Dynamically Feasible Motion Planning through Partial Differential Flatness

Suresh Ramasamy, Guofan Wu, Koushil Sreenath

Abstract—Differential flatness is a property which substantially reduces the difficulty involved in generating dynamically feasible trajectories for underactuated robotic systems. However, there is a large class of robotic systems that are not differentially flat, and an efficient method for computing dynamically feasible trajectories does not exist. In this paper we introduce a weaker but more general form of differential flatness, termed partial differential flatness, which enables efficient planning of dynamic feasible motion plans for an entire new class of systems. We provide several examples of underactuated systems which are not differentially flat, but are partially differentially flat. We also extend the notion of partial differential flatness to hybrid systems. Finally, we consider the infamous cart-pole system and provide a concrete example of designing dynamically feasible trajectories in the presence of obstacles.

I. INTRODUCTION

In recent years, there has been a strong need for generating dynamically feasible motion plans for underactuated robotic systems. Doing this for any general system is extremely hard, involving high dimensional numerical integration and constrained nonlinear optimizations to choose a feedforward input such that the system trajectory evolves as desired while enforcing some constraints. For a particular class of systems, called differentially flat systems, this problem is relatively easy. Briefly, a system is differentially flat if we can find a set of outputs such that both the states and the inputs can be determined from these outputs and their higher-order derivatives without any integration [6]. This enables posing the problem of finding dynamically feasible trajectories as something as simple as a constrained quadratic program with no integration. This produces both a dynamically feasible trajectory that satisfies a set of constraints, and the nominal feedforward input that causes the system to evolve along the trajectory.

Differential flatness is a strong system property that can be used for generating dynamically feasible trajectories for underactuated robotic systems. In recent years, quadrotors systems were shown to be differentially flat, leading to significant progress in trajectory generation and control of quadrotor systems. For instance, in [2], a flatness-based flight planning / replanning strategy is proposed for a quadrotor; while in [7], differential flatness of position dynamics of quadrotor is exploited to design a controller via feedforward linearization; and recently in [21], the underactuated dynamical system comprising of multiple quadrotors cooperatively carrying a cable-suspended load was shown to be differentially flat.

Differential flatness has also been used to generate and track feasible trajectories in many other robotic and mechanical systems. In [10], a point to point pose tracking controller based on differential flatness for a steer-and-drive omnidirectional mobile robot is presented. In [9], a differential flatness based model predictive controller is proposed to reduce the fatal damage of tire blowout during vehicle operation. In [22], a differentially driven wheeled mobile robot is controlled by a differential flatness based controller.

However, the class of systems that are differentially flat is rather small. For instance, common underactuated systems such as the cart-pole, segway, ballbot, satellite with moment gyros, etc., are not differentially flat and thus we require a better way to plan dynamically feasible trajectories. In this paper we introduce a weaker notion of differential flatness, termed *partial differential flatness* that applies to a broader set of systems. Roughly speaking, a system is partially differentially flat if we can find a set of outputs and a state partition, such that the inputs and a partition of the states can be obtained from the outputs and their higher-order derivatives without integration. The remaining (typically low-dimensional) partition of the state can be found through integration if the initial condition is known. As we will see, partial differential flatness will enable dynamically feasible trajectory generation for a larger class of underactuated robotic systems.

We will also briefly note the similarities / differences / relations between our notion of partial differential flatness and relative flatness [19], defect of a nonlinear system [6], shape-accelerated underactuated balancing systems [15], and partial feedback linearization [20].

The rest of the paper is structured as follows. Section II introduces the concept of partial differential flatness, Section III presents several examples of systems that are partially differentially flat, Section IV extends the notion of partial differential flatness to hybrid systems, Section V performs a numerical simulation to illustrate dynamical trajectory generation for one particular system, and finally Section VI provides some concluding remarks and summarizes the key benefits and limitations of partial differential flatness.

II. PARTIAL DIFFERENTIAL FLATNESS

The goal of this section is to introduce the new concept of partially differentially flat systems, wherein a part of the system is differentially flat and the remaining part of the system has a special structure corresponding to that of multiple chains of integrators. If the flat output of the system and the initial condition at time t_0 of the non-differentially flat part of
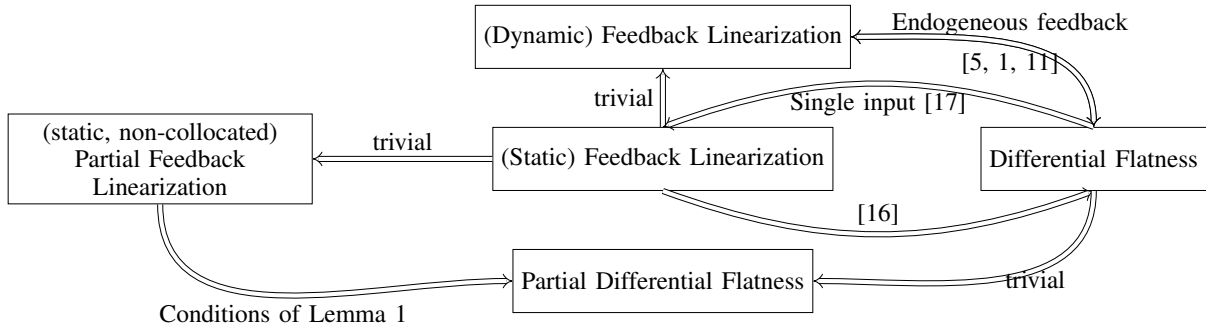
Fig. 1: Illustration depicting various system properties and their implication relations. Lemma 1 in the paper establishes conditions under which (static, non-collocated) partial feedback linearization implies partial differential flatness.

the system is known, then everything about the system can be obtained. This enables dynamically feasible motion planning for an entire class of systems that are not differentially flat. Examples of such systems include the cart-pole, the triaxial spacecraft attitude testbed [3], ballbot [14], cubli [12], etc.

Definition 1. Differentially-flat system [13]: *A system* $\dot{x} = f(x, u)$, $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, *is differentially flat if there exists outputs* $y \in \mathbb{R}^m$ *of the form* $y = y(x, u, \dot{u}, \cdots, u^{(p)})$, *such that the states and the inputs can be expressed as* $x = x(y, \dot{y}, \cdots, y^{(q)})$, $u = u(y, \dot{y}, \cdots, y^{(q)})$, *where* p, q *are finite integers.*

Definition 2. Partial Differential Flatness: *A system* $\dot{x} = f(x, u)$, $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, *is partially differentially flat if there exists a partition of the state,* $x = \begin{bmatrix} s \\ r \end{bmatrix}$, $s \in \mathbb{R}^k, r \in \mathbb{R}^{n-k}, k \leq n$, *(called s-states and r-states) and outputs* $y \in \mathbb{R}^m$ *of the form* $y = y(s, u, \dot{u}, \cdots, u^{(p)})$, *such that the state-partition s and input* u *can be expressed as* $s = s(y, \dot{y}, \cdots, y^{(q)})$, $u = u(y, \dot{y}, \cdots, y^{(q)})$ *respectively, where* p, q *are finite integers, and the r-state dynamics are that of one or more chains of integrators, i.e.,*

$$\dot{r} = \begin{bmatrix} A_1 & 0 & \cdots & 0 \\ 0 & A_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & A_l \end{bmatrix} r + \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_l \end{bmatrix}, \quad (1)$$

with A_i *and* b_i *given as*

$$A_i = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ 0 & \mathbf{0} \end{bmatrix}, \quad b_i = \begin{bmatrix} \mathbf{0} \\ h_i(y, \dot{y}, \ldots, y^{(q)}) \end{bmatrix}, \quad (2)$$

where h_i *is a smooth scalar function, and the bold constants are either row/column vectors or matrices of appropriate size.*

Remark 1. *For* $k = n$, *the above definition is equivalent to differential flatness. Thus, partial differential flatness encompasses a larger class of systems.*

Remark 2. *This definition renders a part of the system differentially flat, and thus the name partial differential flatness. In particular, the s-states of the system can be obtained from the flat output and their higher order derivatives without integration. Furthermore, we also stress that, just as in differential flatness, no differential equations need to be integrated to obtain the open-loop control either.*

Remark 3. *Clearly the entire state trajectory,* $x(t)$ *can not be determined from the flat output and its higher-order derivatives. Only the s-states,* $s(t)$, *can be determined. However, if the initial condition of the r-states,* $r(0)$ *is known, then the entire state trajectory can be determined through integration of the chain of integrators given by (1). However, note that we are not integrating the system dynamics nor do we require the system input for this integration. Moreover, since the dimension of* r *is typically significantly smaller than the state dimension, this integration is very efficient. In a practical setting, even if this integration can not be performed analytically, it can be carried out numerically (potentially as a simple trapezoidal integration), enabling online trajectory planning.*

Remark 4. *For numerical trajectory optimization, integration can be avoided all together if we plan trajectories for both the s-states,* $s(t)$, *and r-states,* $r(t)$. *From the s-states and its higher order derivatives, the control input can be obtained, along with the functions* h_i *in (2). Then, a (nonlinear) equality constraint on* $\dot{r}$, *as in (1)-(2), can be placed on the trajectory generation optimization. This completely avoids integration.*

Remark 5. *Partial differential flatness is similar to* relative flatness *as defined in [19], wherein a subsystem is differentially flat. However, unlike relative flatness, for partial differential flatness, the entire set of inputs can be determined from the flat outputs and its higher-order derivatives.*

Remark 6. *The system we have considered above can also be seen as a special form of a nonlinear system with defect equal to the dimension of* r, *where defect is as defined in [6].*

Remark 7. *Partial differential flatness can also be compared with* shape-accelerated underactuated balancing systems *as defined in [15], wherein the configuration space is equally partitioned into unactuated shape variables and actuated position variables. However, in [15], trajectory tracking is performed by considering a position trajectory, and inverting the non-holonomic dynamic constraint between the acceleration of the position variables and the shape variables and their higher order derivatives, under the assumption that the shape variables are constant. This results in a shape variable trajectory point-wise in time. In particular, this is like using a trajectory* $r(t)$ *and then inverting the scalar* h *map under a quasi-static assumption to obtain the configuration variable that is part of* s. *In general, unlike the notion of partial differential flatness being presented here, the method of shape-*

accelerated underactuated balancing systems does not produce dynamically feasible trajectories.

***Remark* 8.** *It is well known that systems that are (static) feedback linearizable are differentially flat [16]. This raises the question of whether systems that are (static, non-collocated) partial feedback linearizable [20], where only a part of the system is feedback linearized, are partially differentially flat? Although partial feedback linearization does introduce a state partition and does create the integrator chain as in (1), however, in general, the input may not be expressed as a function of the flat output and its higher-order derivatives. Lemma 1 provides conditions under which a partial feedback linearizable system is partially differentially flat. Figure 1 summarizes various implication relations between various kinds of feedback linearization and differential flatness.*

***Lemma* 1.** *Consider the n-DOF underactuated system with unactuated states $q_u \in \mathbb{R}^l$, actuated states $q_a \in \mathbb{R}^m$, with $l + m = n$, and control input $\tau \in \mathbb{R}^m$ given by,*

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_u \\ \ddot{q}_a \end{bmatrix} + \begin{bmatrix} H_1 \\ H_2 \end{bmatrix} = \begin{bmatrix} 0 \\ \tau \end{bmatrix}. \tag{3}$$

Suppose (a) there exists a partition of the actuated states such that $M_{12}\ddot{q}_a = \begin{bmatrix} M_{12}^c & M_{12}^d \end{bmatrix} \begin{bmatrix} q_a^c & q_a^d \end{bmatrix}^T$, with $q_a^c \in \mathbb{R}^l$, $q_a^d \in \mathbb{R}^{m-l}$, (b) the above system is (static, non-collocated) partial feedback linearizable with $\operatorname{rank}(M_{12}^c) = l$, (c) q_a^c are cyclic variables, and (d) H_1, H_2 are independent of $\dot{q}_a^c$, then the system is partially differentially flat with flat outputs $\mathcal{Y} = \begin{bmatrix} q_u & q_a^d \end{bmatrix} \in \mathbb{R}^m$.

Proof: The above underactuated system can be (static, collocated) partial feedback linearized as [20],

$$M_{11}\ddot{q}_u + H_1 = -\begin{bmatrix} M_{12}^c & M_{12}^d \end{bmatrix} \begin{bmatrix} v_a^c \\ v_a^d \end{bmatrix}, \quad \begin{bmatrix} \ddot{q}_a^c \\ \ddot{q}_a^d \end{bmatrix} = \begin{bmatrix} v_a^c \\ v_a^d \end{bmatrix}. \tag{4}$$

By assumption (b), we can choose $v_a^c = -(M_{12}^c)^{-1}(M_{11}v_u + H_1 - M_{12}^d v_a^d)$, to obtain the (static, non-collocated) partial feedback linearized form,

$$\ddot{q}_u = v_u, \tag{5}$$
$$\ddot{q}_a^c = -(M_{12}^c)^{-1}(M_{11}v_u + H_1 - M_{12}^d v_a^d), \tag{6}$$
$$\ddot{q}_a^d = v_a^d, \tag{7}$$

with the actual control input τ being a function of v_u, v_a^d. Assumption (c) implies $M_{ij}(q), H_i(q,\dot{q})$ are independent of q_a^c, since $\frac{\partial L(q,\dot{q})}{\partial q_a^c} = 0$, where $L(q,\dot{q})$ is the Lagrangian of the system. Assumption (d) implies $H(q,\dot{q})$ is independent of $\dot{q}_c$. Then, from the flat output and its higher order derivatives, we can determine M_{ij}, H_i; while from (5)-(7) we can determine v_u, v_a^d, thereby determining the actual control input τ; and finally we can determine $\ddot{q}_a^c$ from (6). ∎

Having introduced the abstract concept of partial differential flatness and providing some comments to compare / contrast / relate with other notions such as relative flatness, defect of a nonlinear system, shape-accelerated underactuated balancing systems, and partial feedback linearization, we now look to illustrate the concept through several examples. In particular, we will demonstrate that the following underactuated mechanical systems are partially differentially flat: cart-pole, planar ballbot, triaxial attitude control test bed, and cubli
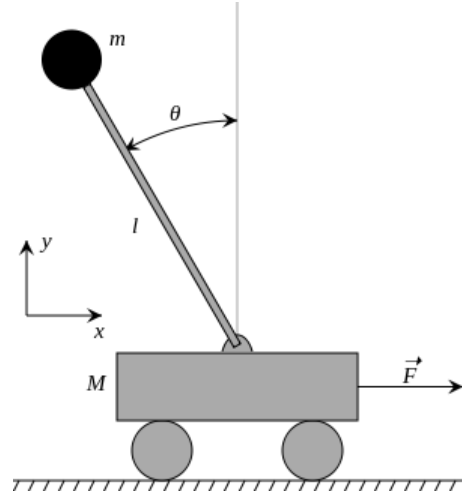

Fig. 2: Cart-Pole system from Wikipedia. Considering θ as your flat output yields input force and all the state variables except cart velocity and position as a function of the flat output and its higher order derivatives.

balancing on its edge and on its corner. This is just a small subset of such systems. Other potential systems are the segway, 3D ballbot, etc.

III. EXAMPLES

This section will present several underactuated mechanical systems and demonstrate that they are partially differentially flat. We will use results from this section to demonstrate dynamically feasible trajectory generation for one particular system in Section V.

A. Cart-Pole

The cart-pole system is a classical underactuated mechanical system that serves as a benchmark example for control design. Here, we will consider the cart-pole system and demonstrate that it is partially differentially flat. The cart-pole system is illustrated in Figure 2 and makes use of the variables defined below.

Symbol	
M	Mass of the cart
m	Mass of the pendulum bob
θ	Angle made by the cart with the vertical
F	Force applied on the cart
x	Displacement of the cart in horizontal direction
g	Magnitude of gravity vector

The equations of motion for this system are given by,

$$\begin{bmatrix} (M+m) & -ml\cos\theta \\ -ml\cos\theta & ml^2 \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} ml\dot{\theta}^2\sin\theta \\ 0 \end{bmatrix}$$
$$+ \begin{bmatrix} 0 \\ -mlg\sin\theta \end{bmatrix} = \begin{bmatrix} F \\ 0 \end{bmatrix} \tag{8}$$

If we consider $\mathcal{Y} = \theta$ to be our flat output, from the equations of motion we get,

$$
\begin{aligned}
F &= (M+m)\frac{(l\ddot{\theta} - g\sin\theta)}{\cos\theta} - ml\ddot{\theta}\cos\theta \\
&\quad + ml\dot{\theta}^2\sin\theta, \tag{9} \\
\ddot{x} &= \frac{(l\ddot{\theta} - g\sin\theta)}{\cos\theta}. \tag{10}
\end{aligned}
$$

Thus, we get the control input F and the s-states $[\theta, \dot{\theta}]$ as a function of our flat output and its higher order derivatives. Furthermore, since the cart acceleration $\ddot{x}$ is also know from (10), if the initial condition of the r-states, $[x(0), \dot{x}(0)]$ is known, then by integrating (10) we can find all the system states.

B. Planar Ballbot

The Ballbot is a mobile robot that moves on a single spherical wheel. The ballbot can be modeled as a rigid cylinder on top of a rigid sphere. In [14], a planar model of the ballbot is developed. The model assumes no slip between the spherical wheel and the floor. Furthermore, we assume that there is no friction between the rollers and the ball, which means we can neglect the damping term in the equations of motion. The ballbot system is illustrated in Figure 3 and makes use of the variables defined below.

Symbol	
ϕ	Body angle
θ	Angle between body and the ball
m_{ball}	Mass of the ball
m_{body}	Mass of the body
r	Radius of the ball
l	Distance of centre of mass from the ball
τ	Input torque
I_{ball}	Moment of inertia of the ball
I_{body}	Moment of inertia of the body
α	$I_{ball} + (m_{ball} + m_{body})r^2$
β	$m_{body}rl$
γ	$I_{body} + m_{body}l^2$

Euler Lagrange equations are used to derive the dynamic equations of motion, which are given by,

$$
M(q)\ddot{q} + C(q,\dot{q}) + G(q) = \begin{bmatrix} \tau \\ 0 \end{bmatrix} \tag{11}
$$

where $q = [\theta, \phi]$

$$
M(q) = \begin{bmatrix} \alpha & \alpha + \beta\cos\phi \\ \alpha + \beta\cos\phi & \alpha + \gamma + 2\beta\cos\phi \end{bmatrix}, \tag{12}
$$

$$
C(q,\dot{q}) = \begin{bmatrix} -\beta\sin\phi\dot{\phi}^2 \\ -\beta\sin\phi\dot{\phi}^2 \end{bmatrix}, \tag{13}
$$

$$
G(q) = \begin{bmatrix} 0 \\ -\frac{\beta g\sin\phi}{r} \end{bmatrix}. \tag{14}
$$

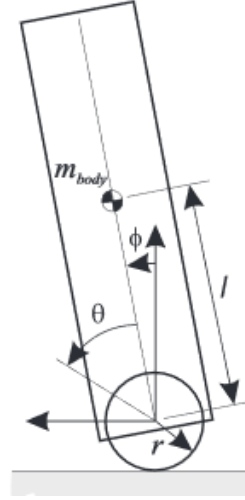where $\alpha = I_{ball} + (m_{ball} + m_{body})r^2$, $\beta = m_{body}rl$, $\gamma = I_{body} + m_{body}l^2$

Fig. 3: Planar Ballbot obtained from [18]. Considering ϕ as your flat output yields input torque and $\ddot{\theta}$ as a function of the flat output and its higher order derivatives. θ and $\dot{\theta}$ can be obtained by integrating the expression for $\ddot{\theta}$ if the initial conditions are known.
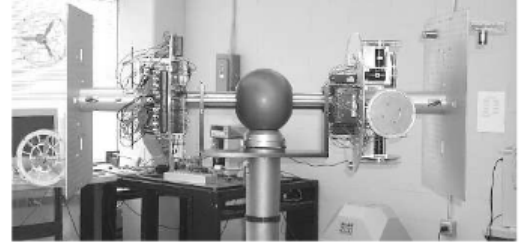

Fig. 4: Figure obtained from [4]. Tri-Axial Attitude Control Testbed: Considering the rotation matrix to be the flat output, yields input τ_s and $\dot{q}$ as functions of the flat output and its higher order derivatives and the state variable q can be found out by integration if the initial conditions are known.

If we choose $\mathcal{Y} = \phi$ as our flat output, from equation (11), we get,

$$
\ddot{\theta} = \frac{\beta\sin\phi\dot{\phi}^2 + \frac{\beta g\sin\phi}{r} - (\alpha + \gamma + 2\beta\cos\phi)\ddot{\phi}}{\alpha + \beta\cos\phi}, \tag{15}
$$

$$
\tau = \beta\sin\phi\dot{\phi}^2 - (\alpha + \beta\cos\phi)\ddot{\phi} - \alpha\ddot{\theta}. \tag{16}
$$

Thus we get the input torque τ and the s-states $[\phi, \dot{\phi}]$ explicitly as a function of the flat output and its higher order derivatives and we also know $\ddot{\theta}$ from (15). Thus with knowledge of the initial conditions, we can integrate (15) to get the r-states $[\theta, \dot{\theta}]$.

C. Triaxial Attitude Control Test Bed

The Triaxial Attitude Control testbed (TACT) was developed to study spacecraft multibody rotational dynamics and control. Here we prove that the TACT is a partially differentially flat system when actuated by three reaction wheels. The

TACT system is illustrated in Figure 4 and makes use of the variables defined below.

Symbol	
R	Rotation matrix from base sphere frame to inertial frame
τ_s	Generalised forces and moments that act to change TACT shape dynamics
ω	Angular velocity of base body
Γ	Reduced attitude vector

The equations of motion for the TACT actuated by three reaction wheels is obtained from [3], and are given by,

$$\dot{R} = R\hat{\omega}, \tag{17}$$

$$\begin{bmatrix} J & B \\ B^T & M \end{bmatrix} \begin{bmatrix} \dot{\omega} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} J\omega \times \omega + B\dot{q} \times \omega + m_\tau g \rho_s \times \Gamma \\ 0 \end{bmatrix} \tag{18}$$

$$+ \begin{bmatrix} 0 \\ \tau_s \end{bmatrix}. \tag{19}$$

where,

$$\Gamma = R^T e_3 \tag{20}$$

From equation (21),

$$\begin{bmatrix} \dot{\omega} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} J & B \\ B^T & M \end{bmatrix}^{-1} \begin{bmatrix} J\omega \times \omega + B\dot{q} \times \omega + m_\tau g \rho_s \times \Gamma \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \tau_s \end{bmatrix} \tag{21}$$

If we consider $\mathcal{Y} = R$ to be our flat output, from equation (17), we get ω to be a function of our flat outputs and it's higher order derivatives. From equation (21) and assuming B is invertible, we get $\dot{q}, \ddot{q}, \tau_s$ as function of our flat outputs and its higher order derivatives. Thus we obtain the input torque τ_s, the s-states $[R, \omega, \dot{q}]$ and $\ddot{q}$ as a function of our flat outputs and its higher order derivatives and if we know the initial conditions of our system, we can obtain the r-state $[q]$ by integrating the expression for $\dot{q}$.

D. Cubli about its edge

Cubli is a 3D inverted pendulum test bed with three reaction wheels mounted orthogonally to each other. This is an interesting hybrid system that is capable transitioning from being on its face to balancing on its edge, to balancing on a corner. The Cubli system is illustrated in Figure 5 and makes use of the variables defined below.

Symbol	
ϕ, ψ	Angles that describe position of cubli
θ_o	Total moment of inertia in the body fixed coordinate frame
θ_w	Reaction wheel's moment of inertia in the body fixed coordinate frame
p_ϕ, p_ψ	Generalized momenta
m_{tot}	total mass
l	distance between the pivot point to the center of gravity of the whole system
m	$m_{tot}g$
T	Input torque

The equations of motion are obtained from [12]. The generalized momenta are defined by:

$$p_\phi = \theta_o \dot{\phi} + \theta_w \dot{\psi}, \tag{22}$$

$$p_\psi = \theta_w(\dot{\phi} + \dot{\psi}). \tag{23}$$

The equations of motion are given by,

$$\begin{bmatrix} \dot{\phi} \\ \dot{p_\phi} \\ \dot{p_\psi} \end{bmatrix} = \begin{bmatrix} \hat{\theta}_o^{-1}(p_\phi - p_\psi) \\ mg\sin\phi \\ T \end{bmatrix} \tag{24}$$

where T is the torque and $\hat{\theta}_o$ is equal to $\theta_o - \theta_w$. Differentiating equation (22) and (23),

$$\dot{p}_\psi = \theta_w(\ddot{\phi} + \ddot{\psi}), \tag{25}$$

$$\dot{p}_\phi = \theta_o \ddot{\phi} + \theta_w \ddot{\psi}. \tag{26}$$

By solving equations (24),(25) and (26), the following can be obtained,

$$\dot{p}_\psi = T = \ddot{\phi}(\theta_w - \theta_o) + mg\sin\phi \tag{27}$$

Therefore if we choose our flat output to be $\mathcal{Y} = \phi$, the input torque T, and the s-state $[\phi]$ have been obtained in terms of the flat outputs and its higher order derivatives. Also from equations (24) and (27) we know $\dot{p}_\phi$ and $\dot{p}_\psi$. Now given the initial conditions of the system, by integration we can find our r-states $[p_\psi, p_\phi]$.

E. Cubli Balancing About a Corner

The equations of motion when the cubli is balancing about a corner as shown in the Figure 5c are given in [12], and they utilize the following symbols.

Symbol	
ω_w	Reaction wheel angular velocity
ω_h	Body angular velocity
$\overrightarrow{g}$	Gravity vector
p_{ω_h}, p_{ω_w}	Generalized momenta
θ_o	Total moment of inertia of the full cubli about the pivot point
θ_w	Moment of inertia of the reaction wheels in the body fixed frame
$\overrightarrow{m}$	Position vector from the pivot point to the center of gravity multiplied by the total mass
T	Input torque

The equations of motion are given as,

$$\dot{\overrightarrow{g}} = 0, \tag{28}$$

$$\dot{\overrightarrow{p}}_{\omega_h} = \overrightarrow{m} \times \overrightarrow{g}, \tag{29}$$

$$\dot{p}_{\omega_w} = T, \tag{30}$$

where,

$$p_{\omega_h} =: \theta_o \omega_h + \theta_w \omega_w, \tag{31}$$

$$p_{\omega_w} =: \theta_w(\omega_h + \omega_w). \tag{32}$$

If we consider our flat output to be $\mathcal{Y} = \overrightarrow{m}$ in the world frame, then we can obtain our input torque T from equations

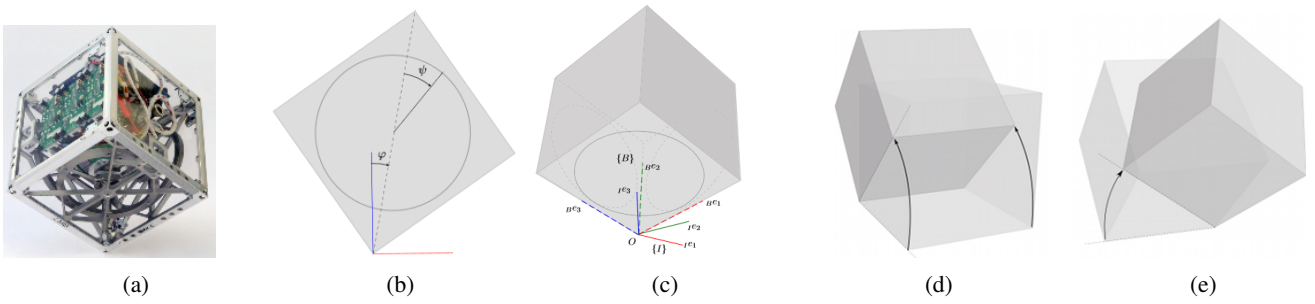(a)   (b)   (c)   (d)   (e)

Fig. 5: Figures obtained from [12, 8]. (a) The Cubli experimental test bed. (b) Cubli balancing about an edge. In this configuration it is effectively a reaction wheel based 1D inverted pendulum. Considering ϕ as your flat output yields input torque and derivatives of generalised momenta as functions of the flat output and its higher order derivatives. (c) Cubli balancing about a point. B^{e_i} and I^{e_i} denote the principle axis of the body fixed frame B and inertial frame I. The pivot point O is the common origin of coordinate frames I and B. (d) The cubli jumping up to balance on its edge. (e) The cubli goes from balancing on an edge to balancing on a corner.

	Cart-pole	Planar Ballbot	TACT	Cubli about an Edge	Cubli about a Corner
Variable	θ, x	ϕ, θ	R, q	ϕ, p_ϕ, p_ψ	$\overrightarrow{m}, p_{w_h}, p_{w_w}$
No. of Degrees of Freedom	2	2	6	2	6
No. of Actuators	1	1	3	1	3
Partially Flat output	θ	ϕ	R	ϕ	$\overrightarrow{m}$
r	$x, \dot{x}$	$\theta, \dot{\theta}$	q	p_ϕ, p_ψ	p_{w_h}, p_{w_w}

TABLE I: Summary of the presented examples of partially differentially flat system and their various properties.
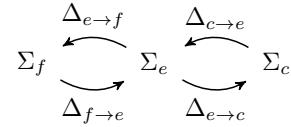

Fig. 6: Transition between the different dynamical modes of Cubli. The subscripts f, e, c represent *face*, *edge* and *corner* respectively

(29),(30) and substituting the time-derivatives of equations (31),(32) as follows,

$$T = \dot{p}_{\omega_w} = \hat{m}g - (\theta_o - \theta_w)\dot{\omega}_h \qquad (33)$$

Since from $\overrightarrow{m}$, we can calculate $\dot{\omega}_h$, we can get $\dot{p}_{\omega_w}$ and hence we can obtain our input torque as a function of the flat outputs and its higher order derivatives. We also know $\dot{\overrightarrow{p}}_{w_h}, \dot{p}_{w_w}$ from equations (29) and (30). Now given the initial conditions of the system the r-states $[\overrightarrow{p}_{w_h}, p_{w_w}]$ can be obtained by integration.

IV. HYBRID SYSTEMS

In the previous section, we presented several examples of underactuated mechanical systems and showed that they were partially differentially flat, as summarized in Table I. Our goal in this section is to extend the concept of partial differential flatness to hybrid systems, so as to plan dynamically feasible trajectories that span multiple dynamical hybrid modes. We will also provide a concrete example of an underactuated hybrid system and show that it is a partially differentially-flat hybrid system.

Definition 3. *A* partially differentially-flat hybrid system *is a hybrid system where each subsystem is partially differentially-flat, with the guards being functions of the flat outputs, their higher-order derivatives, and the r-states (as in Definition 2), and with the transition maps being sufficiently smooth.*

Remark 9. *In contrast to differentially-flat hybrid systems, as defined in [21], for partially differentially-flat hybrid sys-*tems, the guards may not be determined only from the flat outputs and their higher-order derivatives. Instead, r-states would need to be known, which can only be obtained through integration. Moreover, the transition maps need not map from the flat output space of one subsystem to the flat output space of the subsequent subsystem either.*

For the purposes of a concrete illustration of this concept, we will consider the Cubli system which can transition from being on its face, to balancing on its edge, and then to balancing on its corner, with the dynamics switching at the transition (see Figure 5d and 5e), making it a hybrid system. The transitions occurs when a brake is applied to the internal moment gyros spinning at high speed causing a transfer of angular momentum from the gyros to the cube due to conservation of angular momentum. For the purpose of this discussion, we will assume the brake is applied when the moment gyros reach a particular angular momentum.

The three distinct dynamical subsystems or modes of Cubli are: resting on a face (Σ_f), balancing on an edge (Σ_e), and balancing on a corner (Σ_c). This is illustrated in Figure 6 with appropriate transition maps between the various subsystems. We can summarize a single set of equations of motion that are valid for all three cases with the initial conditions ensuring that the system maintains the constrains of each mode. For all three subsystems, the state of the system can be defined as $x = [\theta_\omega \quad g \quad p_{w_h} \quad p_{w_\omega}] \in \mathcal{X}$, where $\theta_\omega \in \mathbb{R}^3$ represents the angles of the three moment wheels, and the other variables are as defined in Section III-E. Then, the equations of motion can

be written as (see (24), (28)-(30)),

$$\Sigma_{f,e,c} : \frac{d}{dt} \begin{bmatrix} \theta_\omega \\ \overrightarrow{g} \\ \overrightarrow{p}_{\omega_h} \\ p_{\omega_\omega} \end{bmatrix} = \begin{bmatrix} \omega_\omega \\ 0 \\ \overrightarrow{m} \times \overrightarrow{g} \\ T \end{bmatrix}, \tag{34}$$

with the guard surfaces defined as

$$\begin{aligned}
\mathcal{S}_{f \to e} &= \{x \in \mathcal{X} \mid p_{\omega_\omega} \cdot e_1 = c_1\}, \tag{35} \\
\mathcal{S}_{e \to c} &= \{x \in \mathcal{X} \mid p_{\omega_\omega} \cdot (e_1 + e_2) = c_2\}. \tag{36}
\end{aligned}$$

This indicates that the face to edge balancing transition occurs when the angular momentum of one of the internal gyros reaches c_1, and the edge to corner transition occurs when the sum of angular momentum of the remaining moment gyros reaches c_2. The transition maps are defined as follows:

$$\begin{aligned}
\Delta_{f \to e} \quad & : \quad p_{\omega_\omega}^+ \cdot e_1 = 0, \quad p_{\omega_h}^+ \cdot e_1 = p_{\omega_\omega}^- \cdot e_1 \tag{37} \\
\Delta_{e \to c} \quad & : \quad p_{\omega_\omega}^+ \cdot e_2 = 0, \quad p_{\omega_\omega}^+ \cdot e_3 = 0, \\
& \quad p_{\omega_h}^+ \cdot e_2 = p_{\omega_\omega}^- \cdot e_2, \quad p_{\omega_h}^+ \cdot e_3 = p_{\omega_\omega}^- \cdot e_3, \tag{38}
\end{aligned}$$

where the subscripts $+,-$ indicate the values of the variables before and after transition respectively.

The flat outputs for the three subsystems are (the edge and corner is shown in the previous section),

$$\begin{aligned}
\mathcal{Y}_f &= \{\theta_\omega \cdot e_1, \theta_\omega \cdot e_2, \theta_\omega \cdot e_3\}, \tag{39} \\
\mathcal{Y}_e &= \{\psi, \theta_\omega \cdot e_2, \theta_\omega \cdot e_3\}, \tag{40} \\
\mathcal{Y}_c &= \{m\}. \tag{41}
\end{aligned}$$

Now, starting with the Cubli resting on its face ($g = 0, p_{\omega_\omega} = 0$, we can plan trajectories for the flat output $\mathcal{Y}_f$ and know exactly when the system transitions to balancing on its edge and when the system transitions to balancing on its face, and so on. This is because the Cubli is trivially partially differentially flat (its actually differentially flat) when its resting on its face. Then $\mathcal{Y}_f(t)$ lets us determine when the state intersects the guard surface $\mathcal{S}_{f \to e}$. Then, the transition map $\Delta_{f \to e}$ gives us the initial post transition state. Further, planning a trajectory for $\mathcal{Y}_e(t)$ and the initial state gives us the full state as a function of time, and we can once again exactly plan when the transition to balancing on its edge occurs. Thus, the Cubli system is a partially differentially flat hybrid system.

V. SIMULATION RESULTS

Having demonstrated that several underactuated mechanical systems are partially differentially flat and even one which is a partially differentially flat hybrid system, we now illustrate an example of using the partial differential flatness property to generate dynamically feasible trajectories. In particular, we will consider the cart-pole system, and use the fact that it is a partially differentially flat system to generate dynamically feasible trajectories around obstacles.

The problem statement is to generate dynamically feasibly trajectories that take a cart-pole system from position A at rest to position B at rest while ensuring that the pendulum avoids an obstacle (see Figure 7).
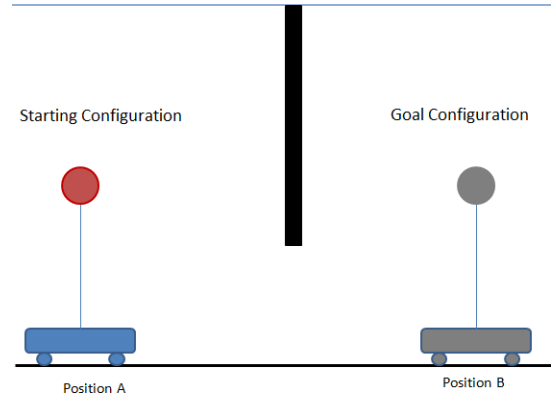

Fig. 7: Cart Pole Planning problem. The objective is to plan a dynamically feasible trajectory from Position A to position B which is 5m away which avoids collision of the pendulum bob with the wall.
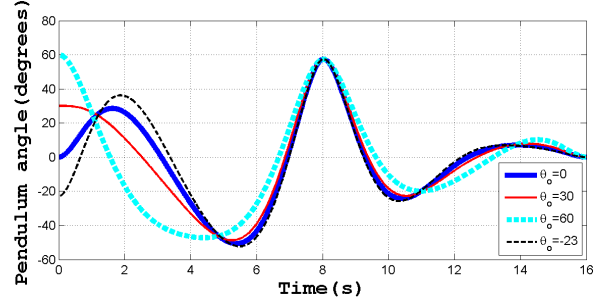

Fig. 9: Trajectory of the pendulum angle (in degrees): This corresponds to the stick figure plot of the pendulum and cart shown in Figure 8.

We solve this problem by posing it as a nonlinear constrained optimization problem. We will utilize the fact that the Cart-Pole system is a partially differentially flat system with θ being the flat output. So, any planned trajectory in the flat output space will yield corresponding trajectories for the control input and s-states of the system. Furthermore, if the initial condition of the r-states of the system is known, which is the case here, we can find the entire state trajectory.

We parametrize the flat output $\theta(t)$ using Legendre polynomials as follows,

$$\theta(t) = \sum_{k=1}^{6} \alpha_i B_i(t) \tag{42}$$

where, α_i is the coefficient of the ith Legendre polynomial and $B_i(t)$ is the ith Legendre polynomial. The starting and ending configurations are posed as equality constraints, and the object collision avoidance is posed by having the pendulum cart pass through a set point, which would ensure the cart leans enough to avoid the wall, and another constraint to ensure once the cart crosses the wall, it doesnt return. The cost function is the

(a) $\theta=0°$

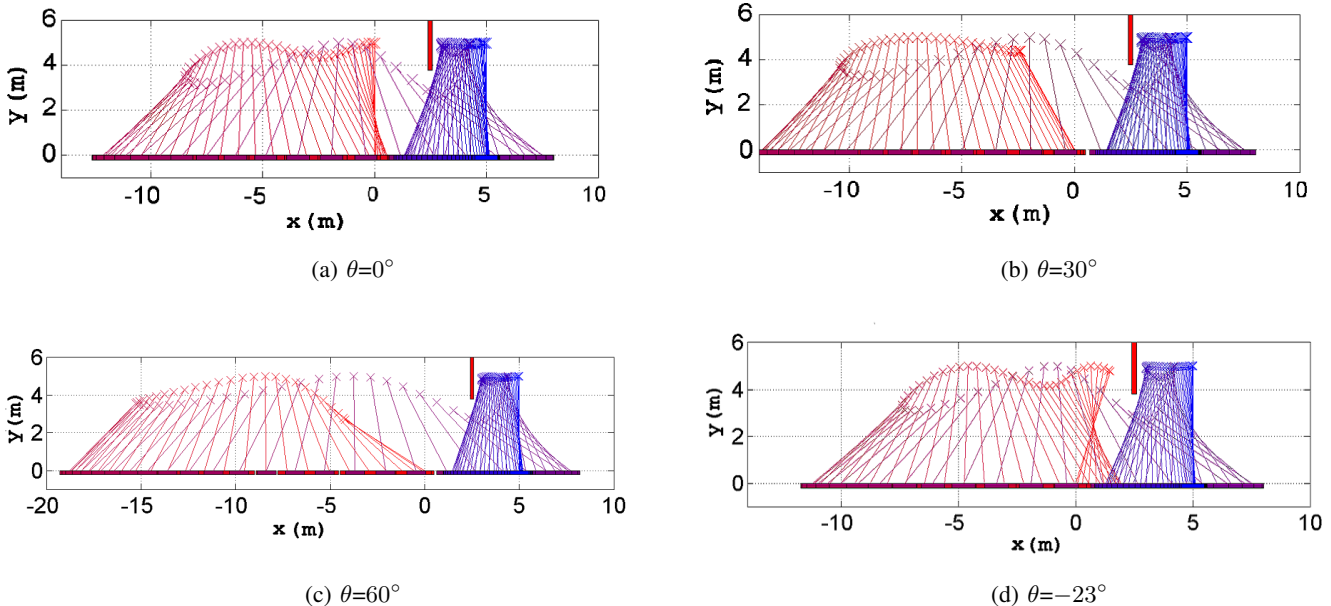(b) $\theta=30°$

(c) $\theta=60°$

(d) $\theta=-23°$

Fig. 8: Stick figure of the pendulum cart trajectory obtained for different initial pendulum lean angles. The color of the snapshots transition from red (initial configuration) to blue (final configuration), while the density is inversely proportional to the speed of the pendulum cart. As is clearly visible the pendulum bob avoids the wall.

$\mathcal{L}_2$ norm of the force applied on the cart,

$$J = \int_0^T |F(t)|^2 \, dt. \tag{43}$$

Thus, the flat output gives us the control input, $F(t)$, the s-states, $[\theta(t), \dot{\theta}(t)]$, and then using the initial condition for the r-states, $r(0) = [x(0), \dot{x}(0)] = [0, 0]$, numerical integration of (10) provides the full state trajectory which the optimization uses to compute an optimal solution that meets the constraints.

The resulting optimal trajectory of the pendulum bob on a cart is shown as stick figure plots in Figure 8 for various initial pendulum lean angles. It can be seen that the pendulum bob clearly avoids collision with the obstacle. The corresponding trajectory of the pendulum angle θ is shown in Figure 9, with the pendulum swinging almost 60 degrees on either side.

VI. CONCLUSION

We have introduced the notion of partial differential flatness, that enables efficient generation of dynamically feasible trajectories for a large class of underactuated systems that are not differentially flat. Several underactuated systems, such as the cart-pole, planar ballbot, tri-axial satellite attitude control testbed, Cubli balancing about both its edge and corner were shown to be partially differentially flat. We have also extended the notion of partial differential flatness to hybrid systems enabling dynamically feasible trajectory generation that spans multiple hybrid modes. We have illustrated a simple example of designing dynamically feasible trajectories in the presence of obstacles for the cart-pole system. In the near future, we

intend to demonstrate online planning of dynamically feasible trajectories for more complex systems in experiments.

The key benefit of partial differential flatness is in extending the concept and advantages of differential flatness to systems that are not differentially flat. It offers a way to analytically compute the input and a subset of the states using the flat outputs. Although integration is required to compute the remaining subset of the flat outputs, the system dynamics do not have to be integrated. Moreover, a differential constraint can be posed directly in the trajectory optimization process to completely avoid integration. Since partial differential flatness reduces the dimension of the integration and essentially the trajectory optimization, it has the potential of being faster. A more thorough theoretical and / or numerical analysis needs to be done to clearly establish this.

Partial differential flatness inherits all the limitations of differential flatness. These include not being robust and requiring a perfect model, requiring several time-derivatives of variables which makes physical implementation hard, not being a generally applicable method since not all systems are partially differentially flat, and non existence of a constructive method to find the flat outputs. In addition, partial differential flatness does not provide the complete state trajectory as a function of time without integration. Moreover, in most cases, the trajectory optimization is still a constrained nonlinear program, albeit of reduced dimension.

REFERENCES

[1] E. Aranda-Bricaire, C. H. Moog, and J.-B. Pomet, "A linear algebraic framework for dynamic feedback linearization," *IEEE Transactions on Automatic Control*, vol. 40, no. 1, pp. 127–132, January 1995.

[2] A. Chamseddine, Y. Zhang, C. Rabbath, C. Join, and D. Theilliol, "Flatness-based trajectory planning/replanning for a quadrotor unmanned aerial vehicle," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 2832–2848, October 2012.

[3] S. Cho, J. Shen, N. McClamroch, and D. Bernstein, "Equations of motion for the triaxial attitude control testbed," in *IEEE Conference on Decision and Control*, vol. 4, 2001, pp. 3429–3434.

[4] S. Cho, J. Shen, and N. H. Mcclamroch, "Mathematical models for the triaxial attitude control testbed," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 9, no. 2, pp. 165–192, 2003.

[5] M. Fliess, J. Levine, P. Martin, F. Ollivier, and P. Rouchon, "Flatness and dynamic feedback linearizability: Two approaches," in *European Control Conference*, Munich, Germany, September 1995, pp. 649–654.

[6] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, "Flatness and defect of non-linear systems: introductory theory and examples," *International journal of control*, vol. 61, pp. 1327–1361, 1995.

[7] S. Formentin and M. Lovera, "Flatness-based control of a quadrotor helicopter via feedforward linearization," in *IEEE Conference on Decision and Control and European Control Conference*, Dec 2011, pp. 6171–6176.

[8] M. Gajamohan, M. Muehlebach, T. Widmer, and R. D'Andrea, "The cubli: A reaction wheel based 3d inverted pendulum," in *Proc. European Control Conference*, 2013, pp. 268–274.

[9] H. Guo, F. Wang, H. Chen, and D. Guo, "Stability control of vehicle with tire blowout using differential flatness based mpc method," in *World Congress on Intelligent Control and Automation*, July 2012, pp. 2066–2071.

[10] S.-Y. Jiang and K.-T. Song, "Differential flatness-based motion control of a steer-and-drive omnidirectional mobile robot," in *IEEE International Conference on Mechatronics and Automation*, Aug 2013, pp. 1167–1172.

[11] P. Martin, R. M. Murray, and P. Rouchon, "Flat systems, equivalance and trajectory generation," Control and Dynamical Systems, California Institute of Technology, Tech. Rep. CaltechCDSTR:2003.008, April 2003.

[12] M. Muehlebach, G. Mohanarajah, and R. D'Andrea, "Nonlinear analysis and control of a reaction wheel-based 3d inverted pendulum," in *IEEE Conference on Decision and Control*, Florence, Italy, December 2013, pp. 1283–1288.

[13] R. M. Murray, M. Rathinam, and W. Sluis, "Di erential Flatness of Mechanical Control Systems : A Catalog of Prototype Systems," in *ASME International Mechanical Engineering Congress*, 1995, pp. 1–9.

[14] U. Nagarajan, G. Kantor, and R. Hollis, "Trajectory planning and control of an underactuated dynamically stable single spherical wheeled mobile robot," in *IEEE International Conference on Robotics and Automation*, 2009, pp. 3743–3748.

[15] U. Nagarajan and R. Hollis, "Shape space planner for shape-accelerated balancing mobile robots," *International Journal of Robotics Research*, vol. 32, no. 11, pp. 1323–1341, September 2013.

[16] F. Nicolau and W. Respondek, "Multi-input control-affine systems linearizable via one-fold prolongation and their flatness," in *IEEE Conference on Decision and Control*, Florence, Italy, December 2013, pp. 3249–3254.

[17] M. V. Nieuwstadt, M. Rathinam, and R. M. Murray, "Differential flatness and absolute equivalence of nonlinear control systems," *SIAM Journal of Control and Optimization*, vol. 36, no. 4, pp. 1225–1239, July 1998.

[18] M. Shomin and R. Hollis, "Differentially flat trajectory generation for a dynamically stable mobile robot," in *IEEE International Conference on Robotics and Automation*, 2013.

[19] P. S. P. D. Silva and C. C. Filho, "Relative flatness and flatness of implicit systems," *SIAM Journal of Control and Optimization*, vol. 39, no. 6, pp. 1929–1951, April 2001.

[20] M. Spong, "Partial feedback linearization of underactuated mechanical systems," in *International Conference on Intelligent Robots and Systems*, Munich, Germany, September 1994, pp. 314–321.

[21] K. Sreenath and V. Kumar, "Dynamics, control and planning for cooperative manipulation of payloads suspended by cables from multiple quadrotor robots," in *Robotics: Science and Systems*, 2013.

[22] C. P. Tang, "Differential flatness-based kinematic and dynamic control of a differentially driven wheeled mobile robot," in *IEEE International Conference on Robotics and Biomimetics*, Dec 2009, pp. 2267–2272.