



**HAL**  
open science

# Introducing Force Feedback in Model Predictive Control

Sébastien Kleff, Ewen Dantec, Guilhem Saurel, Nicolas Mansard, Ludovic Righetti

► **To cite this version:**

Sébastien Kleff, Ewen Dantec, Guilhem Saurel, Nicolas Mansard, Ludovic Righetti. Introducing Force Feedback in Model Predictive Control. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2022), Oct 2022, Kyoto, Japan. hal-03594295v1

**HAL Id: hal-03594295**

**<https://hal.science/hal-03594295v1>**

Submitted on 2 Mar 2022 (v1), last revised 10 Nov 2022 (v2)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Introducing Force Feedback in Model Predictive Control

Sébastien Kleff<sup>1,2</sup>, Ewen Dantec<sup>2,3</sup>, Guilhem Saurel<sup>1</sup>, Nicolas Mansard<sup>2,3</sup>, Ludovic Righetti<sup>1,4</sup>

**Abstract**—In the literature about model predictive control (MPC), contact forces are planned rather than controlled. In this paper, we propose a novel paradigm to incorporate effort measurements into a predictive controller, hence allowing to control them by direct measurement feedback. We first demonstrate why the classical optimal control formulation, based on position and velocity state feedback, cannot handle direct feedback on force information. Following previous approaches in force control, we then propose to augment the classical formulations with a model of the robot actuation, which naturally allows to generate online trajectories that adapt to sensed position, velocity and torques. We propose a complete implementation of this idea on the upper part of a real humanoid robot, and show through hardware experiments that this new formulation incorporating effort feedback outperforms classical MPC in challenging tasks where physical interaction with the environment is crucial.

## I. INTRODUCTION

Many tasks accomplished by humans in everyday life require a sense of touch. For instance feeling external forces is of primal importance when handing over an object, sanding a rough surface or kneading dough. While the importance of haptic feedback in robotic manipulation or locomotion tasks is well acknowledged, recent progress in advanced control methodology based on optimal control have reduced our capability to account for an artificial sense of touch. As a matter of fact, fast numerical optimal control solvers [1] combined with torque-control capabilities of modern robots have made nonlinear MPC a mature technology for manipulation and locomotion, thanks to its ability to react and reason about the future at the low control level [2], [3]. But such controllers reveal their brittleness during contact situations: they rely on simplistic contact models with limited capability to predict future interactions, so the resulting policies are not meaningful. Indeed, when creating contact with an object to fulfill a task, deciding what action to take next should imply some awareness of the force that is currently being applied on that object and how it may evolve in the future.

Consequently these controllers require tedious hand-tuning with no guarantees of success. This fundamental problem remains open for MPC practitioners in robotics: how to control contact interactions? A common practice is to discard force and torques sensory information in the online optimization and to treat contacts as kinematic constraints. Although this

approach circumvents the need for a potentially complex model of interactions with a priori unknown environments, it comes at the price of only being able to control forces in a feedforward sense. Yet there are solid reasons to believe that closing the loop on force measurements inside a MPC framework could increase the robustness and versatility of robots in contact tasks [4]–[6]. First, the superiority of MPC over instantaneous control methods in online motion generation is now well established [7], [8]. Second, it is known from the field of interaction control that stabilizing a contact interaction requires to take into account both motion and forces [9]. Thirdly, modern robots are increasingly equipped with diverse sensor modalities which can be exploited, in particular force and torque sensors [10]–[13]. In this paper we propose to model the actuation as a low-pass filter in order to allow feedback control on forces.

In the MPC and Trajectory Optimization (TO) literature, contact forces are *planned* rather than *controlled*. For instance, standard approaches in locomotion consist in optimizing contact forces as control inputs to the centroidal dynamics [14], [15], or as auxiliary variables associated with kinematic constraints of the whole-body dynamics [16], [17]. Other works have also proposed to discover online optimized rigid contacts by solving LCPs [17], [18] or smooth contact force models that are compatible with standard TO algorithms [19]. While these works report experimental evidence of the importance of utilizing predictive models of contact forces to generate contact-rich behaviors, the perceived forces are never fed-back and compared against their predictions, which prevents from controlling physical interactions.

From a different perspective, the controller proposed in [20] presents similarities with our approach. It enforces the actuation’s bandwidth limitation by using a frequency-dependent cost function. In that sense, our approach resembles the special case of a low-pass shaping function. But the conceptual difference is that frequency-shaping leaves the relation between state and control unchanged during the optimization, while our controller reasons over higher dimensional dynamics, which enables to naturally derive control policies that depend on torque measurements. The same model augmentation is exploited in [21] in order to derive contact-aware policies for complementarity systems.

In another line of work, some authors have proposed to combine nonlinear MPC with interaction control. In [22] the authors propose an output feedback control strategy combined with path-following MPC in order to track desired forces and motion under nonlinear constraints. In [23], MPC is used to achieve a desired impedance while ensuring con-

<sup>1</sup>Tandon School of Engineering, New York University, Brooklyn, NY

<sup>2</sup>LAAS-CNRS, Université de Toulouse, CNRS, Toulouse

<sup>3</sup>Artificial and Natural Intelligence Toulouse Institute (ANITI), Toulouse

<sup>4</sup>Max Planck Institute for Intelligent Systems, Tübingen

We would like to thank Pierre Fernbach for his valuable support during hardware experiments.

straints. In [24], indirect adaptive control is used to adjust the impedance while interacting with an unknown environment.

In this paper we propose a new MPC paradigm that enables to re-optimize online motions based on measured joint torques. Our approach is based on an augmented dynamic model of the robot that includes an actuation model, and doesn't require any general model of the contact forces. We propose a comparative analysis of performance in simulations and hardware experiments of a sanding task on two torque-controlled robots. We also draw connections with the existing literature of MPC and interaction control, and discuss the implications of this proof-of-concept for the practice of nonlinear MPC on torque-controlled robots. Our work shows that a simple actuation model suffices to treat forces or torques as controlled variables and that incorporating it into MPC leads to increased performance compared to the classical approach.

## II. BACKGROUND

We start the paper by recalling the classical rigid contact model and the consequent classical OCP used to control complex robots in contact. Based on the formulation, we will then draw a paradox when trying to feedback on force measurements with such an approach, which we will use to justify our contribution in the following section.

### A. Rigid contact dynamics

A *rigid contact* is a kinematic constraint acting at the point of contact between the robot and the environment. The equations of motion of a fully-actuated robot in rigid contact with the environment can be derived from the KKT conditions of the convex optimization problem corresponding to Gauss' least constraint principle [25]

$$\min_{\dot{v}} \frac{1}{2} \|\dot{v} - \dot{v}_f\|_{M(q)}^2 \quad (1a)$$

$$\text{s.t. } J(q)\dot{v} + \dot{J}(q)v = 0 \quad (1b)$$

where  $q, v \in \mathbb{R}^n$  are the vectors of joint positions and velocities,  $M(q) \in \mathbb{S}_+^n$  is the generalized inertia matrix,  $J(q) \in \mathbb{R}^{6 \times n}$  is the contact Jacobian,  $\dot{v}_f = M(q)^{-1}(\tau - h(q, v)) \in \mathbb{R}^n$  is the free acceleration,  $h(q, v) \in \mathbb{R}^n$  is the vector of centrifugal, Coriolis and gravity forces and  $\tau \in \mathbb{R}^n$  is the vector of joint torques. As explained in [26] the generalized contact forces  $\lambda \in \mathbb{R}^6$  appear in the KKT conditions of (1) as the Lagrange multiplier associated with the rigid contact constraint (1b). The solution to the KKT system has the form of a constrained dynamical system with state  $x \triangleq (q, v)$  and control  $u \triangleq \tau$

$$\dot{x} = f(x, u) \quad (2a)$$

$$\lambda = g(x, u) \quad (2b)$$

### B. Classical MPC formulation

The following OCP is solved online

$$\begin{aligned} \min_{u(\cdot), x(\cdot)} \int_0^T l(x(t), u(t), \lambda(t), t) dt + l_T(x(T)) \quad (3) \\ \text{s.t. } \begin{cases} \dot{x}(t) = f(x(t), u(t)) \\ \lambda(t) = g(x(t), u(t)) \\ x(0) = x_0 \\ x(t) \in \mathcal{X} \\ u(t) \in \mathcal{U} \end{cases} \end{aligned}$$

where  $x_0$  is the initial state,  $l, l_T$  the running and terminal costs,  $\mathcal{X}, \mathcal{U}$  represent state and control constraints. We transcribe this OCP into the following NLP

$$\begin{aligned} \min_{X, U} \sum_{k=0}^{N-1} L_k(x_k, u_k, \lambda_k) + L_N(x_N) \quad (4) \\ \text{s.t. } \begin{cases} x_{k+1} = F(x_k, u_k) \\ \lambda_k = G(x_k, u_k) \\ x_0 = x_0 \end{cases} \end{aligned}$$

where  $L, L_N, F, G$  are the discretized cost and dynamics with sampling step  $\Delta t$ . The inequality constraints are handled by penalization in the cost. We use the FDDP algorithm implemented in [1] to solve (4). The algorithm outputs a locally optimal control policy of which we only apply the first element to the measured state  $\hat{x}$

$$\pi_0^*(\hat{x}) = u_0^* + K_0(\hat{x} - x_0^*) \quad (5)$$

The feedback gain  $K_0$  compensates small deviations around  $x_0^*$  and can serve to interpolate the control trajectory between two MPC updates as the control rate usually is higher than the update rate. For more details the reader can refer to [3].

### C. Rigid contacts accept no predictive feedback

In this context, explicit force feedback is not possible because contact forces and control torques are algebraically coupled. While  $\lambda$  is used as a *prediction*, it is possible to choose a control action  $u$ . But if  $\lambda$  is used as a measurement (like the state  $x$ ), it can be seen from (2b) that  $u$  is completely determined and cannot be chosen. In fact,  $\lambda$  appears as the output of a nonlinear system in with instantaneous transfer from the input, since the feedthrough  $\frac{\partial \lambda}{\partial u} = (JM^{-1}J^T)^{-1}JM^{-1}$  is non-zero. In this context, attempting to control  $\lambda$  with some policy  $u(\lambda)$  would create an algebraic loop as  $u$  and  $\lambda$  would influence each other instantaneously. On way to be able to break this loop is to introduce delay between force and control input as suggested in [21]. Another way is to model the contact as non-rigid as proposed e.g. in [22]. We follow [21] and introduce delay by modeling the actuation abstractly.

## III. FORCE FEEDBACK MPC

In this section, we formulate the augmented dynamics and OCP. The MPC scheme with force feedback is then introduced.

### A. Augmented dynamics

The actuation dynamics is modeled as low-pass filter (LPF). The command torque  $w$  is filtered into an actuation torque  $\tau$  according to

$$\dot{\tau} = \omega_c(w - \tau) \quad (6)$$

where  $\omega_c > 0$  is the cut-off angular frequency in  $\text{rad s}^{-1}$ , i.e.  $\omega_c = 2\pi f_c$  where  $f_c$  is the ordinary cut-off frequency in Hz. The augmented robot dynamics reads

$$\dot{y} := \begin{bmatrix} \dot{q} \\ \dot{v} \\ \dot{\tau} \end{bmatrix} = \begin{bmatrix} v \\ M^{-1}(\tau - h + J^T \lambda) \\ \omega_c(w - \tau) \end{bmatrix} \quad (7)$$

where  $y$  denotes the augmented state of the robot including the classical state  $x = (q, v)$  and the filtered torques  $\tau$ , and  $w$  is the new control input corresponding to the unfiltered torque, that is torque reference sent to the motor. The discretized actuation dynamics is an exponential moving average (first order infinite impulse response filter)

$$\tau_{k+1} = \alpha \tau_k + (1 - \alpha)w_k \quad (8)$$

where  $\alpha$  is related to cut-off frequency and discretization step  $\Delta t$ . Applying a zero-order hold in the Laplace domain on the CT filter

$$H(s) = \frac{\omega_c}{s + \omega_c} \quad (9)$$

leads to the exponential formula  $\alpha = e^{-2\pi f_c \Delta t}$ . Note that  $f_c \rightarrow \infty$  corresponds to the case where there is no filtering, i.e. the actuator is perfect and  $w$  goes entirely through ( $\alpha \rightarrow 0$ ). On the opposite,  $f_c \rightarrow 0$  corresponds to an infinite filtering effect, i.e.  $w$  is fully blocked ( $\alpha \rightarrow 1$ ).

While the actuation model introduced here is simple and abstract - a single scalar and a linear equation express the actuation bandwidth and delay - we will show that it led to good experimental results. In fact, this model is in some sense generic as it should capture the linear behavior of most other actuation models. Besides, the proposed method goes beyond this abstract model, and other actuation models could be considered, which we will investigate in the future.

### B. Augmented OCP

The new optimal control problem yields

$$\begin{aligned} \min_{w(\cdot), y(\cdot)} \quad & \int_0^T l(y(t), w(t), \lambda(t), t) dt + l_T(y(T)) \\ \text{s.t.} \quad & \begin{cases} \dot{x}(t) = f(x(t), \tau(t)) \\ \dot{\tau}(t) = \omega_c(w(t) - \tau(t)) \\ \lambda(t) = g(x(t), \tau(t)) \\ y(0) = y_0 \\ y(t) \in \mathcal{Y} \\ w(t) \in \mathcal{W} \end{cases} \end{aligned} \quad (10)$$

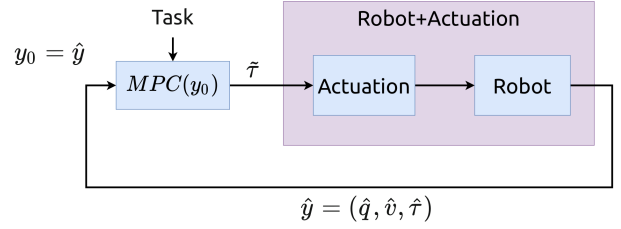


Fig. 1: MPC scheme with augmented dynamics. The measured torque is injected into the MPC and an interpolation of the optimal torque is sent to the actuators.

which is transcribed using the same method as for (4), using an explicit integration scheme for the augmented state

$$\min_{Y, W} \sum_{k=0}^{N-1} L_k(y_k, w_k, \lambda_k) + L_N(y_N) \quad (11)$$

$$\text{s.t.} \begin{cases} x_{k+1} = F(x_k, \tau_k) \\ \tau_{k+1} = \alpha \tau_k + (1 - \alpha)w_k \\ \lambda_k = G(x_k, \tau_k) \\ x_0 = \hat{x} \end{cases} \quad (12)$$

### C. MPC with force feedback

The numerical solver (FDDP [1]) computes optimal state and control sequences

$$\begin{aligned} Y^* &= \{y_0^*, \dots, y_N^*\} \\ W^* &= \{w_0^*, \dots, w_{N-1}^*\} \end{aligned}$$

as well as Riccati feedback gains  $K_k$  defining a locally optimal stabilizing policy around each shooting node

$$\pi_k(\hat{y}) = w_k^* + K_k(y_k - \hat{y}) \quad (13)$$

We recall that  $y = (q, v, \tau)$  gathers all joint measurements (see Figure 1). At each MPC cycle, the initial state of the OCP is set to the measured state  $y_0 \leftarrow \hat{y}$ . We could be tempted to apply the unfiltered optimal torque  $w_0^*$  but it seems safer to send a filtered torque that represents our "best" available reference (the torque we actually want the actuator to produce). In fact, if the real actuator is perfectly modeled by the LPF, it would not make any difference to send one or the other. But if the LPF model is not accurate, the optimizer could overestimate the filtering effect and pick an overly aggressive torque during the optimization.

Considering that we use a filtered torque as the actual control input, we need to be careful not to send the last measured torque as the current command to the robot. Indeed we have  $\tau_0^* \leftarrow \hat{\tau}$  at each MPC cycle (see Figure 1). Furthermore, it may not be satisfactory either to use the prediction  $\tau_1^*$  which lies too far in the future if the OCP sampling rate  $\Delta t$  is greater than the MPC update rate  $\Delta t_{MPC}$ . In fact it makes more sense to use a linear interpolation of the optimal filtered torque :

$$\tilde{\tau} = \tau_0^* + \epsilon(\tau_1^* - \tau_0^*) \quad (14)$$

where  $\epsilon = \frac{\Delta t_{MPC}}{\Delta t}$ .

As shown in [3], the Riccati gains computed by DDP correspond to the first-order derivatives of the optimal control with respect to the initial state, i.e.

$$K_0 = \frac{\partial w_0^*}{\partial y_0} \quad (15)$$

Let us split the Riccati gains between a position-velocity part and a torque part:  $K_0^x = \frac{\partial w_0^*}{\partial x_0}$  and  $K_0^\tau = \frac{\partial w_0^*}{\partial \tau_0}$ . We can express locally optimal feedback gains associated with control (14), namely  $\tilde{K} = [\tilde{K}^x, \tilde{K}^\tau]$  in terms of  $K_0^x, K_0^\tau$

$$\tilde{K} = \frac{\partial \tilde{\tau}}{\partial y_0} = \frac{\partial \tau_0^*}{\partial y_0} + \epsilon \left( \frac{\partial \tau_1^*}{\partial y_0} - \frac{\partial \tau_0^*}{\partial y_0} \right) \quad (16)$$

where  $\frac{\partial \tau_0^*}{\partial x} = 0$  and  $\frac{\partial \tau_0^*}{\partial \tau} = I$ . Using (8)

$$\tilde{K}^x = \epsilon(1 - \alpha)K_0^x \quad (17)$$

$$\tilde{K}^\tau = I + \epsilon(1 - \alpha)(K_0^\tau - I) \quad (18)$$

We can now interpolate (14) at the control rate by using the following policy

$$\tilde{\pi}(\hat{y}) = \tilde{\tau} + \tilde{K}^x(x_0 - \hat{x}) + \tilde{K}^\tau(\tau_0 - \hat{\tau}) \quad (19)$$

#### IV. SIMULATION ON A MANIPULATOR

In this section we compare the classical MPC and the force feedback MPC in simulation with the 7-DoF KUKA LBR iiwa. The task consists in drawing a circle on a flat horizontal surface (10 cm-radius,  $1.5 \text{ rad s}^{-1}$ ) while exerting a constant normal force (20 N). The contact model used here is a 1-D contact model that constrains the robot motion only along the direction normal to the contact plane.

##### A. Sanding task formulation

1) *Classical MPC*: The cost function used in OCP (3) is defined as

$$l(x, u, \lambda) = c_x \|A_x(x - x_0)\|^2 + c_u \|A_u(\tau - \tau_g(q))\|^2 + c_\lambda \|A_\lambda(\lambda - \bar{\lambda})\|^2 + c_R \|A_R(R(q) \ominus \bar{R})\|^2 + c_p \|A_p(p(q) - \bar{p}(t))\|^2 \quad (20)$$

where  $p, R$  are the end-effector frame 3D position and rotation respectively,  $\tau_g(q)$  is the gravity torque,  $A_x, A_u, A_p, A_\lambda, A_R$  are activation weight matrices,  $c_x, c_u, c_p, c_\lambda, c_R$  are scalar costs weights,  $\ominus$  represents the difference in  $\mathcal{SO}(3)$ .

2) *Force feedback MPC*: The cost function used in OCP (10) is defined as

$$l(y, w, \lambda) = c_y \|A_y(y - y_0)\|^2 + c_y^{lim} \|B_y(y)\|^2 + c_\lambda \|A_\lambda(\lambda - \bar{\lambda})\|^2 + c_R \|A_R(R(q) \ominus \bar{R})\|^2 + c_p \|A_p(p(q) - \bar{p}(t))\|^2 + c_w \|A_w w\|^2 + c_w^{lim} \|B_w(w)\|^2 \quad (21)$$

$A_y, A_w$  are activation weight matrices on the augmented state and unfiltered torque respectively,  $c_y, c_y^{lim}, c_w, c_w^{lim}$  are scalar costs weights,  $B_y, B_w$  are weighted quadratic barriers.

	Classical MPC	Force feedback MPC
Avg. abs. position err. (m)	0.026	<b>0.021</b>
Avg. abs. force err. (N)	9.45	<b>2.88</b>
Max. abs. force (N)	102.4	<b>34.01</b>
Time not in contact (%)	5.58	<b>0.00</b>

TABLE I: Performance on simulated sanding task with KUKA iiwa - imperfect actuation, perfect contact model

##### B. Imperfect actuation and perfect contact model

We used PyBullet as a simulation environment which doesn't allow the simulation of low-level actuation. In order to simulate a sensible torque measurement, we perturb the desired torque by scaling, noise and delays as follow

$$\hat{\tau}(t) = a\tilde{\tau}(t - \delta_{OCP}) + b + \eta_\tau \quad (22a)$$

$$\hat{x}(t) = \hat{x}(t - \delta_{SIM}) + \eta_x \quad (22b)$$

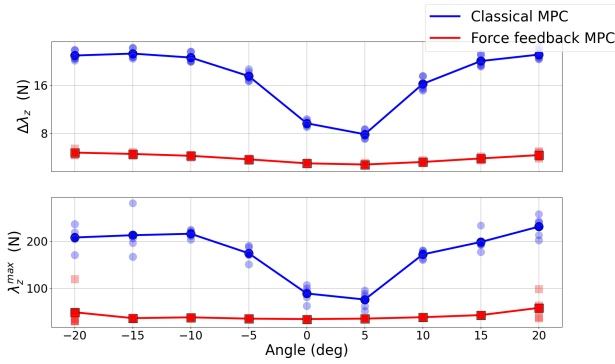
where  $\eta_x \sim \mathcal{N}(0, \sigma_x), \eta_\tau \sim \mathcal{N}(0, \sigma_\tau)$  are Gaussian noise signals capturing sensing and actuation noise,  $a, b$  are scaling coefficients uniformly drawn from  $[\underline{a}, \bar{a}] \times [\underline{b}, \bar{b}]$ ,  $t$  is the simulation time,  $\delta_{OCP}$  is a delay due to the computation time of the MPC and  $\delta_{SIM}$  is a delay introduced by e.g. transmission or sensing. We set  $\underline{a} = 0.95, \bar{a} = 1.05, \underline{b} = -0.1 \text{ N m}, \bar{b} = 0.1 \text{ N m}, \delta_{OCP} = 1 \text{ ms}, \delta_{SIM} = 2$  simulation cycles.

We tuned each controller separately to achieve the best performance, to the best of our ability. We found that MPC parameters were very dependent on the controller. Eventually we choose the classical MPC frequency as 500 Hz with  $\Delta t = 20 \text{ ms}$  and  $N = 50$  nodes (1 s horizon), while the force feedback MPC frequency was set to 250 Hz with  $\Delta t = 10 \text{ ms}$  and  $N = 100$  nodes (1 s horizon), and the cutoff frequency is set to  $f_c = 5 \text{ Hz}$ . In both cases, the simulation frequency is 1 kHz and we allow a maximum of 5 DDP iterations at each MPC update. The policies used are (5), (19).

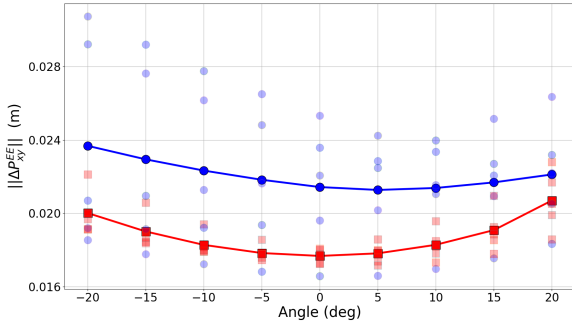
We choose as performance metric the average position tracking error, the average normal force error, the peak normal force and the percentage of the simulation time spent not in contact, as reported in Table I. We observe a higher tracking performance in position and force in the case of the force feedback MPC. Also the proposed controller results in a lower maximum force, and maintains contact with the environment throughout the task.

##### C. Imperfect actuation and disturbed contact model

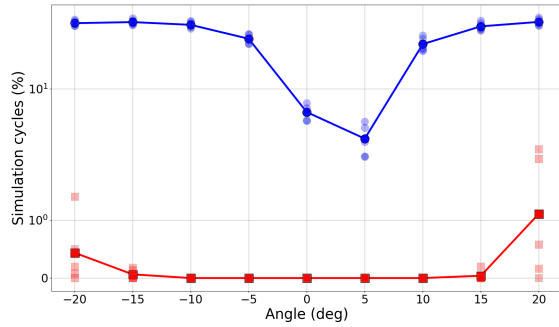
In order to assess the benefit of torque feedback, we compare the performances when the contact model is disturbed and under different actuation models. The contact surface is tilted about the  $y$ -axis by an angle ranging from  $-20^\circ$  to  $+20^\circ$ , and for 5 different actuation models (i.e. random pairs of  $(a, b)$  in (22)). The results are shown in Figure 2. The position and force tracking performances are better with the torque feedback MPC. Moreover, the maximum force is lower than with classical MPC (about 50 N vs 220 N for the  $20^\circ$  angle), which can be explained by the more frequent contact breaking (about 40% of the simulation time is spent not in contact without torque feedback vs less than 10%



(a) Average absolute normal force error (top) and maximum normal force magnitude (bottom)



(b) 2-norm of the position tracking error in  $(x, y)$



(c) Percentage of the time spent not in contact

Fig. 2: Performance on the sanding task of classical MPC (blue) and force feedback MPC (red) for a range of tilting angles averaged over several actuation models. The proposed controller achieves higher performance in position and force.

for the proposed approach). We also observed a smaller standard deviation in the position tracking performance over the actuation models (i.e. when tilting angle is fixed), which suggests our control scheme is less sensitive to actuation model uncertainty.

## V. EXPERIMENT ON A HUMANOID

We propose now to validate the approach with hardware experiments of the sanding task on the TALOS humanoid robot [12] with a perturbed contact model.

### A. Experimental setup

We use the same ROS-based real-time control architecture as [3]. The MPC runs at 100 Hz with 3 DDP iterations. The OCP is solved using the Crocoddyl library [1] and rigid-body computations are done using Pinocchio [27]. The lower level is controlled at 2 kHz. The robot model is reduced to a 6-dimensional model including the torso and right arm. All other joints are position-controlled in a fixed posture. A soft material is taped to the robot forearm to damp impacts. The joint torque measurements are filtered with a moving average filter.

At the time of the experiments, FT sensors in the wrist were not available so we used joint torque measurements to estimate the contact force  $\hat{\lambda}$

$$\hat{\lambda} = (\hat{J}\hat{M}^{-1}\hat{J}^T)^{-1}(\hat{\gamma} - \hat{J}\hat{M}^{-1}(\hat{\tau} - \hat{h})) \quad (23)$$

where the spatial acceleration  $\hat{\gamma}$  is neglected. This estimation is not accurate enough to legitimate a thorough quantitative analysis of performance as proposed in the previous section. Therefore we leave this for future work and restrict ourselves to a qualitative discussion for now, which is nevertheless relevant for this proof of concept.

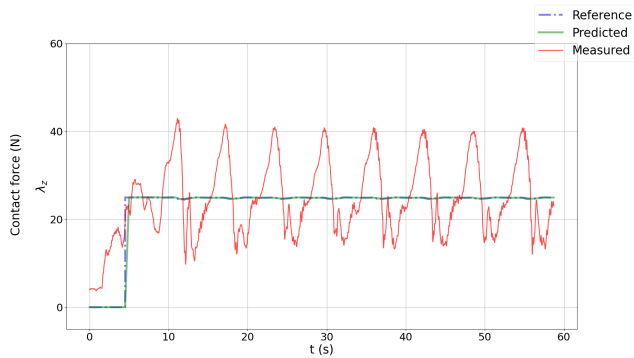
### B. Sanding task

The general cost function used for both task is similar the cost functions (20) and (21) used in simulation, the only difference being the introduction of time phases: the main task is divided into sub-tasks with fixed time duration (posture regularization phase, table reaching phase, contact stabilizing phase, circle tracking phase, etc.). Hence the cost weights  $c_p, A_p, c_R, c_\lambda$  and the rigid contact constraint activeness are made time-varying. Also for each phase switch, the OCP is updated progressively starting from the last node in the horizon - it empirically led to more stability than updating all the nodes at once.

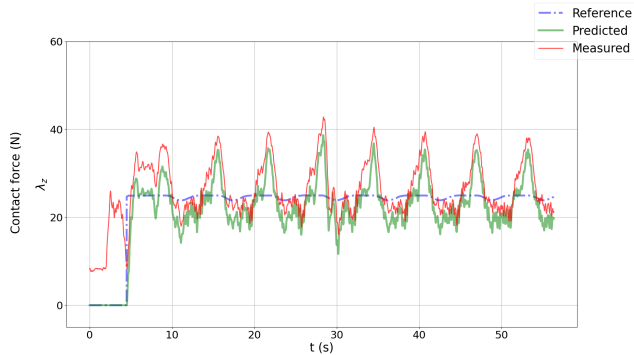
### C. Results

The robot must draw 7 cm-radius circles at 1 rad s<sup>-1</sup> on a table. The contact model is assumed to be perfect by the controller (perfectly horizontal table) but the real table is tilted by an unknown angle. As in simulation, each controller was tuned separately to the best of our abilities. We didn't use the torque feedback gains of equation (19) as TALOS already has a low-level torque control loop with fixed feedback gains.

As seen in the supplementary video, the motions generated by the force feedback MPC are smoother than with the classical controller. Our controller maintains contact throughout the whole task while the classical controller often breaks contact, creates more vibrations with the table. As seen on Fig. 3a,3b, the force variations have a greater amplitude in the case of classical MPC. The predictions of the classical MPC (green) are perfect, which explains the large mismatch between actual and predicted force. Our controller predicts forces that match more closely the actual ones thanks to its force feedback, which explains the observed improved performance. Fig. 4a,4b show that the circle tracking is improved with the force feedback MPC, while the classical



(a) Classical MPC



(b) Force feedback MPC

Fig. 3: Normal force during the sanding task with perturbed contact model. The robot maintains contact despite perturbations with force feedback MPC (bottom plot) whereas it breaks contact several times with classical MPC (top plot).

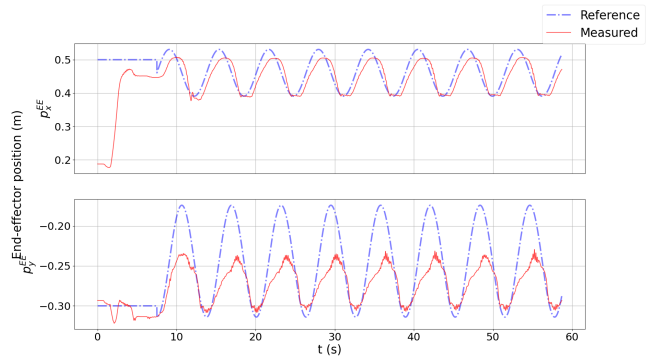
controller struggles to track the circle properly because of frequent contact breaking. We couldn't achieve a higher accuracy with the classical MPC as increasing the position gains led to instability.

## VI. CONCLUSION

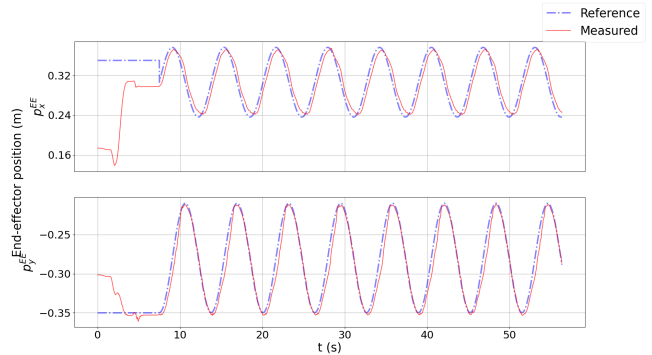
### A. Discussion

1) *Contact force measurements*: We have used joint torque sensors as feedback but we further intend to use Cartesian force measurements as proposed in [22] and retrieve the measured torques as  $\tau = M\dot{v} + h - J^T\lambda$ . One interesting challenge to overcome is the excitation of the robot structural dynamics and the apparition of coupled instabilities due to the non-collocation of sensor and actuators. This will pave the way for a more general inclusion of force sensing in MPC.

2) *Riccati feedback gains*: TALOS has a low-level torque control loop with fixed feedback gains. But as Section III-C shows, our approach naturally provides locally optimal feedback gains on the torques. Future work therefore includes an empirical evaluation with these gains to replace other low-level controllers and investigate how this improves tracking performance.



(a) Classical MPC



(b) Force feedback MPC

Fig. 4: The position tracking is more accurate with the force feedback MPC (bottom) than with the classical controller (top) thanks to proper control of contact interaction through force and motion feedback.

3) *Improved actuation model*: A simple LPF already shows performance improvement compared to classical MPC, therefore we could expect even better results with a more sophisticated actuation model.

### B. Controlling physical interaction

We draw here a parallel with the existing interaction control literature. During physical interaction, motion and forces are coupled through the exchange of mechanical work [28]. An important consequence is that controlling an interaction requires to regulate force and motion dynamically. When the mechanical energy is small enough this coupling can be neglected but in many interesting cases it cannot (e.g. deforming an object, scraping, etc.).

With this in mind, impedance control [29] aims at modulating the robot reaction to physical interaction by regulating the dynamic relation between force and motion. While high impedance robots have been historically predominant, in part for technological reasons, low impedance is required for many tasks [30]. Explicit force feedback is an efficient way of doing so because it scales the apparent inertia of the robot [31]. But it can render the system non-passive and prone to coupled instabilities [28], [32]. Therefore passivity is classically used as a sufficient criteria for stability [28], [33]–[35]. This condition limits the magnitude of the force

gains, and thereby the achievable bandwidth [36], [37] which can be overly conservative: some tasks may require precisely behaviors that are stable but not necessarily passive (e.g. deforming an object, grinding).

This observation raises the following question: what particular impedance should a robot realize in order to execute a given task? Note that impedance control is agnostic to this "inverse" impedance problem. In practice, designing a suitable impedance is done empirically through expert fine-tuning - which is tedious and vulnerable to uncertainties. While we are not claiming to provide a formal solution to this problem, we like to think of optimal control as a way to automatize control gains synthesis, e.g. by relating impedance modulation during contacts to a trade-off between disturbance rejection and measurement uncertainty [38]. This suggests that incorporating force in the optimization may result in an optimized impedance, trading off motion and force performance to achieve a higher-level objective.

### C. Summary and future prospects

We proposed a novel paradigm to exploit force measurements in MPC. By introducing an actuation dynamics between desired torques and actual torques, we were able to allow force feedback in optimal control which led to a significant improvement in performances for contact tasks compared to classical state-based MPC. We demonstrated the benefit of this new approach through simulations and hardware experiments. This proof of concept confirms our intuition that MPC should tend toward multimodality by including more sensors. We intend to continue investigating this subject in the future.

### REFERENCES

- [1] C. Mastalli, R. Budhiraja, W. Merkt, G. Saurel, B. Hammoud, M. Naveau, J. Carpentier, L. Righetti, S. Vijayakumar, and N. Mansard, "Crocodyl: An efficient and versatile framework for multi-contact optimal control," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2020.
- [2] S. Kleff, A. Meduri, R. Budhiraja, N. Mansard, and L. Righetti, "High-frequency nonlinear model predictive control of a manipulator," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.
- [3] E. Dantec, M. Taïx, and N. Mansard, "First order approximation of model predictive control solutions for high frequency feedback," *IEEE Robotics and Automation Letters (RAL)*, vol. 7, no. 2, 2022.
- [4] D. Kim, J. Di Carlo, B. Katz, G. Bledt, and S. Kim, "Highly Dynamic Quadruped Locomotion via Whole-Body Impulse Control and Model Predictive Control," *arXiv:1909.06586*, 2019.
- [5] M. Dominici and R. P. D. Cortesão, "Model predictive control architectures with force feedback for robotic-assisted beating heart surgery," *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.
- [6] S. Husmann, S. Stemmler, S. Hähnel, S. Vogelgesang, D. Abel, and T. Bergs, "Model Predictive Force Control in Grinding based on a Lightweight Robot," *IFAC-PapersOnLine*, vol. 52, no. 13, 2019.
- [7] R. Grandia, F. Farshidian, R. Ranftl, and M. Hutter, "Feedback MPC for Torque-Controlled Legged Robots," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.
- [8] F. Farshidian, E. Jelavic, A. Satapathy, M. Gifftaler, and J. Buchli, "Real-time motion planning of legged robots: A model predictive control approach," in *IEEE-RAS International Conference on Humanoid Robotics (Humanoids)*, 2017.
- [9] N. Hogan, "Impedance control: An approach to manipulation," in *American Control Conference*, 1984, pp. 304–313.
- [10] Z. Kappasov, J. A. Corrales, and V. Perdereau, "Tactile sensing in dexterous robot hands - Review," *Robotics and Autonomous Systems*, vol. 74, 2015.
- [11] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, K. Bodie, P. Fankhauser, M. Bloesch, R. Diethelm, S. Bachmann, A. Melzer, and M. Hoepflinger, "ANYmal - A highly mobile and dynamic quadrupedal robot," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.
- [12] O. Stasse, T. Flayols, R. Budhiraja, K. Giraud-Esclasse, J. Carpentier, J. Mirabel, A. Del Prete, P. Souères, N. Mansard, F. Lamiraux, J.-P. Laumond, L. Marchionni, H. Tome, and F. Ferro, "Talos: A new humanoid research platform targeted for industrial applications," in *IEEE-RAS International Conference on Humanoid Robotics (Humanoids)*, 2017.
- [13] V. Chawda and G. Niemeyer, "Toward torque control of a KUKA LBR IIWA for physical human-robot interaction," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.
- [14] F. Farshidian, M. Neunert, A. W. Winkler, G. Rey, and J. Buchli, "An efficient optimal planning and control framework for quadrupedal locomotion," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.
- [15] A. Meduri, P. Shah, J. Viereck, M. Khadiv, I. Havoutis, and L. Righetti, "Biconmp: A nonlinear model predictive control framework for whole body motion planning," *arXiv:2201.07601v1*, 2022.
- [16] J. Koenemann, A. Del Prete, Y. Tassa, E. Todorov, O. Stasse, M. Bennewitz, and N. Mansard, "Whole-body model-predictive control applied to the hrp-2 humanoid," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015.
- [17] M. Posa, S. Kuindersma, and R. Tedrake, "Optimization and stabilization of trajectories for constrained dynamical systems," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016.
- [18] S. L. Cleac'h, T. Howell, M. Schwager, and Z. Manchester, "Fast contact-implicit model-predictive control," *arXiv:2107.05616*, 2021.
- [19] M. Neunert, M. Stäuble, M. Gifftaler, C. D. Bellicoso, J. Carius, C. Gehring, M. Hutter, and J. Buchli, "Whole-body nonlinear model predictive control through contacts for quadrupeds," *IEEE Robotics and Automation Letters (RAL)*, vol. 3, no. 3, 2018.
- [20] R. Grandia, F. Farshidian, A. Dosovitskiy, R. Ranftl, and M. Hutter, "Frequency-aware model predictive control," *IEEE Robotics and Automation Letters (RAL)*, vol. 4, no. 2, 2019.
- [21] A. Aydinoglu, P. Sieg, V. M. Preciado, and M. Posa, "Stabilization of complementarity systems via contact-aware controllers," *IEEE Transactions on Robotics*, 2021.
- [22] J. Matschek, J. Bethge, P. Zometta, and R. Findeisen, "Force Feedback and Path Following using Predictive Control: Concept and Application to a Lightweight Robot," *IFAC-PapersOnLine*, vol. 50, no. 1, 2017.
- [23] M. Bednarczyk, H. Omran, and B. Bayle, "Model Predictive Impedance Control," *Proceedings - IEEE International Conference on Robotics and Automation*, no. 1, 2020.
- [24] M. V. Minniti, R. Grandia, K. Fähr, F. Farshidian, and M. Hutter, "Model predictive robot-environment interaction control for mobile manipulation tasks," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.
- [25] F. E. Udwardia and R. E. Kalaba, "A new perspective on constrained motion," *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, vol. 439, no. 1906, pp. 407–410, 1992.
- [26] R. Budhiraja, J. Carpentier, C. Mastalli, and N. Mansard, "Differential dynamic programming for multi-phase rigid contact dynamics," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2018.
- [27] J. Carpentier, G. Saurel, G. Buondonno, J. Mirabel, F. Lamiraux, O. Stasse, and N. Mansard, "The Pinocchio C++ library: A fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives," in *IEEE/SICE International Symposium on System Integration*, 2019.
- [28] E. D. Fasse and N. Hogan, "Control of Physical Contact and Dynamic Interaction," in *Robotics Research*, G. Giral and G. Hirzinger, Eds. London: Springer London, 1996, pp. 28–38.
- [29] N. Hogan, "Impedance Control Part1-3," *Transaction of the ASME, Journal of Dynamic Systems, Measurement, and Control*, vol. 107, no. March 1985, pp. 1–24, 1985.
- [30] G. A. Pratt, "Low Impedance Walking Robots," *Integrative and Comparative Biology*, vol. 42, no. 1, 2002.



- [31] R. Volpe and P. Khosla, "The equivalence of second order impedance control and proportional gain explicit force control: Theory and experiments," *Lecture Notes in Control and Information Sciences*, vol. 190, pp. 1–24, 1993.
- [32] D. Whitney, "Historical perspective and state of the art in robot force control," in *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, 1985, pp. 262–268.
- [33] J. E. Colgate, "The Control of Dynamically Interacting Systems," Ph.D. dissertation, Massachusetts Institute of Technology (MIT), 1988.
- [34] A. Albu-Schäffer, C. Ott, and G. Hirzinger, "A unified passivity-based control framework for position, torque and impedance control of flexible joint robots," *International Journal of Robotics Research (IJRR)*, vol. 26, no. 1, jan 2007.
- [35] M. Focchi, G. A. Medrano-Cerda, T. B. Cunha, M. Frigerio, C. Semini, J. Buchli, and D. G. Caldwell, "Robot impedance control and passivity analysis with inner torque and velocity feedback loops," *Control Theory and Technology*, vol. 14, 2014.
- [36] S. Eppinger and W. Seering, "Understanding bandwidth limitations in robot force control," in *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 4, 1987, pp. 904–909.
- [37] M. Dohring and W. Newman, "The passivity of natural admittance control implementations," *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 3, 2003.
- [38] B. Hammoud, M. Khadiv, and L. Righetti, "Impedance optimization for uncertain contact interactions through risk sensitive optimal control," *IEEE Robotics and Automation Letters (RAL)*, vol. 6, no. 3, 2021.