Abstract—This paper is concerned with a novel computed-torque technique for trajectory tracking, which enhances robustness by replacing some of the system nonlinearities in the inner-loop feedback by the signals of the desired trajectory. This substitution leads to a nonlinear outer-loop feedback in the form of parallel distributed compensation, whose gains are solved in terms of linear matrix inequalities derived from the direct Lyapunov method and exact convex modelling of nonlinearities. The proposal makes use of a recently appeared factorization which explicitly allows extracting the error signal in the difference of identical expressions that depend on a different set of variables. Simulations suggest that the magnitude of the computed-torque signal can be significantly reduced when compared with ordinary schemes while augmenting robustness by employing user-generated signals instead of measurements where appropriate.

Index Terms—Computed-Torque Control, Tracking, Parallel Distributed Compensation, Linear Matrix Inequality.

I. INTRODUCTION

Trajectory tracking consists in designing a feedback controller such that the output $y(t)$ of a nonlinear plant asymptotically tracks a reference signal $r(t)$ while preserving boundedness of the states $x(t)$, i.e., $\lim_{t \to \infty} y(t) = r(t)$, $\|x(t)\| < \infty$ [1]. A special case of trajectory tracking arises in the context of rigid robot manipulators with Lagrange-Euler dynamics as these are 2nd-order systems with respect to the joint vector $q(t) \in \mathbb{R}^n$ for which the well-known computed-torque technique can be applied [2]. This technique guarantees $\lim_{t \to \infty} q(t) = q_d(t)$, $\|q(t)\|$, $\|\dot{q}(t)\| < \infty$ for any feasible trajectory $q_d(t)$ in the joint space [3], [4], by means of two signals: a nonlinear inner-loop feedback which is based on model inversion to cancel out the system nonlinearities [5] and a linear outer-loop feedback in the form of $n$ decoupled PD controllers that drive the error system to zero [6]. A variety of real-time applications of this popular approach can be found in the literature [7], [8], [9].

As any other model-based technique [10], computed-torque critically depends on the precise knowledge of the plant model and parameters; this is particularly crucial in model inversion [11] as little discrepancies may lead to instability [12]; thus, a variety of robust modifications have been implemented to maintain the advantages of computed-torque techniques, namely, adding integral terms [13], perturbation analysis (both structured and unstructured [14]), robustness analysis [15] and bounded guaranteed errors (uniform and ultimate [16]). In this work, robustness is sought by relaxing the dependency of the inner-loop feedback on the plant signals, as they are usually noisy or perturbed; moreover, these imprecisions are amplified when the signal appears in nonlinear operations as those in the inertia matrix $M(q)$, Coriolis vector $V(q, \dot{q})$, and gravity vector $G(q)$: it is shown that replacing $q$ and $\dot{q}$ by $q_d$ and $\dot{q}_d$, respectively, in the latter expressions, leads to a significant improvement of the robustness properties of the scheme as well as to reduced control effort, an idea close to the spirit of [11].

The proposal comes at a price: using $q_d$ and $\dot{q}_d$ instead of $q$ and $\dot{q}$ in $V(q, \dot{q})$ and $G(q)$ produces a nonlinear error system which requires to be stabilized by the outer-loop feedback; thus, linear techniques no longer apply. Since the expressions involved are all bounded in a compact set of the joint space, an exact convex rewriting of the error system is possible via a recently appeared factorization [17] and the sector nonlinearity approach [18]; this algebraic rearrangement allows applying the direct Lyapunov method to derive conditions for controller design in the form of linear matrix inequalities (LMIs) [19], which are solved in polynomial time via commercially available software [20], [21]. The control law is a nonlinear one better known as parallel distributed compensation (PDC) [22], which employs the system nonlinearities to drive the tracking error to zero [23]; thanks to the LMI nature of its design, input saturation limits, decay rate, and other performance measures can be easily incorporated [24].

This work is organized as follows: section II introduces the ordinary computed-torque technique and the methodology for exact factorization of an error signal $e = q - q_d$ in expressions of the form $f(q) - f(q_d)$; section III presents the new computed-torque control law showing how it induces a nonlinear error system; a detailed Lyapunov-based analysis is performed to derive the LMI conditions for designing of the outer-loop PDC feedback that drives the error system to zero; as a way of illustration, in section IV the proposed methodology is put at test in a two-link planar elbow arm; simulation results are contrasted with those of ordinary computed-torque; concluding remarks are given in section V.
II. PRELIMINARIES

For comparison purposes as well as background, the standard computed-torque technique is now presented; it is followed by an introduction to the factorization recently appeared in [17], which will be instrumental for some of the developments in this paper.

A. The standard computed-torque technique

A fully actuated robotic manipulator consisting of rigid beams along with translational and rotational joints, grouped in a generalized coordinate vector \( q \in \mathbb{R}^n \), is always amenable to the following Lagrange-Euler form:

\[
M(q)\ddot{q}(t) + N(q, \dot{q}) = \tau(t),
\]

(1)

where \( M(q) \in \mathbb{R}^{n \times n} \) is the inertia matrix, \( N(q, \dot{q}) \in \mathbb{R}^n \) is the sum of the Coriolis \( V(q, \dot{q}) \), friction \( F(q, \dot{q}) \), and gravity \( G(q) \) vectors, and \( \tau \in \mathbb{R}^n \) is the generalized torque vector.

Given a desired trajectory \( q_d(t) \) whose first and second time derivatives are available, the tracking error is defined as

\[
e(t) = q_d(t) - q(t),
\]

(2)

which after differentiation and substitution of (1) leads to

\[
\begin{bmatrix}
\dot{e}(t) \\
\ddot{e}(t)
\end{bmatrix} =
\begin{bmatrix}
0 & I \\
0 & 0
\end{bmatrix}
\begin{bmatrix}
e(t) \\
\dot{e}(t)
\end{bmatrix} +
\begin{bmatrix}
0 \\
F
\end{bmatrix} u(t),
\]

(3)

for which linear control techniques such as pole placement [25] or LMI formulations [24] can be applied to guarantee \( \lim_{t \to \infty} e(t) = 0 \), which is equivalent to performing trajectory tracking. Since \( u \) is a fictitious input also known as the outer-loop feedback, this signal is transferred to \( \tau \) which, in addition, incorporates the inverse dynamics of the system, i.e.,

\[
\tau(t) = M(q)\ddot{q}_d(t) - u(t)) + N(q, \dot{q}).
\]

(4)

The control signal \( \tau \) is known as the inner-loop feedback; its dependency on the accuracy of the model is clear from the direct use of \( M(q) \) and \( N(q, \dot{q}) \). The whole scheme is shown in Fig. 1 with \( N(q, \dot{q}) \) and \( u \) in black; our proposal will substitute these terms by \( N(q_d, \dot{q}_d) \) and \( M^{-1}(q) u \), respectively (in red).

B. Factorization of error signals

In the context of trajectory tracking as well as observer design, we are often faced with the task of factorizing the error signal (2) from expressions of the form \( f(q_d) - f(q) \), with \( f(\cdot) : \mathbb{R}^n \to \mathbb{R}^n \). Though the differential mean-value theorem (DMVT) guarantees that this is always possible, no explicit formula for the factorization is available in the literature. Recently, in [17], it has been proven that expressions having a convergent Taylor series can be explicitly written as:

\[
f(q_d) - f(q) = F(q_d, q) (q_d - q) = F(q_d, q) e,
\]

(5)

where \( F : \mathbb{R}^{2n} \to \mathbb{R}^{n \times n} \) is a function whose entries are multivariate polynomials of the entries of \( q \) and \( q_d \).

To achieve such factorization, a simplified guideline is the following:

1) Any term containing multivariate polynomials \( p(\cdot) \) of the form \( p(q_d) - p(q) \) can be put into the form \( pT(q_d, q)e(t) \) by adding and subtracting all the monomials whose powers are combinations of those involved in the multivariate polynomials \( p(q_d) \) and \( p(q) \); the resulting \( p(\cdot) \in \mathbb{R}^{n \times 1} \) has multivariate polynomials as entries.

2) Any term containing non-polynomial expressions can be arbitrarily approximated by a polynomial of arbitrary degree via its Taylor series, after which it can be treated as before.

To illustrate this procedure, consider \( q = [q_1 \ q_2]^T \) along with expression \( f(q) = [q_1 \ q_2]^T \). It is clear that:

\[
q_{d1}q_{d2} - q_1q_2
\]

\[
= aq_{d1}(q_{d2} - q_2) + aq_2(q_{d1} - q_1) + (1-a)q_{d2}(q_{d1} - q_1) + (1-a)q_1(q_{d2} - q_2)
\]

\[
= aq_2 + (1-a)q_{d2} aq_{d1} + (1-a)q_1 \begin{bmatrix} e_1 \\ e_2 \end{bmatrix},
\]

for any \( a \in \mathbb{R} \), which means

\[
f(q_d) - f(q) = \begin{bmatrix} aq_2 + (1-a)q_{d2} & aq_{d1} + (1-a)q_1 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix},
\]

where \( F(q_d, q) \) has been highlighted according to (5), \( e_1 = q_{d1} - q_1 \), and \( e_2 = q_{d2} - q_2 \).

Needless to say, this factorization is key for Lyapunov-based stabilization of nonlinear systems, especially in the framework of convex optimization and quasi-LPV systems, also known as exact Takagi-Sugeno models; the interested reader is referred to [23], [26]. All these tools will be employed in the next section as substituting \( N(q, \dot{q}) \) by \( N(q_d, \dot{q}_d) \) leads to nonlinear expressions that can be arranged in a nonlinear error system replacing (3) by means of the factorization just presented.

III. MAIN RESULTS

Let us consider the following variation of the inner-loop feedback of the computed-torque control law

\[
\tau(t) = M(q) (\ddot{q}_d(t) - M^{-1}(q) u(t)) + N(q_d, \dot{q}_d),
\]

(6)
where the introduction of \( N(q_d, \dot{q}_d) \) alleviates its dependency on plant signals and pre-multiplying \( u \) by \( M^{-1}(q) \) intends to facilitate the factorization of the error signal at the rightmost side of the error system.

Indeed, from double differentiation of (2), inclusion of the robot dynamics (1), and substitution of the inner-loop feedback (6), we have that (omitting arguments where appropriate)

\[
\dot{e}(t) = \dot{q}_d(t) - M^{-1}(q)(\tau(t) - N(q, \dot{q})) = \dot{q}_d - M^{-1}(q)([M(q)(\dot{q}_d - M^{-1}(q)u) + N(q_d, \dot{q}_d)] - N(q, \dot{q})) = \dot{q}_d - \dot{q}_d + M^{-1}(q)u - M^{-1}(q)N(q_d, \dot{q}_d) - N(q, \dot{q}) = M^{-1}(q)u - M^{-1}(q)(N(q_d, \dot{q}_d) - N(q, \dot{q})).
\]

(7)

The term \( N(q_d, \dot{q}_d) - N(q, \dot{q}) \) can be treated as described in section II-B because it is a difference of similar expressions depending on a different set of variables, i.e.,

\[
N(q_d, \dot{q}_d) - N(q, \dot{q}) = [N^1(q, q_d, \dot{q}_d)] N^2(q, \dot{q}_d, \dot{q})[e] = [e]
\]

where \( N^1 : \mathbb{R}^{4n} \rightarrow \mathbb{R}^{n \times n}, N^2 : \mathbb{R}^{4n} \rightarrow \mathbb{R}^{n \times n} \) are nonlinear matrix functions that may depend on \( q, q_d, \dot{q}_d, \) and \( \dot{q}_d \).

Thus, the error dynamics can be described as:

\[
\frac{d}{dt}[e] = \begin{bmatrix} 0 & I \\ -M^{-1}(q)N(\dot{q}) & -M^{-1}(q)N^2(\dot{q}) \end{bmatrix}[e] + \begin{bmatrix} 0 \\ M^{-1}(q)u \end{bmatrix},
\]

(8)

where \( \ddot{q} = [q^T \quad \dot{q}_d^T \quad \dot{q}_d^T] \).

Since \( M(q) \) is an inertia matrix, it is positive-definite, i.e., \( M(q) > 0 \), which implies \( \text{diag}\{I, M^{-1}(q)\} > 0 \); therefore, stabilization of the error system above depends solely on stabilizing the pair

\[
\begin{bmatrix} 0 & -N^2(q, q_d, \dot{q}_d, \dot{q}) \\ -N^1(q, q_d, \dot{q}_d) & -N^2(q, \dot{q}_d, \dot{q}) \end{bmatrix}, \quad \begin{bmatrix} 0 \\ I \end{bmatrix},
\]

(9)

by means of \( u \), which requires nonlinear techniques as the entries of \( N^1 \) and \( N^2 \) might be nonlinear.

Since \( q, q_d, \dot{q}, \) and \( \dot{q}_d \) are physically bounded, let say in a compact set \( \Omega \subset \mathbb{R}^{4n} \), it is reasonable to assume the entries of \( N^1 \) and \( N^2 \) can be decomposed into bounded expressions; too; let us denote these expressions as \( z_1(\ddot{q}) \in [z_1^0, z_1^1], z_2(\ddot{q}) \in [z_2^0, z_2^1], \ldots, z_p(\ddot{q}) \in [z_p^0, z_p^1] \). It can be algebraically verified that, defining

\[
w_i^0(\ddot{q}) = \frac{z_i^1 - z_i(\ddot{q})}{z_i^1 - z_i^0}, \quad w_i^1(\ddot{q}) = \frac{z_i(\ddot{q}) - z_i^0}{z_i^1 - z_i^0},
\]

(10)

the non-constant terms can be written as:

\[
z_i(\ddot{q}) = w_i^0(\ddot{q})z_i^0 + w_i^1(\ddot{q})z_i^1,
\]

(11)

where \( \forall \ddot{q} \in \Omega, w_i^0(\ddot{q}) + w_i^1(\ddot{q}) = 1, i \in \{1, 2, \ldots, p\}, w_i^j(\ddot{q}) \in [0, 1] \).

Due to convexity of functions \( w_i^j(\ddot{q}) \), it is possible to stack convex sums at the leftmost side of expressions; therefore, the first element of the pair (9) can be written as (omitting arguments):

\[
\sum_{i_0=0}^1 \sum_{i_p=0}^1 \sum_{i_p=0}^1 w_{i_0}^1 w_{i_1}^2 \cdots w_{i_p}^p \left[ -N^1_{i_0, i_1, \ldots, i_p} - N^2_{i_1, i_2, \ldots, i_p} \right],
\]

where

\[
N^1_{i_1, i_2, \ldots, i_p} = N^1(\ddot{q})w_{i_0}^1 w_{i_1}^2 \cdots w_{i_p}^p = 1 \quad \text{(12)}
\]

\[
N^2_{i_1, i_2, \ldots, i_p} = N^2(\ddot{q})w_{i_0}^1 w_{i_1}^2 \cdots w_{i_p}^p = 1 \quad \text{(13)}
\]

If notation \( \mathbb{P} \equiv \{0, 1\}, i \equiv (i_1, i_2, \ldots, i_p), w_i(\ddot{q}) = w_{i_0}^1(\ddot{q})w_{i_1}^2(\ddot{q}) \cdots w_{i_p}^p(\ddot{q}) \), is introduced, the pair (9) can be further simplified to

\[
\begin{bmatrix} 0 & -N^1(\ddot{q}) & -N^2(\ddot{q}) \end{bmatrix} = \sum_{i \in \mathbb{P}^p} w_i(\ddot{q})[0 \quad I \quad -I],
\]

(14)

Stabilization of the pair (9) (which implies stability of (8)) can be achieved by using the following nonlinear control law

\[
u(t) = F(\ddot{q}) \begin{bmatrix} e(t) \\ \dot{e}(t) \end{bmatrix} = \sum_{j \in \mathbb{P}^p} w_j(\ddot{q})F_j \begin{bmatrix} e(t) \\ \dot{e}(t) \end{bmatrix},
\]

(15)

where \( F_j \in \mathbb{R}^{2n \times 2n} \) are gains to be found via the following result, which is a direct generalization of the ordinary quasi-LPV stabilization in [23]:

**Theorem 1:** The origin of the nonlinear error system (8) under the control law (15), where the first element of the pair (9) can be written as (14) on the induced compact set \( \Omega_e \equiv \{(e, \dot{e}) : \ddot{q} \in \Omega\} \) such that \( (e, \dot{e}) = (0, 0) \in \Omega_e \) is asymptotically stable if there exist matrices \( X \in \mathbb{R}^{2n \times 2n} \), \( X = X^T > 0 \) and \( K_j \in \mathbb{R}^{2n \times 2n} \) such that LMI

\[
\sum_{i \in \mathbb{P}^p} \text{He} \begin{bmatrix} 0 & -I \\ -N_i & -I \end{bmatrix} X + \begin{bmatrix} 0 \\ I \end{bmatrix} K_j \end{bmatrix} < 0,
\]

(16)

hold \( \forall k \in \mathbb{P}^{2p} \) with \( \mathbb{P}^{2p} \) being the set of ordered \( 2p \)-bit binary numbers such that every bit is less or equal than the following, \( \mathbb{P}(k) \) being the set of indexes such that \( w_i(\ddot{q})w_j(\ddot{q}) = w_{k_i}(\ddot{q})w_{k_j}(\ddot{q}) \), and \( \text{He}(E) = E + E^T \). The controller gains in (15) are thus determined by \( F_j = K_jX^{-1} \).

**Proof 1:** If LMIs (16) hold, then convexity of \( w_i(\ddot{q}) \) guarantees that

\[
\text{He} \begin{bmatrix} \sum_{i \in \mathbb{P}^p} w_i(\ddot{q}) [0 & -I \\ -N_i & -I] X + \begin{bmatrix} 0 \\ I \end{bmatrix} \sum_{j \in \mathbb{P}^p} w_j(\ddot{q})K_j \end{bmatrix} < 0,
\]

which can be pre- and post-multiplied by \( P = X^{-1} \), renaming \( K_jX^{-1} \) as \( F_j \), to obtain

\[
\text{He} \left( P \begin{bmatrix} \sum_{i \in \mathbb{P}^p} w_i(\ddot{q}) [0 & -I \\ -N_i & -I] \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} F(\ddot{q}) \right) < 0,
\]
but the positive definiteness of $\text{diag}\{I, M^{-1}(q)\}$ along with the latter inequality implies that:

$$\text{He} \left( P \begin{bmatrix} I & 0 \\ 0 & M^{-1}(q) \end{bmatrix} \begin{bmatrix} 0 \\ -N^1(q) \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} F(q) \right) < 0,$$

which by Lyapunov theory guarantees stability of the nonlinear error system (8) under the control law (15) as the quadratic Lyapunov function candidate $V = \tilde{e}^T P \tilde{e}$, $P = P^T > 0$, with $\tilde{e} = [\tilde{e}^T \ e^T]^T$ has just been proven to be a valid Lyapunov function of the closed-loop nonlinear error system, thus concluding the proof. \hfill \Box

Therefore, when the outer-loop feedback (15), obtained by the previous theorem, is substituted in the novel inner-loop feedback (6), asymptotic tracking of $q_d(t)$ by $q(t)$ is guaranteed for the Lagrange-Euler system (1).

**Remark 1:** The proposal includes ordinary computed-torque rate as well as other performance specifications such as input constraints can be suitably incorporated by changes in the LMI conditions of theorem 1, e.g., developing for condition $V \leq -2\alpha V$ instead of $V \leq 0$ with $\alpha > 0$ guarantees decay rate of $\alpha$ in the asymptotic convergence of the Lyapunov function $V$; similarly, adding the LMIs

$$\begin{bmatrix} X & K_j^T \\ K_j & \mu^2 I \end{bmatrix} \geq 0, \quad \begin{bmatrix} 1 \\ \tilde{e}(0) \end{bmatrix} \begin{bmatrix} e^T(0) \\ X \end{bmatrix} \leq 0 \quad (17)$$

ensures the outer-loop feedback holds $\|u(t)\| \leq \mu$ for initial conditions $\tilde{e}(0)$ [23].

**IV. Example**

In this section, an example is fully worked out in order to test the proposed technique against ordinary computed-torque. The plant chosen for this purpose is a well-known rigid robot manipulator, namely, the two-link planar elbow arm, which is schematically shown in Fig. 2; the link masses under consideration are $m_1 = m_2 = 2$kg while their lengths are given by $a_1 = a_2 = 0.5$m; the joint variables are the angles $q_1$ (measured from the “floor” level to the first link) and $q_2$ (measured from the first link to the second), grouped in the joint vector $q \equiv [q_1 \ q_2]^T$. Modelling the links as point masses as well as ignoring the effects of friction, the corresponding Lagrange-Euler model is given by [2] (arguments omitted when convenient; elements of equation (1) identified with underbraces):

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{12} & m_{22} \end{bmatrix} \ddot{q}(t) + \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix},$$

where $g = 9.81 m/s^2$, $m_{11} \equiv (m_1 + m_2)a_1^2 + m_2a_2^2 + 2m_2a_1a_2 \cos q_2$, $m_{12} \equiv 2m_2a_1a_2 \cos q_2$, $m_{22} \equiv m_2a_2^2$, $n_1 \equiv -m_2a_1a_2(2q_1q_2 + \dot{q}_2^2) \sin q_2 + (m_1 + m_2)ga_1 \cos q_1 + m_2ga_2 \cos (q_1 + q_2)$, and $n_2 \equiv m_2a_1a_2q_1^2 \sin q_2 + m_2ga_2 \cos (q_1 + q_2)$; $\tau_1$ and $\tau_2$ are the torques applied at the first and second joints, respectively.

The trajectory $q(t)$ is intended to track is given by

$$q_d(t) = \begin{bmatrix} \frac{2}{3\pi} \sin(3t) + \frac{2}{\pi} \sin t \\ 0.1 \sin(5t) - 0.3 \cos(6t + \pi/4) \end{bmatrix};$$

The gains of the ordinary computed-torque control (4) usually are determined by pole placement [2]. Choosing the closed-loop poles of the linear error system as $-6 \pm 6i$ and $-3 \pm 3i$, the following outer-loop signal is obtained:

$$u(t) = \begin{bmatrix} 35.9846 & -0.0014 & 8.9979 & -3.0020 \\ -0.0014 & 36.0154 & 2.9980 & 9.0021 \end{bmatrix} \begin{bmatrix} e(t) \\ \dot{e}(t) \end{bmatrix}.$$

Recall this signal will be used only for comparison against our proposal along with ordinary inner-loop feedback (4).

We now turn our attention to the proposed methodology. Factorization of the error signal $[e^T \ e^T]^T$ from the difference $N(q, \dot{q}) - N(q, \dot{q})$ is required. The resulting expressions in $N^1(q, \dot{q}, \ddot{q}, q_d)$ and $N^2(q, \dot{q}, \ddot{q}, \dot{q}_d)$ are too big to be reproduced entirely; yet, for illustration, the expression corresponding to the (1,1) entry of $N^2(q, \dot{q}, \ddot{q}, \dot{q}_d)$ is provided: $-0.00833q_2^2q_2^5 + 0.1667q_2^2q_2^3 - q_2^2q_2^2$. Importantly, the entry (2,2) of $N^2(q, \dot{q}, \ddot{q}, \dot{q}_d)$ is 0.

The following physical constraints will be taken into account in order to rewrite the long expressions resulting from factorization of the error signals in $N^1(q, \dot{q}, \ddot{q}, q_d)$ and $N^2(q, \dot{q}, \ddot{q}, \dot{q}_d)$ in a convex form; those for the angles arise naturally from the scheme in Fig. 2; those for velocities are only for simulation purposes:

$$q_1, q_2, \dot{q}_d, q_{2d} \in [-\pi, \pi], \quad \dot{q}_1, \dot{q}_2, \dot{q}_{1d}, \dot{q}_{2d} \in [-5, 5].$$

Taking into account these intervals, Table 1 provides the induced bounds on the different entries of $N^1(q, \dot{q}, \ddot{q}, q_d)$ and $N^2(q, \dot{q}, \ddot{q}, \dot{q}_d)$; naturally, other arrangements can be done as there is an infinite number of convex embeddings for a given nonlinear expression [27]. Based on these values, the different vertices of the convex representation of $N^1(q, \dot{q}, \ddot{q}, q_d)$ and $N^2(q, \dot{q}, \ddot{q}, \dot{q}_d)$ are easily obtained as the different combinations of maxima/minima. Since there are 7 non-constant
entries, the number of vertices in the convex representations of $N^1(q, \dot{q}, \ddot{q}, \dot{d})$ and $N^2(q, \dot{q}, \ddot{q}, \dot{d})$ is $2^7 = 128$. Again, only some of these vertices are given for illustration purposes; the rest omitted for brevity:

$$N^1_{0000000} = \begin{bmatrix} 59.60 \\ 19.91 \end{bmatrix}, \quad N^2_{0000000} = \begin{bmatrix} 5.02 \\ 5.02 \end{bmatrix}, \quad N^1_{1111111} = \begin{bmatrix} 59.60 \\ 19.91 \end{bmatrix}, \quad N^2_{1111111} = \begin{bmatrix} 5.02 \end{bmatrix}.$$

Based on the previous vertex matrices and considering initial conditions $\bar{v}(0) = [-1 -1.2121 -2.7268 -2.2272]^T$, decay rate of $\alpha = 0.5$, and input constraint of $\mu = 200$, LMI conditions in Theorem 1 and those in (17) are found feasible via the LMI Toolbox in MATLAB®. Running on a 2.40GHz Intel®Core i5 processor; they yield $P = X^{-1} > 0$:

$$P = X^{-1} = \begin{bmatrix} 0.3433 & -0.0366 & 0.0278 & -0.0108 \\ -0.0366 & 0.2369 & 0.0035 & 0.0197 \\ 0.0278 & 0.0035 & 0.0100 & -0.0009 \\ -0.0108 & 0.0197 & -0.0009 & 0.0112 \end{bmatrix},$$

and 128 gains $F_j$, some of which are given below for illustration purposes:

$$F_{0000000} = \begin{bmatrix} -8.0771 \\ 19.3293 \end{bmatrix}, \quad F_{0000001} = \begin{bmatrix} -6.9134 \\ 35.1070 \end{bmatrix}, \quad F_{1111111} = \begin{bmatrix} -89.2262 \\ -4.9665 \end{bmatrix}.$$

With these gains, all the information required to apply the novel computed-torque control law

$$\tau(t) = M(q) \ddot{q}(t) - \sum_{j \in \mathbb{R}^p} w_j(\bar{q}) F_j e_j(t) + N(q, \dot{q}, \ddot{q}, \dot{d}),$$

resulting from the substitution of (15) in (6) is now available, since $w_j(\bar{q})$ are all the different products of the 7 pairs of functions $w_1^1(\bar{q}), w_1^2(\bar{q}), i \in \{1, 2, \ldots, 7\}$, defined as in (10) with each $z_i(\bar{q})$ being a different non-constant entry of $N^1(q, \dot{q}, \ddot{q}, \dot{d})$ and $N^2(q, \dot{q}, \ddot{q}, \dot{d})$.

Figures 3 and 4 show the time evolution of the position ($q_1$ and $q_2$) and velocity variables ($\dot{q}_1$ and $\dot{q}_2$), respectively; the signals in blue result from applying ordinary computed torque (classical) while those in black result from the novel technique (proposal). As expected, both of them achieve the task of tracking the reference signals in dotted lines.

In order to make a fair comparison of the energy invested in achieving this tracking, the poles of the inner-loop feedback of the ordinary computed torque were chosen as to make signals converge to the desired trajectory approximately at the same time of the new proposal. The noisy signal

$$w(t) = \begin{bmatrix} 0.05r(t) \\ 0.05r(t) \\ \sin(6\pi t) + r(t) \\ 0.1(\sin(60\pi t) + r(t)) \end{bmatrix} U(t - 4)$$

was added to the vector $[q^T \dot{q}^T]^T$, where $r(t) \in [-1, 1]$ is the

---

**Table I: Definitions of Nonlinear Terms**

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
<th>Lower bounds $z_i^0$</th>
<th>Upper bounds $z_i^1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$z_1$</td>
<td>$-N^1_{12}(q, \dot{q}, \ddot{q}, \dot{d})$</td>
<td>$-59.6034$</td>
<td>$59.6034$</td>
</tr>
<tr>
<td>$z_2$</td>
<td>$-N^2_{12}(q, \dot{q}, \ddot{q}, \dot{d})$</td>
<td>$-23.6525$</td>
<td>$47.2481$</td>
</tr>
<tr>
<td>$z_3$</td>
<td>$-N^1_{12}(q, \dot{q}, \ddot{q}, \dot{d})$</td>
<td>$-5.0220$</td>
<td>$5.0220$</td>
</tr>
<tr>
<td>$z_4$</td>
<td>$-N^2_{12}(q, \dot{q}, \ddot{q}, \dot{d})$</td>
<td>$-10.0447$</td>
<td>$10.0447$</td>
</tr>
<tr>
<td>$z_5$</td>
<td>$-N^1_{21}(q, \dot{q}, \ddot{q}, \dot{d})$</td>
<td>$-19.9195$</td>
<td>$19.9195$</td>
</tr>
<tr>
<td>$z_6$</td>
<td>$-N^2_{21}(q, \dot{q}, \ddot{q}, \dot{d})$</td>
<td>$-22.3544$</td>
<td>$19.9195$</td>
</tr>
<tr>
<td>$z_7$</td>
<td>$-N^2_{21}(q, \dot{q}, \ddot{q}, \dot{d})$</td>
<td>$-5.0228$</td>
<td>$5.0228$</td>
</tr>
</tbody>
</table>
random function with uniform distribution and \( U(t) \) stands for the step function.

We now compare the torques in Fig. 5: it is clear that the proposed technique requires less energy than the ordinary computed-torque technique. This advantage comes along with another one: should the signals from the encoders be noisy, our proposal will remain less affected as it employs desired signals (generated by the designer) in \( N(q_d, \dot{q}_d) \) instead of those coming from the encoders in \( N(q, \dot{q}) \). Thus, robustness is also better in the new proposal than in the classical approach.

V. CONCLUSIONS

A novel computed-torque technique for trajectory tracking has been presented, which employs the signals of the desired trajectory instead of those of the plant for some of the nonlinearities in the inner-loop feedback. It has been shown that a nonlinear error system is thus obtained, which can be stabilized by an outer-loop feedback in the form of parallel distributed compensation. The design conditions have been expressed as linear matrix inequalities by combining a recently appeared factorization, the direct Lyapunov method, and exact convex modelling of nonlinearities. The methodology has been successfully tested against ordinary computed-torque control on a two-link planar elbow arm, proving to be less energy-demanding while increasing robustness.

REFERENCES