicted change $\Delta\boldsymbol{\lambda}^{\text{flex}}$.

We utilize an admittance scheme to compute an additional Cartesian velocity command for disabled effectors $\boldsymbol{\nu}^{\text{adm}}$:

$$\boldsymbol{\nu}^{\text{adm}} = \mathsf{filtering}\left(K_{\text{adm}}\boldsymbol{\lambda}^{\text{read}}_{\text{off}}\right), \tag{14}$$

where filtering involves a deadband and output clamping. This control law aims to reduce interaction wrenches to zero for disabled effectors, preventing large unintended and unmodeled forces during contact establishment, facilitating foot alignment with surface orientation, and minimizing residual wrenches after contact removal. Implemented at input of the Retargeting level, this approach seamlessly integrates with operator command processing (8).

V. Experimental Evaluation

A. Implementation Details

We implemented SEIKO in C++ using RBDL [34] and Pinocchio [35] rigid body libraries. More specifically, Pinocchio efficiently computes the analytical derivatives of the terms appearing in the differentiated equation (2). We solve the QP problems using the QuadProg [36] solver.

The entire control pipeline operates at a frequency of 500 Hz, with joint position commands interpolated at 2 kHz before being transmitted to the robot's actuators. The median computing times observed on the internal computer of the Talos robot are 0.50 ms and 0.40 ms for SEIKO Retargeting and SEIKO Controller, respectively. The maximum measured times for each were 0.56 ms and 0.43 ms, respectively.

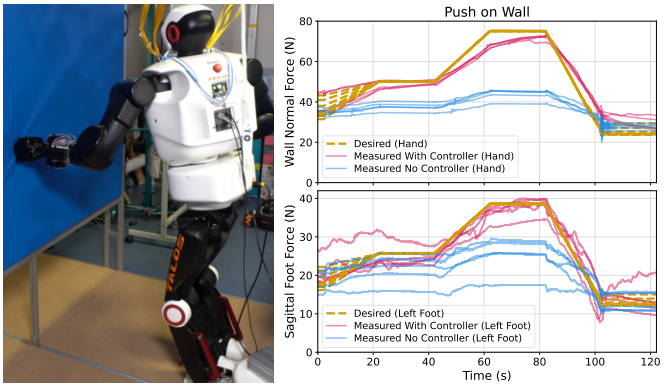

Figure 3. Force distribution tracking during pushing tasks. The Talos robot (left) pushes a vertical wall using its left hand, following a predefined hand force target trajectory. Plots display the desired and measured normal force for the left hand (top) and the sagittal tangential force for the left foot (bottom) ; comparing with control enabled (5 trials) and without (5 trials).

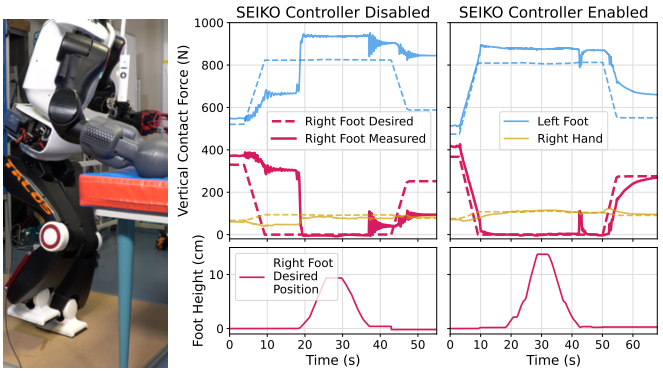

Figure 4. Comparison of contact switch trials with and without SEIKO Controller. Initially, both feet and right hand are in contact. The operator teleoperated the robot to disable the right foot contact, lift the foot, and re-establish contact. Vertical contact forces $\boldsymbol{\lambda}^{\mathrm{d}}$,$\boldsymbol{\lambda}^{\mathrm{read}}$ (top row) and the desired vertical position of the right foot $\boldsymbol{X}^{\mathrm{d}}_{\mathrm{right\ foot}}$ (bottom row) are displayed.

The Talos robot, manufactured by PAL Robotics, is a humanoid robot of 1.75 m height with 32 DoFs. Externally, we measured its actual total mass to be 99.7 kg, while the URDF model provided by PAL assumes a mass of 93.4 kg. This discrepancy of 6 kg can be seen by the Force-Torque sensors in the feet, which enable our controller to adapt to this model error. We changed the robot's right hand and forearm with a 3D printed part that replaced the gripper and wrist joints beyond the elbow joint. The ball-shaped hand allows us to apply high contact forces (up to 30 kg) on the arm during multi-contact tests. After removing the right forearm joints and excluding the head joints, our QP solver works with $n=25$ joints.

Throughout all our evaluations, we employed as flexibility model $\boldsymbol{K}$ the position-control P gains imported from PAL's Gazebo simulation of the Talos robot. Unlike other works [10] that estimate precise flexibility model, our approach does not heavily depend on model accuracy. This is because our differentiated formulation utilizes only the approximate "gradient" direction for whole-body control.

In all subsequent experiments, an expert operator issued velocity commands for each robot's effectors using dedicated 6-DoF input devices[1], with one device assigned to each effector. Teleoperation was conducted with a clear, direct line of sight to the robot and its surrounding environment.

B. Wrench Distribution Tracking

In Fig. 3, we illustrate the role of SEIKO Controller in realizing multi-contact wrench distribution during a hand pushing task. The robot initiates a point contact with a vertical wall using its left hand. The "pushing mode" of SEIKO is employed to command a target trajectory for the normal force applied on the wall. Retargeting adjusts the robot's posture slightly forward to apply a large force (75 N), and generates the desired contact wrenches, including opposing tangential forces on the feet in the sagittal plane.

It is worth noting that we didn't perform any identification or tuning of the robot flexibility model on the actual hardware,

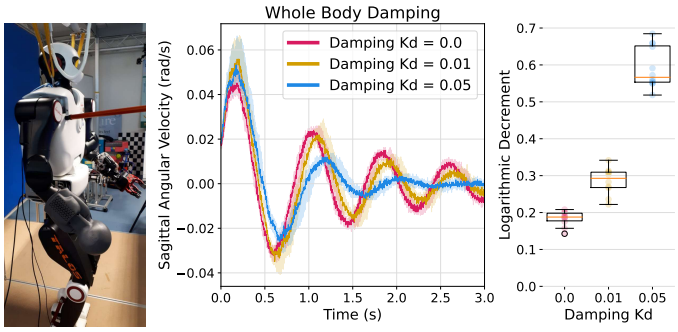

Figure 5. Impact of damping gain K_d on Talos's torso oscillations. Short pushes are applied (left), and IMU's gyroscope measures sagittal plane oscillation for varying K_d values (middle). Damping effect is quantified using logarithmic decrement metric from oscillation peaks (right).

which may have significant errors. Estimating this flexibility [10] could enhance tracking accuracy, given that we observed near-perfect tracking performance in the Gazebo simulator which uses an ideal model.

The attached video[2] demonstrates additional multi-contact scenarios, such as stair climbing and stepping on a sloped surface (Fig. 1).

C. Contact Switch

Fig. 4 illustrates the foot contact switch capabilities, showcasing the Talos robot being teleoperated to lift and then re-establish contact with the right foot. Without the Controller, weight transfer from the right to the left foot and hand occurs abruptly during the foot lift. The robot did not fall as it was operating far from its feasibility boundaries. Conversely, when the controller and admittance scheme (equation (14)) were enabled, the redistribution of contact wrenches became smooth and controlled. Additionally, at $t=43$ s, when the foot collided with the ground, the admittance control sightly lifted the foot to prevent unwanted ground forces before contact was re-established.

[1]3Dconnexion SpaceMouse: https://3dconnexion.com/uk/spacemouse/

[2]https://hucebot.github.io/seiko_controller_website/

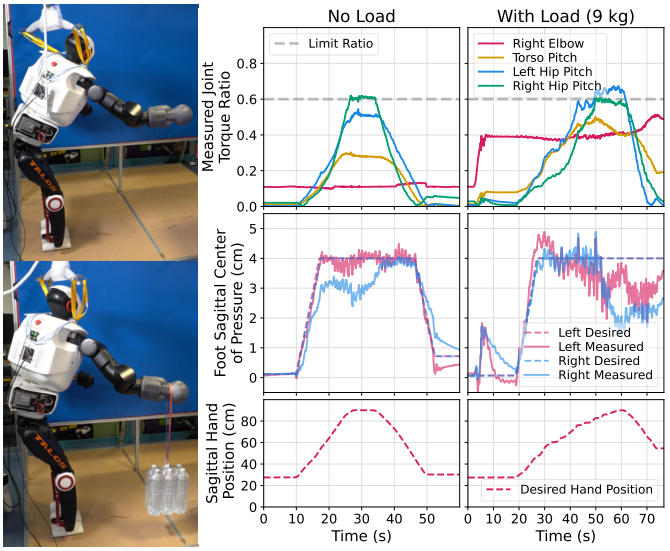

Figure 6. Far reaching task with and without adding a large unmodeled mass (9 kg) on the hand. The controller enforces joint torque ratio limits (top row, set to 0.6) and tracks the foot contact wrenches (middle row) to ensure balance.

D. Whole Body Damping

Imperfect stiff position control and flexibilities lead to small oscillations when disturbed, particularly noticeable on Talos in the sagittal plane, causing forward-backward oscillations. In equation (10), the controller's feedback law employs a damping term with the gain parameter K_d. We show in Fig. 5 that this unique feedback law on contact wrenches effectively attenuates these whole-body oscillations.

In double support, we applied short pushes (10-12 pushes, Fig.5 left) to the robot's torso and observed oscillations until energy dissipation. Using the controller, we tested various damping gain ($K_d = 0.0, 0.01, 0.05$). We recorded unfiltered angular velocity in sagittal plane with pelvis IMU's gyroscope since it does not rely on model nor unobserved joint positions. Fig. 5 (center) shows median and $20\%-80\%$ deciles confidence interval of sagittal motion velocity. To quantify damping (Fig. 5 right), we estimated the averaged logarithmic decrement from oscillation peaks ($\delta = \text{avg}\left(\log\left(\frac{\omega(t)}{\omega(t+T)}\right)\right)$), reflecting damping of oscillation amplitudes and linked to the damping ratio for under-damped systems.

In following experiments, the damping gain is set to $K_d = 0.02$, as higher values tended to be unstable near feasibility boundaries where model errors had a more pronounced effect.

E. Far Reaching with Model Errors

Fig. 6 illustrates the capability of our approach to perform challenging far-reaching tasks near feasibility limits, even in the presence of large model errors. We teleoperated the right hand of the Talos robot for a forward-reaching motion as far as allowed by the controller, and added a 9 kg load during operation on the hand to induce mass model errors. The robot remained stable thanks to the tracking of foot contact wrenches and adaptation of the whole body posture. Additionally, the Controller through equation (12) prevents excessive violation of joint torques, with a limit ratio set to $\frac{|\tau^{\text{read}}|}{\tau^{\max}} < 0.6$.

F. Robustness Evaluation

We performed a comprehensive analysis of our approach's robustness using the MuJoCo simulator, as summarized in Fig. 7. The focus was on evaluating the impact of model errors and motion speed on system's balance. We simulated the Talos robot in double support, executing 10 motion sequences reaching a distant target with the left hand and returning to the initial posture. The number of successful trials without fall for three conditions are reported: (i) without SEIKO Controller, (ii) with SEIKO Controller but without considering joint torque limits (11i), (12), and (iii) using the full control method. Variations included hand Cartesian motion velocity (slow 2 cm/s to fast 40 cm/s) and additional mass on the left hand (none to 12 kg).

We observed that MuJoCo's soft contact model produces a more pronounced flexibility behavior than Gazebo or even the actual robot. The presented results implicitly incorporate flexibility model errors, although they are not quantified.

SEIKO Retargeting without whole-body control (left) operates in open-loop and is partially robust to motion speed but struggles with model errors. Introducing SEIKO Controller (middle) significantly improves success rates, adapting joint position commands to handle additional hand mass for balance. However, unplanned posture changes and model errors near full extension reach actuator torque limits, leading to loss of control authority. Considering actuator torque limits in the controller (right) enhances robustness by optimizing posture and avoiding infeasible hand pose commands. Challenges persist at high speeds and heavy masses, where inertial effects violate the quasi-static assumption.

VI. Discussion and Conclusion

Our control architecture's robustness is showcased at moderate motion speeds (Fig. 7), but it inherently relies on the quasi-static assumption and is unsuitable for highly dynamic motions. Establishing contact with stiff position-controlled robots requires precise and slow operator commands, even if effectors admittance (14) helps mitigating this problem. Future work could explore applying the proposed approach to robots using joint impedance control. As analyzed in [10], we noted greater leg flexibility in the Talos robot than in our basic model. Although our controller enables successful contact transitions in teleoperated tasks, this significant difference hampers the quick contact switches needed for walking. Refining the flexibility model may allow walking capabilities.

The robot fell when attempting to climb large 20 cm stairs due to exceeding arm joint torque limits during the challenging contact switch. Despite being theoretically feasible according to the retargeting model, the adaptation of joint torque limits (12) is insufficient to ensure robustness if an infeasible contact transition is attempted due to model errors (e.g., underestimating the robot's weight).

Our approach overcomes the inherent lack of direct control authority over contact forces of position-controlled by explicitly considering flexibilities. While torque-controlled robots are traditionally used to perform pushing and multi-contact tasks, our SEIKO control pipeline extends these capabilities to position-controlled robots. We also demonstrate robustness

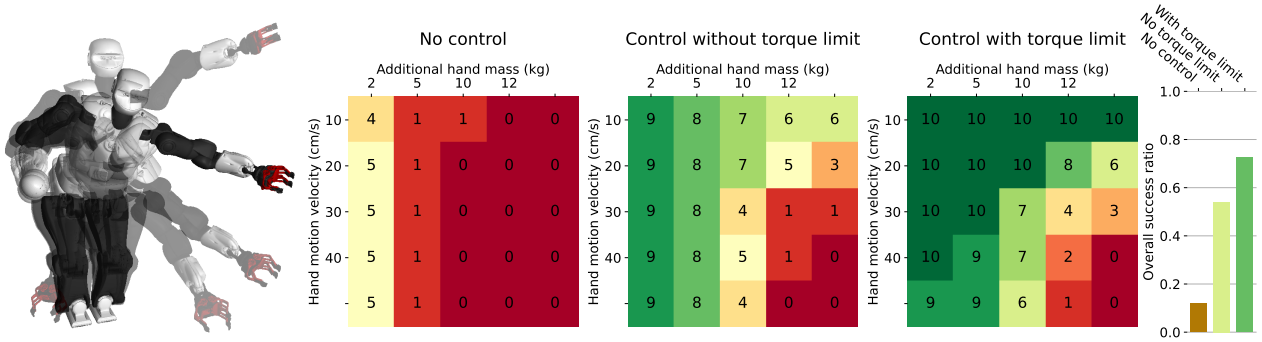

Figure 7. Comparison of our controller's robustness against model errors and motion velocity. The Talos robot performs in double support 10 far-reaching tasks at the edge of the feasibility boundary in the MuJoCo simulator (left). The number of successful trials without falling is indicated (out of 10). Different combinations of hand motion velocity and added mass on the robot's hand are compared (middle). The comparison includes scenarios with the SEIKO controller disabled (only open-loop SEIKO retargeting), the SEIKO controller with only foot wrenches control, and the full controller also considering joint torque limits. Overall success ratio comparing the three controllers is given on right panel.

to model errors, safely carrying substantial unmodelled loads at arm's length. The unified formulation employs a single feedback law on contact forces, effectively leveraging both posture change (i.e., CoM displacement) and contact force redistribution to regulate whole-body balance. Given that the primary advantage of humanoids and other multi-limbed robots lies in their strong versatility, this research paves the way for broadening the application and deployment of real-world scenarios, utilizing more capable and adaptable multi-contact systems in uncertain contexts and environments.

References

[1] C. G. Atkeson *et al.*, "What happened at the darpa robotics challenge finals," *The DARPA robotics challenge finals: Humanoid robots to the rescue*, 2018.

[2] R. Featherstone, *Rigid body dynamics algorithms*. Springer, 2014.

[3] R. Cisneros-Limon *et al.*, "An inverse dynamics-based multi-contact locomotion control framework without joint torque feedback," *Advanced Robotics*, 2020.

[4] B. Henze, M. A. Roa, and C. Ott, "Passivity-based whole-body balancing for torque-controlled humanoid robots in multi-contact scenarios," *Int. Journal of Robotics Research*, 2016.

[5] F. Abi-Farraj *et al.*, "Torque-based balancing for a humanoid robot performing high-force interaction tasks," *IEEE RA-L*, 2019.

[6] N. Ramuzat *et al.*, "Actuator model, identification and differential dynamic programming for a talos humanoid robot," in *IEEE ECC*, 2020.

[7] G. Romualdi *et al.*, "A benchmarking of dcm-based architectures for position, velocity and torque-controlled humanoid robots," *International Journal of Humanoid Robotics*, 2020.

[8] G. Nava, D. Pucci, and F. Nori, "Momentum control of humanoid robots with series elastic actuators," in *IEEE/RSJ IROS*, 2017.

[9] O. Stasse *et al.*, "Talos: A new humanoid research platform targeted for industrial applications," in *IEEE-RAS Humanoids*, 2017.

[10] N. A. Villa *et al.*, "Torque controlled locomotion of a biped robot with link flexibility," in *IEEE-RAS Humanoids*, 2022.

[11] Q. Rouxel, K. Yuan, R. Wen, and Z. Li, "Multicontact motion retargeting using whole-body optimization of full kinematics and sequential force equilibrium," *Trans. on Mechatronics*, 2022.

[12] R. Wen, Q. Rouxel, M. Mistry, Z. Li, and C. Tiseo, "Collaborative bimanual manipulation using optimal motion adaptation and interaction control," *IEEE RAM*, 2023.

[13] Q. Rouxel, R. Wen, Z. Li, C. Tiseo, J.-B. Mouret, and S. Ivaldi, "Feasibility retargeting for multi-contact teleoperation and physical interaction," *2nd Workshop Toward Robot Avatars, ICRA*, 2023.

[14] K. Darvish *et al.*, "Teleoperation of humanoid robots: A survey," *IEEE Trans. on Robotics*, 2023.

[15] A. Di Fava *et al.*, "Multi-contact motion retargeting from human to humanoid robot," in *IEEE-RAS Humanoids*, 2016.

[16] S. McCrory, S. Bertrand, D. Calvert, J. Pratt, and R. Griffin, "A virtual-reality driven approach for generating humanoid multi-contact trajectories," *arXiv preprint arXiv:2303.08232*, 2023.

[17] F. Ruscelli *et al.*, "A multi-contact motion planning and control strategy for physical interaction tasks using a humanoid robot," in *IEEE/RSJ IROS*, 2020.

[18] M. P. Polverini *et al.*, "Multi-contact heavy object pushing with a centaur-type humanoid robot: Planning and control for a real demonstrator," *IEEE RA-L*, 2020.

[19] Z. Li *et al.*, "Stabilization for the compliant humanoid robot coman exploiting intrinsic and controlled compliance," in *IEEE ICRA*, 2012.

[20] J. Vaillant *et al.*, "Multi-contact vertical ladder climbing with an hrp-2 humanoid," *Autonomous Robots*, 2016.

[21] S. Kajita *et al.*, "Biped walking stabilization based on linear inverted pendulum tracking," in *IEEE/RSJ IROS*, 2010.

[22] S. Caron, A. Kheddar, and O. Tempier, "Stair climbing stabilization of the hrp-4 humanoid robot using whole-body admittance control," in *IEEE ICRA*, 2019.

[23] R. Cisneros *et al.*, "Qp-based task-space hybrid/parallel control for multi-contact motion in a torque-controlled humanoid robot," in *IEEE-RAS Humanoids*, 2019.

[24] S. Samadi *et al.*, "Balance of humanoid robots in a mix of fixed and sliding multi-contact scenarios," in *IEEE ICRA*, 2020.

[25] ——, "Humanoid control under interchangeable fixed and sliding unilateral contacts," *IEEE RA-L*, 2021.

[26] O. Khatib *et al.*, "Torque-position transformer for task control of position controlled robots," in *IEEE ICRA*, 2008.

[27] A. Del Prete *et al.*, "Implementing torque control with high-ratio gear boxes and without joint-torque sensors," *Int. Journal of Humanoid Robotics*, 2016.

[28] E. Farnioli, M. Gabiccini, and A. Bicchi, "Optimal contact force distribution for compliant humanoid robots in whole-body loco-manipulation tasks," in *IEEE ICRA*, 2015.

[29] ——, "Toward whole-body loco-manipulation: Experimental results on multi-contact interaction with the walk-man robot," in *IEEE/RSJ IROS*, 2016.

[30] N. Hiraoka *et al.*, "Online generation and control of quasi-static multi-contact motion by pwt jacobian matrix with contact wrench estimation and joint load reduction," *Advanced Robotics*, 2021.

[31] E. Rimon and J. Burdick, *The Mechanics of Robot Grasping*. Cambridge University Press, 2019.

[32] S. Caron, Q.-C. Pham, and Y. Nakamura, "Stability of surface contacts for humanoid robots: Closed-form formulae of the contact wrench cone for rectangular support areas," in *IEEE ICRA*, 2015.

[33] K. H. Koch, K. Mombaur, O. Stasse, and P. Soueres, "Optimization based exploitation of the ankle elasticity of hrp-2 for overstepping large obstacles," in *IEEE-RAS Humanoids*, 2014.

[34] M. L. Felis, "Rbdl: an efficient rigid-body dynamics library using recursive algorithms," *Autonomous Robots*, 2016.

[35] J. Carpentier *et al.*, "The pinocchio c++ library – a fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives," in *IEEE SII*, 2019.

[36] D. Goldfarb and A. Idnani, "A numerically stable dual method for solving strictly convex quadratic programs," *Mathematical programming*, 1983.