2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Nonlinear Motion Control of CPG-based Movement with Applications to a Class of Swimming Robots

Yannick Morel, Mathieu Porez, Alexander Leonessa, and Auke J. Ijspeert

Abstract— **In bio-inspired robotics, use of a Central Pattern Generator (CPG) to coordinate actuation is fairly common. The gait achieved depends on a number of CPG parameters, which can be adjusted to control the robot's motion. This paper presents an output feedback motion control framework, addressing issues encountered when dealing with this type of control problem, including partial state measurements and system uncertainty. Efficacy of the presented approach is illustrated by results of numerical simulations in the case of a swimming robot.**

I. Introduction

Use of unmanned vehicles has been steadily increasing over the past decade, due in large parts to their aptitude in fulfilling missions proving possibly too dull, dirty, or dangerous for humans to pursue. While all aspects of unmanned systems' technology have seen considerable improvements as a result, aspects relating to their agility and autonomy have received special attention. Indeed, utility of such platforms is oftentimes in direct correlation with their capacity to evolve in cluttered and possibly hazardous environments, for extended periods of time.

The most direct manner in which one may enhance agility and autonomy of a vehicle is by improving its actuation system. In pursuing such a goal, designers have found that they could turn to nature to find elegant solutions, leading for example to the design of flapping flying robots ([1]), swimming robots ([2], [3]), as well as legged ground robots ([4], [5]). A common characteristic of these systems is that movement is generated through coordination of a number of actuated degrees of freedom, in order to, for instance, generate propelling body and/or appendage oscillations for swimming robots ([3]), or walking gaits for legged robots ([6]). Such coordination can be achieved using a number of different approaches. For example, one can rely on a Central Pattern Generator (CPG, [7–10]), which uses coupled nonlinear oscillators to ensure coordination between actuated degrees of freedom. CPGs enjoy a growing popularity, and have been successfully applied to a wide variety of systems ([6], [11–14]).

However, while use of a CPG allows to generate movement, it does not necessarily provide precise control over the vehicle's motion. For example, if the CPG-based limb coordination and resulting gait for a legged robotic system are such that the robot is set in motion, the pattern generator does not play the role of motion controller. In particular, while movement is produced, what precise movement is created is a byproduct of a number of CPG-specific parameters characterizing the particular gait followed. Thus, direction and speed of motion will depend on some CPG parameters and, while adjustment of these parameters allows to steer the vehicle in real time (that is, allows to perform *motion control*), that is not a role performed by the CPG itself.

In the literature, the manner in which these parameters are adjusted to achieve motion control is not always clear. When acknowledged, the issue is oftentimes only addressed in vague, qualitative terms ([14]), using ad hoc, heuristic (bio-inspired) solutions ([15]), or by resorting to somewhat simplistic control techniques ([16]). However, for the class of systems considered, this control problem is non-trivial. Indeed, a system model for the type of robotic platform under consideration is typically nonlinear, can prove fairly complicated, and feature significant levels of uncertainty. While the addition of a CPG does not increase uncertainty, it does extend the system's relative degree ([17]) in most instances. Furthermore, it is rarely practical (if even possible) to include sensors measuring (or allowing to reconstruct) all state variables for a given mobile robotic platform. Hence, control design has to account for the fact that only partial state measurements are available, which constitutes another complicating factor.

To address system uncertainty in the control design for nonlinear systems, one can resort to adaptive control techniques (see [18–21]). Classical direct and indirect adaptive control techniques are commonly used to account for linearly parameterized system uncertainty ([18–20]). If, however, the uncertainty is structural (that is, beyond what could be framed as linearly parametric), which is likely to be the case for the type of robots investigated in the following, then one typically turns to Neural Network-based (NN-based) adaptive control techniques ([21]). While NN-based techniques allow to handle greater system uncertainty, they suffer from the drawback that, when replacing in the control design the system model with the output of a neural network, one effectively discards most available system knowledge.

Yet, while the systems we are interested in are bound to

This work was supported by the European Commission, Information Society and Media, Future and Emerging Technologies (FET), ANGELS project, contract number: 231845.

Y. Morel is with the BioRobotics Laboratory, Bioengineering Institute, Ecole Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland, yannick.morel@epfl.ch

M. Porez is with the IRCCyN Laboratory, Mines de Nantes, 44307 Nantes, France, mathieu.porez@mines-nantes.fr

A. Leonessa is with the Faculty of Mechanical Engineering (Vibration and Acoustics Laboratory), Virginia Polytechnic and State University, Blacksburg, VA 24061, USA, leonessa@vt.edu

A. J. Ijspeert is with the Faculty of Bioengineering (BioRobotics Laboratory), Ecole Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland, auke.ijspeert@epfl.ch

feature some level of uncertainty, there still remains system knowledge available, which should prove valuable to control design. In particular, the manner in which CPG parameters affect motion is often qualitatively known. For illustration, consider the example of an anguilliform swimming robot such as that described in [3]. Its swimming gait is characterized by a number of CPG parameters, such as frequency of body undulation, noted ν, and average body curvature α. While characterizing a precise model of the system may prove difficult, simulation results, experiments and common sense indicate that "*the faster it wiggles, the faster it moves,*" meaning that greater values of body undulation frequency ν lead to faster displacement, and "*the more it bends, the more it turns,*" that is, non-zero values of average body curvature α result in non-zero angular velocity in yaw.

To exploit such qualitative knowledge of the system's input/output behavior, while simultaneously addressing issues stemming from system uncertainty and partial state measurements, we propose the use of an output predictor-based control technique, such as that presented in [22], [23]. More specifically, available system knowledge is used in the construction of an output predictor, whose structure is not constrained to be Luenberger-like ([18], [24]), meaning that the output predictor's form is not limited to strictly replicating that of an actual model of the system (plus correction term). This flexibility allows to accommodate and exploit various types of system information, including qualitative and very general insights ("α *positive means it goes right*"), or if need be, more specific and quantitative model information. Through the use of nonlinear control techniques, one is able to design such a predictor so that it provides stability and convergence guarantees. More specifically, we construct it so that, accepting the same control input as the actual system's, it is guaranteed to predict the (measured) system output with arbitrary accuracy. As a result, the prediction error, which acts as a measure of prediction quality, is ultimately bounded ([25], [26]), with arbitrarily small ultimate bound.

Once such an output predictor is created, solving the control problem becomes straightforward. Indeed, while the output predictor shares (by design) a number of characteristics with the actual system, the predictor has the advantage of being a precisely known virtual system, integrated locally. Because of these features, designing a control input so as to achieve convergence of the predictor's output to a given desired trajectory is a rather simple proposition. Depending on the used predictor structure, different control techniques may be used to derive an appropriate control action. In the following, we use a simple nonlinear control law, guaranteeing convergence of the predictor's output to a given desired trajectory. Combining this tracking result with the fact that, by design, the predictor's output is guaranteed to remain arbitrarily close to the actual system's, we are able to guarantee bounded-error tracking of the desired trajectory by the actual system's output.

This paper is structured as follows. Section II describes in greater detail the control strategy pursued. Application of this strategy to a class of swimming robots is presented in Section III, including results of numerical simulations. Section IV concludes this paper.

II. Control Framework

Consider the following general class of systems,

$$\dot{x}(t) = f(x(t),u(t),t), \quad x(0)=x_0, \quad t \geqslant 0, \quad (1)$$
$$y(t) = h(x(t),t), \quad (2)$$

where $x(t) \in \mathbb{R}^n$, $t \geqslant 0$, $f(x,u,t)$ is locally Lipschitz in x and piecewise continuous in t, the control input $u(t) \in \mathbb{R}^m$ belongs to the set $\mathcal{U}$ of admissible control inputs, such that for all $u(t) \in \mathcal{U}$, equation (1) has a single solution forward in time ([26]), and the system output $y(t) \in \mathbb{R}^p$.

The general type of systems which we are interested in, that is, unmanned vehicles augmented with a CPG, can be described by a model of the form of (1)–(2). In particular, their state vector $x(t)$, $t \geqslant 0$, includes a number of state variables characterizing position, attitude and velocity of a given point of the robot (typically either that of its center of gravity or of an extremity, end limb or head), state variables characterizing body configuration, such as angle and angular velocity of joints, and additional state variables relating to limb motion coordination, such as CPG oscillator states, for instance. The overall system can prove quite large and difficult to deal with in a classical model-based control design procedure.

In the following, we will use the approach laid out in [22], [23] to design a control law for this type of systems. While the approach is fairly general and able to solve the motion control problem for a wide variety of systems, we will, for ease of exposition, limit considerations to a subset of all systems described by (1)–(2). In particular, we will consider a class of systems fitting our targeted application (swimming robot AmphiBotIII, [3]). More specifically, we will investigate systems which are minimum phase, affine in command, and have an external (that is, actuated) dynamics of the form

$$\dot{x}_{\mathrm{e}1}(t) = x_{\mathrm{e}2}(t), \quad x_{\mathrm{e}1}(0)=x_{\mathrm{e}10}, \quad t \geqslant 0, \quad (3)$$
$$\dot{x}_{\mathrm{e}2}(t) = f_{\mathrm{e}}(x_{\mathrm{e}}(t),x_{\mathrm{i}}(t),t) + g_{\mathrm{e}}(x_{\mathrm{e}}(t),x_{\mathrm{i}}(t),t)u(t), \quad x_{\mathrm{e}2}(0)=x_{\mathrm{e}20}, \quad (4)$$
$$y(t) = x_{\mathrm{e}1}(t), \quad (5)$$

where $x_{\mathrm{e}}(t) \triangleq \begin{bmatrix} x_{\mathrm{e}1}^{\mathrm{T}}(t) & x_{\mathrm{e}2}^{\mathrm{T}}(t) \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{2m}$, $t \geqslant 0$, represents the state of the system's external dynamics, while $x_{\mathrm{i}}(t) \in \mathbb{R}^{n-2m}$ is that of the internal dynamics ([17]). In addition, we assume that (3)–(5) is controllable, and that the system output $y(t)$ is sufficiently smooth.

The control objective is to design the control input $u(t)$, $t \geqslant 0$, so that the system output $y(t)$ follows a desired pattern, for instance, tracking of a desired output trajectory $y_{\mathrm{d}}(t)$. To address this problem, we begin by constructing an output predictor, which aims to capture the input/output dynamical behavior of (3)–(5). The following theorem provides the tools to design one such predictor. In latter stages, we will take advantage of this result to facilitate design of a control law for (3)–(5).

Theorem 2.1: Consider the external dynamics given by (3)–(5), along with the following output predictor,

$$\dot{\hat{x}}_1(t) = v(t), \qquad \hat{x}_1(0) = x_{\mathrm{e}10}, \quad t \geqslant 0, \quad (6)$$

$$\dot{v}(t) = \varphi(t) + \eta(\hat{x}_1(t), v(t), t) + \gamma(y(t), v(t), t)u(t), \quad v(0) = 0_m, \quad (7)$$

where $\hat{x}_1(t) \in \mathbb{R}^m$, $t \geqslant 0$, is a prediction of the actual, measured output $y(t)$, $\gamma(y, v, t) \in \mathbb{R}^{m\times m}$ is chosen nonsingular for all y, $v \in \mathbb{R}^m$, $t \geqslant 0$, and $\eta(\cdot) \in \mathbb{R}^m$ is a continuous function of its arguments, included to accommodate known system dynamics, if available. In addition,

$$\begin{aligned}\varphi(t) \triangleq\ & -2\zeta\omega_{\mathrm{n}}\dot{y}_{\mathrm{f}}(t) + \omega_{\mathrm{n}}^2(y(t) - u_{\mathrm{f}}(t) - y_{\mathrm{f}}(t)) - \beta_1 u_{\mathrm{f}}(t)\\ & -\beta_2\dot{u}_{\mathrm{f}}(t) - \bar{A}_{1\mathrm{p}}\big(\bar{A}_{1\mathrm{p}}e_{1\mathrm{p}}(t) + e_{2\mathrm{p}}(t)\big) - A_{2\mathrm{p}}e_{2\mathrm{p}}(t)\\ & +P_{2\mathrm{p}}^{-1}P_{1\mathrm{p}}e_{1\mathrm{p}}(t) + \bar{A}_{1\mathrm{p}}\bar{A}_{1\mathrm{p}}^{\mathrm{T}}P_{2\mathrm{p}}e_{2\mathrm{p}}(t)/2, \quad t \geqslant 0, \quad (8)\end{aligned}$$

with $A_{1\mathrm{p}}$, $A_{2\mathrm{p}} \in \mathbb{R}^{m\times m}$ chosen Hurwitz, $\bar{A}_{1\mathrm{p}} \triangleq A_{1\mathrm{p}} - P_{1\mathrm{p}}/2$, filter constants β_1, β_2, ζ, $\omega_{\mathrm{n}} > 0$, prediction errors $e_{1\mathrm{p}}(t)$, $e_{2\mathrm{p}}(t)$, $t \geqslant 0$, defined as follows,

$$e_{1\mathrm{p}}(t) \triangleq y(t) - \hat{x}_1(t), \quad t \geqslant 0, \quad (9)$$

$$e_{2\mathrm{p}}(t) \triangleq \dot{y}_{\mathrm{f}}(t) + \dot{u}_{\mathrm{f}}(t) - (A_{1\mathrm{p}} - P_{1\mathrm{p}}/2)e_{1\mathrm{p}}(t) - v(t), \quad (10)$$

with matrices $P_{1\mathrm{p}}$, $P_{2\mathrm{p}} \in \mathbb{R}^{m\times m}$ obtained from the Lyapunov equations

$$A_{1\mathrm{p}}^{\mathrm{T}}P_{1\mathrm{p}} + P_{1\mathrm{p}}A_{1\mathrm{p}} = -Q_{1\mathrm{p}}, \ A_{2\mathrm{p}}^{\mathrm{T}}P_{2\mathrm{p}} + P_{2\mathrm{p}}A_{2\mathrm{p}} = -Q_{2\mathrm{p}}, \quad (11)$$

where $Q_{1\mathrm{p}}$, $Q_{2\mathrm{p}} > 0$. Finally, the filtered input and output variables $u_{\mathrm{f}}(t)$, $y_{\mathrm{f}}(t)$, $t \geqslant 0$, in (8) are obtained from

$$\begin{aligned}\ddot{u}_{\mathrm{f}}(t) =\ & -\beta_1 u_{\mathrm{f}}(t) - \beta_2\dot{u}_{\mathrm{f}}(t) + \gamma(y(t), v(t), t)u(t)\\ & +\eta(\hat{x}_1(t), v(t), t), \ \dot{u}_{\mathrm{f}}(0) = u_{\mathrm{f}}(0) = 0_m, \ t \geqslant 0, \quad (12)\\ \ddot{y}_{\mathrm{f}}(t) =\ & -2\zeta\omega_{\mathrm{n}}\dot{y}_{\mathrm{f}}(t) + \omega_{\mathrm{n}}^2(y(t) - u_{\mathrm{f}}(t) - y_{\mathrm{f}}(t)),\\ & \dot{y}_{\mathrm{f}}(0) = 0_m, \quad y_{\mathrm{f}}(0) = x_{\mathrm{e}10}. \quad (13)\end{aligned}$$

The solution $v(t)$, $t \geqslant 0$, to (7) guarantees uniform ultimate boundedness of (9)–(10).

Proof: The time derivative of the prediction error $e_{1\mathrm{p}}(t)$, $t \geqslant 0$, is given by

$$\dot{e}_{1\mathrm{p}}(t) = \dot{y}(t) - v(t), \quad t \geqslant 0. \quad (14)$$

Note that, $y(t)$, $t \geqslant 0$, being sufficiently smooth, one can design filter parameters β_1, β_2, ζ, ω_{n}, such that $\dot{y}_{\mathrm{f}}(t) = \dot{y}(t) - \dot{u}_{\mathrm{f}}(t) - \varepsilon(t)$, with bounded higher-frequency filtering residue $\varepsilon(t)$. In particular, we design our filters so that there exists $\epsilon \in \mathbb{R}$ such that $\|\varepsilon(t)\| \leqslant \sqrt{\epsilon/2}$. Decomposing the system input derivative in (14) into its filtered counterpart and (bounded) high-frequency residue, we obtain

$$\begin{aligned}\dot{e}_{1\mathrm{p}}(t) &= \dot{y}_{\mathrm{f}}(t) + \dot{u}_{\mathrm{f}}(t) - v(t) + \varepsilon(t)\\ &= (A_{1\mathrm{p}} - P_{1\mathrm{p}}/2)e_{1\mathrm{p}}(t) + e_{2\mathrm{p}}(t) + \varepsilon(t), \quad t \geqslant 0. \quad (15)\end{aligned}$$

At this stage of the proceedings, it is important to highlight the fact that, while we do not include the higher-frequency residual term $\varepsilon(t)$, $t \geqslant 0$, within the control law design (among other reasons, to improve smoothness of control action), we account for its influence on system stability, as required by a rigorous stability analysis.

Next, consider the time derivative of $e_{2\mathrm{p}}(t)$, $t \geqslant 0$, which is of the form

$$\begin{aligned}\dot{e}_{2\mathrm{p}}(t) &= \ddot{y}_{\mathrm{f}}(t) + \ddot{u}_{\mathrm{f}}(t) - \bar{A}_{1\mathrm{p}}\dot{e}_{1\mathrm{p}}(t) - \dot{v}(t)\\ &= -2\zeta\omega_{\mathrm{n}}\dot{y}_{\mathrm{f}}(t) + \omega_{\mathrm{n}}^2(y(t) - u_{\mathrm{f}}(t) - y_{\mathrm{f}}(t)) - \beta_1 u_{\mathrm{f}}(t)\\ &\quad -\beta_2\dot{u}_{\mathrm{f}}(t) + \gamma(y(t), v(t), t)u(t) + \eta(\hat{x}_1(t), v(t), t)\\ &\quad -\bar{A}_{1\mathrm{p}}\big(\bar{A}_{1\mathrm{p}}e_{1\mathrm{p}}(t) + e_{2\mathrm{p}}(t)\big) - \gamma(y(t), v(t), t)u(t)\\ &\quad -\varphi(t) - \eta(\hat{x}_1(t), v(t), t) - \bar{A}_{1\mathrm{p}}\varepsilon(t)\\ &= -2\zeta\omega_{\mathrm{n}}\dot{y}_{\mathrm{f}}(t) + \omega_{\mathrm{n}}^2(y(t) - u_{\mathrm{f}}(t) - y_{\mathrm{f}}(t)) - \beta_1 u_{\mathrm{f}}(t)\\ &\quad -\beta_2\dot{u}_{\mathrm{f}}(t) - \bar{A}_{1\mathrm{p}}\big(\bar{A}_{1\mathrm{p}}e_{1\mathrm{p}}(t) + e_{2\mathrm{p}}(t)\big) - \bar{A}_{1\mathrm{p}}\varepsilon(t)\\ &\quad -\varphi(t), \quad t \geqslant 0. \quad (16)\end{aligned}$$

Substituting (8) into (16), we obtain

$$\begin{aligned}\dot{e}_{2\mathrm{p}}(t) =\ & -P_{2\mathrm{p}}^{-1}P_{1\mathrm{p}}e_{1\mathrm{p}}(t) + (A_{2\mathrm{p}} - \bar{A}_{1\mathrm{p}}\bar{A}_{1\mathrm{p}}^{\mathrm{T}}P_{2\mathrm{p}}/2)e_{2\mathrm{p}}(t)\\ & -\bar{A}_{1\mathrm{p}}\varepsilon(t), \quad t \geqslant 0. \quad (17)\end{aligned}$$

We then consider the following Lyapunov function candidate,

$$V(e_{1\mathrm{p}}, e_{2\mathrm{p}}) = e_{1\mathrm{p}}^{\mathrm{T}}P_{1\mathrm{p}}e_{1\mathrm{p}} + e_{2\mathrm{p}}^{\mathrm{T}}P_{2\mathrm{p}}e_{2\mathrm{p}}. \quad (18)$$

The time derivative of (18) along the trajectories of (15) and (16) is given by

$$\begin{aligned}\dot{V}(t) =\ & -e_{1\mathrm{p}}^{\mathrm{T}}(t)Q_{1\mathrm{p}}e_{1\mathrm{p}}(t) - e_{2\mathrm{p}}^{\mathrm{T}}(t)Q_{2\mathrm{p}}e_{2\mathrm{p}}(t)\\ & -e_{1\mathrm{p}}^{\mathrm{T}}(t)P_{1\mathrm{p}}^2e_{1\mathrm{p}}(t) - e_{2\mathrm{p}}^{\mathrm{T}}(t)P_{2\mathrm{p}}\bar{A}_{1\mathrm{p}}\bar{A}_{1\mathrm{p}}^{\mathrm{T}}P_{2\mathrm{p}}e_{2\mathrm{p}}(t)\\ & +2e_{1\mathrm{p}}^{\mathrm{T}}(t)P_{1\mathrm{p}}\varepsilon(t) - 2e_{2\mathrm{p}}^{\mathrm{T}}(t)P_{2\mathrm{p}}\bar{A}_{1\mathrm{p}}\varepsilon(t), \ t \geqslant 0. \quad (19)\end{aligned}$$

Using the completion of the square rule, we obtain

$$\begin{aligned}2e_{1\mathrm{p}}^{\mathrm{T}}P_{1\mathrm{p}}\varepsilon &= -(P_{1\mathrm{p}}e_{1\mathrm{p}} - \varepsilon)^{\mathrm{T}}(P_{1\mathrm{p}}e_{1\mathrm{p}} - \varepsilon) + \varepsilon^{\mathrm{T}}\varepsilon\\ &\quad +e_{1\mathrm{p}}^{\mathrm{T}}P_{1\mathrm{p}}^2e_{1\mathrm{p}}, \quad (20)\\ -2e_{2\mathrm{p}}^{\mathrm{T}}P_{2\mathrm{p}}\bar{A}_{1\mathrm{p}}\varepsilon &= -(\bar{A}_{1\mathrm{p}}^{\mathrm{T}}P_{2\mathrm{p}}e_{2\mathrm{p}} + \varepsilon)^{\mathrm{T}}(\bar{A}_{1\mathrm{p}}^{\mathrm{T}}P_{2\mathrm{p}}e_{2\mathrm{p}} + \varepsilon)\\ &\quad +e_{2\mathrm{p}}^{\mathrm{T}}P_{2\mathrm{p}}\bar{A}_{1\mathrm{p}}\bar{A}_{1\mathrm{p}}^{\mathrm{T}}P_{2\mathrm{p}}e_{2\mathrm{p}} + \varepsilon^{\mathrm{T}}\varepsilon. \quad (21)\end{aligned}$$

Substituting (20)–(21) into (19) yields

$$\dot{V}(t) \leqslant -e_{1\mathrm{p}}^{\mathrm{T}}(t)Q_{1\mathrm{p}}e_{1\mathrm{p}}(t) - e_{2\mathrm{p}}^{\mathrm{T}}(t)Q_{2\mathrm{p}}e_{2\mathrm{p}}(t) + \epsilon, \ t \geqslant 0. \quad (22)$$

It follows that $\dot{V}(t)$, $t \geqslant 0$, remains strictly negative outside of $\{(e_{1\mathrm{p}}, e_{2\mathrm{p}}) : e_{1\mathrm{p}}^{\mathrm{T}}Q_{1\mathrm{p}}e_{1\mathrm{p}} + e_{2\mathrm{p}}^{\mathrm{T}}Q_{2\mathrm{p}}e_{2\mathrm{p}} \leqslant \epsilon\}$, which allows to conclude ultimate boundedness of $(e_{1\mathrm{p}}(t), e_{2\mathrm{p}}(t))$ ([26]).

In addition, the ultimate bound is $\chi \triangleq \min(e_{1\mathrm{p}}^{\mathrm{T}}P_{1\mathrm{p}}e_{1\mathrm{p}} + e_{2\mathrm{p}}^{\mathrm{T}}P_{2\mathrm{p}}e_{2\mathrm{p}})$, subject to the constraint $e_{1\mathrm{p}}^{\mathrm{T}}Q_{1\mathrm{p}}e_{1\mathrm{p}} + e_{2\mathrm{p}}^{\mathrm{T}}Q_{2\mathrm{p}}e_{2\mathrm{p}} = \epsilon$. To better characterize χ, define

$$N \triangleq \begin{bmatrix} Q_{1\mathrm{p}}^{-1}P_{1\mathrm{p}} & 0_{m\times m}\\ 0_{m\times m} & Q_{2\mathrm{p}}^{-1}P_{2\mathrm{p}}\end{bmatrix}. \quad (23)$$

Using Lagrange multipliers, we obtain $\chi = \epsilon\lambda_{\max}(N)$, where $\lambda_{\max}(N)$ denotes the maximum eigenvalue of N. ■

What we achieve in using Theorem 2.1 is constructing a new system (6)–(7), distinct from the original system (3)–(5), but sharing a number of key features with it. In particular, both systems accept the same input $u(t)$, $t \geqslant 0$, and their output are ultimately arbitrarily close to one another. In addition, one should note that we have constructed the output predictor (6)–(7) in such a way that it is controllable, and

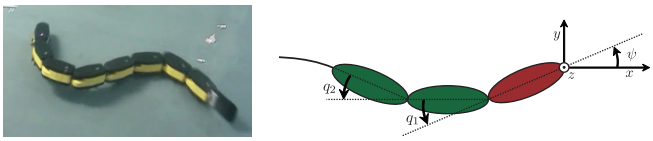

Fig. 1. Left: AmphibotIII in an eight-module configuration; right: schematics of three-module swimming configuration.

designing a control law so that $\hat{x}_1(t)$, $t \geqslant 0$, tracks a given desired output $y_{\mathrm{d}}(t)$ is straightforward. More specifically, consider the following control law,

$$u(t) = \gamma^{-1}(y(t), v(t), t)(-\varphi(t) - \eta(\hat{x}_1(t), v(t), t) + \ddot{y}_{\mathrm{d}}(t) - k_1 e_{\mathrm{t}}(t) - k_2 \dot{e}_{\mathrm{t}}(t)), \quad t \geqslant 0, \tag{24}$$

where $e_{\mathrm{t}}(t) \triangleq \hat{x}_1(t) - y_{\mathrm{d}}(t)$, $t \geqslant 0$, and k_1, $k_2 \in \mathbb{R}$ are chosen so that

$$A \triangleq \begin{bmatrix} 0 & 1 \\ -k_1 & -k_2 \end{bmatrix}, \tag{25}$$

is Hurwitz. Substituting (24) into (7), it directly follows that $e_{\mathrm{t}}(t)$, $t \geqslant 0$, converges exponentially to the origin.

Combining prediction and tracking results, we obtain that $e_{1\mathrm{p}}(t) + e_{\mathrm{t}}(t) = y(t) - y_{\mathrm{d}}(t)$ converges to a neighborhood of the origin, meaning that the system output $y(t)$ converges to a neighborhood of the specified desired trajectory $y_{\mathrm{d}}(t)$. The advantage of designing the control law for the output predictor, rather than for the actual system, is significant. More specifically, while the system's dynamics (3)–(5) may be complicated, feature a number of unmeasured and unobservable state variables, and most likely be uncertain, that is not the case for the output predictor (6)–(7). Indeed, the predictor is not required to reproduce the system's general complexity to approximate its output. Furthermore, because the predictor is constructed by the control designer, its dynamics are precisely known (given by (7) and (8)), and its structure can be designed so as to facilitate derivation of a control law. In addition, it is a virtual, numerical system, integrated locally, whose state variables are readily available for feedback, thus elegantly circumventing the issue of output feedback.

Remark 2.1: Note that we do not provide a closed form for $\eta(\cdot)$ in (7). This function is included for flexibility, to allow the control designer to take advantage of available system information (trying to emulate $f_{\mathrm{e}}(\cdot)$ in (4) for example). If little or no exploitable system information is available, one may select $\eta(\hat{x}_1(t), v(t), t) \equiv 0$, $t \geqslant 0$.

III. Application to Swimming Robot

To illustrate efficacy of the approach, we apply it to address the motion control problem for a specific swimming robot, AmphibotIII (left in Figure 1, [3]). Amphibot is a modular robot, constituted of a number of identical modules, linked together with actuated revolute joints, as described in Figure 1. In the configuration considered hereafter, the robot is constituted of a head module (in red in Figure 1, right), two body modules (in green), and a rectangular, caudal rubber fin. While the application of output predictor-based control

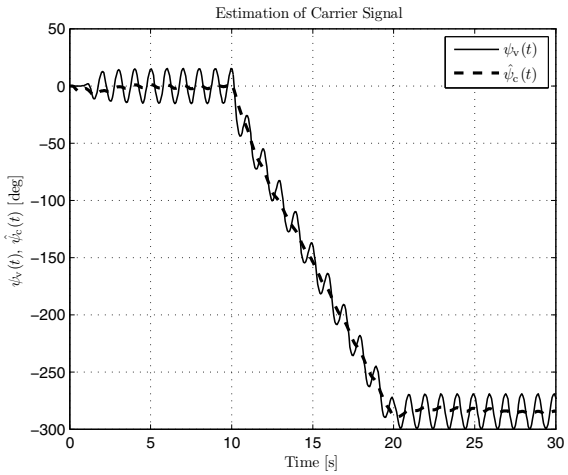

Fig. 2. Prediction of carrier signal, measured motion direction $\psi_{\mathrm{v}}(t)$ (solid), estimation $\hat{\psi}_{\mathrm{c}}(t)$ of carrier signal (dashed)

to classical unmanned vehicles is rather straightforward (see [23]), that is not necessarily the case for this specific type of swimming robot.

In particular, the problem to be addressed is the design of a control law, updating appropriate CPG parameters in real time, to ensure that the robot is moving in a desired manner. Yet, isolating an output variable satisfactorily characterizing the system's motion is not straightforward. In other words, evaluating whether or not the robot is moving in a desired manner is not immediate. Indeed, due to the manner in which the system propels itself (through oscillation of body angles $q_1(t)$ and $q_2(t)$, $t \geqslant 0$, shown in Figure 1), all measurable outputs show an oscillating component, whose frequency is identical to that of body undulations. For instance, velocity of the system's center of mass is oscillating both in amplitude and direction. As a result, it can prove difficult to distinguish the contribution to a given output signal of body undulation-induced oscillations from that of the system's actual net movement. To illustrate this point, consider the measured displacement velocity direction $\psi_{\mathrm{v}}(t)$ shown in Figure 2 (solid). This angle can in practice be calculated based on a linear combination of the head module's yaw, and body angles $q_1(t)$, $q_2(t)$, $t \geqslant 0$. The maneuver shown in Figure 2 corresponds to a right-hand-side turn. In the transition from steady forward motion to right-hand-side turn, after 10 seconds, it is not immediately clear whether the robot is actually turning or whether the decrease in $\psi_{\mathrm{v}}(t)$ is due to body oscillations. Accordingly, and to better assess robot movements, it would be desirable to estimate the underlying carrier signal about which output variables oscillate, as it provides a better measure of the platform's net movement.

As previously mentioned, in practice one may reconstruct with accuracy the direction $\psi_{\mathrm{v}}(t)$, $t \geqslant 0$, of the center of mass' velocity vector using a linear combination of the yaw $\psi(t)$, $t \geqslant 0$ (as described in Figure 1, measure obtained from an electronic compass), and body angles $q_i(t)$, $i = 1, 2$, (measured by encoders on the motor axes). In the following,

in order to better characterize the system's movement, instead of designing an output predictor to construct an estimate of $\psi_{\rm v}(t)$, we decompose this output signal into carrier and oscillating components, $\psi_{\rm v}(t) = \psi_{\rm c}(t) + \psi_{\rm o}(t)$, and reconstruct an estimate of the underlying carrier signal $\psi_{\rm c}(t)$, using measures of $\psi_{\rm v}(t)$.

To that end, it is necessary to estimate the oscillating component $\psi_{\rm o}(t)$, $t \geqslant 0$. Luckily, the frequency of $\psi_{\rm o}(t)$ is known and is the same as that of the CPG. In addition, for a given CPG amplitude, amplitude of $\psi_{\rm o}(t)$ remains constant and can be fairly easily estimated. Hence, to estimate the oscillating component of the output, one needs only to estimate its phase, which can be achieved using a number of different techniques, such as extremum matching (that is, detecting signal extrema and adjusting the estimated signal's phase accordingly).

A. Design of Output Predictor

Ultimately, it is possible to construct an estimate $\hat{\psi}_{\rm o}(t)$, $t \geqslant 0$, of the oscillating component $\psi_{\rm o}(t)$ so that there exists $\varepsilon_1, t_1 \in \mathbb{R}^+$ such that $\|\psi_{\rm o}(t) - \hat{\psi}_{\rm o}(t)\| \leqslant \varepsilon_1$, $t \geqslant t_1$. We are then able to take advantage of that estimate to construct a predictor for $\psi_{\rm c}(t)$ using Theorem 2.1. The first step in doing so consists in assessing what available knowledge regarding the system's dynamics is available, and exploiting that knowledge to construct appropriate $\eta(\cdot)$ and $\gamma(\cdot)$ functions. In our case, simulation and experimental results show that for a given body curvature $\alpha(t)$, one obtains a given turning speed $\dot{\psi}_{\rm c}(t)$. See for instance the simulation results reported in Figure 2, where body curvature $\alpha(t)$ is kept at 25° for $10{\rm s} \leqslant t \leqslant 20{\rm s}$. As a result, we observe a somewhat steady turning speed. Meaning that $\dot{\psi}_{\rm c}(t) \simeq g\alpha(t)$, for some $g \in \mathbb{R}$. In addition, we observe in some circumstances a delay between a change in $\alpha(t)$ and the resulting change in turning speed $\dot{\psi}_{\rm c}(t)$. We conclude that it may be possible to approximate the input/output relationship between $\alpha(t)$ and $\psi_{\rm c}(t)$ by having $g\alpha(t)$ acting on $\dot{\psi}_{\rm c}(t)$ through, for instance, a first order low-pass filter, with some time constant $\tau \in \mathbb{R}$. Or, more specifically, $\ddot{\psi}_{\rm c}(t) \simeq \tau(g\alpha(t) - \dot{\psi}_{\rm c}(t))$, $t \geqslant 0$.

Note that the above analysis is essentially qualitative. More specifically, we have no strict equality or closed form model but only general trends, and we do not necessarily have values to attach to g and τ. Yet, we are able to exploit this knowledge in our design, by adjusting the form of $\eta(\cdot)$ and $\gamma(\cdot)$ in (7) accordingly. In particular, we construct the following output predictor,

$$\begin{aligned}\dot{\hat{\psi}}_{\rm c}(t) &= \varphi(t) - \tau\dot{\hat{\psi}}_{\rm c}(t) + \tau g\alpha(t), \quad \dot{\hat{\psi}}_{\rm c}(0) = 0,\\ &\qquad \hat{\psi}_{\rm c}(0) = y(0) - \hat{\psi}_{\rm o}(0), \quad t \geqslant 0, \quad (26)\\ \hat{y}(t) &= \hat{\psi}_{\rm c}(t) + \hat{\psi}_{\rm o}(t), \quad (27)\end{aligned}$$

where $\hat{\psi}_{\rm c}(t)$ is an estimate of the carrier signal $\psi_{\rm c}(t)$, $g, \tau \in \mathbb{R}$, $\alpha(t)$ is the control input and represents average body curvature,

$$\begin{aligned}\varphi(t) = {} & -2\zeta\omega_{\rm n}\dot{y}_{\rm f}(t) + \omega_{\rm n}^2(\psi_{\rm v}(t) - \hat{\psi}_{\rm o}(t) - u_{\rm f}(t) - y_{\rm f}(t))\\ & -u_{\rm f}(t) - \dot{u}_{\rm f}(t) - \bar{a}_1\big(\bar{a}_1 e_{1{\rm p}}(t) + e_{2{\rm p}}(t)\big) - a_2 e_{2{\rm p}}(t)\\ & + e_{1{\rm p}}(t) + \bar{a}_1^2 e_{2{\rm p}}(t)/2, \ \ t \geqslant 0, \quad (28)\end{aligned}$$

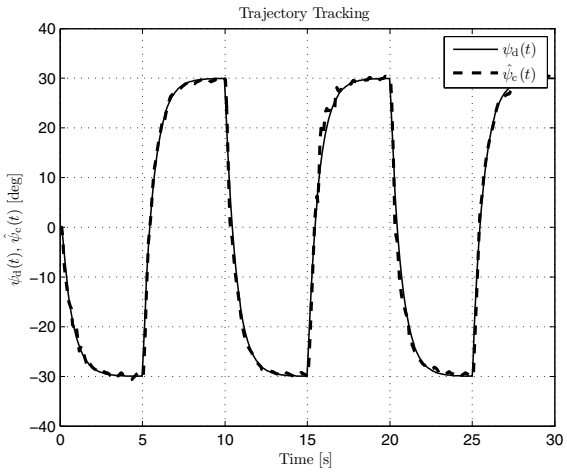

Fig. 3. Tracking of desired motion direction $\psi_{\rm d}(t)$ (solid), by the estimated carrier signal $\hat{\psi}_{\rm c}(t)$ (dashed).

where a_1, $a_2 < 0$, $\bar{a}_1 = a_1 - 1/2$, $y_{\rm f}(t)$ is obtained from

$$\begin{aligned}\ddot{y}_{\rm f}(t) = {} & -2\zeta\omega_{\rm n}\dot{y}_{\rm f}(t) + \omega_{\rm n}^2(\psi_{\rm v}(t) - \hat{\psi}_{\rm o}(t) - u_{\rm f}(t) - y_{\rm f}(t)),\\ & \dot{y}_{\rm f}(0) = 0_m, \quad y_{\rm f}(0) = y(0) - \hat{\psi}_{\rm o}(0), \quad (29)\end{aligned}$$

$u_{\rm f}(t)$ is obtained from (12) with $\beta_1 = \beta_2 = 1$, and

$$e_{1{\rm p}}(t) \triangleq \psi_{\rm v}(t) - \hat{y}(t), \quad t \geqslant 0, \quad (30)$$

$$e_{2{\rm p}}(t) \triangleq \dot{y}_{\rm f}(t) + \dot{u}_{\rm f}(t) - \bar{a}_1 e_{1{\rm p}}(t) - \dot{\hat{\psi}}_{\rm c}(t). \quad (31)$$

Using Theorem 2.1, we can show that the prediction errors (30)–(31) are ultimately bounded, and thus there exists $\varepsilon_2, t_2 \geqslant 0$ such that $\|y(t) - \hat{y}(t)\| \leqslant \varepsilon_2$, $t \geqslant t_2$.

B. Simulation Results and Tracking Performance

The dynamical model used to simulate the system's behavior relies on two different results. In particular, the robot's dynamics are modeled using the recursive algorithms in [27], which are based on Newton-Euler's equations. In addition, the hydrodynamic forces and interactions between robot and surrounding fluid are represented using the analytical hydrodynamic model for three dimensional self-propelled fish swimming introduced in [28]. A detailed presentation of the model is beyond the scope of the present paper.

The CPG algorithm used to generate motion is detailed within [3]. The parameters used are the following, a body undulation frequency $\nu = 1$ Hz, amplitude $A = 35^\circ$, phase difference between neighbors $\Delta\phi = 11\pi/40$ radians, amplitude convergence gain $a = 100$, and coupling gain $\omega = 10$. In addition, we used the following values for the predictor, $g = -1.15$, $\tau = 12.5$, $\zeta = \sqrt{2}$, $\omega_n = 4$, $a_1 = -10$, $a_2 = -0.1$.

Results of numerical simulations show that (26)–(27) allows to reconstruct a close estimate of the carrier signal $\psi_{\rm c}(t)$, $t \geqslant 0$, as seen in Figure 2. The output predictor (26)–(27) captures the essential input/output features of the actual system, and, as discussed in a previous section, use of this

predictor makes the design of a control law for the system straightforward. In particular, using (24), we select

$$\alpha(t) = \dddot{\hat{\psi}}_{\mathrm{c}}(t)/g + (\ddot{\psi}_{\mathrm{d}}(t) - k_1 e_{\mathrm{t}}(t) - k_2 \dot{e}_{\mathrm{t}}(t) - \varphi(t))/\tau g, \quad t \geqslant 0, \quad (32)$$

where $\psi_{\mathrm{d}}(t)$ defines a chosen desired direction of motion whose trajectory is commensurate with the actuation system's capacities, $e_{\mathrm{t}}(t) = \hat{\psi}_{\mathrm{c}}(t) - \psi_{\mathrm{d}}(t)$, and $k_1,\ k_2 \in \mathbb{R}$ are chosen so that (25) is Hurwitz. The control law (32) guarantees exponential convergence of the tracking error $e_{\mathrm{t}}(t)$ to the origin ([26]), and hence of $\hat{\psi}_{\mathrm{c}}(t)$ to $\psi_{\mathrm{d}}(t)$.

Results of numerical simulation illustrate the algorithm's tracking performance. In particular, Figure 3 shows tracking of a given $\psi_{\mathrm{d}}(t)$ (solid, low-pass filtered series of steps at $\pm 30^\circ$, alternating every 5s). The control gains used are $k_1 = 4.5$, $k_2 = 1$. Combining this tracking result with the output prediction result previously achieved allows to guarantee convergence of the actual, unmeasured carrier signal $\psi_{\mathrm{c}}(t)$ to an ϵ-neighborhood of the desired direction $\psi_{\mathrm{d}}(t)$ and positive invariance of this neighborhood ([25]); that is, there exists $t_3 \geqslant 0$ so that $\psi_{\mathrm{c}}(t) \in \mathcal{D}$, $t \geqslant t_3$, where $\mathcal{D} \triangleq \{x(t) \in \mathbb{R} : \|\psi_{\mathrm{d}}(t) - x(t)\| \leqslant \varepsilon_1 + \varepsilon_2, t \geqslant 0\}$.

IV. Conclusion

The presented control framework solves the output feedback motion control problem for a class of uncertain unmanned vehicles. The approach makes use of an output predictor to address issues stemming from system uncertainty and partial state measurements. The algorithm was applied to control movement of a swimming robot, and results of numerical simulations illustrate the achieved control performance. Implementation on the actual robot is currently underway, and experimental results are expected to confirm the proposed control law's efficacy. In addition, the proposed control algorithm will in the future be extended to address the position trajectory tracking problem. In particular, and assuming that position of the system may be measured (in a laboratory using cameras, on the field using an acoustic positioning system), we will exploit the fact that we are able to accurately control the speed direction to treat the system as a unicycle, thus facilitating control design.

References

[1] D. Lentink and A. A. Biewener, "Nature-inspired flight: Beyond the leap," *Bioinspiration and Biomimetics*, vol. 5, no. 4, pp. 247–269, 2010.

[2] J. M. Anderson, M. S. Triantafyllou, and P. A. Kerrebrock, "Concept design of a flexible-hull unmanned undersea vehicle," in *Proc. 1997 IEEE Int. Offshore and Polar Eng. Conf.*, (Honolulu, HI), pp. 82–88, 1997.

[3] A. Crespi and A. Ijspeert, "Amphibot II: An amphibious snake robot that crawls and swims using a central pattern generator," in *Proc. 9th Int. Conf. on Climbing and Walking Robots*, (Brussels, Belgium), pp. 19–27, 2006.

[4] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, "The intelligent ASIMO: System overview and integration," in *Proc. 2002 IEEE Int. Conf. on Intelligent Robots and Systems*, (Lausanne, Switzerland), pp. 2478–2483, 2002.

[5] M. Raibert, K. Blankespoor, G. Nelson, and R. Playter, "Bigdog, the rough-terrain quadruped robot," in *Proc. IFAC 17th World Congress*, (Seoul, Korea), pp. 10822–10825, 2008.

[6] S. Rutishauser, A. Spröwitz, L. Righetti, and A. J. Ijspeert, "Passive compliant quadruped robot using central pattern generators for locomotion control," in *Proc. 2nd IEEE/RAS-EMBS Int. Conf. on Biomedical Robotics and Biomechatronics*, (Scottsdale, AZ), pp. 710–715, 2008.

[7] J. S. Bay and H. Hemami, "Modeling of a neural pattern generator with coupled nonlinear oscillators," *IEEE Trans. on Biomedical Engineering*, vol. BME-34, no. 4, pp. 297–306, 1987.

[8] A. J. Ijspeert and J. Kodjabachian, "Evolution and development of a central pattern generator for the swimming of a lamprey," *Artificial Life*, vol. 5, no. 3, pp. 247–269, 1999.

[9] K. Seo and J.-J. E. Slotine, "Models for global synchronization in CPG-based locomotion," in *Proc. 2007 IEEE Int. Conf. on Robotics and Automation*, (Roma, Italy), pp. 281–286, 2007.

[10] A. J. Ijspeert, "Central pattern generators for locomotion control in animals and robots: A review," *Neural Networks*, vol. 21, pp. 642–653, 2008.

[11] M. Wang, L. Sun, P. Yuan, and Q. Meng, "Periodicity locomotion control based on central pattern generator," in *Proc. 6th IEEE World Congress on Intelligent Control and Automation*, (Dalian, China), pp. 3144–3148, 2006.

[12] L. Righetti and A. J. Ijspeert, "Pattern generators with sensory feedback for the control of quadruped locomotion," in *Proc. 2008 IEEE Int. Conf. on Robotics and Automation*, (Pasadena, CA), pp. 819–824, 2008.

[13] R. Ding, J. Yu, Q. Yang, M. Tan, and J. Zhang, "CPG-based dynamics modeling and simulation for a biomimetic amphibious robot," in *Proc. 2009 IEEE Int. Conf. on Robotics and Biomimetics*, (Guilin, China), pp. 1658–1662, 2009.

[14] K. Seo, S.-J. Chung, and J.-J. E. Slotine, "CPG-based control of a turtle-like underwater vehicle," *Autonomous Robots*, vol. 28, pp. 247–269, 2010.

[15] G. Taga, "A model of the neuro-musculo-skeletal system for anticipatory adjustment of human locomotion during obstacle avoidance," *Biological Cybernetics*, vol. 78, no. 1, pp. 9–17, 1998.

[16] J. Morimoto, G. Endo, J. Nakanishi, S.-H. Hyon, G. Cheng, D. Bentivegna, and C. G. Atkeson, "Modulation of simple sinusoidal patterns by a coupled oscillator model for biped walking," in *Proc. 2006 IEEE Int. Conf. on Robotics and Automation*, (Orlando, FL), pp. 1579–1584, 2006.

[17] S. Sastry, *Nonlinear Systems, Analysis, Stability and Control*. Springer-Verlag New York, NY, 1999.

[18] K. S. Narendra and A. M. Annaswamy, *Stable Adaptive Systems*. Prentice Hall, 1989.

[19] M. Krstić, I. Kanellakopoullos, and P. Kokotović, *Nonlinear and Adaptive Control Design*. New York, NY: John Wiley and Sons, Inc, 1995.

[20] P. A. Ioannou and J. Sun, *Robust Adaptive Control*. Prentice-Hall, Upper Saddle River, NJ, 1996.

[21] F. L. Lewis, S. Jagannathan, and A. Yesilderik, *Neural Network Control of Robot Manipulators and Nonlinear Systems*. Boca Raton, FL: Taylor and Francis, 1999.

[22] Y. Morel and A. Leonessa, "Observer-based output feedback control of nonlinear systems non-affine in the unmeasured states," in *Proc. 2009 ASME Dynamic Systems and Control Conference*, (Hollywood, CA), 2009.

[23] Y. Morel and A. Leonessa, "Nonlinear predictor-based output feedback control for a class of uncertain nonlinear systems," in *Proc. 2010 ASME Dynamic Systems and Control Conference*, (Boston, MA), 2010.

[24] D. G. Luenberger, "Observer for multivariable systems," *IEEE Trans. on Automatic Control*, vol. AC-11, no. 2, pp. 190–197, 1966.

[25] N. P. Bhatia and G. P. Szegö, *Stability Theory of Dynamical Systems*. Springer-Verlag, 1970.

[26] H. K. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ: Prentice Hall, third ed., 2002.

[27] W. Khalil, G. Gallot, and F. Boyer, "Dynamic modeling and simulation of 3-D serial eel-like robot," *IEEE Trans. Systems, Man and Cybernetics - Part C: Applications and reviews*, vol. 37, pp. 1259–1268, 2007.

[28] F. Boyer, M. Porez, and A. Leroyer, "Poincaré-Cosserat equations for the Lighthill three-dimensional large amplitude elongated body theory: Application to robotics," *Journal of Nonlinear Science*, vol. 20, pp. 47–79, 2010.