Momentum Control of an Underactuated Flying Humanoid Robot

Daniele Pucci, Silvio Traversaro, and Francesco Nori

***Abstract*—The letter takes the first step toward the development of a control framework for underactuated flying humanoid robots. These robots may thus have the capacities of *flight*, *contact locomotion*, and *manipulation*, and benefit from technologies and methods developed for *Whole-Body Control* and *Aerial Manipulation*. As in the case of quadrotors, we assume that the humanoid robot is powered by four thrust forces. For convenience, these forces are placed at the robot hands and feet. The control objective is defined as the asymptotic stabilization of the robot centroidal momentum. This objective allows us to track a desired trajectory for the robot center of mass and keep *small* errors between a reference orientation and the robot base frame. Stability and convergence of the robot momentum are shown to be in the sense of Lyapunov. Simulations carried out on a model of the humanoid robot iCub verify the soundness of the proposed approach.**

***Index Terms*—Aerial systems, mechanics and control, humanoid robots, motion control.**

I. Introduction

THE general purpose of providing humanoid robots with some degree of locomotion has driven most of the recent research in the humanoid robotics community. Legged and wheeled locomotion, for instance, have proven to be feasible on various humanoid platforms, which can now be envisioned as user assistants in several domains (see, e.g., [1], [2]). The robot underactuation combined with the (usually) large number of the robot degrees-of-freedom are among the main challenges for robot locomotion control. This letter takes the first step towards the extension of the humanoid robot locomotion problem to the flight case, and proposes a framework for underactuated flying humanoid robots. The studied robots may have the capacity of *flight*, *contact locomotion*, and *manipulation*, and belong to the domain of interests of both *Whole-Body Loco-Manipulation* and *Aerial Manipulation* [3], [4].

Humanoid and flying robot control has developed along different directions, and suffer from specific limitations.

Humanoid robot control is often addressed assuming the robot attached to ground, and in this case the robot is referred to as *fixed-base* [5]. To weaken this assumption, one can apply the Euler-Poincarè formalism that provides one with singularity free equations of motion for the humanoid robot [6, Chapter 13]: in this case, the robot is referred to as *floating base*. When considering these robot equations of motion, however, the mechanical system representing the humanoid robot is underactuated, which forbids full feedback linearization of the underlying system [7]. The system underactuation is usually dealt with by means of constraints arising from the contacts between the robot and the environment. From the control design perspective, instead, a common strategy for humanoid robots is based on the so called *stack-of-task* [8]. These strategies usually consider several control objectives organized in a hierarchical or weighted prioritization. Often, the high-priority task is the stabilization of the robot momentum [8]–[10]: the objective of this task is the control of the robot center-of-mass and angular momentum by means of contact forces, while guaranteeing stable zero-dynamics [11]. Quadratic programming (QP) solvers can be used to monitor contact forces while achieving momentum control [12], [13].

The literature on flying vehicle control is much vaster and richer than that of humanoid robot control. Besides classical flight control applications [14], the nonlinear control community has been attracted by *small and versatile* aircraft, sometimes referred to as Vertical Take Off and Landing (VTOL) vehicles (see, e.g., [15], [16]). One of the main assumptions for VTOLs control is that the flying robot is powered by a body-fixed thrust force, and moves at *relatively* small velocities. This latter assumption renders the aerodynamic forces negligible when compared to gravity, and drag effects are but seldom taken into account [16]. Then, a common approach for VTOL control is the so-called *vectored-thrust control paradigm* [15], [16]: the aircraft angular velocity is considered as a control input, and its main role is to align thrust and gravity forces.

We believe that there is a strong technological benefit in bringing humanoid and flying robots closer: a platform combining these two robot natures may have the capacities of *flight*, *contact locomotion*, and *manipulation*, eventually not all used at the same time. A first step in this direction has already been taken by the so-called *Aerial Manipulation*, which conceives platforms capable to *fly* and *manipulate* [4]. Most robots having these two capacities are composed of a VTOL equipped with one or several robotic arms [17], [18]. These robots, however, are not energetically efficient when flying in confined spaces, and an additional capacity of *contact locomotion* may lower energy consumption considerably. Also, the additional end-effector of the studied platforms may be used to make contacts with the environment, thus robustifying the system against perturbations during manipulation tasks.

Manuscript received February 15, 2017; accepted July 8, 2017. Date of publication July 31, 2017; date of current version August 25, 2017. This letter was recommended for publication by Associate Editor P. Pounds and Editor J. Roberts upon evaluation of the reviewers' comments. This work was supported by the FP7 EU project CoDyCo under Grant 600716 ICT 2011.2.1 Cognitive Systems and Robotics. *(Corresponding author: Daniele Pucci.)*

The authors are with the iCub Facility Department, Istituto Italiano di Tecnologia, Genoa 16163, Italy (e-mail: Daniele.Pucci@iit.it; silvio.traversaro@iit.it; francesco.nori@iit.it).

Digital Object Identifier 10.1109/LRA.2017.2734245

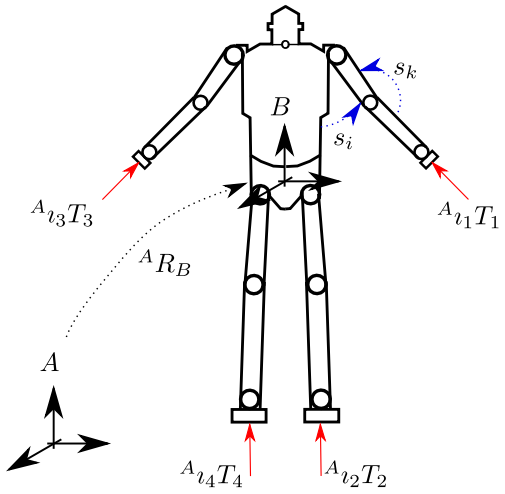

Fig. 1. Notation representation: inertial frame A, robot base frame B, thrust forces ${}^A\imath_kT_k$, $k=\{1,2,3,4\}$, rotation matrix ${}^AR_B$, and s_i, s_k two of the joint angles, with $i,k\in\mathbb{N}$.

This letter takes a further step in bringing humanoid and flying robots closer, and proposes a control architecture for flying humanoid robots. We assume that a humanoid robot is powered by four thrust forces installed at the robot end effectors, namely the robot hands and feet. Four turbo engines exemplify the robot actuation assumed in this letter. The control objective is the asymptotic stabilization of the robot *centroidal momentum* [19], which allows us to stabilize a reference trajectory for the center of mass and keep *small*, *bounded* errors between a desired orientation and the robot base frame. Reminiscent of the *vectored-thrust control paradigm* in VTOL control, we assume that the robot joint velocities can be considered as control input, and we then use them to align the effect of thrust forces with gravity effects. Aerodynamic effects are here neglected, and stability and convergence is shown to be in the sense of Lyapunov. A quadratic programming (QP) solver is employed to consider actuation limits. Finally, the robot joint torques that stabilize the (desired) joint velocities are generated via an high-gain joint control. Simulation results performed on the humanoid robot iCub verify the soundness of the proposed approach.

The paper is structured as follows. Section II introduces notation, equations of motion, and robot actuation. Section III describes the strategy for momentum, center of mass position, and orientation control. Simulations verifying the soundness and robustness of the control laws are presented in Section IV. Remarks and perspectives conclude the paper.

II. Background

A. Notation

1) A denotes an inertial frame, composed of a point (*origin*) and an *orientation frame* (see Fig. 1) and $g\in\mathbb{R}$ the norm of gravitational acceleration. Given two orientation frames A and B, and vectors of coordinates expressed in these frames, i.e. ${}^Ap, {}^Bp\in\mathbb{R}^3$, respectively, the rotation matrix ${}^AR_B$ is such that ${}^Ap={}^AR_B{}^Bp+{}^Ao_B$, with ${}^Ao_B\in\mathbb{R}^3$ the origin of the frame B w.r.t. A.
2) $1_n\in\mathbb{R}^{n\times n}$ is the identity matrix of size n; $0_{m\times n}\in\mathbb{R}^{m\times n}$ is the zero matrix of size $m\times n$ and $0_n=0_{n\times 1}$.
3) $S(x)\in\mathbb{R}^{3\times 3}$ is the skew-symmetric matrix such that $S(x)y=x\times y$, with $\times$ the cross product operator in $\mathbb{R}^3$.

B. Robot Modelling

We assume that the humanoid robot is composed of $n+1$ rigid bodies – called links – connected by n joints with one degree of freedom each. The configuration space of the multi-body system can then be characterized by the *position* and the *orientation* of a frame attached to a robot link – called *base frame* B – and the joint configurations. More precisely, the robot configuration space is defined by: $\mathbb{Q}=\mathbb{R}^3\times SO(3)\times\mathbb{R}^n$. An element of the set $\mathbb{Q}$ is a triplet $q=({}^Ao_B, {}^AR_B, s)$, where $({}^Ao_B, {}^AR_B)$ denotes the origin and orientation of the *base frame* expressed in the inertial frame, and s – which characterizes the robot *shape* – denotes the *joint angles*. The *velocity* of the multi-body system can then be characterized by the set $\mathbb{V}$ defined by: $\mathbb{V}=\mathbb{R}^3\times\mathbb{R}^3\times\mathbb{R}^n$. An element of $\mathbb{V}$ is then $\nu=(\mathrm{v}_B,\dot{s})$, where $\mathrm{v}_B=({}^A\dot{o}_B, {}^A\omega_B)$ is the linear and angular velocity of the base frame w. r. t. the frame A, i.e. ${}^A\dot{R}_B=S({}^A\omega_B){}^AR_B$.

Applying the Euler-Poincaré formalism [20, Ch. 13.5] to the robot yields the following equations of motion:

$$M(q)\dot{\nu}+C(q,\nu)\nu+G(q)=\begin{bmatrix}0_6\\ \tau\end{bmatrix}+\sum_{k=1}^{m}J_k^\top F_k, \quad (1)$$

where $M,C\in\mathbb{R}^{n+6\times n+6}$ are the mass and Coriolis matrix, respectively, $G\in\mathbb{R}^{n+6}$ is the gravity vector, τ are the internal actuation torques, and $F_k\in\mathbb{R}^3$ is the kth of the m external forces applied by the environment on robot. In particular, we assume that the application point of the external force F_k is the origin of a frame $\mathcal{C}_k$, which is attached to the robot's link where the force acts; the external force F_k is expressed in a frame whose orientation coincides with that of the inertial frame A. The Jacobian $J_k=J_k(q)$ is the map between the robot's velocity ν and the linear velocity ${}^A\dot{o}_{\mathcal{C}_k}\in\mathbb{R}^3$ of the origin of $\mathcal{C}_k$, i.e. ${}^A\dot{o}_{\mathcal{C}_k}=J_k(q)\nu$.

C. Robot Actuation

We assume that the robot is powered by four thrust forces $T_1,T_2,T_3,T_4\in\mathbb{R}$ that act along the directions ${}^A\imath_1, {}^A\imath_2, {}^A\imath_3, {}^A\imath_4\in\mathbb{R}^3$, with $|{}^A\imath_i|=1$, $\forall i\in\{1,2,3,4\}$. The application points ${}^Ao_i\in\mathbb{R}^3$ of the thrust forces are the origins of four frames attached to the robot end-effectors, e.g., the robot hands and feet. The thrust force directions, instead, move accordingly to the robot end-effectors, since thrust forces are assumed to be attached to the end-effector links. Fig. 1 depicts the notation used for the robot actuation. Four turbo engines installed at the robot end-effectors exemplify the actuation assumed for the humanoid robot[1]. We also assume that each thrust force T_i is measurable: this assumption holds when force sensors are installed in series with turbo engines.

By defining $T:=(T_1,T_2,T_3,T_4)^\top$, the effects of the external thrust forces on the right hand side of the equations of motion (1) can be compactly written as follows:

$$\sum J_k^\top F_k=\sum J_k^\top(q){}^A\imath_k(q)T_k:=f(q,T). \quad (2)$$

[1]For the humanoid robot iCub, we are considering the turbo engine JETCAT P100 RX [21] (weight: 1050gr; length: 245mm; thrust: 100N). Since iCub weight is of about 30 Kg, four engines suffice for hovering.

III. Control Design

A. Problem Statement

In light of Section II-C, the equations of motion

$$M(q)\dot{\nu} + C(q,\nu)\nu + G(q) = \begin{bmatrix} 0_6 \\ \tau \end{bmatrix} + f(q,T), \qquad (3)$$

are powered by $n+4$ control inputs. This in turn implies that controlling the entire robot configuration space $\mathbb{Q} = \mathbb{R}^3 \times SO(3) \times \mathbb{R}^n$, which is of *dimension*[2] $n+6$, may not be straightforward being system (3) underactuated and thus not feedback linearisable. Let us remark that one may attempt at the stabilization of the configuration space $\mathbb{Q}$ by applying advanced techniques developed for underactuated systems evolving on Lie Groups [22], [23]. The application of these techniques, however, is beyond the scope of this letter.

Assume that the control objective is the asymptotic stabilization of a frame – i.e. origin and orientation – associated with a robot link. Without loss of generality, assume that one wants to control the base frame B of the humanoid robot. The equations of motion of the base frame are given by:

$$\dot{\mathrm{v}}_B = \dot{J}_B\nu + J_B\dot{\nu}, \qquad (4)$$

with $\mathrm{v}_B \in \mathbb{R}^6$ the linear and angular velocity of the base frame, $J_B \in \mathbb{R}^{6\times n+6}$ the Jacobian of the base frame, and $\dot{\nu}$ the robot acceleration obtained from the robot equations of motion (3). More precisely, by using the (3), one can evaluate the robot acceleration $\dot{\nu}$ and substitute it into (4), thus obtaining an instantaneous relationship between the base acceleration $\dot{\mathrm{v}}_B$ and the control inputs (τ, T), i.e.

$$\dot{\mathrm{v}}_B = \mathrm{v}(q,\nu,T,\tau). \qquad (5)$$

Then, one may attempt at the control of a reference frame by performing feedback linearisation of the above function with proper feedback correction terms to achieve asymptotic stability. One of the main drawbacks of this approach is that the joint torques τ have *little* influence on the base acceleration $\dot{\mathrm{v}}_B$, which may render the associated control laws ill-posed especially close to constant reference position-and-orientation for the base frame. As a matter of fact, at *low* joint velocities, a first approximation for the robot equations of motion is given by the Newton-Euler equations, which corresponds to assuming that the robot behaves as a single rigid body. In this case, the dynamics $\dot{\mathrm{v}}_B$ can be obtained from the Newton-Euler equations, which clearly do not depend upon the (internal) joint torques τ but only on the four (external) thrust forces. Having only four inputs in this dynamics, one cannot perform feedback linearisation of the six-dimensional dynamics $\dot{\mathrm{v}}_B$.

Now, since the mass matrix $M(q)$ is positive definite, and thus invertible, the (3) point out that the joint dynamics $\ddot{s}$ can be feedback-linearized via a proper choice of the joint torques τ. So, any differentiable, desired joint velocity $\dot{s}_d \in \mathbb{R}^n$ can be stabilized with any desired settling time. We then make the following assumption.

[2]The set $\mathbb{Q}$ is a Lie Group. Hence, the dimension of $\mathbb{Q}$ is the dimension of the algebra associated with it.

Assumption 1: The joint velocity $\dot{s} := u_2$ and the thrust forces rate-of-change, i.e. $\dot{T} := (\dot{T}_1, \dot{T}_2, \dot{T}_3, \dot{T}_4)^\top := u_1$, can be chosen at will and then considered as control inputs.

In the language of Automatic Control, assuming $\dot{s}$ as control variable is a typical *backstepping* assumption. Then, the production of the joint torques associated with the desired joint velocities can be achieved via classical nonlinear control techniques [24, p. 589] or high-gain control.

B. Control Objective and Error Dynamics

This section shows that despite the aforementioned underactuation, it is possible to conceive control laws for the Newton-Euler equation of the multibody system without any approximation. Then, the control objective for the reminder of this section is defined as follows:

1) Asymptotic stabilization of the robot centroidal momentum $\mathrm{h} \in \mathbb{R}^6$ about desired, smooth values $\mathrm{h}_d(t) \in \mathbb{R}^6$.

when complemented with integral correction terms, the laws for the momentum control allows us to:

i) stabilize a reference trajectory $r(t) \in \mathbb{R}^3$ for the robot center of mass ${}^A c \in \mathbb{R}^3$;
ii) keep *small, bounded* errors between a reference orientation $R_d(t) \in SO(3)$ and the robot base frame.

Note that the choice of the above objective renders the dimension of the control task equal to six. This means that under Assumption 1, we may attempt at achieving momentum control by means of the control inputs $(\dot{s}, \dot{T})$. More precisely, let $\tilde{\mathrm{h}} \in \mathbb{R}^3$ denote the momentum error, i.e.

$$\tilde{\mathrm{h}} := \mathrm{h} - \mathrm{h}_d(t). \qquad (6)$$

Then, by recalling that the rate-of-change of the centroidal momentum equals the summation of all external wrenches acting on the robot [25], one has the following dynamics:

$$\dot{\tilde{\mathrm{h}}} = A(q)T - mge_3 - \dot{\mathrm{h}}_d(t), \qquad (7a)$$

$$\dot{T} = u_1 \qquad \dot{s} = u_2 \qquad (7b)$$

with $e_3 := (0,0,1,0,0,0)^\top$,

$$A(q) = \left(\bar{S}(r_1)^A \imath_1,\ \bar{S}(r_2)^A \imath_2,\ \bar{S}(r_3)^A \imath_3,\ \bar{S}(r_4)^A \imath_4\right) \qquad (8a)$$

$$r_i := {}^A o_i - {}^A c, \quad \forall i \in \{1,2,3,4\} \qquad (8b)$$

$$\bar{S}(r_i) := \begin{pmatrix} 1_3 \\ S(r_i) \end{pmatrix}. \qquad (8c)$$

The main advantage of controlling the dynamics (7a) over (5) resides in its independence from both the joint velocity $\dot{s}$ and joint torques τ. This independence allows us to identify the role of the robot joint dynamics for momentum control. More precisely, (7a) highlights that at the equilibrium configuration $(\tilde{\mathrm{h}}, \dot{\tilde{\mathrm{h}}}) = (0,0)$, the effect of the thrust intensities must oppose the gravity plus $\dot{\mathrm{h}}_d(t)$, i.e.

$$0 = A(q)T - mge_3 - \dot{\mathrm{h}}_d(t). \qquad (9)$$

The above equation is reminiscent of the so-called *vectored-thrust control paradigm* used in recent flight dynamics control techiniques. In this literature, in fact, the aircraft angular velocity is assumed as control input, and then exploited to align the thrust

force against the effect of gravity and, in general, of external forces [15], [16]. This in turn emphasizes the role of the joint velocities $\dot{s}$ in establishing (9): they are in charge of aligning the total thrust force $A(q)T$ to the gravity and desired momentum rate-of-change effect.

More precisely and generally, define

$$\tilde{\xi} := A(q)T + F, \tag{10a}$$

$$F := -mge_3 - \dot{\mathrm{h}}_d(t) + K_D\tilde{\mathrm{h}} + K_P I(t), \tag{10b}$$

with $K_P, K_D \in \mathbb{R}^{6\times 6}$ two symmetric positive definite matrices, and the variable $I(t)$ –representing the integral of $\tilde{\mathrm{h}}$– governed by $\dot{I} = \tilde{\mathrm{h}}$. Then, (7a) can be rewritten as follows:

$$\dot{I} = \tilde{\mathrm{h}} \tag{11a}$$

$$\dot{\tilde{\mathrm{h}}} = \tilde{\xi} - K_D\tilde{\mathrm{h}} - K_P I, \tag{11b}$$

$$\dot{\tilde{\xi}} = Au_1 + \Lambda_s u_2 + \Lambda_b \mathrm{v}_B - \ddot{\mathrm{h}}_d(t) + K_D\dot{\tilde{\mathrm{h}}} + K_P\tilde{\mathrm{h}} \tag{11c}$$

with the matrix A given by (8a),

$$\Lambda_b := \Lambda \begin{pmatrix} 1_6 \\ 0_{n\times 6} \end{pmatrix}, \quad \Lambda_s := \Lambda \begin{pmatrix} 0_{6\times n} \\ 1_n \end{pmatrix}, \tag{12a}$$

$$\Lambda := -\left(\tilde{S}_1, \tilde{S}_2, \tilde{S}_3, \tilde{S}_4\right) J_r, \tag{12b}$$

$$\tilde{S}_i := T_i \begin{pmatrix} 0_3 & S({}^A\imath_i) \\ S({}^A\imath_i) & S(r_i)S({}^A\imath_i) \end{pmatrix}, \tag{12c}$$

and $J_r \in \mathbb{R}^{24\times n+6}$ the Jacobian matrix mapping the robot velocity ν into the velocities $\Omega := (\dot{r}_1, \omega_1, \dot{r}_2, \omega_2, \dot{r}_3, \omega_3, \dot{r}_4, \omega_4) \in \mathbb{R}^{24}$, i.e. $\Omega = J_r(q)\nu$, where $\omega_i \in \mathbb{R}^3$ is the angular velocity associated with the ith end-effector frame, i.e. $\frac{d}{dt}{}^A\imath_i = S(\omega_i){}^A\imath_i$.

C. Momentum Control

In view of the dynamics (11), the main role of the control inputs (u_1, u_2) is to bring the variable $\tilde{\xi}$ to zero: the effect of the thrust forces must oppose the *apparent* force F (see (10a)). To introduce the control laws accomplishing this task, let us recall that the centroidal momentum h is linear versus the robot velocity ν. Namely, there exists a Jacobian matrix $J_{\mathrm{h}}(q) \in \mathbb{R}^{6\times n+6}$ such that

$$\mathrm{h} = J_{\mathrm{h}}(q)\nu = \begin{pmatrix} J_{\mathrm{h}}^b & J_{\mathrm{h}}^s \end{pmatrix} \nu = J_{\mathrm{h}}^b(q)\mathrm{v}_B + J_{\mathrm{h}}^s(q)u_2, \tag{13}$$

with $J_{\mathrm{h}}^b \in \mathbb{R}^{6\times 6}$ an invertible matrix, and $J_{\mathrm{h}}^s \in \mathbb{R}^{6\times n}$. The matrix J_{h} is usually referred to as *centroidal momentum matrix*. We can now present the control laws for system (11).

Theorem 1: Assume that Assumption 1 holds and define

$$\delta := \left(\Lambda_b + \tilde{K}J_{\mathrm{h}}^b\right)\mathrm{v}_B + (K_D + 1_3)\dot{\tilde{\mathrm{h}}} + K_P I - \ddot{\mathrm{h}}_d - \tilde{K}\mathrm{h}_d, \tag{14a}$$

$$B := \Lambda_s + \tilde{K}J_{\mathrm{h}}^s, \tag{14b}$$

$$\tilde{K} := K_P + K_D + K_O^{-1}, \tag{14c}$$

with $K_O \in \mathbb{R}^{6\times 6}$ a symmetric positive definite matrix.

If there exist smooth control inputs $(u_1, u_2) \in \mathbb{R}^{4+n}$ such that

$$\delta + Au_1 + Bu_2 = 0_6, \tag{15}$$

then the closed loop equilibrium point $(I, \tilde{\mathrm{h}}, \tilde{\xi}) = (0, 0, 0)$ of system (11) is globally asymptotically stable.

The proof is given in the Appendix. The main condition of the above theorem is (15). Let us remark that the number of control inputs to satisfy this condition is $n + 4$, which means that as long as

$$\mathrm{rank}(A\ B) = 6,$$

one is left with a redundancy of dimension $n - 2$ to render (15) satisfied. This redundancy is later exploited to attempt the stabilization of the system *zero dynamics*, i.e. the evolution of system (3) at $(I, \tilde{\mathrm{h}}, \tilde{\xi}) = (0, 0, 0)$.

The explicit form of the state feedback laws $(u_1(q, \nu), u_2(q, \nu))$ is omitted because the expression (15) is later exploited into the formulation of a *quadratic programming problem*, which allows us to take into account inequality constraints on the control inputs (u_1, u_2).

Let us remark that the control laws satisfying (15) are similar to those obtained by applying pure-feedback linearization techniques with output I. In particular, the output I is of *relative degree* equal to three, and one can then apply feedback linearization on the obtained third-order dynamics. The laws deduced with this approach, however, usually forbid to choose control gains independently from each other since the relative degree is higher than two. Hence, we believe that the laws proposed here may be more effective once implemented on a real platform than those obtained by applying a pure feedback linearization technique.

D. Velocity and Position Control

The control laws satisfying (15) can also be used to stabilize a (smooth) desired velocity $v_d(t) \in \mathbb{R}^3$ for the robot center-of-mass. To this purpose, let us recall that the centroidal momentum h can be decomposed into a linear and angular component $\mathrm{h}^l, \mathrm{h}^\omega \in \mathbb{R}^3$, respectively. Recall also that the linear component h^l is given by the robot center-of-mass velocity ${}^A\dot{c} \in \mathbb{R}^3$ times its mass m. In formule,

$$\mathrm{h} = \begin{pmatrix} \mathrm{h}^l \\ \mathrm{h}^\omega \end{pmatrix} = \begin{pmatrix} m{}^A\dot{c} \\ \mathrm{h}^\omega \end{pmatrix}. \tag{16}$$

Consequently, the control laws stabilizing a desired velocity v_d for the robot center-of-mass are those satisfying (15) with the desired momentum $\mathrm{h}_d(t)$ defined as follows:

$$\mathrm{h}_d = \begin{pmatrix} mv_d \\ \mathrm{h}_d^\omega \end{pmatrix}, \tag{17}$$

with $h_d^\omega \in \mathbb{R}^3$ the desired angular momentum. Clearly, stability and convergence statements of Theorem 1 are retained under the same assumptions.

Analogously, one can exploit the control laws satisfying (15) to stabilize a (smooth) desired trajectory $r(t) \in \mathbb{R}^3$ for the robot center-of-mass ${}^Ac \in \mathbb{R}^3$. In particular, it suffices to choose the desired momentum h_d as given by (17), but with $v_r = \dot{r}$ and the integral initial condition $I(0)$ such that

$$I(t) = \begin{pmatrix} m({}^Ac - r(t)) \\ I^\omega(0) + \int_0^\infty \tilde{\mathrm{h}}^\omega(s)ds \end{pmatrix}. \tag{18}$$

Again, stability and convergence statements of Theorem 1 are retained under the same assumptions.

E. Orientation Control

This section proposes modifications to the (15) for the control of the robot base frame ${}^A R_B \in SO(3)$ towards desired values $R_d \in SO(3)$. Let us first make a short digression on the orientation control problem on $SO(3)$.

The problem of stabilizing a desired orientation $R_d \in SO(3)$ for a rigid body orientation R may not be straightforward. For instance, it is known that the topology of $SO(3)$ forbids the design of smooth controllers that globally asymptotically stabilize a reference orientation R_d [26]. Then, quasi-global asymptotic stability is a common feature that orientation controllers guarantee. Just to recall a result:

Lemma 1 ([27] p. 173): Let $skew(A) := \frac{1}{2}(A - A^\top)$ for any matrix $A \in \mathbb{R}^3$, and the operator $(\cdot)^\vee$ defined by $x = S(x)^\vee$. Consider the orientation dynamics $\dot{R} = S(\omega)R$, where $\omega \in \mathbb{R}^3$ is considered as control input. Assume that the control objective is the asymptotic stabilization of a (constant) desired attitude $R_d \in SO(3)$. Then,

$$\omega = -k \left(\mathrm{skew}(R_d^\top R)\right)^\vee, \quad k > 0, \tag{19}$$

renders the equilibrium point $R = R_d$ quasi globally stable.

Let us recall that to evaluate the control torques generating the angular velocity (19), the correction terms (19) are first multiplied by the body inertia, and then complemented with additional velocity correction terms [27, p. 178].

The orientation control for the robot base frame proposed in this letter builds upon the above digression, and consists in modifying the term I in (15) so as to take into account the orientation correction term (19). More precisely, recall that the robot centroidal angular momentum h^ω can be expressed in terms of the total robot inertia $\bar{\mathbb{I}} \in \mathbb{R}^{3\times 3}$ as follows, $\mathrm{h}^\omega = \bar{\mathbb{I}}(q)\omega_o$, where $\omega_o \in \mathbb{R}^3$ is the so-called *locked* angualr velocity [19]: the terminology encompasses the fact that when joint velocities are blocked, $\dot{s} \equiv 0$, then ω_o corresponds to the angular velocity of the humanoid robot, which behaves as a rigid body. This suggests that the control laws (15) with

$$I = \begin{pmatrix} I^l(0) + \int_0^\infty \tilde{\mathrm{h}}^l(s)ds \\ \bar{\mathbb{I}}(q) \left(\mathrm{skew}(R_d^\top {}^A R_B)\right)^\vee \end{pmatrix} \tag{20}$$

may guarantee good tracking performance of the base frame ${}^A R_B$ towards the reference orientation R_d. It is important to emphasize that the asymptotic stability of the equilibrium point containing ${}^A R_B = R_d$ is not guaranteed in this case. Simulations that we have performed, however, tend to show that the suggested control law for the orientation control of the base frame can guarantee good tracking performance even if asymptotic stability is not guaranteed. Control laws ensuring stability properties of the equilibrium point containing ${}^A R_B = R_d$ will be the subject of forthcoming studies.

F. Orientation and Position Control

In light of the above, the control of the robot center-of-mass and base frame can be attempted by using the control laws satisfying (15) with h_d given by (17), $v_r = \dot{r}$, and

$$I = \begin{pmatrix} m({}^A c - r(t)) \\ \bar{\mathbb{I}}(q) \left(\mathrm{skew}(R_d^\top {}^A R_B)\right)^\vee \end{pmatrix}. \tag{21}$$

G. Zero Dynamics and Optimization Problem

Now, assume that the humanoid robot center of mass and base frame are stabilized about the reference position and orientation, respectively. This constrains only six out of the $n + 6$ *degrees of freedom* of the robot, thus leaving an n-dimensional *free-motion* of the system at the desired values: this free motion should be at least bounded. More precisely, this section discusses how to deal with the *boundedness* of the system zero dynamics by exploiting the input redundancy when satisfying (15).

Let us recall that stability and convergence in Theorem 1 are shown when (15) holds, i.e. $\delta + Au_1 + Bu_2 = 0_6$. Finding the control inputs $(u_1, u_2) \in \mathbb{R}^{n+4}$ such that this equation holds in general leaves an $n - 2$ dimensional input redundancy, which can be used for other purposes. We here use this redundancy so as the joint velocity $u_2 = \dot{s}$ are *as close as possible* to a *postural task* of the following form:

$$p := -K_P^p(s - s_r), \tag{22}$$

with $K_P^p \in \mathbb{R}^{n\times n}$ a symmetric positive definite matrix, and $s_r \in \mathbb{R}^n$ a reference position for the joint configuration. If $u_2 = p$, the joint configurations tend to the reference value s_r, thus reducing the risk of unstable zero dynamics.

We combine the tasks of satisfying (15) and $u_2 = p$ in a weighted optimization problem of the following form:

$$(u_1^*, u_2^*) = \underset{(u_1,u_2)}{\mathrm{argmin}} \ \lambda_m |\delta + Au_1 + Bu_2|^2 + \lambda_p |u_2 - p|^2 + \lambda_s |u_2|^2 + u_1^\top W u_1 \tag{23a}$$

$$\text{s.t.} \qquad lb_1 < u_1 < ub_1, \quad lb_2 < u_2 < ub_2 \tag{23b}$$

where $\lambda_m, \lambda_p, \lambda_s$ are positive weighting constants, and $lb_1, ub_1 \in \mathbb{R}^4$ and $lb_2, ub_2 \in \mathbb{R}^n$ are the lower and upper bounds of u_1 and u_2, respectively. Note that the cost function of the optimization problem (23a) contains also some regularization terms depending on u_2 and u_1, with the weighting matrix $W \in \mathbb{R}^4 : W = W^\top > 0$. These terms are paramount to have minimum thrust-intensity variation at the equilibrium.

H. Torque Control for Joint Velocity Stabilisation

The solution to the problem (23a) is a pair (u_1^*, u_2^*), namely the instantaneous rate-of-change of the thrust intensities $\dot{T}^*$ and the joint velocities $\dot{s}^*$. This latter value is then interpreted as a desired value to be stabilized by a torque-control law. More precisely, the route we follow is high-gain control for the stabilization of $\dot{s}^*$. Now, partition (3) as follows $M = \begin{pmatrix} \mathbb{I} & \mathbb{F} \\ \mathbb{F}^\top & \mathbb{H} \end{pmatrix}$ with $\mathbb{I} \in \mathbb{R}^{6\times 6}$, $\mathbb{F} \in \mathbb{R}^{6\times n}$, $\mathbb{H} \in \mathbb{R}^{n\times n}$, $b := \begin{pmatrix} lb_b \\ b_s \end{pmatrix} := C(q,\nu)\nu + G(q)$, $f := \begin{pmatrix} f_b \\ f_s \end{pmatrix}$ with $b_b, f_b \in \mathbb{R}^6$ and $b_s, f_s \in \mathbb{R}^n$. Then, from (3), one gets

$$\bar{M}\, \ddot{s} + \bar{b} = \tau, \tag{24a}$$

$$\bar{M} := \mathbb{H} - \mathbb{F}^\top \mathbb{I}^{-1}\mathbb{F}, \quad \bar{b} := b_s - f_s + \mathbb{F}^\top \mathbb{I}^{-1}(f_b - b_b). \tag{24b}$$

In view of (24a), the stabilization of a desired joint velocity u_2^* is attempted by means of the following high-gain control law:

$$\tau = \bar{b} + \bar{M}\left(K_P^s(u_2^* - \dot{s}) + K_I^s \int_0^t (u_2^* - \dot{s})dt\right). \quad (25)$$

IV. Simulation Results

This section presents simulations results obtained by applying the control algorithm discussed in the previous section. These simulations have been carried out by using the humanoid robot iCub with 25 DoFs [28]. In particular, the robot joint angles $s \in \mathbb{R}^{25}$ can be partitioned into five robot parts, which correspond to the robot torso $s_t \in \mathbb{R}^3$, left arm $s_{la} \in \mathbb{R}^5$, right arm $s_{ra} \in \mathbb{R}^5$, left leg $s_{ll} \in \mathbb{R}^6$, and right leg $s_{rl} \in \mathbb{R}^6$, i.e. $s = (s_t, s_{la}, s_{ra}, s_{ll}, s_{rl})$.

A. Simulation Environment and Control Parameters

The custom simulation environment is in charge of integrating the equations of motion (3) with the joint torques τ given by (25) and the thrust intensities T provided by the time integration of u_1^*, which are in turn generated by (23a).

For time-integration purposes of (3), we parametrize $SO(3)$ by means of a quaternion representation $\mathcal{Q} \in \mathbb{R}^4$. The resulting state space system, which is integrated through time, is then: $\chi := ({}^A o_B, \mathcal{Q}, s, \dot{p}_B, \omega_B, \dot{s})$, and its derivative is given by $\dot{\chi} = ({}^A \dot{o}_B, \dot{\mathcal{Q}}, \dot{s}, \dot{\nu})$. The constraints $|\mathcal{Q}| = 1$ are enforced during the integration phase [29]. The system evolution is obtained by integrating the constrained dynamical system with the numerical integrator MATLAB *ode15s* thanks to our software abstraction interfaces – described in [30] – which provide us with the elements of the equations of motion (3), e.g. the mass matrix. The humanoid robot initial condition is $\chi(0) = (0_3, 1, 0_3, 0_3, 0, 25, 0_3, 0, 25, 0_3, 0_{12})$, which corresponds to having the robot arms open (25° on the robot shoulders) and the base frame corresponding to the inertial frame. The robot legs are at the zero position, which corresponds to having the robot legs straight.

We then set the control gains as follows: for the momentum control described in Theorem 1, $K_P = (1_3 5, 0_3; 0_3, 1_3 50)$, $K_D = 2\sqrt{K_P}$, $K_O = 10$; for the optimization problem (23a): $\lambda_m = 50$, $\lambda_p = 1$, $\lambda_s = 50$, K_P^p =diag$(1, 1, 1, 5, 5, 5, 10, 1, 5, 5, 5, 10,$ ones$(1, 12))$, with limits $ub_1 = -lb_1 =$ 45 ones$(4, 1)$ $[°/s]$, $ub_2 = -lb_2 = 100$ ones$(25, 1)$ $[N/s]$, and $W = 1_2 + W_s$, with W_s, $|W_s| = 0.34$, a positive-semi definite matrix such that all thrust-intensity variations are equal if $u_1^\top W_s u_1 = 0$; and for (25): $K_I^s = 10^3$ diag(ones$(25, 1)$), $K_P^s = 2\sqrt{K_I^s}$.

In the following simulations, noise is not considered. The main reason why is that given the robot actuation, we expect that the noise produced by turbo engines may be higher than that produced by the robot joint motors. Hence, noise will be considered after proper experimental campaigns will be carried on to identify noise nature and amplitude.

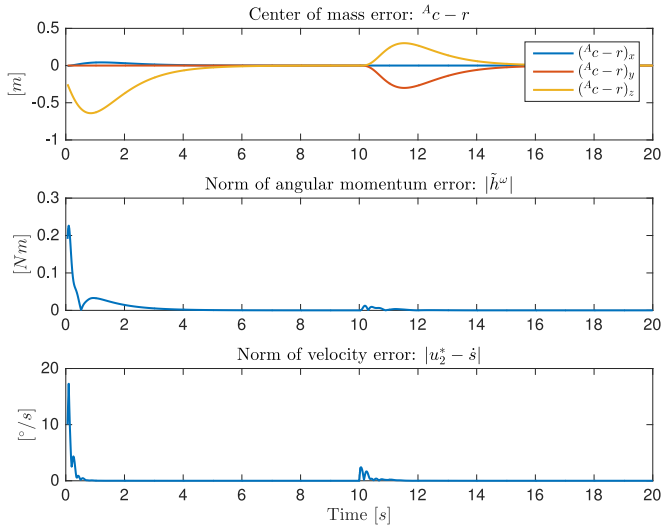

Fig. 2. Data of simulation 1. From top to bottom, center-of-mass tracking error, norm of angular momentum error, and joint velocity tracking error are depicted.

B. Simulation 1: Vertical Takeoff and Horizontal Flight with Constant Velocity Trajectory and no Orientation Control

The first simulation concerns the stabilization of a reference trajectory $r(t)$ $[m/s]$ obtained by integrating the following piece-wise constant reference velocity:

$$\dot{r}(t) = \begin{cases} (0,\ 0,\ 1) \text{ if} & 0 \le t < 10\ [s] \\ (1,\ 0,\ 0) \text{ if} & 10 \le t \qquad [s], \end{cases} \quad (26)$$

with $r(0) = 0_3$. To stabilize the resulting trajectory, we then apply the algorithm (23a)–(25), with the integral term I given by (18) and $\mathrm{h}_d^\omega = 0_3$. Data associated with this simulation are depicted in Fig. 2–4. In particular, from top to bottom, Fig. 2 depicts the tracking error of the robot center of mass, the angular momentum error, and the norm of the joint velocity error $u_2^* - \dot{s}$. Since all tracking errors converge to zero, the simulation results shown in Fig. 2 verify the statements of Theorem 1 when combined with the high-gain control (25). The control gains K_P and K_O influence the settling time with which both tracking errors of the center-of-mass and the angular momentum tend to zero. Still in Fig. 2, observe that the norm of the velocity error $u_2^* - \dot{s}$ tends to zero meaning that the high-gain control (25) for joint velocity stabilization is working properly. The joint torques (25) to obtain this stabilization are depicted in Fig. 3. Observe that despite the high-gain control chosen in this letter, the joint torques remain relatively *small* during the task

$$|u_2^* - \dot{s}| \to 0.$$

The outcome $u_1^* = \dot{T}$ of the optimization problem (23a) is depicted in Fig. 4. Note how the optimizer tends to choose solutions that equalizing all thrust intensity variations thanks to the term W_s. The smaller the norm of this regularization term, the more different the rate-of-change u_1^* of the jet intensities.

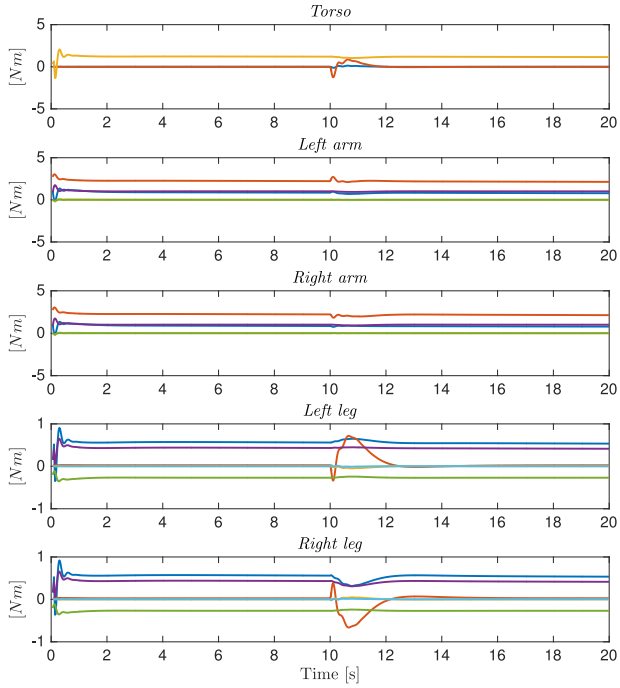

Fig. 3. Data of simulation 1. This picture depicts the joint torques τ necessary for stabilizing the joint velocity error $u_2^* - \dot{s}$ to zero. From top to bottom, joint torques of the robot torso, left and right arms, and left and right legs.

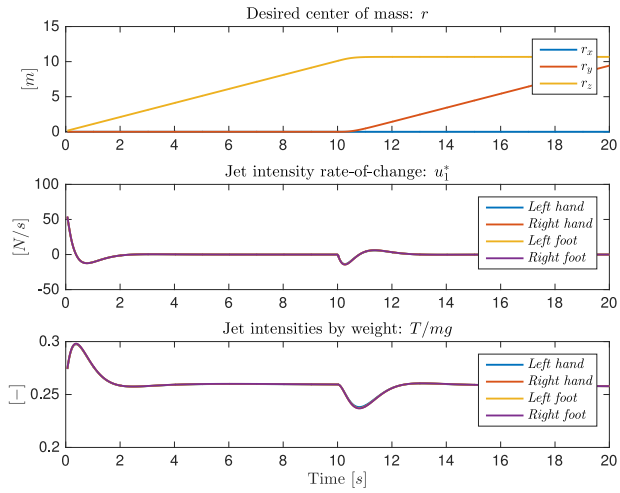

Fig. 4. Data of simulation 1. From top to bottom, desired center-of-mass trajectory, jet intensity rate-of-change u_1^*, and thrust-to-weight ratio are depicted.

C. Simulation 2: Helicoidal Flight with Orientation Control and Non-Perfect Control Model to Test Controller Robustness

The second simulation consists in tracking the reference trajectory given by

$$r(t) = \bar{A}(t)\cos(0.3\pi t)e_1 + \bar{A}(t)\sin(0.3\pi t)e_2 + te_3, \quad (27)$$

with

$$\bar{A}(t) = \begin{cases} 2\frac{t}{10} & \text{if} \quad 0 \le t < 10\ [s] \\ 2 & \text{if} \quad 10 \le t \qquad [s] \end{cases}. \quad (28)$$

After 10 $[s]$, it represents a helicoidal trajectory of radius 2 $[m]$ with vertical speed of 1 $[m/s]$. Differently to the previous section, we here test the trajectory tracking controller with orientation control. So, we apply (23a)–(25) with the

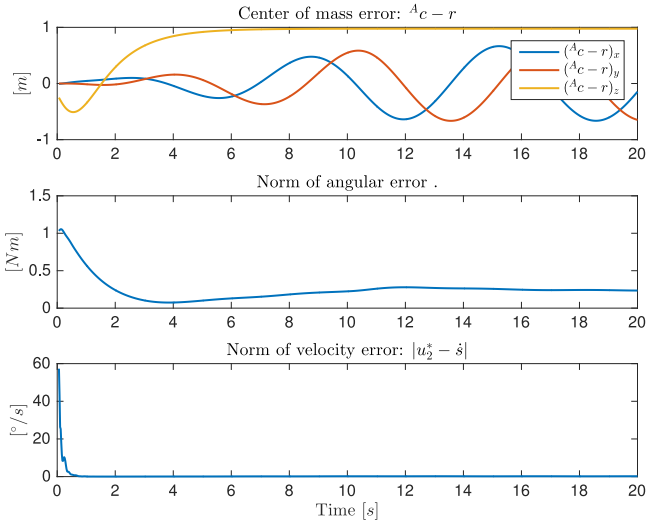

Fig. 5. Data associated with simulation 2. From top to bottom, center-of-mass tracking error, norm of angular error, and joint velocity tracking error are depicted.

integral term I given by (21) and R_d obtained by rotating ${}^A R_B(0)$ about the vertical axis of 60°. Then, the robot is expected to rotate about the vertical axis to reduce the orientation error. Furthermore, to test the control robustness, the controller has been calculated with estimated parameters that differ from the real ones of 10%. More precisely, the controller is evaluated by over-estimating the real model, i.e. $\hat{M}(q) = 1.1M(q)$, $\hat{C}(q,\nu)\nu + \hat{G}(q) = 1.1(C(q,\nu)\nu + G(q))$, and $\hat{J}_h(q) = 1.1J_h(q)$.

As depicted in Fig. 5, modeling errors lead to non convergence of the center-of-mass tracking error and orientation error to zero. However, *relatively small* bounded errors between actual and reference signals are kept, despite the controller (23a)–(25) (21) is not guaranteed to possess stability and convergence properties. This shows a degree of robustness versus modeling errors of the control laws presented in this letter. Let us remark that the orientation control performance degrades quickly for a time-varying reference orientation, and this calls for studies and extensions of the control laws presented in this letter for time-varying orientation tracking. Due to lack of space, plots depicting the control inputs of this simulations are omitted.

V. Conclusion

This letter has proposed extensions of the so-called *vectored-thrust control paradigm* used in VTOL control (see, e.g., [15], [16]) to the case of an underactuated flying humanoid robot. The main assumption is that the humanoid robot is powered by four thrust forces placed at the robot end effectors: these forces reduce but do not eliminate the well-known humanoid robot underactuation. Within this actuation framework, we have presented control laws guaranteeing stability and convergence properties for the robot centroidal momentum. Slight modifications to these laws allows us to track a desired reference trajectory for the center of mass and keep *small* tracking errors between a reference orientation and the robot base frame.

In this respect, future work consists in proposing control laws guaranteeing stability and convergence not only of the robot

centroidal momentum, but also of a robot frame. Furthermore, the model of the external forces acting on the robot neglects the aerodynamic phenomena. Hence, future work will consist in extending the *vectored-thrust control paradigm* [31]–[33], which considers aerodynamic effects, to the case of flying humanoid robots.

APPENDIX: PROOF OF THEOREM 1

Consider the following Lyapunov function candidate

$$V(I,\tilde{h},\tilde{\xi}) := \frac{1}{2} I^\top K_P I + \frac{1}{2}|\tilde{\mathrm{h}}|^2 + \frac{1}{2}\tilde{\xi}^\top K_O \tilde{\xi}. \tag{29}$$

Note that $V = 0 \iff (I,\tilde{h},\tilde{\xi}) = 0$. Direct calculations show that $\dot{V}$ along the trajectories of (11) is given by:

$$\dot{V} = -\tilde{\mathrm{h}}^\top K_D \tilde{\mathrm{h}} + \tilde{\xi}^\top K_O \left(\dot{\tilde{\xi}} + K_O^{-1}\tilde{\mathrm{h}}\right). \tag{30}$$

It is clear that $\dot{V} \leq 0$ if the term in the parenthesis on the right hand side of the above equation is equal to $-\tilde{\xi}$. In view of (11c) and (10), imposing $-\tilde{\xi} = \dot{\tilde{\xi}} + K_O^{-1}\tilde{\mathrm{h}}$ yields:

$$\begin{aligned} -\tilde{\xi} &= Au_1 + \Lambda_s u_2 + \Lambda_b \mathrm{v}_B - \ddot{\mathrm{h}}_d(t) + K_D \dot{\tilde{\mathrm{h}}} + \left(K_P + K_O^{-1}\right)\tilde{\mathrm{h}} \\ &= -A(q)T + mge_3 + \dot{\mathrm{h}}_d(t) - K_D\tilde{\mathrm{h}} - K_P I. \end{aligned} \tag{31}$$

By substituting $\tilde{\mathrm{h}} = \mathrm{h} - \mathrm{h}_d$ and $\mathrm{h} = J_{\mathrm{h}}^b(q)\mathrm{v}_B + J_{\mathrm{h}}^s(q)u_2$ in (31), one obtains (15) of Theorem 1. Hence, if (15) – and consequently (31) – are satisfied, then $\dot{V}$ in (30) becomes

$$\dot{V} = -\tilde{\mathrm{h}}^\top K_D \tilde{\mathrm{h}} - \tilde{\xi}^\top K_O \tilde{\xi} \leq 0.$$

And $\dot{V} \leq 0$ implies the stability of the equilibrium point and also global boundedness of the system trajectories.

Now, as long as (31) is satisfied, the closed loop dynamics is given by (11a) (11b) and $\dot{\tilde{\xi}} = -\tilde{\xi} - K_O^{-1}\tilde{\mathrm{h}}$: then, the closed loop dynamics is autonomous, and we can use LaSalle Theorem to conclude that $\dot{V} \to 0$. This implies that $(\tilde{\mathrm{h}},\tilde{\xi}) \to 0$ and $(\dot{\tilde{\mathrm{h}}},\dot{\tilde{\xi}}) \to 0$. By using these implications with (11b), i.e. $\dot{\tilde{\mathrm{h}}} = \tilde{\xi} - K_D\tilde{\mathrm{h}} - K_P I$, one has $I \to 0$. Hence, the equilibrium point $(I,\tilde{\mathrm{h}},\tilde{\xi}) = 0$ is asymptotically stable. Global asymptotic stability comes from the radial unboundedness of V.

REFERENCES

[1] M. Simon, "Boston dynamics new rolling, leaping robot is an evolutionary marvel," *Wired*, 2017. [Online]. Available: https://www.wired.com/2017/03/boston-dynamics-new-rolling-leaping-robot -evolutionary-marvel/

[2] P.-B. Wieber, R. Tedrake, and S. Kuindersma, *Modeling and Control of Legged Robots*. Cham, Germany: Springer, 2016, pp. 1203–1234.

[3] E. Farnioli, M. Gabiccini, and A. Bicchi, "Toward whole-body loco-manipulation: Experimental results on multi-contact interaction with the walk-man robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2016, pp. 1372–1379.

[4] K. Kondak *et al.*, "Unmanned aerial systems physically interacting with the environment: Load transportation, deployment, and aerial manipulation," in *Handbook of Unmanned Aerial Vehicles*. New York, NY, USA: Springer, 2015, pp. 2755–2785.

[5] R. Featherstone, *Rigid Body Dynamics Algorithms*. Secaucus, NJ, USA: Springer, 2007.

[6] J. E. Marsden and T. Ratiu, *Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems*. New York, NY, USA: Springer, 2013, vol. 17.

[7] J. A. Acosta and M. Lopez-Martinez, "Constructive feedback linearization of underactuated mechanical systems with 2-DOF," in *Proc. IEEE 44th Conf. Decis. Control, Eur. Control Conf.*, 2005, pp. 4909–4914.

[8] B. Stephens and C. Atkeson, "Dynamic balance force control for compliant humanoid robots," in *Proc. Intell. Robots Syst., IEEE/RSJ Int. Conf.*, 2010, pp. 1248–1255.

[9] T. Koolen *et al.*, "Design of a momentum-based control framework and application to the humanoid robot atlas," *Int. J. Humanoid Robot.*, vol. 13, no. 1, p. 1650007, Mar. 2016.

[10] D. Pucci, F. Romano, S. Traversaro, and F. Nori, "Highly dynamic balancing via force control," in *Proc. IEEE-RAS 16th Int. Conf. Humanoid Robots*, 2016, pp. 141–141.

[11] G. Nava, F. Romano, F. Nori, and D. Pucci, "Stability analysis and design of momentum-based controllers for humanoid robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2016, pp. 680–687.

[12] C. Ott, M. Roa, and G. Hirzinger, "Posture and balance control for biped robots based on contact force optimization," in *Proc. 11th IEEE-RAS Int. Conf. Humanoid Robots*, Oct. 2011, pp. 26–33.

[13] M. Hopkins, R. Griffin, A. Leonessa, B. Lattimer, and T. Furukawa, "Design of a compliant bipedal walking controller for the darpa robotics challenge," in *Proc. IEEE-RAS 15th Int. Conf. Humanoid Robots*, Nov. 2015, pp. 831–837.

[14] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation*, 2nd ed. Hoboken, NJ, USA: Wiley, 2003.

[15] R. Naldi, M. Furci, R. G. Sanfelice, and L. Marconi, "Robust global trajectory tracking for underactuated VTOL aerial vehicles using inner-outer loop control paradigms," *IEEE Trans. Autom. Control*, vol. 62, no. 1, pp. 97–112, Jan. 2017.

[16] M. D. Hua, T. Hamel, P. Morin, and C. Samson, "Introduction to feedback control of underactuated VTOL vehicles," *IEEE Control Syst. Mag.*, vol. 33, no. 1, pp. 61–75, Feb. 2013.

[17] H. Yang and D. Lee, "Dynamics and control of quadrotor with robotic manipulator," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2014, pp. 5544–5549.

[18] H. N. Nguyen, S. Park, and D. Lee, "Aerial tool operation system using quadrotors as rotating thrust generators," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2015, pp. 1285–1291.

[19] D. Orin, A. Goswami, and Lee, "Centroidal dynamics of a humanoid robot," *Autonom. Robots*, vol. 35, no. 2/3, pp. 161–176, 2013.

[20] J. E. Marsden and T. S. Ratiu, *Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems*. New York, NY, USA: Springer, 2010.

[21] JetCat, "JETCAT P100 RX," 2017. [Online]. Available: http://www.dreamworksrc.com/catalog/Jetcat-P100-Rx. Accessed on: May 3, 2017.

[22] P. Morin and C. Samson, "Practical stabilization of driftless systems on Lie groups: The transverse function approach," *IEEE Trans. Autom. Control*, vol. 48, no. 9, pp. 1496–1508, Sep. 2003.

[23] P. Morin and C. Samson, "Practical and asymptotic stabilization of chained systems by the transverse function approach," *SIAM J. Control Optim.*, vol. 43, no. 1, pp. 32–57, 2004.

[24] H. Khalil, *Nonlinear Systems*, 3rd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2002.

[25] D. Orin, A. Goswami, and S.-H. Lee, "Centroidal dynamics of a humanoid robot," *Autonom. Robots*, vol. 35, no. 2/3, pp. 161–176, 2013.

[26] S. P. Bhat and D. S. Bernstein, "A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon," *Syst. Control Lett.*, vol. 39, no. 1, pp. 63–70, 2000.

[27] R. Olfati-Saber, "Nonlinear control of underactuated mechanical systems with application to robotics and aerospace vehicles," Ph.D. dissertation, Massachusetts Instit. Technol.Cambridge, MA, USA, 2000.

[28] G. Metta *et al.*, "The iCub humanoid robot: An open-systems platform for research in cognitive development," *Neural Netw.*, vol. 23, no. 8, pp. 1125–1134, 2010.

[29] S. Gros, M. Zanon, and M. Diehl, "Baumgarte stabilisation over the *SO*(3) rotation group for control," in *Proc. 54th IEEE Conf. Decis. Control*, Dec. 2015, pp. 620–625.

[30] F. Romano, S. Traversaro, D. Pucci, J. Eljaik, A. D. Prete, and F. Nori, "A whole-body software abstraction layer for control design of free-floating mechanical systems," in *Proc. IEEE Int. Conf. Robotic Comput.*, Apr. 2017, pp. 148–155.

[31] D. Pucci, T. Hamel, P. Morin, and C. Samson, "Nonlinear control of aerial vehicles subjected to aerodynamic forces," in *Proc. IEEE Conf. Decision Control*, 2013, pp. 4839–4846.

[32] D. Pucci, "Towards a unified approach for the control of aerial vehicles," Ph.D. dissertation, , Université de Nice-Sophia Antipolis and "Sapienza" Universita di Roma, Rome, Italy, 2013.

[33] D. Pucci, T. Hamel, P. Morin, and C. Samson, "Nonlinear feedback control of axisymmetric aerial vehicles," *Automatica*, vol. 31, no. 31, pp. 781–806, 2015.