Momentum Control of an Underactuated Flying Humanoid Robot

Daniele Pucci, Silvio Traversaro, Francesco Nori

Abstract—This paper takes the first step towards the development of a control framework for underactuated flying humanoid robots. We assume that the robot is powered by four thrust forces placed at the robot end effectors, namely the robot hands and feet. Then, 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.

I. INTRODUCTION

The general purpose of providing humanoid robots with some degree of locomotion has driven most of research in the humanoid robotics community of the last decade. Legged and wheeled locomotion, for instance, have proven to be feasible on various humanoid robot platforms, which can now be envisioned as interfaces for user assistance in several domains (see, e.g., [1], [2]). The intrinsic humanoid robot underactuation, however, combined with the (usually large) number of the robot degrees of freedom partially account for the continued attention the robotics community is paying to the locomotion problem. This paper takes the first step towards the extension of the humanoid robot locomotion problem to the flight case, and proposes a control framework for underactuated flying humanoid robots.

Nonlinear control techniques for humanoid and flying robots have developed along different directions, and suffer from specific limitations. Besides the morphological differences between aerial and humanoids robots, one of the reasons accounting for this divergence is that humanoids robot control is often addressed assuming the robot attached to ground. In this case, the robot is referred to as fixed-base [3]. The limitations of this approach arise when attempting to tackle the general locomotion control problem, which also requires the robot to make and break contacts to achieve locomotion.

At the modelling level, the Euler-Poincaré formalism provides us with singularity free equations of motion for the humanoid robot, without assuming that the robot is attached to ground [4, 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, and this forbids full feedback linearization of the underlying system [5]. The system underactuation is usually dealt with by means of constraints that arise from the contacts between the robot and the environment, and also by applying task-based control strategies [6]. These strategies usually consider several control objectives organized in a hierarchical or weighted structure, which can also be exploited to combine manipulation and balancing tasks [7]. Often, the high-priority control task is defined as the stabilization of the robot momentum [6], [8], [9], [10], [11], whose essence is that of controlling the robot’s momentum while guaranteeing stable zero-dynamics. Quadratic programming (QP) solvers can be used to monitor the contact forces while achieving the momentum control [12], [13], [14].

The literature on flying vehicle control is vast and rich being Flight Dynamics much older than Robotics. This literature, however, can be roughly divided into two main categories: fixed-wing, e.g. commercial airplane, and rotary-wing, e.g. helicopters, aircraft control techniques.

Fixed-wing aircraft control exploits models of the aerodynamic forces and torques applied on the airplane, and then makes extensive use of linear control techniques [15]. By doing so, it is possible to assess robustness properties of the closed loop system [16]: a property of paramount importance when dealing with fixed-wing aircraft control.

The emergence of small and versatile rotary-wing aircraft – often referred to as Vertical Take Off and Landing (VTOL) vehicles – has driven the recent attention of the control community to aerial vehicle control (see, e.g., [17], [18], [19]). The main assumption of these work is that the flying robot is powered by a body-fixed thrust force and moves at relatively small velocities. This, in turn, renders the aerodynamic forces negligible when compared to gravity forces, and drag effects are but seldom taken into account [20]. A common control approach for position stabilization of VTOL is the so-called vectored-thrust control paradigm: the body angular velocity is considered as a control input, and its main role is to align the thrust force with the gravity force.

We believe that there is a strong technological benefit in conceiving robotic platforms capable of contact locomotion, flight, and manipulation. In this respect, flight and manipulation have already been implemented on many platforms contributing to the so-called aerial manipulation (see, e.g., [21] and the references therein) but robots combining the three aforementioned capacities are still missing to the best of the authors knowledge. This paper takes the first step in this direction by assuming 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 paper. The control objective is the asymptotic stabilization of the robot centroidal
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. In addition, the multi-body system is assumed to be free floating, i.e. none of the links has a constant pose with respect to the inertial frame. 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 $q = (A_{OB}, A_{RB}, s)$, where $(A_{OB}, A_{RB})$ denotes the origin and orientation of the base frame expressed in the inertial frame, and $s$ – which characterizes the shape of the robot – 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 $\nu = (v_B, s)$, where $v_B = (A_{OB} \dot{A}_{OB}, A_{RB} \dot{A}_{RB})$ is the linear and angular velocity of the base frame w.r.t. the inertial frame, i.e. $A_{RB} = S(A_{OB}) A_{RB}$.

Applying the Euler-Poincaré formalism [23, Ch. 13.5] to the robot yields the following equations of motion:

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

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, $\tau$ are the internal actuation torques, and $F_k \in \mathbb{R}^3$ is the $k$th 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 $C_k$, 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 $\nu$ and the linear velocity $A_{OB} \dot{c}_k \in \mathbb{R}^3$ of the origin of $C_k$, i.e. $A_{OB} \dot{c}_k = J_k(q) \nu$.

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 $A_{i1}, A_{i2}, A_{i3}, A_{i4} \in \mathbb{R}^3$, with $|A_{i1}| = 1$, $\forall i \in \{1, 2, 3, 4\}$, respectively. The application points $A_{Oi} \in \mathbb{R}^3$ of the thrust forces are the origins of four frames attached to the robot end-effectors, e.g., the robot hands and foot. 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. Figure 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. In addition, we also assume that each thrust force $T_i$ is measurable. This latter assumption holds, for instance, when force sensors are installed in series with the turbo engines.

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

$$\sum_{k=1}^{4} J_k^T F_k = \sum_{k=1}^{4} J_k^T (q) A_{ik}(q) T_k := f(q, T). \quad (2)$$
III. CONTROL DESIGN

A. Problem statement

In light of section II-C the equations of motion

\[ M(q)\ddot{\nu} + C(q, \nu)\dot{\nu} + G(q) = \begin{bmatrix} 0_n \\ \tau \end{bmatrix} + f(q, T), \]  

(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 \( n + 6 \), 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 [24], [25]. The application of these techniques, however, is beyond the scope of this paper.

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 of the humanoid robot. The equations of motion of the base frame are given by:

\[ \dot{v}_B = \dot{J}_Bv + J_B\nu, \]

(4)

with \( v_B \in \mathbb{R}^6 \) 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 \( \nu \) the robot acceleration obtained from the robot equations of motion \( 3 \). More precisely, by using the equations \( 3 \), one can evaluate the robot acceleration \( \dot{\nu} \) and substitute it into Eq. \( 4 \), thus obtaining an instantaneous relationship between the base acceleration \( \dot{v}_B \) and the control inputs \( (\tau, T) \), i.e.

\[ \dot{v}_B = v(q, \nu, T, \tau). \]

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 \( \tau \) have little influence on the base acceleration \( \dot{v}_B \), 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 clearly do not depend upon the (internal) joint torques \( \tau \) 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 \( \dot{v}_B \).

Now, since the mass matrix \( M(q) \) is positive definite, and thus invertible, the equations \( 3 \) point out that the joint dynamics \( \dot{s} \) can be feedback-linearized via a proper choice of the joint torques \( \tau \). 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.

\[ \text{Assumption 1. The joint velocity } \dot{s} := u_2 \text{ and the thrust forces } \text{rate-of-change, i.e. } \dot{T} := (T_1, T_2, T_3, T_4) := u_1, \text{ can be chosen at will and then considered as control inputs}. \]

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 [26, p. 589] or high-gain control.

B. Control objective and centroidal momentum 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:

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

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 this task by means of the control inputs \( (\dot{s}, \dot{T}) \). More precisely, let \( h \in \mathbb{R}^3 \) denote the momentum error defined by

\[ \dot{h} := h - \dot{h}_d(t). \]

Then, by recalling that the rate-of-change of the centroidal momentum equals the summation of all external wrenches acting on the robot [27], one has the following dynamics:

\[ \dot{h} = A(q)T - mge\dot{h}_3 - \dot{h}_d(t), \]  

(6a)

\[ \dot{T} = u_1 \]  

(6b)

with

\[ A(q) = \left( S(r_1)A_{t1}, S(r_2)A_{t2}, S(r_3)A_{t3}, S(r_4)A_{t4} \right) \]  

(7a)

\[ r_i := A_{qi}, \quad \forall i \in \{1, 2, 3, 4\} \]  

(7b)

\[ \dot{S}(r_i) := \begin{bmatrix} 1_3 \\ S(r_i) \end{bmatrix} \]  

(7c)

Eq. \( 6a \) points out that the dynamics \( \dot{h} \in \mathbb{R}^6 \) are underactuated even if one assumes that the thrust intensities \( T \in \mathbb{R}^4 \) can be considered as control input, and this renders the control of these dynamics not straightforward.

The equation \( 6a \) highlights also that at the equilibrium configuration \( (\dot{h}, \dot{h})= (0, 0) \), the effect of the thrust intensities must oppose the gravity plus \( \dot{h}_d(t) \), i.e.

\[ 0 = A(q)T - mge\dot{h}_3 - \dot{h}_d(t). \]  

(8)

The above equation is reminiscent of the so-called vectored-thrust control paradigm used in recent flight dynamics control techniques. 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 [19], [18]. This in turn emphasizes the role of the joint velocities \( \dot{\mathbf{s}} \) in establishing Eq. (3): 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
\[
\dot{\mathbf{s}} := A(q)\mathbf{T} + \mathbf{F},
\]

(9a)
and the variable \( \mathbf{I}(t) \) –representing the integral of \( \mathbf{h} \)– governed by \( \dot{\mathbf{I}} = \mathbf{h} \). Then, Eqs. (6) can be rewritten as follows:
\[
\dot{\mathbf{I}} = \mathbf{h},
\]

(10a)
\[
\dot{\mathbf{h}} = \mathbf{\xi} - K_{D}\mathbf{h} - K_{P}\mathbf{I},
\]

(10b)
\[
\dot{\mathbf{\xi}} = A\mathbf{u}_1 + A_{\omega}\mathbf{u}_2 + A_{\nu}\mathbf{v}_{\mathbf{B}} - \mathbf{\tilde{h}}_{d}(t) + K_{D}\dot{\mathbf{h}} + K_{P}\mathbf{h},
\]

(10c)
with the matrix \( A \) given by (7).

\[
A_{b} := \Lambda \begin{pmatrix}
1_{4} \\
0 \times n
\end{pmatrix}, \quad A_{\omega} := \Lambda \begin{pmatrix}
0 \times n \\
1_{n}
\end{pmatrix}, \quad A := \begin{pmatrix}
A_{b} \\
A_{\omega}
\end{pmatrix},
\]

(11a)
\[
\Lambda := - (S_{1}, S_{2}, S_{3}, S_{4}) J_{r},
\]

(11b)
\[
\tilde{S}_{i} := \begin{pmatrix}
0_{3} \\
T_{i}S^{(4)}(a_{i})
\end{pmatrix},
\]

(11c)
and \( J_{r} \) the Jacobian matrix mapping the robot velocity \( \mathbf{\nu} \) into the velocities \( \Omega := (r_{1}, \omega_{1}, r_{2}, \omega_{2}, r_{3}, \omega_{3}, r_{4}, \omega_{4}) \in \mathbb{R}^{24} \), i.e. \( \Omega = J_{r}(\mathbf{\nu}) \), where \( \omega_{i} \in \mathbb{R}^{4} \) is the angular velocity associated to the \( i \)th end-effector frame, i.e. \( A_{\nu} = S(\omega_{i}) A_{\nu} \).

C. Momentum control

In view of the dynamics (10), the main role of the control inputs \( (u_{1}, u_{2}) \) is to bring the variable \( \dot{\mathbf{\xi}} \) to zero, which means that the effect of the thrust forces must oppose the apparent force \( \mathbf{F} \) (see Eq. (9a)). To introduce the control laws accomplishing this task and stabilizing the desired momentum \( \mathbf{h}_{d}(t) \), let us recall that the centroidal momentum \( \mathbf{h} \) is linear versus the robot velocity \( \mathbf{\nu} \). Namely, there exists a Jacobian matrix \( J_{h}(q) \in \mathbb{R}^{6 \times 6} \) such that
\[
\mathbf{h} = J_{h}(q)\mathbf{\nu} = (J_{h}^{1} \quad J_{h}^{2}) \mathbf{\nu} = J_{h}^{1}(q)\mathbf{v}_{\mathbf{B}} + J_{h}^{2}(q)\mathbf{u}_{2},
\]

(12)
with \( J_{h}^{1} \in \mathbb{R}^{6 \times 6} \) an invertible matrix, and \( J_{h}^{2} \in \mathbb{R}^{6 \times n} \). The matrix \( J_{h} \) is usually referred to as centroidal momentum matrix. We can now present the control laws for system (10).

\textbf{Theorem 1.} Assume that Assumption 7 holds and define
\[
\delta := \left( A_{\omega} + \tilde{K} J_{h}^{2} \right) \mathbf{\nu}_{B} + (K_{D} + 1_{3}) \tilde{h} + K_{P} I - \tilde{h}_{d} - \tilde{K}_{h} d,
\]

(13a)
\[
B := A_{b} + \tilde{K} J_{h}^{1},
\]

(13b)
\[
\tilde{K} := K_{P} + K_{D} + K_{O}^{-1},
\]

(13c)
where \( K_{O} \in \mathbb{R}^{6 \times 6} \) is a symmetric positive definite matrix.

If there exist control inputs \( (u_{1}, u_{2}) \in \mathbb{R}^{4+n} \) such that
\[
\delta + A\mathbf{u}_1 + B\mathbf{u}_2 = 0_{6},
\]

(14)
then the closed loop equilibrium point \((\mathbf{I}, \dot{\mathbf{h}}, \dot{\mathbf{\xi}}) = (0, 0, 0)\) of system (10) is globally asymptotically stable.

The proof is given in the Appendix. The main condition of the above theorem is Eq. (14). Let us remark that the number of control inputs to satisfy this condition is \( n + 4 \), which means that as long as
\[
\text{rank}(A B) = 6,
\]
one is left with a redundancy of dimension \( n - 2 \) to render Eq. (14) satisfied. This redundancy is later exploited to attempt at the stabilization of the system zero dynamics, i.e. the evolution of system (3) at \((I, \mathbf{h}, \dot{\mathbf{\xi}}) = (0, 0, 0)\).

The explicit form of the state feedback laws, namely \((u_{1}, u_{2}) = C(q, \mathbf{\nu}, t)\), is here omitted because the expression (14) 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 Eq. (14) 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 dynamics. The laws deduced with this approach, however, usually forbid to choose control gains independently from each other when the relative degree is higher than two. Also, simulations we have performed tend to show that the law proposed here are more robust w.r.t. to modeling errors than those obtained from pure feedback linearization.

D. Velocity and position control

The control laws satisfying Eq. (14) can also be used to stabilize a (smooth) desired velocity \( \mathbf{v}_{d}(t) \in \mathbb{R}^{3} \) for the robot center-of-mass. To this purpose, let us recall that the centroidal momentum \( \mathbf{h} \) can be decomposed into a linear and angular component \( h^{l}, h^{\omega} \in \mathbb{R}^{3} \), respectively. Recall also that the linear component \( h^{l} \) is given the robot center-of-mass velocity \( A_{c} \) times its mass \( m \). In formule,
\[
\mathbf{h} = \begin{pmatrix}
h^{l} \\
h^{\omega}
\end{pmatrix} = \begin{pmatrix}
m A_{c}^{l} \\
m A_{c}^{\omega}
\end{pmatrix}.
\]

(15)
Consequently, the control laws stabilizing a desired velocity \( \mathbf{v}_{d} \) for the robot center-of-mass are those satisfying Eq. (14) with the desired momentum \( h_{d}(t) \) defined as follows:
\[
\dot{h}_{d} = \begin{pmatrix}
(m v_{d})^{l} \\
(m v_{d})^{\omega}
\end{pmatrix},
\]

(16)
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 Eq. (14) to stabilize a (smooth) desired position \( \mathbf{r}(t) \in \mathbb{R}^{3} \) for the robot center-of-mass. In particular, it suffices to choose the desired momentum \( h_{d} \) as in Eq. (16) with \( v_{r} = \dot{r} \) and to set the integral initial condition \( I(0) \) such that
\[
\begin{pmatrix}
I^{l}(0) \\
I^{\omega}(0) + \int_{0}^{\infty} h^{\omega}(s) ds
\end{pmatrix}.
\]

(17)
Again, stability and convergence statements of Theorem 1 are retained under the same assumptions.
E. Orientation control

This section proposes modifications to Eq. (14) for the control of the robot base frame $^AR_B \in SO(3)$ towards desired values $R_d \in SO(3)$. Let us first make a short digression on orientation control 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 $R_0$ \cite{28}. Then, quasi-global asymptotic stability is a common feature that orientation controllers guarantee. Just to recall a result:

**Lemma 1** (\cite{29} p. 173). Let $\text{skew}(A) := \frac{1}{2}(A - A^\top)$ for any matrix $A \in \mathbb{R}^3$, and the operator $(\cdot)^\top$ defined by $x = S(x)^\top$. 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,

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

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

Let us recall that to evaluate the control torques generating the angular velocity (13), the correction terms (13) are first multiplied by the body inertia, and then complemented with additional velocity correction terms \cite{29, 178}.

The orientation control for the robot base frame proposed in this paper builds upon the above digression, and consists in modifying the term $I$ in (14) so as to take into account the orientation correction term (18). More precisely, recall that the robot centroidal angular momentum $\dot{h}_o$ can be expressed in terms of the total robot inertia $I \in \mathbb{R}^{3 \times 3}$ as follows, $\dot{h}_o = I(q)\dot{\omega}_o$, where $\omega_o \in \mathbb{R}^3$ is the so-called locked angular velocity \cite{22}: the terminology encompasses the fact that when joint velocities are blocked, $\dot{s} \equiv 0$, then $\omega_o$ corresponds to the angular velocity of the humanoid robot, which behaves as a rigid body. This suggests that the control laws (14) with

$$I = \left( I(q) \left(\text{skew}(R_d^\top A R_B)\right)^\top \right)$$

(19)

can guarantee good tracking performance of the base frame $^AR_B$ towards the reference orientation $R_d$. It is important to note that the asymptotic stability of the equilibrium point containing $^AR_B = R_d$ 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 $^AR_B = R_d$ 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 Eq. (14) with $h_v$ given by (16), $v_r = \dot{r}$, and

$$I = \left( m(Ac - r(t)) \right)$$

$$\left( I(q) \left(\text{skew}(R_d^\top A R_B)\right)^\top \right).$$

(20)

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 Eq. (14).

Let us recall that stability and convergence in Theorem 1 are shown when Eq. (14) holds, i.e. $\delta + A u_1 + B u_2 = 0$. Finding the control inputs $(u_1, u_2) \in \mathbb{R}^{n+4}$ such that this equation holds in general leaves a $n - 2$ 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:

$$p := -K_p^s(s - s_r),$$

(21)

with $K_p^s \in \mathbb{R}^{n \times n}$ 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 Eq. (14) and $u_2 = p$ in a weighted optimization problem of the following form:

$$(u_1^a, u_2^a) = \text{argmin}_{(u_1, u_2)} \lambda_m |\delta + A u_1 + B u_2|^2 + \lambda_p |u_2 - p|^2$$

+ $\lambda_s |u_2|^2 + \lambda_T |u_1|^2$

(22a)

subject to $lb_1 < u_1 < ub_1$, $lb_2 < u_2 < ub_2$ (22b)

where $\lambda_m, \lambda_p, \lambda_s, \lambda_T$ 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 for the thrust-intensity variations $u_1$ and joint velocities $u_2$, respectively. Note that the cost function of the optimization problem (22) contains also some regularization terms depending on $|u_2|^2$ and $|u_1|^2$.

H. Torque control for joint velocity stabilisation

The solution to the problem (22) is a pair $(u_1^a, u_2^a)$, 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 to perform a high-gain control for the stabilization of $\dot{s}$. Now, partition (3) as follows

$$M = \begin{pmatrix} I & F \end{pmatrix}$$

with $I \in \mathbb{R}^{6 \times 6}$, $F \in \mathbb{R}^{5 \times n}$, $H \in \mathbb{R}^{n \times n}$,

$b := \begin{pmatrix} b_0 \\ b_1 \end{pmatrix} := C(q, \nu)\nu + G(q)f := \begin{pmatrix} f_b \\ f_a \end{pmatrix}$ with $b_0, f_0 \in \mathbb{R}^6$ and $b_1, f_1 \in \mathbb{R}^n$. Then, from Eq. (3), one gets

$$M \ddot{s} + \dot{b} + \tau = 0$$

(23a)

$$M := H - F^\top I^{-1} F, \quad \dot{b} := b_1 - f_a + F^\top I^{-1} (f_0 - b_0)$$

(23b)

In view of Eq. (23a), the stabilization of a desired joint velocity $u_2^a$ is then attempted by means of the following high-gain torque control law:

$$\tau = \ddot{b} + M \left( K_p^s (u_2^a - \dot{s}) + K_s^s \int_0^t (u_2^a - \dot{s}) dt \right).$$

(24)
For time-integration purposes of (3), we parametrize $SO(3)$ by means of a quaternion representation $q \in \mathbb{R}^4$. The resulting state space system, which is integrated through time, is then: $\chi := (A_{OB}, q, s, \dot{p}_B, \omega_B, \dot{s})$, and its derivative is given by $\dot{\chi} = (A_{OB}, \dot{q}, \dot{s}, \dot{\nu})$.

The constraints $|q| = 1$ are then enforced during the integration phase, and additional correction terms have been added to achieve this enforcement [31]. 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 [32] – which provide us the elements of the equations of motion (24), e.g. the mass matrix.

The humanoid robot initial condition is $\chi(0) = (0_3, 1, 0_3, 0_3, 0, 25, 0_3, 0_3, 0_3, 0_3, 0_3, 0_3)$, which corresponds to having the robot arms open just a little ($25^\circ$ 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 $K_P = (1_3, 0_3, 0_3, 1_3, 1_50)$, $K_D = 2\sqrt{K_P}$, $K_O = 10$; for the optimization problem (22): $\lambda_m = 50$, $\lambda_p = 1$, $\lambda_s = 50$, $\lambda_T = 1$, $K^{\text{opt}}_P = \text{diag}(1, 1, 1, 2, 5, 5, 5, 10, 1, 5, 5, 5, 5, 5, 10, 10, 10)$, with limits $ub_1 = -lb_1 = 45 \text{ ones}(4, 1)$ [°/s], $ub_2 = -lb_2 = 100 \text{ ones}(25, 1)$ [N/m]; and for the high-gain torque control (24): $K^*_P = 10^3 \text{ diag}(\text{ones}(25, 1))$, $K^*_P = 2\sqrt{K^*_P}$.

### B. Simulation 1: piece-wise constant velocity trajectory without and with 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:

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

with $r(0) = 0_3$. To stabilize the resulting trajectory, we then apply the algorithm (22)-(24), with the integral term $I$ given by (17) and $h^2_{\nu|} = 0_3$. Data associated with this simulation are...
depicted in Figures 2-4. In particular, from top to bottom, Figure 2 depicts the tracking error of the robot center of mass, the angular momentum, and the norm of the joint velocity error $u_2^* - \dot{s}$. These results verify the statements of Theorem 1. The joint torques (24) stabilizing the joint reference velocities $u_2^*$ are depicted in Figure 3. Observe that despite the high-gain control chosen in this paper, the joint torques remain relatively small during the task $|u_2^* - \dot{s}| \rightarrow 0$.

We then attempt at stabilizing a desired orientation for the base frame while tracking the reference trajectory obtained from (25). In particular, we apply the algorithm (22)-(24), with the integral term $I$ given by (20) and $R_d$ obtained by rotating $\dot{A} R_B(0)$ about the vertical axis of $60^\circ$. Then, the robot is expected to rotate about the vertical axis to reduce the orientation error. In fact, despite stability and convergence of the control law (22)-(24) are not ensured, Figure 5 shows that the orientation error is brought to zero. Let us remark that the orientation control performance degrades quickly for time-varying reference orientation, and this calls for studies and extensions of the control laws presented in this paper for time-varying orientation tracking.

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

$$r(t) = A(t) \cos(0.3\pi t)e_1 + A(t) \sin(0.3\pi t)e_2 + te_3,$$  \hspace{1cm} (26)

with

$$A(t) = \begin{cases} 2 & \text{if } 0 \leq t < 10 \ [s] \\ 1 & \text{if } 10 \leq t \leq 20 \ [s] \end{cases}.$$ \hspace{1cm} (27)

After $10 \ [s]$, it represents a helicoidal trajectory of radius $2 \ [m]$ with vertical speed of $1 \ [m/s]$. Analogously to the previous section, we here test the trajectory tracking controller with orientation control: so we apply (22)-(24) with the integral term $I$ given by (20) and $R_d$ obtained by rotating $\dot{A} R_B(0)$ about the vertical axis of $60^\circ$. Differently from before, 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. $M(q) = 1.1 M(q)$, $\dot{C}(q, \nu) + G(q) = 1.1 (C(q, \nu) + G(q))$, and $\dot{J}_h(q) = 1.1 J_h(q)$. As depicted in Figure 6, 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 (22)-(24) (20) 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 paper.

Fig. 5: Data associated with simulation 1: orientation control.

Fig. 6: Data associated with simulation 2.

V. CONCLUSIONS

This paper has proposed extensions of the so-called vectored-thrust control paradigm used in VTOL control 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 of proposing control laws guaranteeing stability and convergence not only for desired center of mass position, but also for the reference orientation of the base frame. Also, the model of the external forces acting on the robot neglects the aerodynamic phenomena. Hence, future work will also consist in extending vectored-thrust control paradigm with aerodynamic effects to the case of the considered flying humanoid [33], [34], [35].

APPENDIX: PROOF OF THEOREM 1

Consider the following Lyapunov function candidate

$$V(I, \dot{h}, \dot{\tilde{\xi}}) := \frac{1}{2} I^\top K_P I + \frac{1}{2} |\dot{h}|^2 + \frac{1}{2} \dot{\xi}^\top K_O \dot{\xi}.$$ \hspace{1cm} (28)

Note that $V = 0 \iff (I, \dot{h}, \dot{\tilde{\xi}}) = 0$. Direct calculations show that the time derivative of $V$ along the trajectories of system (10) is given by:

$$\dot{V} = -\dot{h}^\top K_D \dot{h} + \dot{\xi}^\top K_O \dot{\xi} + K_O^{-1} \dot{\dot{h}}.$$ \hspace{1cm} (29)

Now, 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 $-\dot{\xi}$. By imposing this condition, one gets in view of (10b) and (9):
\[ \dot{\xi} = \xi + K_O \dot{h} \iff (30a) \]
\[ \dot{\xi} = A(t) \xi + u_I + \sum_{\alpha=1}^{i} (A_t \xi + \dot{X}_\alpha) \iff (30b) \]

By substituting \[ \ddot{h} = \dot{h} - K_O \dot{h} \] in (30b), one gets the condition (14) of Theorem 1. As a consequence, if (14) − and consequently (30b) − are always satisfied, then \( V \) in Eq. (29) becomes
\[ V = -\dot{h} + K_D h = K_O \xi \leq 0. \]

And this implies the stability of the equilibrium point and also global boundedness of the system trajectories.

Now, as long as \( (30b) \) is satisfied, the closed loop dynamics is given by \( \ddot{\xi} = -\dot{\xi} - K_O \dot{\xi} \) and (10a) (10b); therefore, the closed loop dynamics is autonomous, and we can use LaSalle's Theorem to conclude that \( \dot{V} \to 0 \). This implies that \( \dot{\xi} \to 0 \), \( \ddot{\xi} \to 0 \), and \( \dot{\xi} \to 0 \). By using these implications with (10b), namely \( \dot{\xi} = -K_O \dot{\xi} - K_P I \), one obtains that \( I \to 0 \). Hence, the equilibrium point \((I, \dot{h}, \dot{\xi} = (0, 0, 0)\) is asymptotically stable. Global asymptotic stability comes from radial unboundedness of \( V \).

REFERENCES
[1] W. Chung and K. Iagnemma, Wheeled Robots. Cham: Springer International Publishing, 2016, pp. 575–594.
[2] P.-B. Wieber, R. Tedrake, and S. Kuindersma, Modeling and Control of Legged Robots. Cham: Springer International Publishing, 2016, pp. 1203–1234.[Online]. Available: http://dx.doi.org/10.1007/978-3-319-32552-1_48
[3] R. Featherstone, Rigid Body Dynamics Algorithms. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.
[4] J. E. Marsden and T. Ratiu, Introduction to mechanics and symmetry: a basic exposition of classical mechanical systems. Springer Science & Business Media, 2013, vol. 17.
[5] J. A. Acosta and M. Lopez-Martinez, “Constructive feedback linearization of underactuated mechanical systems with 2-DOF,” Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC ’05, 44th IEEE Conference on, 2005.
[6] B. Stephens and C. Atkeson, “Dynamic balance force control for compliant humanoid robots,” in Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, 2010, pp. 1248–1255.
[7] E. Farnioli, M. Gabiccini, and A. Bicchi, “Optimal contact force distribution for compliant humanoid robots in whole-body locomotion tasks,” 2013 IEEE International Conference on Robotics and Automation (ICRA), pp. 5675–5681, May 2015.
[8] A. Herzog, L. Righetti, F. Grimminger, P. Pastor, and S. Schaal, “Balancing experiments on a torque-controlled humanoid with hierarchical inverse dynamics,” in Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on, Sept 2014, pp. 981–988.
[9] T. Koolen, S. Bertrand, G. Thomas, T. de Boer, T. Wu, J. Smith, J. Englsberger, and J. Pratt, “Design of a momentum-based control framework and application to the humanoid robot atlas,” International Journal of Humanoid Robotics, vol. 13, p. 34, March 2016.
[10] G. Nava, F. Romano, F. Nori, and D. Pucci, “Stability analysis and design of momentum-based controllers for humanoid robots,” Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, Oct 2016.
[11] D. Pucci, F. Romano, S. Traversaro, and F. Nori, “Highly dynamic balancing via force control,” in 2016 IEEE-RAAS 16th International Conference on Humanoid Robots (Humanoids), 2016, pp. 141–141.
[12] C. Ott, M. Roa, and G. Hirzinger, “Posture and balance control for biped robots based on contact force optimization,” in Humanoid Robots (Humanoids), 2011 11th IEEE-RAAS International Conference on, Oct 2011, pp. 26–33.
[13] P. Wensing and D. Orin, “Generation of dynamic humanoid behaviors through task-space control with conic optimization,” in Robotics and Automation (ICRA), 2013 IEEE International Conference on, May 2013, pp. 3103–3109.
[14] M. Hopkins, R. Griffen, A. Leonessa, B. Lattimer, and T. Furukawa, “Design of a compliant bipedal walking controller for the darpa robotics challenge,” in Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on, Nov 2015, pp. 831–837.
[15] B. L. Stevens and F. L. Lewis, Aircraft Control and Simulation, 2nd ed. Wiley-Interscience, 2003.
[16] M. Roos, C. Doll, and J.-M. Biannic, “Flight control laws: recent advances in the evolution of their robustness properties,” Aerospace Lab, vol. 4, p. 5, 2012.
[17] P. Castillo, R. Lonzano, and A. E. Dzul, Modelling and Control of Mini-Flying Machines. Springer Verlag, 2005.
[18] R. Naldi, M. Fuci, R. G. Sanfelice, and L. Marconi, “Robust global trajectory tracking for underactuated vtol aerial vehicles using inner−outer loop control paradigms,” IEEE Transactions on Automatic Control, vol. 62, no. 1, pp. 97–112, Jan 2017.
[19] M. D. Hu, T. Hamel, P. Morin, and C. Samson, “Introduction to feedback control of underactuated vtol vehicles,” IEEE Control Systems Magazine, vol. 33, pp. 61–75, 2013.
[20] M.-D. Hu, T. Hamel, P. Morin, and C. Samson, “A control approach for thrust-propelled underactuated vehicles and its application to VTOL drones,” IEEE Trans. on Automatic Control, vol. 54(8), pp. 1837–1853, 2009.
[21] K. Kondak, A. Ollero, I. Maza, K. Krieger, A. Albu-Schaeffer, M. Schwarzbach, and M. Laaiker, “Unmanned aerial vehicles physically interacting with the environment: Load transportation, deployment, and aerial manipulation,” in Handbook of Unmanned Aerial Vehicles. Springer, 2015, pp. 2755–2785.
[22] D. Orin, A. Goswami, and Lee, “Centroidal dynamics of a humanoid robot,” Autonomous Robots, vol. 35, no. 2-3, pp. 161–176, 2013.
[23] J. E. Marsden and T. S. Ratiu, Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems. Springer Publishing Company, Incorporated, 2010.
[24] P. Morin and C. Samson, “Practical stabilization of driftless systems on Lie groups: the transverse function approach,” vol. 48, pp. 1496–1508, 2003.
[25] ——, “Practical and asymptotic stabilization of chained systems by the transverse function control approach,” vol. 43, no. 1, pp. 32–57, 2004.
[26] H. Khalil, Nonlinear systems, 3rd ed. Prentice Hall, 2002.
[27] D. Orin, A. Goswami, and S.-H. Lee, “Centroidal dynamics of a humanoid robot,” Autonomous Robots, vol. 35, no. 2-3, pp. 161–176, 2013.
[28] S. P. Bhat and D. S. Bernstein, “A topological obstruction to a discontinuous global stabilization of rotational motion and the unwinding phenomenon,” Systems Control Letters, vol. 39, no. 1, pp. 63 – 70, 2000.[Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167691199000000
[29] R. Offati-Saber, “Nonlinear Control of Underactuated Mechanical Systems with Application to Robotics and Aerospace Vehicles,” Ph.D. dissertation, 2000.
[30] G. Metta, L. Natale, F. Nori, G. Sandini, D. Vernon, L. Fadiga, C. Von Hofsten, K. Rosander, M. Lopes, J. Santos-Victor et al., “The icub humanoid robot: An open-systems platform for research in cognitive development,” Neural Networks, vol. 23, no. 8, pp. 1125–1134, 2010.
[31] S. Gros, M. Zanon, and M. Diehl, “Baumgarte stabilization over the SO(3) rotation group for control,” 2015 54th IEEE Conference on Decision and Control (CDC), pp. 620–625, December 2015.
[32] 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,” Proceedings of the 2017 IEEE International Conference on Robotic Computing, Apr 2017.
[33] D. Pucci, T. Hamel, P. Morin, and C. Samson, “Nonlinear control of aerial vehicles subjected to aerodynamic forces,” in IEEE Conf. on Decision and Control (CDC), 2013, pp. 4839 – 4846.
[34] D. Pucci, “Towards a unified approach for the control of aerial vehicles,” Ph.D. dissertation, Universidad de Nice-Sophia Antipolis and “Sapienza” Universita di Roma, 2013.
[35] 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.