Nonlinear Analysis and Control of a Reaction Wheel-based 3D Inverted Pendulum

Michael Muehlebach and Raffaello D'Andrea

Abstract—This article presents control and learning algorithms for a reaction wheel-based 3D inverted pendulum. The inverted pendulum system has two main features: the ability to balance on its edge or corner and to jump from lying flat to its corner by suddenly braking its reaction wheels. Algorithms which address both features are presented. For balancing, a backstepping based controller providing global stability (almost everywhere) is derived, together with a simple tuning method based on the analysis of the resulting closed-loop system. For jump-up, a computationally efficient, gradient-based learning algorithm is provided, which is shown experimentally to converge to the correct angular velocities enabling a successful jump-up. Moreover, a controller based on feedback linearization is derived and used to track an ideal trajectory during jump-up, increasing robustness and reliability.

I. INTRODUCTION

This article presents control and learning algorithms for a reaction wheel-based 3D inverted pendulum. The inverted pendulum system consists of three perpendicular reaction wheels embedded in a cubic housing. Due to its relatively small footprint, i.e. a side length of $150\,\text{mm}$, it is called Cubli, which is derived from the Swiss German diminutive for cube. Figure 1 shows the Cubli balancing on a corner. Unlike other inverted pendulum test beds, [2], [4], [5], [18], [19], [21], [22], and references therein, it has the ability to jump-up from a resting position without any external support by suddenly braking its reaction wheels rotating at high angular velocities. While the mechatronic design is covered in [7], and a linear controller is discussed in [8], this paper presents nonlinear control strategies and a learning algorithm enabling a successful jump-up.[1]

In [13] several design variants of a reaction wheel-based 3D inverted pendulum are compared. Moreover, a swing-up control strategy is presented based on feedforward and linear state feedback, for which local stability is shown. However, no braking system is used, which has the drawback that the design is not capable of swinging up from arbitrary positions, as the electric motors provide only limited torques.

Based on a reduced system description two nonlinear controllers are proposed herein. The first control design is based on backstepping and provides a smooth, globally (almost everywhere) stabilizing control law characterized by four tuning parameters. In contrast to earlier work, e.g. [3], [13], [16] the full 3D case is treated and global stability is proved (almost everywhere). The work presented in [15] is extended by relating these parameters to the closed-loop behavior, leading to a simple tuning strategy suitable for implementation.

The second control design is based on feedback linearization; an appropriate state transformation is introduced allowing for feedback linearization in the 3D case. This extends the result of [20], where the 1D (planar) case is discussed.

Both controllers are implemented on the Cubli: The controller based on backstepping is used for balancing. The controller based on feedback linearization is used for tracking predefined non-equilibrium motions; compared to other methods, such as time-varying LQR control, feedback linearization has the advantage of providing a time-invariant feedback law.

Additionally, a low-complexity model describing the jump-up is derived. The model is used to apply a gradient-based learning algorithm, similar to [12], to the Cubli and is shown experimentally to converge. To enhance the reliability of the jump-up, a predefined jump-up trajectory is tracked using the controller based on feedback linearization.

The remainder of this article is structured as follows: The dynamics are introduced in Section II, followed by the control design in Section III. Aspects related to the jump-up are covered in Section IV. Finally, experimental results are presented in Section V, and the conclusions are summarized in Section VI.
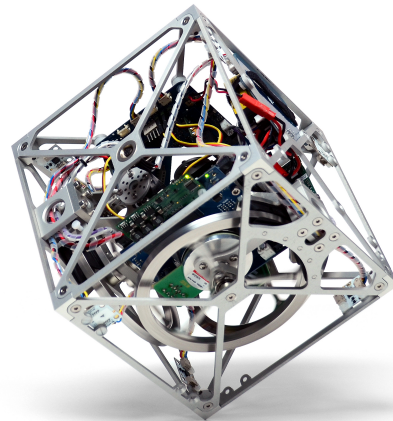

Figure 1. The Cubli balancing on a corner.

II. DYNAMICS OF THE REACTION WHEEL-BASED 3D INVERTED PENDULUM

In this section the reaction wheel-based 3D inverted pendulum dynamics are briefly outlined. After introducing the notation, the equations of motion are presented and are used to demonstrate the conservation of angular momentum. As will be pointed out, this has important consequences for control design. Additionally, in the absence of motor torques energy is conserved. This

The authors are with the Institute for Dynamic Systems and Control, ETH Zurich, Switzerland (michaemu@ethz.ch, rdandrea@ethz.ch).

[1]A video showing the Cubli can be found under https://www.youtube.com/watch?v=n_6p-1J551Y.

will become important in Section IV, where an ideal jump-up trajectory is determined via the conservation of energy.

A. Notation

Let Θ_{wi}, $i = 1, 2, 3$ denote the moment of inertia of each reaction wheel (in the direction of the corresponding rotation axis, referred to the corresponding suspension point), and define $\Theta_w := \text{diag}(\Theta_{w1}, \Theta_{w2}, \Theta_{w3})$. Let $\Theta_0 + \Theta_w$ denote the total moment of inertia of the Cubli around the pivot point O (see Figure 2). Next, let $\vec{m}$ denote the position vector from the pivot point to the center of gravity multiplied by the total mass and $\vec{g}$ denote the gravity vector. The projection of a tensor onto a particular coordinate frame is denoted by a preceding superscript, i.e. ${}^{\text{K}}\Theta_0 \in \mathbb{R}^{3\times3}$, ${}^{\text{K}}m \in \mathbb{R}^3$. The arrow notation is used to emphasize that a vector (and tensor) should be a priori thought of as a linear object in a normed vector space detached from its coordinate representation in a particular coordinate frame. The transformation matrix $R_{\text{IK}} \in SO(3)$ relates vectors from the body-fixed frame to their representation in the inertial frame, that is ${}^{\text{I}}v = R_{\text{IK}} {}^{\text{K}}v$, for all vectors ${}^{\text{K}}v \in \mathbb{R}^3$. Moreover, the skew symmetric matrix corresponding to a vector $a \in \mathbb{R}^3$, denoted by $\widetilde{a}$, is defined as $a \times b = \widetilde{a}b$, for all $b \in \mathbb{R}^3$, where $a \times b$ refers to the cross product of the two vectors a and b. The Euclidean norm is referred as $|\cdot|$, i.e. $|a|^2 = a^\mathsf{T}a$, and $a \parallel b$ is used to indicate that the two vectors $a \in \mathbb{R}^3$ and $b \in \mathbb{R}^3$ are parallel (that is $a \times b = 0$). Additionally, the sphere of radius $|g|$ is denoted by S^2.

Since the body-fixed coordinate frame $\{K\}$ is the most commonly projected coordinate frame, its preceding superscript is usually removed for ease of notation. That is, ${}^{\text{K}}m = m$, ${}^{\text{K}}\Theta_0 = \Theta_0$, etc.

Moreover, vectors are expressed as n-tuples $(x_1, x_2, \ldots, x_n)$ with dimension and stacking clear from context.

B. Equations of Motion

It was derived in [8] and [15] that the equations of motion are given by

$$\dot{p}_{\omega_h} = -\widetilde{\omega}_h p_{\omega_h} + \widetilde{m}g, \qquad \dot{p}_{\omega_w} = T, \qquad \dot{R}_{\text{IK}} = R_{\text{IK}}\widetilde{\omega}_h,$$
$$p_{\omega_h} := \Theta_0\omega_h + \Theta_w(\omega_h + \omega_w), \qquad p_{\omega_w} := \Theta_w(\omega_h + \omega_w), \quad (1)$$

where $\omega_h \in \mathbb{R}^3$ denotes the angular velocity of the Cubli housing, $\omega_w \in \mathbb{R}^3$ the angular velocity of the reaction wheels, and $T \in \mathbb{R}^3$ the motor torque applied to the reaction wheels. The fixed-body coordinate frame is aligned with the Cubli housing and therefore the first component of ω_w denotes the angular velocity of the reaction wheel pointing in ${}_K\vec{e}_1$ direction, the second component the reaction wheel pointing in ${}_K\vec{e}_2$, etc. The components of the motor torque T have a similar interpretation.

The following observations are worth pointing out: The dynamics are invariant to the initial reaction wheel positions, leading to the conservation of the angular momentum p_{ω_w} in the absence of motor torques. Moreover, the evolution of all possible initial conditions over time[2] is symmetric around the

---

[2]Commonly referred to as the flow of the system.

---

gravity vector leading to the conservation of angular momentum $p_{\omega_h}^\mathsf{T} g$. This can be easily checked by explicit calculation:

$$\frac{d}{dt}\left(p_{\omega_h}^\mathsf{T} g\right) = \dot{p}_{\omega_h}^\mathsf{T} g + p_{\omega_h}^\mathsf{T} \dot{g} = p_{\omega_h}^\mathsf{T} \widetilde{\omega}_h g - p_{\omega_h}^\mathsf{T} \widetilde{\omega}_h g = 0, \quad (2)$$

where $\dot{g}$ is expressed by $\dot{g} = \dot{R}_{\text{IK}}^\mathsf{T} {}^{\text{I}}g = -\widetilde{\omega}_h g$, or by noting that gravity exerts no torque in direction ${}_{\text{I}}\vec{e}_3$. The conservation of the angular momentum $g^\mathsf{T} p_{\omega_h}$ has an important consequence for control design: Independent of the control input applied, the momentum in direction $\vec{g}$ is conserved and, depending on the initial condition, it may be impossible to bring the system to rest. For example, a yaw motion in the upright position can be slowed down by increasing the velocity of the reaction wheels. However, the yaw motion and the reaction wheel velocity cannot be driven to zero at the same time. Note that the conservation of angular momentum in direction $\vec{g}$ is independent of the mass distribution or inertia of the Cubli and independent of the motor torque T.

In the presence of friction between the pivot point and the ground, exerting a friction torque about ${}_{\text{I}}\vec{e}_3$, the angular momentum $p_{\omega_h}^\mathsf{T} g$ is no longer conserved, and as a result, a yaw motion in the upright position will slowly decay.

In addition, in the absence of motor torques, the total energy given by

$$\mathcal{H} = \frac{1}{2}\omega_h^\mathsf{T}\Theta_0\omega_h + \frac{1}{2}(\omega_h + \omega_w)^\mathsf{T}\Theta_w(\omega_h + \omega_w) - m^\mathsf{T}g - |m|\,|g|, \quad (3)$$

is conserved. Due to the fact that $p_{\omega_w} = \Theta_w(\omega_h + \omega_w)$ is constant for $T = 0$, the energy related to the Cubli housing, given by

$$\mathcal{H}_h = \frac{1}{2}\omega_h^\mathsf{T}\Theta_0\omega_h - m^\mathsf{T}g - |m|\,|g|, \quad (4)$$

is conserved as well. Note that the energy is normalized such that it attains zero for the upright equilibrium. The conservation of energy will become important in Section IV, where it will be used to derive an ideal jump-up trajectory.

Using the gravity vector expressed in the Cubli's body-fixed coordinate frame, i.e. $g = R_{\text{IK}}^\mathsf{T} {}^{\text{I}}g$, to represent the attitude, the dynamics given by (1) can be reduced to

$$\dot{p}_{\omega_h} = -\widetilde{\omega}_h p_{\omega_h} + \widetilde{m}g, \qquad \dot{p}_{\omega_w} = T, \qquad \dot{g} = -\widetilde{\omega}_h g,$$
$$p_{\omega_h} = \Theta_0\omega_h + \Theta_w(\omega_h + \omega_w), \qquad p_{\omega_w} = \Theta_w(\omega_h + \omega_w). \quad (5)$$

This comes however at the cost of losing the yaw information. A formal treatment of this reduction step can, for example, be found in [6].

C. Equilibria

In this section the equilibria of the Cubli are briefly discussed. The reduced equations of motion (5) give rise to equilibria corresponding to limit cycles in the full configuration, so called relative equilibria, [1].

The relative equilibria are obtained by setting the right-hand side of (5) to zero, leading to

$$-\bar{\omega}_h \times \bar{p}_{\omega_h} + m \times \bar{g} = 0, \qquad \bar{T} = 0, \qquad -\bar{\omega}_h \times \bar{g} = 0, \quad (6)$$
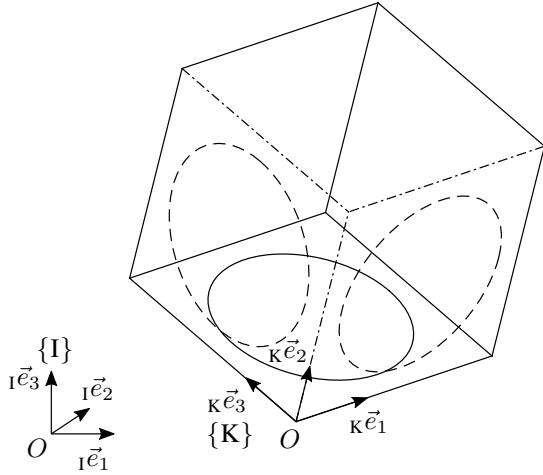
Figure 2. The Cubli balancing on its corner. The vectors $_K\vec{e}_i$ and $_I\vec{e}_i$, $i = 1, 2, 3$, denote the principle axes of the body fixed frame {K} and inertial frame {I}. The pivot point O is the common origin of coordinate frames {I} and {K}. For illustration purposes the coordinate system {I} is shifted to the left.

where $\bar{g}$, $\bar{p}_{\omega_h}$, and $\bar{\omega}_h$ denote the equilibrium configurations. The last equation implies that $\bar{\omega}_h \parallel \bar{g}$ or likewise $\bar{\omega}_h = \lambda_1 \bar{g}$, with $\lambda_1 \in \mathbb{R}$. Thus, the relative equilibria are characterized by

$$\bar{\omega}_h = \lambda_1 \bar{g}, \quad \lambda_1 \bar{p}_{\omega_h} + m = \lambda_2 \bar{g}, \quad \bar{T} = 0, \qquad (7)$$

with $\lambda_1, \lambda_2 \in \mathbb{R}$, $\bar{g} \in S^2$, and $\bar{\omega}_h, \bar{p}_{\omega_h}, \bar{T} \in \mathbb{R}^3$. The hanging and upright equilibria, which are of interest for the remainder of this article, are obtained by setting $\lambda_1 = 0$ implying $\bar{g} \parallel m$. As expected, a linear analysis reveals that the upright equilibrium is unstable, while the hanging equilibrium is marginally stable.

III. NONLINEAR CONTROL

In the next section two different control strategies are presented, which asymptotically stabilize the upright equilibrium. The first approach is based on backstepping and provides a smooth control law characterized by four tuning parameters. In a subsequent step the tuning parameters are related to the closed-loop behavior, extending the result presented in [15]. The second approach is based on feedback linearization and extends the result in [20] to the 3D case.

For the control design and subsequent analysis the reduced dynamics (5) are used. The state space is chosen to be $(g, p_{\omega_h}, p_{\omega_w}) \in \mathcal{X} := S^2 \times \mathbb{R}^3 \times \mathbb{R}^3$. By using the reduced attitude representation, the feedback control laws derived next will naturally be invariant to the orientation around the gravity vector and to the reaction wheel positions.

Since the component of the angular momentum p_{ω_h} in the direction of gravity is a conserved quantity, only the component of p_{ω_h} that is orthogonal to g can be affected by feedback control. Hence, it is convenient to split the angular momentum p_{ω_h} into two parts: one in the direction of gravity, and one orthogonal to it, i.e.

$$p_{\omega_h} =: p_{\omega_h}^\perp + p_{\omega_h}^g \frac{g}{|g|}, \qquad p_{\omega_h}^g := p_{\omega_h}^\mathsf{T} \frac{g}{|g|}. \qquad (8)$$

The control objective consists of balancing the Cubli in the upright position, and at the same time requiring $\omega_h \to 0$

together with $p_{\omega_h}^\perp \to 0$ as time goes to infinity. Thus, the control objective for balancing can be formulated as driving the system to the closed invariant set

$$\mathcal{T} = \{(g, p_{\omega_h}, p_{\omega_w}) \in \mathcal{X} \mid g^\mathsf{T} m = -|g| \, |m|,$$
$$p_{\omega_h}^\perp = 0, \; p_{\omega_h} = p_{\omega_w}\}. \qquad (9)$$

Note that ω_h is given by $\Theta_0^{-1}(p_{\omega_h} - p_{\omega_w})$ and therefore $p_{\omega_h} = p_{\omega_w}$ implies zero angular velocity of the Cubli housing.

A. Backstepping Approach

In the following section a nonlinear controller is presented, which stabilizes the set $\mathcal{T}$ asymptotically. In a subsequent step its closed-loop behavior is analyzed leading to a geometric interpretation of closed-loop trajectories and a simple tuning strategy.

For ease of notation, the hanging relative equilibria with $\omega_h = 0$ are denoted by x^-, i.e.

$$x^- = \{(g, p_{\omega_h}, p_{\omega_w}) \in \mathcal{X} \mid g = \frac{|g|}{|m|}m, \; p_{\omega_h}^\perp = 0, \; p_{\omega_h} = p_{\omega_w}\}.$$

Next, the control law

$$T = K_1 \widetilde{m} g + K_2 \omega_h + K_3 p_{\omega_h} - K_4 p_{\omega_w}, \qquad (10)$$

with

$$K_1 = I + (\alpha + \beta\gamma + \delta)\Theta_0,$$
$$K_2 = \Theta_0 (\alpha \widetilde{p}_{\omega_h}^\perp + \beta \widetilde{m} \widetilde{g}) + \widetilde{p}_{\omega_h},$$
$$K_3 = \gamma(I + \alpha\Theta_0(I - \frac{gg^\mathsf{T}}{|g|^2})),$$
$$K_4 = \gamma I, \quad \alpha, \beta, \gamma, \delta > 0,$$

and $I \in \mathbb{R}^{3\times3}$ the identity matrix, is shown to asymptotically stabilize the upright equilibrium. More precisely:

Theorem 3.1: The controller (10) renders the closed invariant set $\mathcal{T}$ of the system (5) stable and asymptotically stable on $x \in \mathcal{X} \setminus x^-$.

Proof Consider the following Lyapunov candidate function $V : \mathcal{X} \to \mathbb{R}$,

$$V(x) = \frac{1}{2}\alpha p_{\omega_h}^{\perp\mathsf{T}} p_{\omega_h}^\perp + m^\mathsf{T} g + |m| \, |g| + \frac{1}{2\delta}z^\mathsf{T}\Theta_0^{-2}z, \quad (11)$$

with $z := \Theta_0 (\alpha p_{\omega_h}^\perp + \beta \widetilde{m} g) + p_{\omega_h} - p_{\omega_w}$.

Clearly, there exists a $\mathcal{K}_\infty$ function[3] $a : [0, \infty) \to [0, \infty)$ such that $V(x) \geq a(|x - x_0|)$ for all $x \in \mathcal{X}$ and all $x_0 \in \mathcal{T}$. Furthermore $V(x = x_0) = 0$ implies $x = x_0$, where $x_0 \in \mathcal{T}$. Therefore V is a positive definite function and a valid Lyapunov candidate.

Next, $\dot{V}$ is evaluated along trajectories of the closed-loop system:

$$\dot{V}(x) = \alpha p_{\omega_h}^{\perp\mathsf{T}} \dot{p}_{\omega_h}^\perp + m^\mathsf{T}\dot{g} + \frac{1}{\delta}z^\mathsf{T}\Theta_0^{-2}\dot{z}$$
$$= m^\mathsf{T}\widetilde{g}(\alpha p_{\omega_h}^\perp + \omega_h) + \frac{1}{\delta}z^\mathsf{T}\Theta_0^{-2}\dot{z}.$$

[3]A continuous function belongs to class $\mathcal{K}_\infty$ if it is strictly increasing and radially unbounded, see e.g. [10, Definition 4.2, p. 144].

From the identity $\Theta_0^{-1}z = \alpha p_{\omega_h}^\perp + \beta \widetilde{m}g + \omega_h$ it follows that

$$\dot{V}(x) = m^\mathsf{T}\widetilde{g}(\beta\widetilde{g}m + \Theta_0^{-1}z) + \frac{1}{\delta}z^\mathsf{T}\Theta_0^{-2}\dot{z}$$
$$= -\beta(\widetilde{g}m)^\mathsf{T}(\widetilde{g}m) + z^\mathsf{T}\Theta_0^{-1}\widetilde{m}g + \frac{1}{\delta}z^\mathsf{T}\Theta_0^{-2}\dot{z}.$$

Moreover, the control input T can be rewritten as

$$T = \frac{d}{dt}(z + p_{\omega_w}) + \gamma z + \delta\Theta_0\widetilde{m}g. \tag{12}$$

Using the fact that $\dot{p}_{\omega_w} = T$, the closed loop evolution of the auxiliary variable z is given by

$$\dot{z} = -\gamma z - \delta\Theta_0\widetilde{m}g, \tag{13}$$

which can be used to simplify $\dot{V}$ to

$$\dot{V}(x) = -\beta(\widetilde{g}m)^\mathsf{T}(\widetilde{g}m) - \frac{\gamma}{\delta}z^\mathsf{T}\Theta_0^{-2}z \le 0, \qquad \forall x \in \mathcal{X}.$$

Since $\dot{V}(x) \le 0$, for all $x \in \mathcal{X}$, we conclude from Lyapunov's stability theorem, [10, Theorem 4.8] that the equilibria $x_0 \in \mathcal{T}$ are stable.

To prove asymptotic stability of the set $\mathcal{T}$ for $x \in \mathcal{X} \setminus x^-$, the set

$$\mathcal{R} := \{x \in \mathcal{X} \setminus x^- \mid \dot{V}(x) = 0\} \tag{14}$$

is considered in more detail. From $\dot{V}(x) < 0$ for all $x \in \mathcal{X} \setminus (\mathcal{R} \bigcup x^-)$ it can be inferred that any trajectory in $\mathcal{X} \setminus x^-$ is converging to an invariant set contained in $\mathcal{R}$. The condition $\dot{V}(x) = 0$ leads to $z = 0$, m parallel g, such that $\mathcal{R}$ can be rewritten as $\mathcal{R} = \{x \in \mathcal{X} \setminus x^- \mid m \parallel g,\ p_{\omega_w} = \alpha\Theta_0 p_{\omega_h}^\perp + p_{\omega_h}\}$. The dynamics on $\mathcal{R}$ can be simplified to:

$$g \parallel m \Rightarrow g = -\frac{m}{|m|}|g| \Rightarrow \dot{g} = 0$$
$$\Rightarrow \omega_h \parallel g \quad \text{because} \quad \dot{g} = -\widetilde{\omega}_h g \tag{15}$$
$$g \parallel m,\ z = 0 \Rightarrow \omega_h = \alpha p_{\omega_h}^\perp$$
$$\Rightarrow \omega_h \parallel p_{\omega_h}^\perp \tag{16}$$

However, since $p_{\omega_h}^\perp$ is orthogonal to g by definition, equations (15) and (16) imply $\omega_h = 0$ and $p_{\omega_h}^\perp = 0$. Therefore $\mathcal{T}$ is the largest invariant set contained in $\mathcal{R}$. This implies by the Krasovskii–LaSalle principle [10, Theorem 4.4], that for any trajectory $x(t)$,

$$\lim_{t \to \infty} x(t) = x_f, \quad x(0) \in \mathcal{X} \setminus x^-, \quad x_f \in \mathcal{T}.$$

∎

1) Remarks:
a) Interpretation of the Lyapunov Function:
The Lyapunov function given by (11) can be found via a backstepping approach, see for example [10] or [11] for an introduction to backstepping. The reduced Lyapunov function

$$V_R(x) = \frac{1}{2}\alpha p_{\omega_h}^{\perp\mathsf{T}}p_{\omega_h}^\perp + m^\mathsf{T}g + |m|\,|g|, \tag{17}$$

which is independent of the momentum p_{ω_w} can be used to demonstrate stability given that $p_{\omega_w} = \alpha\Theta_0 p_{\omega_h}^\perp + p_{\omega_h} + \beta\widetilde{m}g$ (corresponding to $z = 0$). Therefore, z accounts for the momentum p_{ω_w} and penalizes indirectly non-zero wheel velocities.

b) Extension of the Controller:
In practice, modeling errors can cause steady-state deviations, e.g. an erroneous estimate of the center of gravity leads to non-vanishing steady-state reaction wheel velocities when balancing. Integral control can be used to prevent these steady-state deviations. Therefore the controller is extended with the state z_{int}, i.e. $\hat{u} = u + \nu z_{\text{int}}$, where

$$z_{\text{int}}(t) = z_0 + \int_0^t z(\tau)d\tau$$

and $\nu > 0$. In that case, closed-loop stability can be proved by augmenting the Lyapunov function given by (11):

$$V_I(x) = V(x) + \frac{\nu}{2\delta}z_{\text{int}}^\mathsf{T}\Theta_0^{-2}z_{\text{int}}.$$

In [15] an alternative approach to account for non-zero steady-state wheel velocities is presented, which has the advantage of directly providing an estimate of the center of gravity.

c) Interpretation of the Control Law:
Rewriting (10) yields

$$u = \dot{p}_{\omega_h} + \gamma p_{\omega_h} + \alpha\Theta_0(\dot{p}_{\omega_h}^\perp + \gamma p_{\omega_h}^\perp)$$
$$+ \Theta_0\widetilde{m}(\beta\dot{g} + (\delta + \gamma\beta)g) - \gamma p_{\omega_w}, \tag{18}$$

where

$$p_{\omega_w} = u_0 + \int_0^t u(\tau)d\tau. \tag{19}$$

Therefore the controller given by (10) is a linear PID controller in the variables $p_{\omega_w}, p_{\omega_h}^\perp$ and g. The only nonlinearity of the controller lies in the projection of p_{ω_h} into $p_{\omega_h}^\perp$ and $p_{\omega_h}^g$. Nevertheless, the control law guarantees global asymptotic stability (almost everywhere) as has been shown previously.

2) Closed-loop behavior: Due to its smoothness and its dependence on only four tuning parameters, the controller is well-suited for practical implementation. A simple tuning strategy based on the closed-loop behavior is outlined next. We will analyze the closed-loop response subject to two different initial conditions, providing an interpretation of the tuning parameters. In the first case, the Cubli will be released at rest, but with a non-zero inclination angle. For this specific initial condition the closed-loop dynamics of the inclination angle are given by a third-order differential equation, which allows for pole placement. It will be shown that there is a set of tuning parameters matching every desired pole location (provided that the desired poles have negative real parts). This determines three of the four tuning parameters (α, β and δ). In the second case, a pure yaw motion will be analyzed and related to the remaining tuning parameter γ.

Proposition 3.2: Consider the controller (10) applied to the system governed by (5) with initial conditions at $t = 0$ such that $p_{\omega_h}(0)$ and $\omega_h(0)$ are parallel to $m \times g(0) \ne 0$. Then it holds for all $t > 0$ that $\omega_h(t)$, $p_{\omega_h}(t)$, and $m \times g(t)$ remain parallel.

Proof Since $p_{\omega_h}(0) \parallel m \times g(0)$ it implies that $p_{\omega_h}^\perp(t) = p_{\omega_h}(t)$ for all $t > 0$. Moreover, by combining the control law given by (10) with the system dynamics it follows that

$$\dot{\omega}_h = \Theta_0^{-1}(\dot{p}_{\omega_h} - T)$$
$$= \alpha\omega_h \times p_{\omega_h} - (\alpha + \beta\gamma + \delta)m \times g + \beta m \times (\omega_h \times g)$$
$$- \gamma(\alpha p_{\omega_h} + \omega_h), \tag{20}$$

4

together with

$$\dot{p}_{\omega_h} = p_{\omega_h} \times \omega_h + m \times g \quad \text{and} \quad \frac{d}{dt}(m \times g) = m \times (g \times \omega_h).$$

Note also that from the Lagrange identity, [9],

$$m \times (g \times (m \times g)) = -m^\mathsf{T} g \, m \times g \qquad (21)$$

follows. Assume that $p_{\omega_h}(t^*)$, $\omega_h(t^*)$, and $m \times g(t^*)$ are parallel at time $t = t^*$. Together with equations (20)-(21) these assumptions imply that

$$\frac{d}{dt}(m \times g(t^*)) \parallel m \times g(t^*), \qquad (22)$$

$$\dot{\omega}_h(t^*) \parallel m \times g(t^*), \quad \text{and} \qquad (23)$$

$$\dot{p}_{\omega_h}(t^*) \parallel m \times g(t^*). \qquad (24)$$

Hence, $p_{\omega_h}(t)$, $\omega_h(t)$, and $m \times g(t)$ will remain parallel for an infinitesimal time increment $\mathrm{d}t$, that is at time $t = t^* + \mathrm{d}t$. By induction, the vectors $p_{\omega_h}(t)$, $\omega_h(t)$, and $m \times g(t)$ will therefore remain parallel for all times $t > t^*$. Note that the right-hand side of the closed-loop dynamics is locally Lipschitz, which implies the local existence and uniqueness of closed-loop trajectories, [17]. Since the initial conditions at $t = 0$ are such that $p_{\omega_h}(0)$, $\omega_h(0)$, and $m \times g(0)$ are parallel, the result follows. ∎

Note that the previous proposition applies especially in the case where the Cubli is initialized with zero body angular velocity and zero wheel velocity ($\omega_h(0) = \omega_w(0) = 0$), and states that the Cubli's center of mass will never leave the plane normal to $m \times g(0)$ for all times $t > 0$. This sets the stage for deriving a differential equation describing the inclination angle in closed-loop provided that p_{ω_h}, ω_h, and $m \times g$ are parallel at $t = 0$.

It is convenient to introduce the unit vector

$$e_\varphi := \frac{m \times g(0)}{|m \times g|}, \qquad \text{where} \quad m \times g(0) \neq 0, \qquad (25)$$

and define the inclination angle by

$$\varphi := \arccos\left(-\frac{m^\mathsf{T} g}{|m| \, |g|}\right), \qquad (26)$$

with $\varphi \in [0, \pi]$ for $g \in S^2$. Note that

$$\sin \varphi = \frac{|-g \times m|}{|m| \, |g|} = \frac{|m \times g|}{|m| \, |g|} \qquad (27)$$

holds. By Proposition 3.2 it follows that ω_h is parallel to $m \times g$ and e_φ for all times $t > 0$. Furthermore, from (26) and the system dynamics (5) it can be confirmed that $\omega_h = \dot{\varphi} e_\varphi$. Rewriting (20) yields

$$\begin{aligned}
\dot{\omega}_h &= \alpha \omega_h \times p_{\omega_h} - (\alpha + \beta\gamma + \delta)m \times g \\
&\quad + \beta m \times (\omega_h \times g) - \gamma(\alpha p_{\omega_h} + \omega_h) \\
&= -e_\varphi(\alpha + \beta\gamma + \delta)|m| \, |g| \sin\varphi \\
&\quad - e_\varphi \beta |m| \, |g| \dot{\varphi} \cos\varphi - \gamma(\alpha p_{\omega_h} + e_\varphi \dot{\varphi}).
\end{aligned} \qquad (28)$$

Taking the time derivative of the previous equation and using the fact that $\dot{e}_\varphi = 0$ and $\dot{p}_{\omega_h} = e_\varphi |m| \, |g| \sin\varphi$ results in

$$\begin{aligned}
\dddot{\varphi} &+ (\beta|m| \, |g| \cos\varphi + \gamma)\ddot{\varphi} + (\alpha + \beta\gamma + \delta)|m| \, |g| \dot{\varphi} \cos\varphi \\
&- \beta|m| \, |g| \dot{\varphi}^2 \sin\varphi + \gamma\alpha|m| \, |g| \sin\varphi = 0.
\end{aligned} \qquad (29)$$

Linearizing (29) around the upright equilibrium, i.e. $\varphi = 0$ yields

$$\dddot{\varphi} + (\beta|m| \, |g| + \gamma)\ddot{\varphi} + (\alpha + \beta\gamma + \delta)|m| \, |g| \dot{\varphi} + \gamma\alpha|m| \, |g|\varphi = 0 \qquad (30)$$

and provides a method to relate the closed-loop poles to the parameters $\{\alpha, \beta, \gamma, \delta\}$. To simplify notation the following scaling is introduced, $\hat{\alpha} := \alpha|m| \, |g|$, $\hat{\beta} := \beta|m| \, |g|$, $\hat{\gamma} := \gamma$, and $\hat{\delta} := \delta|m| \, |g|$, such that (30) reads as

$$\dddot{\varphi} + (\hat{\beta} + \hat{\gamma})\ddot{\varphi} + (\hat{\alpha} + \hat{\beta}\hat{\gamma} + \hat{\delta})\dot{\varphi} + \hat{\gamma}\hat{\alpha}\varphi = 0. \qquad (31)$$

Moreover, the parameter $\hat{\gamma}$ is related to the closed-loop yaw motion, by considering the case where the Cubli is initialized in an upright relative equilibrium, with non-zero angular velocity, $\omega_h(0) \neq 0$. Hence, it follows that $\omega_h(0) \parallel p_{\omega_h}(0) \parallel g(0) \parallel m$ and that the closed-loop dynamics read as

$$\dot{g} = 0, \quad \dot{p}_{\omega_h} = 0, \quad \text{and} \quad \dot{p}_{\omega_w} = -\hat{\gamma}(p_{\omega_w} - p_{\omega_h}). \qquad (32)$$

This leads to the interpretation of $\hat{\gamma}$ as a time constant prescribing how fast the yaw rotation is slowed down.

Ideally, the parameters $\hat{\alpha}, \hat{\beta}, \hat{\gamma}, \hat{\delta}$ are chosen such that the desired closed-loop poles of the inclination angle are matched and that a prescribed time constant of the closed-loop yaw motion is met. However, it turns out that depending on $\hat{\gamma}$, this might be impossible, i.e. for a fixed $\hat{\gamma} > 0$ it might be impossible to obtain $\hat{\alpha}, \hat{\beta}, \hat{\gamma}, \hat{\delta}$ such that the pole configuration is met, while guaranteeing nonlinear closed-loop stability with the proposed controller. This fact is illustrated in the following.

For given closed-loop pole locations of the inclination angle, let the third order characteristic polynomial corresponding to (31) be denoted by

$$s^3 + As^2 + Bs + C = 0, \qquad (33)$$

where the coefficients $\{A, B, C\}$ are related to the pole locations by a homeomorphism. Therefore it is sufficient to analyze the function[4] $f : \mathbb{R}_+^4 \to \mathbb{R}_+^3$ mapping the tuning parameters $\{\hat{\alpha}, \hat{\beta}, \hat{\gamma}, \hat{\delta}\}$ to the constants $\{A, B, C\}$. According to the Routh-Hurwitz criterion the poles have strictly negative parts if and only if the conditions $A > 0, B > 0, AB > C > 0$ are fulfilled, see e.g. [14]. Clearly, if $\hat{\alpha}, \hat{\beta}, \hat{\gamma}, \hat{\delta} > 0$, it follows that $A > 0, B > 0, AB > C > 0$, which corresponds to a stable pole configuration (as expected, nonlinear closed-loop stability implies linear closed-loop stability). The converse is not true; for a fixed $\hat{\gamma} > 0$ there might be no $\hat{\alpha}, \hat{\beta}, \hat{\delta} > 0$, such that the desired pole location is matched. This fact is illustrated by expressing the level set of $f(\hat{\alpha}, \hat{\beta}, \hat{\gamma}, \hat{\delta}) = (A, B, C)$ as

$$\left\{ (\hat{\alpha}, \hat{\beta}, \hat{\gamma}, \hat{\delta}) \in \mathbb{R}^4 \,\middle|\, \hat{\alpha}\hat{\gamma} = C, \; \hat{\beta} = A - \hat{\gamma}, \right. \\
\left. \hat{\gamma}\hat{\delta} = \hat{\gamma}^3 - A\hat{\gamma}^2 + B\hat{\gamma} - C \right\}. \qquad (34)$$

Hence, given that $A, B, C > 0$ the condition $\hat{\alpha}, \hat{\beta}, \hat{\gamma}, \hat{\delta} > 0$ reduces to

$$A > \hat{\gamma} > 0, \quad h(\hat{\gamma}) := \hat{\gamma}^3 - A\hat{\gamma}^2 + B\hat{\gamma} - C > 0. \qquad (35)$$

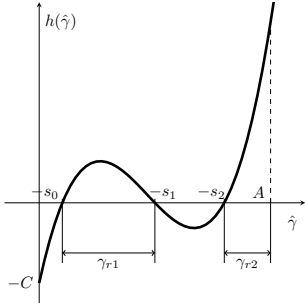[4]The positive real numbers are denoted by $\mathbb{R}_+ := \{x \in \mathbb{R} | x > 0\}$.

5

Figure 3. Example for a desired pole configuration with three real poles. The admissible regions for $\hat{\gamma}$ are denoted by γ_{r1} and γ_{r2}.
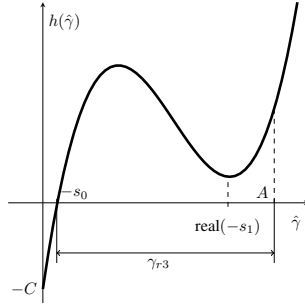

Figure 4. Example for a desired pole configuration with one complex pole pair. The admissible region for $\hat{\gamma}$ is denoted by γ_{r3}.

Note that $h(-s) = -(s^3 + As^2 + Bs + C)$ holds, which implies that the zeros of $h(\hat{\gamma})$ are just the negative values of the desired poles. Thus, if the desired pole locations, s_0, s_1, and s_2, with $s_2 < s_1 < s_0 < 0$, are all real and distinct, then there are two different $\hat{\gamma}$-regions, e.g. $\hat{\gamma} \in (-s_0, -s_1)$, $\hat{\gamma} \in (-s_2, A)$, where $h(\hat{\gamma}) > 0$, see Fig. 3. If there are two complex conjugated poles or non-distinct poles then there might be only one $\hat{\gamma}$-region, where $h(\hat{\gamma}) > 0$, see Fig. 4.

Note that in all cases $\hat{\gamma}$ needs to be greater than $\min_i\{-\text{real}(s_i)\}$, where s_0, s_1, s_2 are the desired pole locations. Hence, the closed-loop yaw motion needs to have a time constant at least as fast as the smallest pole of the (closed-loop) inclination angle dynamics, in order to guarantee global closed-loop stability with the proposed controller.

Summarizing, the following tuning recipe is proposed:

1) Choose the desired pole locations of the closed-loop inclination angle dynamics, which determines possible intervals for $\hat{\gamma}$.
2) Choose $\hat{\gamma}$ within those intervals such that the time constant of the yaw-dynamics matches the desired one as close as possible. Solving (34) yields the parameters $\{\hat{\alpha}, \hat{\beta}, \hat{\gamma}, \hat{\delta}\}$.

B. Feedback Linearization

Next an explicit input-to-state feedback linearization is found extending the result presented in [20]. The generalized momentum p_{ω_h} is chosen to be the virtual output. However, to remove the conserved component (in direction $\vec{g}$), it is convenient to project p_{ω_h} in the inertial frame, where the dynamics of the Cubli are given by

$$\begin{aligned}
{}^{\text{I}}\dot{m} &= {}^{\text{I}}\omega_h \times {}^{\text{I}}m, \\
{}^{\text{I}}\dot{p}_{\omega_h} &= {}^{\text{I}}m \times {}^{\text{I}}g, \\
{}^{\text{I}}\dot{p}_{\omega_w} &= {}^{\text{I}}T + {}^{\text{I}}\omega_h \times {}^{\text{I}}p_{\omega_w}.
\end{aligned} \quad (36)$$

The virtual output y is formed by the first two elements of ${}^{\text{I}}p_{\omega_h}$, i.e.

$$y := \left({}^{\text{I}}p_{\omega_h 1}, {}^{\text{I}}p_{\omega_h 2}\right), \quad (37)$$

since the third component of ${}^{\text{I}}p_{\omega_h}$ is conserved. This choice can be motivated by the feedback linearization of the 1D reaction

wheel-based inverted pendulum presented in [20]. Using the matrices

$$J = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad \text{and} \quad P = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \quad (38)$$

the first two components of the cross product $a \times b$ with $a \in \mathbb{R}^3$, $b \in \mathbb{R}^3$ can be expressed by

$$P(a \times b) = -a_3 J P b + b_3 J P a. \quad (39)$$

Thus, $P({}^{\text{I}}m \times {}^{\text{I}}g)$ simplifies to $P({}^{\text{I}}m \times {}^{\text{I}}g) = -|g| J P\, {}^{\text{I}}m$.

Taking the time derivative of y, $\dot{y}$, and $\ddot{y}$ leads to

$$\dot{y} = -|g| J P\, {}^{\text{I}}m, \quad (40)$$

$$\ddot{y} = -|g| J P({}^{\text{I}}\omega_h \times {}^{\text{I}}m), \quad (41)$$

$$\dddot{y} = -|g| J P({}^{\text{I}}\dot{\omega}_h \times {}^{\text{I}}m + {}^{\text{I}}\omega_h \times {}^{\text{I}}\dot{m}). \quad (42)$$

Additionally, ${}^{\text{I}}\dot{\omega}_h$ is given by

$$\begin{aligned}
{}^{\text{I}}\dot{\omega}_h &= R_{\text{IK}}\Theta_0^{-1}(\dot{p}_{\omega_h} - \dot{p}_{\omega_w}) \\
&= R_{\text{IK}}\Theta_0^{-1}(m \times g - \omega_h \times p_{\omega_h} - T).
\end{aligned} \quad (43)$$

Solving for the input torque T, i.e. using the change of variable $T \to {}^{\text{I}}v$ with

$$T = -\omega_h \times p_{\omega_h} + m \times g - \Theta_0 R_{\text{IK}}^{\mathsf{T}}\, {}^{\text{I}}v \quad (44)$$

leads to ${}^{\text{I}}\dot{\omega}_h = {}^{\text{I}}v$. Using the identity given by (39) allows us to rewrite (42) as

$$\dddot{y} = |g| \left({}^{\text{I}}m_3 P\, {}^{\text{I}}v - {}^{\text{I}}v_3 P\, {}^{\text{I}}m - J P\, {}^{\text{I}}\widetilde{\omega}_h\, {}^{\text{I}}\widetilde{\omega}_h\, {}^{\text{I}}m\right). \quad (45)$$

Choosing the first two components of ${}^{\text{I}}v$ to be

$$P{}^{\text{I}}v = \frac{1}{{}^{\text{I}}m_3}\left({}^{\text{I}}v_3 P\, {}^{\text{I}}m + J P\, {}^{\text{I}}\widetilde{\omega}_h\, {}^{\text{I}}\widetilde{\omega}_h\, {}^{\text{I}}m + \frac{1}{|g|}w\right) \quad (46)$$

with $w \in \mathbb{R}^2$ leads to $\dddot{y} = w$. Note that the transformation is not defined for ${}^{\text{I}}m_3 = 0$. This parallels the 1D case, where it was shown that a feedback linearization exists only for an inclination angle φ such that $\varphi \neq \pm\frac{\pi}{2}$, see [20].

Hence, by choosing the state transformation

$$x = (y, \dot{y}, \ddot{y}, {}^{\text{I}}\omega_{h3}), \quad (47)$$

together with the input transformation given by (44) and (46) the following linear system dynamics are obtained for the case ${}^{\text{I}}m_3 \neq 0$:

$$\dot{x} = \begin{pmatrix}
0_{2\times2} & I_{2\times2} & 0_{2\times2} & 0_{2\times1} \\
0_{2\times2} & 0_{2\times2} & I_{2\times2} & 0_{2\times1} \\
0_{2\times2} & 0_{2\times2} & 0_{2\times2} & 0_{2\times1} \\
0_{1\times2} & 0_{1\times2} & 0_{1\times2} & 0
\end{pmatrix} x + \begin{pmatrix} 0_{4\times3} \\ I_{3\times3} \end{pmatrix} \begin{pmatrix} w \\ {}^{\text{I}}v_3 \end{pmatrix}. \quad (48)$$

IV. JUMP UP

By suddenly braking its reaction wheels spinning at high angular velocities, the Cubli is able to "jump up" from lying flat to its upright equilibrium as shown in Figure 5.

The jump-up is divided into two parts: the *braking phase*, where the reaction wheels are almost instantaneously slowed down and the *guiding phase*, where additional control action is used to guide the Cubli to its upright equilibrium. Identifying
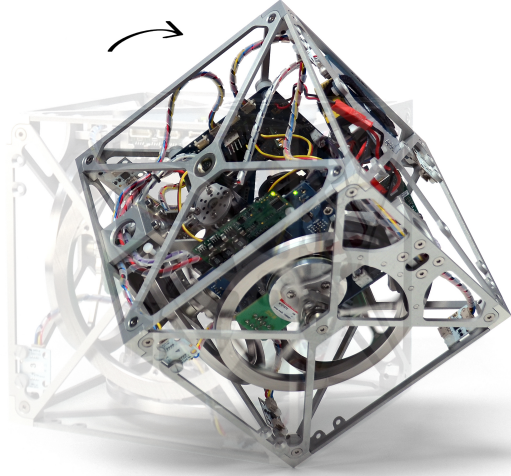
6

Figure 5. The Cubli jumping from lying flat to its upright equilibrium.

and modeling the braking phase exactly is difficult due to large process uncertainties such as the friction between the brake and the wheel, the timing of the different brakes and the inaccuracies in the state estimation due to high accelerations. However, these uncertainties are mostly time invariant and can therefore be circumvented by using a low-order model in combination with a learning algorithm. The learning algorithm accounts therefore for the repeatable modeling errors, and is used to adapt the initial wheel velocities of the reaction wheels.

To further improve the reliability of the jump-up, an ideal trajectory is tracked during the guiding phase using feedback linearization. Compared to a linear reference tracking approach, this has the advantage of providing a time-invariant control law.

The next section is divided into the following parts. First, a low-complexity model for the jump-up is outlined for both the braking and the guiding phase. Then, the learning framework is introduced and discussed in general, before being applied to the Cubli jump-up.

A. Impact-based Braking Model

The jump-up is modeled by assuming that the reaction wheels are stopped instantaneously. To simplify the analysis further, it is assumed that after braking the angular momentum associated with the reaction wheels is zero, that is $p_{\omega_w}(0)^+ = \Theta_w(\omega_h(0)^+ + \omega_w(0)^+) = 0$. This assumption is used to determine an ideal jump-up trajectory; it guarantees the conservation of angular momentum around the figure axis in the absence of control inputs, reducing the Cubli model to a symmetric spherical pendulum (see Section IV-B). Note that this assumption is not entirely fulfilled since in reality the wheel speed $\omega_w(0)^+$ is actually zero after braking. Compared to the reaction wheel momentum before braking, $p_{\omega_w}(0)^+$ is however negligible. The braking is assumed to happen at the time instant 0; $\omega_w(0)^-$ and $\omega_w(0)^+$ denote the left and right limits of the reaction wheel angular velocity ω_w. Note that the

left and right limits of a discontinuous function f (of locally bounded variation) are defined by

$$f(0)^- := \lim_{t\uparrow 0} f(t) \qquad \text{and} \qquad f(0)^+ := \lim_{t\downarrow 0} f(t). \qquad (49)$$

The impact is modeled by using conservation of angular momentum. More formally, an impact torque density $\mathrm{d}\Lambda$ ($[\mathrm{d}\Lambda] = \mathrm{Nms}$) is introduced and the equations of motion given by (5) are integrated over the impact time singleton $\{0\}$. This yields

$$\int_{\{0\}} \mathrm{d}p_{\omega_h} = p_{\omega_h}(0)^+ - p_{\omega_h}(0)^- \qquad (50)$$
$$= \int_{\{0\}} (-\widetilde{\omega}_h p_{\omega_h} + \widetilde{m}g)\, \mathrm{d}t = 0,$$

$$\int_{\{0\}} \mathrm{d}p_{\omega_w} = p_{\omega_w}(0)^+ - p_{\omega_w}(0)^- \qquad (51)$$
$$= \int_{\{0\}} (T\mathrm{d}t + \mathrm{d}\Lambda) = \Lambda(0)^+ - \Lambda(0)^-,$$

where $\mathrm{d}p_{\omega_h}$ and $\mathrm{d}p_{\omega_w}$ are the differential measures of p_{ω_h} and p_{ω_w}, containing a density with respect to the Lebesgue measure $\mathrm{d}t$ and the atomic measure $\mathrm{d}\eta$, i.e.

$$\mathrm{d}p_{\omega_h} = \dot{p}_{\omega_h}\mathrm{d}t + (p_{\omega_h}^+ - p_{\omega_h}^-)\mathrm{d}\eta, \qquad (52)$$
$$\mathrm{d}p_{\omega_w} = \dot{p}_{\omega_w}\mathrm{d}t + (p_{\omega_w}^+ - p_{\omega_w}^-)\mathrm{d}\eta.$$

The time singleton $\{0\}$ has zero Lebesgue measure. By assumption, it holds that $\Theta_w(\omega_h(0)^+ + \omega_w(0)^+) = p_{\omega_w}(0)^+ = 0$. Since the Cubli is at rest when activating the brakes, $\omega_h(0)^- = 0$, and therefore (50) yields

$$p_{\omega_h}(0)^+ = \Theta_0 \omega_h(0)^+ = p_{\omega_h}(0)^- = \Theta_w \omega_w(0)^-, \qquad (53)$$

which relates the body angular momentum after braking to the initial wheel velocity.

B. Guiding Phase

During the guiding phase, the Cubli is guided along an "ideal" trajectory to the upright equilibrium. The trajectory is tracked using feedback linearization, resulting in a time-invariant control law. Next, this predefined trajectory is derived by using first integrals of the equations of motion.

To simplify the analysis, the following assumption is made:

Assumption 4.1: (Symmetric housing inertia) The inertia tensor Θ_0 has an eigenvector in direction m. The associated eigenvalue is denoted by I_3. The remaining two eigenvalues are equal, i.e. $I_1 = I_2$.

In case $T = 0$ and $p_{\omega_w} = 0$, this assumption leads to an additional conserved quantity, which is nothing but the angular momentum around the figure axis, i.e.

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(m^\mathsf{T} p_{\omega_h}\right) = m^\mathsf{T}(\omega_h \times p_{\omega_h}) \qquad (54)$$
$$= m^\mathsf{T}\widetilde{\omega}_h \Theta_0 \omega_h + m^\mathsf{T}\widetilde{\omega}_h p_{\omega_w} = 0,$$

where the first term of the previous expression vanishes due to Assumption 4.1 and the second due to the fact that $p_{\omega_w} = 0$.

The "ideal" trajectory is defined as the trajectory leading from the state just after braking, i.e. the right limit at time $t = 0$, to the upright equilibrium without using any motor torque. By

7

assumption, the right limit of p_{ω_w} vanishes at time $t = 0$, which implies that $p_{\omega_w}(t)$ remains zero for all $t > 0$, see Section II. In the absence of motor torque, energy, the angular momentum in direction $\vec{g}$, and the angular momentum in direction $\vec{m}$ are conserved (see (2), (4), and (54)), that is

$$
\begin{aligned}
\mathcal{H}_h &= \frac{1}{2}\omega_h^{\mathsf{T}}\Theta_0\omega_h - m^{\mathsf{T}}g - |m|\ |g| = \text{const}, \\
p_{\omega_h}^g &= p_{\omega_h}^{\mathsf{T}}\frac{g}{|g|} = \text{const}, \qquad p_{\omega_h}^m = p_{\omega_h}^{\mathsf{T}}\frac{m}{|m|} = \text{const}.
\end{aligned}
\tag{55}
$$

In other words, the Cubli is modeled as a symmetric spherical pendulum during the guiding phase. It has as many first integrals as degrees of freedom. This suggests to parametrize the attitude of the Cubli by the inclination angle

$$
\varphi := \arccos\left(\frac{-m^{\mathsf{T}}g}{|m|\ |g|}\right) \in [0, \pi].
\tag{56}
$$

Since the ideal trajectory is supposed to lead to the upright equilibrium, with $g_0 = -\frac{m}{|m|}|g|$, $p_{\omega_{h_0}} = 0$ and $p_{\omega_{w_0}} = 0$ it follows that $p_{\omega_h}^g = 0$, $p_{\omega_h}^m = 0$, and $\mathcal{H} = 0$ along the motion. Thus, the angular momentum can only have a component orthogonal to g and m, and is therefore simplified to

$$
p_{\omega_h} = p_{\omega_h}^{\varphi}\ e_\varphi,
$$

where the unit vector e_φ is given by

$$
e_\varphi = \frac{m \times g}{|m \times g|}, \qquad \text{for} \quad m \times g \neq 0.
\tag{57}
$$

From the condition that the ideal trajectory lies on the zero energy surface it can be inferred that

$$
(p_{\omega_h}^{\varphi})^2 = \frac{2(m^{\mathsf{T}}g - |m|\ |g|)}{e_\varphi^{\mathsf{T}}\Theta_0^{-1}e_\varphi} = 2I_1|m|\ |g|(1 - \cos\varphi),
\tag{58}
$$

with $I_1 = e_\varphi^{\mathsf{T}}\Theta_0 e_\varphi$, which is constant. Due to a vanishing wheel momentum $p_{\omega_w} = 0$, it follows from $\omega_h = \Theta_0^{-1}p_{\omega_h}$ and the system dynamics that

$$
\omega_h = \frac{1}{I_1}p_{\omega_h}^{\varphi}e_\varphi = \dot{\varphi}\ e_\varphi.
\tag{59}
$$

Hence, along the ideal trajectory the Cubli follows the great circle of S^2 passing through the upright equilibrium represented by the north pole. The trajectory is implicitly parametrized by (58), by prescribing the angular momentum as a function of the inclination angle φ.

This "ideal" trajectory is tracked using the controller presented in Section III-B. To that extent the error $y - y_{\text{des}}$ is introduced, with y defined according to (37). From Section III-B it can be inferred that

$$
\begin{aligned}
\dddot{e} &= \dddot{y} - \dddot{y}_{\text{des}} = {}^{I}w - \dddot{y}_{\text{des}} := u_1 \\
{}^{I}\dot{\omega}_{h3} - {}^{I}\dot{\omega}_{h3_{\text{des}}} &= {}^{I}v_3 - {}^{I}\dot{\omega}_{h3_{\text{des}}} := u_2.
\end{aligned}
\tag{60}
$$

Using $x = (e, \dot{e}, \ddot{e}, {}^{I}\omega_{h3} - {}^{I}\omega_{h3_{\text{des}}})$, the error dynamics are rewritten as

$$
\dot{x} = \begin{pmatrix} 0_{2\times2} & I_{2\times2} & 0_{2\times2} & 0_{2\times1} \\ 0_{2\times2} & 0_{2\times2} & I_{2\times2} & 0_{2\times1} \\ 0_{2\times2} & 0_{2\times2} & 0_{2\times2} & 0_{2\times1} \\ 0_{1\times2} & 0_{1\times2} & 0_{1\times2} & 0 \end{pmatrix} x + \begin{pmatrix} 0_{4\times3} \\ I_{3\times3} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}.
\tag{61}
$$

Thus, a time-invariant state feedback controller, e.g. $u = (u_1, u_2) = Kx$ can be used to stabilize the error dynamics. The controller gain $K \in \mathbb{R}^{3\times7}$ can be found by linear control strategies such as a linear quadratic regulator approach or pole placement. Once the virtual control inputs u_1 and u_2 are determined, the resulting input torque is calculated by solving ${}^{I}w$ and ${}^{I}v_3$ for T. This transformation, given by (44) and (46) is bijective, except when the Cubli is inclined by 90 degrees.[5]

For tracking the ideal jump-up trajectory we impose that ${}^{I}\omega_{h3_{\text{des}}} = 0$ and ${}^{I}\dot{\omega}_{h3_{\text{des}}} = 0$ together with

$$
\begin{aligned}
y_{\text{des}} &= I_1\dot{\varphi}_{\text{des}}\ PR_{IK}e_\varphi, \\
\dot{y}_{\text{des}} &= |m|\ |g|\sin\varphi\ PR_{IK}e_\varphi, \\
\ddot{y}_{\text{des}} &= |m|\ |g|\cos\varphi\ \dot{\varphi}_{\text{des}}\ PR_{IK}e_\varphi, \\
\dddot{y}_{\text{des}} &= \frac{|m|^2|g|^2}{I_1}\sin\varphi\ (3\cos\varphi - 2)\ PR_{IK}e_\varphi, \\
\dot{\varphi}_{\text{des}} &:= \sqrt{\frac{2|m|\ |g|}{I_1}(1 - \cos\varphi)}.
\end{aligned}
\tag{62}
$$

The formulas are obtained by mere differentiation and using (58), which prescribes the desired angular momentum as a function of the inclination angle.

C. Learning Algorithm

For adapting the initial wheel velocities $\omega_w(0)^-$ a learning algorithm is used. The Cubli therefore makes multiple jump trials and evaluates the quality of each jump according to predefined criteria. The initial wheel velocities are adjusted using a model-based gradient descent method. In the next section the learning framework is elaborated in more detail.

1) Gradient-based Learning: The learning strategy used can be seen as a variation of the Newton procedure for finding the roots of a differentiable function. It has recently been presented and successfully implemented in [12].

The underlying process, e.g. the Cubli jump-up, is assumed to be dependent on the parameter vector $\theta \in \mathbb{R}^p$, which can be adjusted, as well as the unknown parameters $s \in \mathbb{R}^q$.[6] The goal is to adjust the parameters θ such that a certain error $e \in \mathbb{R}^m$ vanishes. In the case of the Cubli jump-up, we would like to adapt the initial wheel velocities $\omega_w(0)^-$ such that the upright equilibrium is reached without using additional control. The dependence of the error on the parameters (θ, s) is described by the mapping $E : \mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R}^m$. The error dimension m is assumed to be smaller or equal than the number of parameters p that can be adjusted ($m \leq p$).

A model based on nominal parameters s_0 is assumed to be known, which predicts the error $E(\theta, s_0)$. Based on this model, the parameters θ^0 leading to a vanishing error $E(\theta^0, s_0) = 0$ can be inferred, together with the gradient of E with respect to θ, evaluated at θ^0 and s_0. Still, the parameters of the real system, s^*, are unknown. By performing experiments, e.g. jump-up attempts, we can access noisy measurements of the error, $E^i = E(\theta, s^*) + N^i$, where N^i are bounded disturbances,

---

[5]In practice, an inclination of 90 degrees can never occur.

[6]As pointed out in [12] the vector of unknown parameters can be infinite dimensional.

$|N^i| < D$, $i = 0, 1, 2, \ldots$. The goal is therefore to iteratively find the zero of the function $E(\cdot, s^*)$ for unknown parameters s^*. A natural solution is to use Newton's method. However, since the gradient of E with respect to θ is unknown for $s = s^*$, the model-based approximation is used instead.

This leads to the following, simple and computationally efficient update rule for the parameters θ

$$\theta^{i+1} = \theta^i - \lambda^i \left.\frac{\partial E}{\partial \theta}\right|^{\dagger}_{\theta^0, s_0} E^i, \qquad i = 0, 1, 2, \ldots, \qquad (63)$$

with $\lambda^i \in (0, 2)$ a predefined sequence of step sizes, $i = 0, 1, 2, \ldots$, and where † denotes the pseudoinverse.

2) Application to the Cubli: Next, the learning algorithm is applied to the Cubli jump-up. By suddenly braking its reaction wheels rotating at high speeds the Cubli is able to jump up from lying flat to the edge-balancing position, from the edge-balancing position to the corner balancing position, and from lying flat to the corner balancing position. The analysis is restricted to the face to the corner jump-up (initially lying flat, jump-up to the corner), as the other cases can be treated in a similar manner.

From the modeling in Section IV-A and IV-B it can be concluded that the Cubli has essentially three degrees of freedom. The analysis suggests further to split them into a rotation around its center of mass $\vec{m}$, a rotation around the gravity vector $\vec{g}$ and a rotation around the direction perpendicular to $\vec{m}$ and $\vec{g}$. For a successful jump-up, where the upright equilibrium is reached with zero angular velocity, each degree of freedom must be controlled. Therefore, the error is chosen to be composed of the angular momentum in direction $\vec{m}$, the angular momentum in direction $\vec{g}$ and the energy $\mathcal{H}_h$, each of them evaluated at the top point

$$E(\omega_w(0)^-, s) = \begin{pmatrix} p^m_{\omega_h}(t_t) \\ p^g_{\omega_h}(t_t) \\ \mathcal{H}_h(t_t) \end{pmatrix}.$$

The top point is defined as the time instant $t = t_t$ at which the Cubli has either reached the upright position

$$g(t_t) = -\frac{m}{|m|}|g|$$

or has no angular momentum in direction $\vec{m} \times \vec{g}$, i.e. $p_{\omega_h}(t_t)^{\mathsf{T}}(m \times g(t_t)) = 0$. The parameters to be adjusted are the initial wheel velocities $\omega_w(0)^- \in \mathbb{R}^3$, whereas the vector s contains unknown system parameters, e.g. the inertia, the center of mass, the parameters related to the brake properties, etc. Clearly, the error vanishes only if the Cubli reaches the upright equilibrium.

According to the model derived in Section IV-A and IV-B the error components are all conserved quantities in the absence of the input torque T. Hence

$$\begin{aligned} p^m_{\omega_h}(t^t) &= p^m_{\omega_h}(0)^+ = p^m_{\omega_h}(0)^- = m^{\mathsf{T}}\Theta_w \omega_w(0)^- \\ p^g_{\omega_g}(t^t) &= p^g_{\omega_h}(0)^+ = p^g_{\omega_h}(0)^- = g(0)^{\mathsf{T}}\Theta_w \omega_w(0)^- \end{aligned} \qquad (64)$$

and

$$\begin{aligned} \mathcal{H}_h(t^t) &= \mathcal{H}_h(0)^+ \\ &= \frac{1}{2}(\omega_h(0)^+)^{\mathsf{T}}\Theta_0 \omega_h(0)^+ - m^{\mathsf{T}}g(0) - |m|\,|g| \\ &= \frac{1}{2}(\omega_w(0)^-)^{\mathsf{T}}\Theta_w \Theta_0^{-1}\Theta_w \omega_w(0)^- \\ &\qquad - m^{\mathsf{T}}g(0) - |m|\,|g|. \end{aligned} \qquad (65)$$

This implies that the gradient with respect to $\omega_w(0)^-$ evaluated for the model parameters s_0 yields

$$\left.\frac{\partial E}{\partial \omega_w(0)^-}\right|_{s_0} = \begin{pmatrix} m^{\mathsf{T}}\Theta_w \\ (g(0)^-)^{\mathsf{T}}\Theta_w \\ (\omega_w(0)^-)^{\mathsf{T}}\Theta_w \Theta_0^{-1}\Theta_w \end{pmatrix}. \qquad (66)$$

The initial guess $\theta^0 = (\omega_w(0)^-)^0$ is calculated by requiring the model-based error to vanish. This yields according to Section IV-B

$$\begin{aligned} (\omega_w(0)^-)^0 &= \sqrt{2I_1|m|\,|g|(1 - \cos\varphi_0)}\; e_\varphi(0), \\ e_\varphi(0) &= \frac{m \times g(0)}{|m \times g(0)|}, \end{aligned} \qquad (67)$$

with φ_0 the inclination angle when the Cubli is lying on its face.

3) Compensation for the Guiding Control Action: In the previous section the error function evaluating the quality of a jump-up trial has been introduced and its gradient based on the jump-up model has been derived. Therefore the update rule given by (63) can be applied to learn the initial wheel velocities, which lead the Cubli to its upright equilibrium without any control action.

In practice however, not every jump-up succeeds as the process noise, e.g. the randomness in the braking mechanism is too high. To increase the chances of a successful jump-up the guiding controller introduced in Section IV-B is used. The controller tries to maintain the Cubli on a successful jump-up trajectory and is activated after releasing the brakes. Naturally, the control effort of the guiding controller must be considered when evaluating the error criterion $E(\theta^i, s^*)$. In other words, given the value $E(\theta^i, s^*)$, the jump-up performance $E'(\theta^i, s^*)$ which would have been obtained if no additional control action would have been applied needs to be determined. Since the error E is composed of conserved quantities (in the absence of motor torque) it suffices to estimate their values shortly after braking, which yields

$$E'(\omega_w(0)^-, s) = \begin{pmatrix} p^m_{\omega_h}(t_t) - \int_0^{t_t} \dot{p}^m_{\omega_h}\mathrm{d}t \\ p^g_{\omega_h}(t_t) \\ \mathcal{H}_h(t_t) - \int_0^{t_t} \dot{\mathcal{H}}_h\mathrm{d}t \end{pmatrix}. \qquad (68)$$

Note, that the momentum around the g axis is constant, regardless of the motor torque. The time derivative of the momentum around m is obtained from the reduced system dynamics, (5) and is given by

$$\dot{p}^m_{\omega_h} = \frac{m}{|m|}^{\mathsf{T}}(-\omega_h \times p_{\omega_h}).$$

Moreover the rate of change of the energy related to the Cubli housing, $\mathcal{H}_h$, can be calculated to be $\dot{\mathcal{H}}_h = -\omega_h^{\mathsf{T}}T$.

Clearly, if the jump-up is ideal (in the sense of Section IV-B), no correction is applied and therefore E and E' agree. Moreover, the error E' can be simplified to

$$E'(\omega_w(0)^-, s) = \left(p^m_{\omega_h}(0)^+, p^g_{\omega_h}(0)^+, \mathcal{H}_h(0)^+\right), \qquad (69)$$

leading to the conclusion that the gradient of E' with respect to θ is likewise given by the right hand side of (66) for $s = s_0$.

The jump-up procedure is summarized by Algorithm 1.

Algorithm 1 Cubli Jump Up

1: **procedure** JUMPUP(Initial guess $(\omega_w(0)^-)^0$, Step sizes λ^i)
2:      $\theta^0 \leftarrow (\omega_w(0)^-)^0$
3:      $i = 0$
4:      **while** Not converged **do**
5:          Set θ^i to be the initial wheel velocities
6:          Speed up wheels, brake and apply guiding controller
7:          **while** Top point is not reached **do**
8:              Approximate $\int_0^{t_t} \dot{p}^m_{\omega_h} dt$ and $\int_0^{t_t} \dot{\mathcal{H}}_h dt$ by trapezoidal integration
9:          **end while**
10:         Calculate $E'(\theta^i, s^*)$ according to (68)
11:         $\theta^{i+1} \leftarrow \theta^i - \lambda^i \left.\frac{\partial E'}{\partial \theta}\right|_{\theta^0, s_0}^\dagger E'(\theta^i, s^*)$ according to (63)
12:         $i \leftarrow i + 1$
13:      **end while**
14: **end procedure**

V. EXPERIMENTAL RESULTS

In the following section the experimental results are discussed. The control algorithms are implemented on a Cortex M4 processor with a sampling time of $20\,\mathrm{ms}$, except for the guiding controller, which runs at $10\,\mathrm{ms}$. The algorithm presented in [21] is used for state estimation. The state estimation exploits the fact that there is a single pivot point being always at rest to derive a computationally light-weight, nonlinear attitude estimator. It is therefore "model free", in the sense that the estimation is solely based on a kinematic model and does not require knowledge of the center of gravity nor the inertia.

A. Balancing Performance

For balancing, an additional offset-correction filter is implemented, which accounts for modeling errors in the parameter m. Details of the implementation can be found in [15]. The controller parameters are tuned using the strategy presented in Section III and are chosen to be $\alpha = 15$, $\beta = 18$, $\gamma = 12$ and $\delta = 10^{-5}$. This yields closed-loop poles of the inclination angle located at $-32.7\,\mathrm{rad/s}$, $-12.0\,\mathrm{rad/s}$, and $-0.86\,\mathrm{rad/s}$ and a time constant for the yaw motion of $0.083\,\mathrm{s}$. With those parameters a root mean squared inclination angle error (at steady state) below $0.025\,^\circ$ can be observed.

Disturbance rejection measurements are depicted in Figure 6 and 7. The disturbance was chosen to be 0.17 Nm and was applied to a single wheel for 60 ms. After less than 1.8 s the inclination angle reaches steady state. Note that the reaction wheels are barely turning in steady state (the jitter visible in Figure 7 is due to measurement noise).
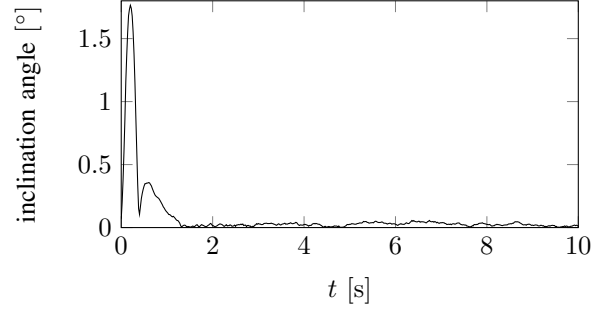

Figure 6. Disturbance rejection measurements. Depicted is the inclination angle over time. Note the inclination angle is not measured directly but estimated using the algorithm presented in [21].

B. Tracking Performance

Next, the tracking performance is evaluated. Simple state feedback in the transformed error variable e is used, that is $u = Kx$ with x and u defined according to (61). The feedback gain K is chosen such that the linearization of the controller around the upright equilibrium agrees with the linearization of the balancing controller.

Figure 8 shows the evolution of $\dot{y}$. Note that according to (40), $\dot{y}_2$ is proportional to $^\mathrm{I}m_1$ and $-\dot{y}_1$ to $^\mathrm{I}m_2$. Therefore the graph can be interpreted as the time evolution of the center of mass in the inertial frame. Although the center of mass is initially away from the ideal trajectory, the tracking controller manages to guide the Cubli back to the desired path. As soon as the center of mass is close enough to the upright equilibrium, i.e. reaches the region indicated by the dotted arc in Figure 8, the balancing controller takes over. Figure 9 shows the time evolution of the controller states y, which is associated to the momentum $^\mathrm{I}p_{\omega_h}$ and $\ddot{y}$, which is proportional to $^\mathrm{I}\omega_h \times {}^\mathrm{I}m$. The reference trajectory is again depicted by the dashed curves. It follows from Figure 9 that the generalized momentum $^\mathrm{I}p_{\omega_h}$ is accurately tracked. The error in the second derivative $\ddot{y}$ is initially larger, but is decreased by the controller as time evolves. However, a slight overshoot can be observed.

C. Learning Performance

The learning algorithm proposed in the previous section is implemented for the face to corner jump. A constant step size of $\lambda^i = 0.8$ for all iterations $i = 0, 1, 2\ldots$ is used. Figure 10 shows the evolution of the initial wheel speeds ω_{w_1} and ω_{w_2}. Due to the geometry of the Cubli, the third reaction wheel is only slightly used to correct for a non-zero momentum $p^g_{\omega_h}$ and is therefore not depicted. The initial wheel speeds were chosen to be around $100\,\mathrm{rad/s}$ away from the angular velocities leading to a successful jump-up. Hence, for the initial wheel speeds the Cubli barely moves or falls on the opposite side. After around 5 trials, the error of the angular momentum $p_{\omega_h}(t_0)^+$ is small enough such that the guiding controller can lead the Cubli to its upright equilibrium. At this point the learning algorithm is only compensating for the control action of the guiding controller leading to small correction steps.
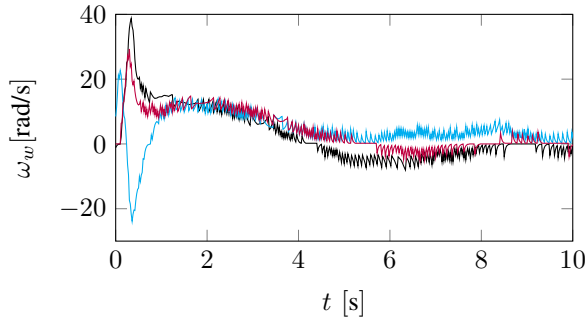
Figure 7. Disturbance rejection measurements. Depicted are the reaction wheel velocities over time, which are directly measured via a hall sensor. The different colors correspond to the different elements of the vector ω_w.
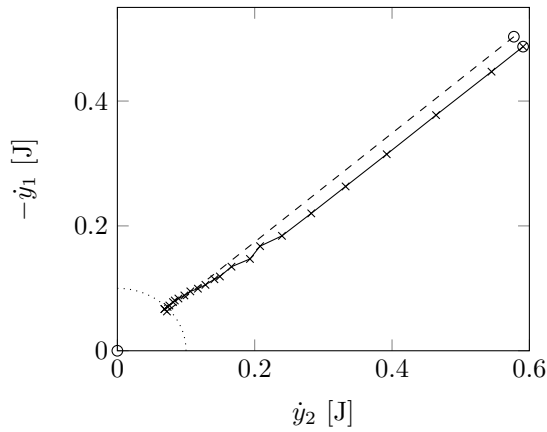

Figure 8. Trajectory tracking: Depicted is the evolution of $\dot{y}_1$ and $\dot{y}_2$ together with the ideal trajectory (dashed) for a successful jump-up. The black crosses indicate the sampling instants. The starting points (right after braking) of the ideal and actual trajectory are marked by black circles. The point $(0,0)$ denotes the upright equilibrium. The area around the upright equilibrium separated by the dotted circle arc represents the balancing region, i.e. the region where the tracking controller is turned off and the balancing controller takes over.

VI. CONCLUSION

This article presents aspects related to the dynamics and control of a reaction wheel-based 3D inverted pendulum. The analysis of the equations of motion revealed the existence of conserved quantities and relative equilibria, and allowed to find a reduced description of the dynamics. In particular, the reduced description was used for the control design. Two different nonlinear control approaches were presented and subsequently discussed. Finally, aspects related to the jump-up were presented, where the effect of repeatable disturbances was decreased by an iterative learning algorithm. To enhance robustness, feedback linearization was used to guide the inverted pendulum system to its upright equilibrium on a predefined trajectory. All control and learning algorithms were evaluated in experiments, which confirmed their effectiveness.

REFERENCES

[1] V.I. Arnold. *Mathematical methods of classical mechanics*. Springer, 1989.
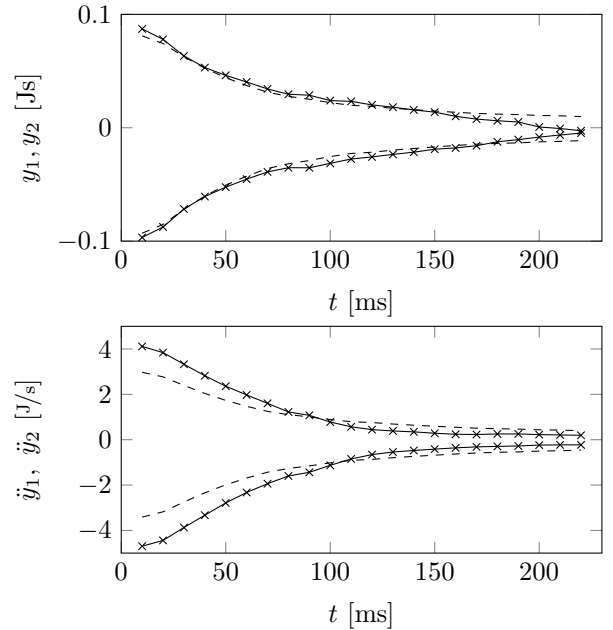[2] K.J. Åström and K. Furuta. Swinging up a pendulum by energy control. *Automatica*, 36(2):287–295, 2000.

Figure 9. Trajectory tracking: Depicted is the evolution of y and $\ddot{y}$ (solid), where the crosses indicate the sampling instants. The ideal trajectories, y_{des} and $\ddot{y}_{\text{des}}$ are shown by the dashed curves.
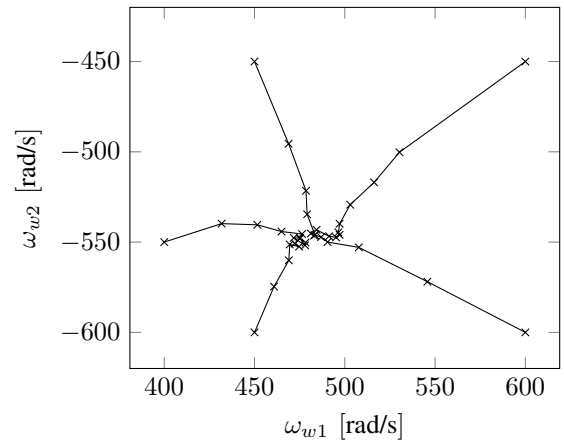

Figure 10. Depicted are the initial wheel speeds of the reaction wheels starting from five different initial conditions. The learning algorithm converges after few iterations to feasible wheel speeds resulting in a successful jump-up.

[3] Bonagiri Bapiraju, K.N. Srinivas, P. Prem. Kumar, and Laxmidhar Behera. On balancing control strategies for a reaction wheel pendulum. In *Annual IEEE India Conference*, pages 199–204, 2004.
[4] Dennis S. Bernstein, N. Harris McClamroch, and Anthony Bloch. Development of air spindle and triaxial air bearing testbeds for spacecraft dynamics and control experiments. In *American Control Conference*, pages 3967–3972, 2001.
[5] Daniel J. Block, Karl J. Åström, and Mark W. Spong. *The reaction wheel pendulum*, volume 1. Morgan & Claypool Publishers, 2007.
[6] Nalin A. Chaturvedi, Taeyoung Lee, Melvin Leok, and N. Harris McClamroch. Nonlinear dynamics of the 3D pendulum. *Journal of Nonlinear Science*, 21:3–32, 2011.
[7] Mohanarajah Gajamohan, Michael Merz, Igor Thommen, and Raffaello D'Andrea. The Cubli: A cube that can jump up and balance. In *International Conference on Intelligent Robots and Systems*, pages 3722–3727, 2012.
[8] Mohanarajah Gajamohan, Michael Muehlebach, Tobias Widmer, and Raffaello D'Andrea. The Cubli: A reaction wheel based 3D inverted

pendulum. In *European Control Conference*, 2013.

[9] K. Itô. *Encyclopedic dictionary of mathematics*, volume 1. MIT press, 1993.

[10] Hassan K. Khalil. *Nonlinear Systems*. Prentice Hall, Upper Saddle River, New Jersey, 1996.

[11] Miroslav Krstic, Ioannis Kanellakopoulos, and Petar Kokotovic. *Nonlinear and Adaptive Control Design*. John Wiley & Sons, Inc., 1995.

[12] Sergei Lupashin and Raffaello D'Andrea. Adaptive fast open-loop maneuvers for quadrocopters. *Autonomous Robots*, 33:89–102, 2012.

[13] Johannes Mayr, Franz Spanlang, and Hubert Gattringer. Mechatronic design of a self-balancing three-dimensional inertia wheel pendulum. *Mechatronics*, 2015.

[14] Leonard Meirovitch. *Methods of analytical dynamics*. Courier Dover Publications, 2010.

[15] Michael Muehlebach, Mohanarajah Gajamohan, and Raffaello D'Andrea. Nonlinear analysis and control of a reaction wheel-based 3D inverted pendulum. In *Conference on Decision and Control*, pages 1283–1288, 2013.

[16] Reza Olfati-Saber. Global stabilization of a flat underactuated system: the inertia wheel pendulum. In *Conference on Decision and Control*, 2001.

[17] Florian Scheck. *Mechanics: From Newton's laws to deterministic chaos*. Springer, 2010.

[18] Jinglai Shen, Amit K. Sanyal, Nalin A. Chaturvedi, Dennis Bernstein, and Harris McClamroch. Dynamics and control of a 3D pendulum. In *Conference on Decision and Control*, pages 323–328, 2004.

[19] Mark W. Spong and Daniel J. Block. The pendubot: A mechatronic system for control research and education. In *Conference on Decision and Control*, pages 555–556, 1995.

[20] Mark W. Spong, Peter Corke, and Rogelio Lozano. Nonlinear control of the reaction wheel pendulum. *Automatica*, 37:1845–1851, 2001.

[21] Sebastian Trimpe and Raffaello D'Andrea. Accelerometer-based tilt estimation of a rigid body with only rotational degrees of freedom. In *International Conference on Robotics and Automation*, pages 2630–2636, 2010.

[22] Wei Zhong and Helmut Röck. Energy and passivity based control of the double inverted pendulum on a cart. In *International Conference on Control Applications*, pages 896–901, 2001.