Alternative Task Definitions for Path Tracking Control of Underactuated Robots

A. Zelei[†], **L. Bencsik**[*], **G. Stépán**[*]

[†] MTA-BME Research Group on Dynamics of Machines and Vehicles
Muegyetem rkp 3., 1111, Budapest, Hungary
{zelei@mm.bme.hu}

[*] Department of Applied Mechanics
Budapest University of Technology and Economics
Muegyetem rkp 3., 1111, Budapest, Hungary
{bencsik@mm.bme.hu, stepan@mm.bme.hu}

Abstract

There exist already several realizations of indoor/domestic robotic applications. Besides the floor based mobile robots, different ceiling based, crane-like robots appeared in the past years. A pendulum-like under-actuated service robot platform called ACROBOTER [4] is shown in Fig 1. The robot is designed to perform pick and place tasks as well as carry other service robots with lower mobility. The ACROBOTER platform extends the workspace of these robots to the whole cubic volume of the indoor environment by utilizing the almost obstacle free ceiling for planar movements. A similar concept, the Winch-Bot is presented in [2] and its structure is shown in Fig 2. A cable winch is the only actuator on the robot, which results its simplicity. Similarly to the ACROBOTER concept, an end-effector can be moved swiftly in a large workspace. Since the cable length can only be actuated, the swinging motion of the the end-effector is induced by parametric excitation when the cable length is varied periodically. Both robots are underactuated, since they have fewer control inputs (l) than degrees of freedom (n).

This work focuses on the *computed torque control* of underactuated systems like the above mentioned ones. For underactuated systems, there are restrictions when the task of the robot is defined. Different approaches can be distinguished: **A:** *one can specify the trajectory of some coordinates in time, but the number of these prescribed coordinates must be equal to the number of control inputs l* **B:** *the motion of all DoFs is specified but one can not prescribe the time histories along the given path.* In both cases we can obtain unique solution for the inverse dynamical calculation and the control inputs.

Concept **B** is used in [2] and the task is given by parametric functions for the coordinates as (1) shows. The path is given by a parametric function $h(p)$ show in Fig. 2. With a more general formalism we can define the endpoint coordinates of the robot by parametric functions $x(p)$ and $y(p)$, as (2) shows. In both (1) and (2) the generalized coordinates are specified as function of parameter $p(t)$, but the time history of p cannot be prescribed. From (1) or (2) $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ can be expressed as a function of parameter p, $\dot{p}$ and $\ddot{p}$. With this, the passive part of the equation of motion can be solved for p as it is detailed in [2].

$$\begin{bmatrix} r \\ \vartheta \end{bmatrix} = \begin{bmatrix} \sqrt{p^2+h(p)^2} \\ \tan^{-1}\frac{p}{h(p)} \end{bmatrix} \quad (1) \qquad \begin{bmatrix} r \\ \vartheta \end{bmatrix} = \begin{bmatrix} \sqrt{x(p)^2+y(p)^2} \\ -\tan^{-1}\frac{x(p)}{y(p)} \end{bmatrix} \quad (2)$$

The same task also can be defined by the so-called servo-constraint [1, 3] using concept **A**. The most obvious usage of servo-constraint concept could be to give $r-r^d=0$ or $\vartheta-\vartheta^d=0$ constraint, where r^d and ϑ^d are appropriate functions of time. Here, instead, parameter p is exterminated from (2) and we obtain a servo-constraint equation for the generalized coordinates of the robot showed in Fig 2:

$$\left[\ r^2-a^2(1+\tan^2\vartheta)\ \right] = \left[\ 0\ \right]. \tag{3}$$

In equation (3) functions $x(p)=p$ and $y(p)=-a$ were defined, which stands for a horizontal path. Note that (2) cannot be reformulated in the form of servo-constraint in closed form for any $x(p)$ and $y(p)$. This is a disadvantage of using servo-constraints.

In this work we compare the the above written concepts **A** and **B** of task definition in general. For this, let us consider a controlled system described by equation of motion (4) with mass matrix $\mathbf{M}(\mathbf{q})\in\mathbb{R}^{n\times n}$, vector of external forces $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\in\mathbb{R}^n$, control input $\mathbf{u}\in\mathbb{R}^l$ and input matrix $\mathbf{H}(\mathbf{q})\in\mathbb{R}^{n\times l}$:

$$\mathbf{M}\ddot{\mathbf{q}}+\mathbf{C}=\mathbf{H}\mathbf{u}. \tag{4}$$

The task specification by the servo-constraint equation (5) used in approach **A** and by parametrized functions (6) [2] used in concept **B** have the following general form:

$$\boldsymbol{\sigma}(\mathbf{q},t)=\mathbf{0}. \quad (5) \qquad \boldsymbol{\psi}(\mathbf{q},\mathbf{p},t), \quad (6)$$

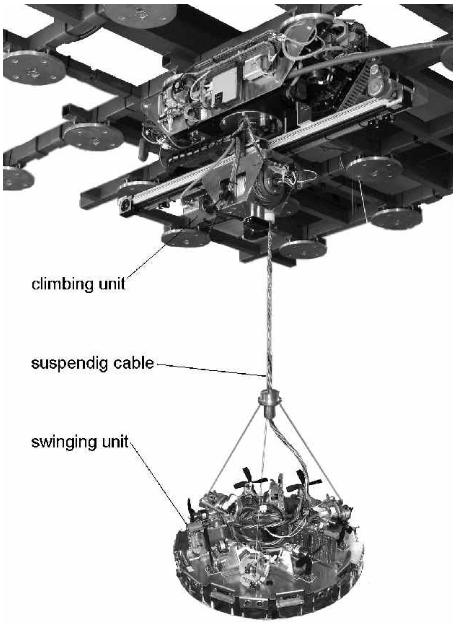

Figure 1: Acroboter system

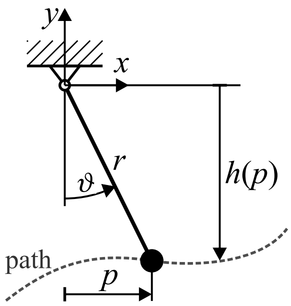

Figure 2: path

We detail an **A** type, servo-constraint based control approach detailed in [3] and used in case of Acroboter [4].

The desired task can only be fulfilled if the dimension of the servo-constraint $\boldsymbol{\sigma} \in \mathbb{R}^l$ equals to the number of control inputs l. As in the method of Lagrange multipliers, the servo-constraint equation $\boldsymbol{\sigma} = \mathbf{0}$ can be written on the level of accelerations as follows:

$$\boldsymbol{\sigma}_{\mathbf{q}}\ddot{\mathbf{q}} + \dot{\boldsymbol{\sigma}}_{\mathbf{q}}\dot{\mathbf{q}} + \dot{\boldsymbol{\sigma}}_t = \mathbf{0}. \tag{7}$$

With (7), equation (8) can be constructed, and the control input $\mathbf{u}$ can be calculated. The acceleration level servo-constraint equation is stabilized by $\mathbf{K}_P$ and $\mathbf{K}_D$ gains, similarly like in Baumgarte method:

$$\begin{bmatrix} \mathbf{M} & -\mathbf{H} \\ \boldsymbol{\sigma}_{\mathbf{q}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} -\mathbf{C} \\ -\dot{\boldsymbol{\sigma}}_{\mathbf{q}}\dot{\mathbf{q}} - \dot{\boldsymbol{\sigma}}_t - \mathbf{K}_D(\boldsymbol{\sigma}_{\mathbf{q}}\dot{\mathbf{q}} + \boldsymbol{\sigma}_t) - \mathbf{K}_P\boldsymbol{\sigma} \end{bmatrix}. \tag{8}$$

If we use parametric functions for the specification of the end-effector path (case **B**), like in [2], the task is defined by $\boldsymbol{\psi}(\mathbf{q},\mathbf{p}) \in \mathbb{R}^n$. The parameters are collected in vector $\mathbf{p} \in \mathbb{R}^{n-l}$. Like in the method of Lagrange multipliers, from the second time derivative of $\boldsymbol{\psi}$ the acceleration $\ddot{\mathbf{p}}$ and $\ddot{\mathbf{q}}$ can be expressed:

$$\boldsymbol{\psi}_{\mathbf{q}}\ddot{\mathbf{q}} + \boldsymbol{\psi}_{\mathbf{p}}\ddot{\mathbf{p}} + \dot{\boldsymbol{\psi}}_{\mathbf{q}}\dot{\mathbf{q}} + \dot{\boldsymbol{\psi}}_{\mathbf{p}}\dot{\mathbf{p}} + \dot{\boldsymbol{\psi}}_t = \mathbf{0}, \tag{9}$$

and the following form can be constructed again:

$$\begin{bmatrix} \mathbf{M} & \mathbf{0} & -\mathbf{H} \\ \boldsymbol{\psi}_{\mathbf{q}} & \boldsymbol{\psi}_{\mathbf{p}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \ddot{\mathbf{p}} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} -\mathbf{C} \\ -\dot{\boldsymbol{\psi}}_{\mathbf{q}}\dot{\mathbf{q}} - \dot{\boldsymbol{\psi}}_{\mathbf{p}}\dot{\mathbf{p}} - \dot{\boldsymbol{\psi}}_t - \mathbf{K}_D(\boldsymbol{\psi}_{\mathbf{q}}\dot{\mathbf{q}} + \boldsymbol{\psi}_{\mathbf{p}}\dot{\mathbf{p}} + \boldsymbol{\psi}_t) - \mathbf{K}_P\boldsymbol{\psi} \end{bmatrix}. \tag{10}$$

After we obtain $\ddot{\mathbf{p}}$, the actual values of $\mathbf{p}$ and $\dot{\mathbf{p}}$ are calculated by numerical integration and $\mathbf{u}$ is determined. The method can be extended to be applicable for systems with geometric constraints, like in [3].

In case of domestic applications the execution time of the task is not the key problem, so control concept **B** may be appropriate instead of approach **A**. Nevertheless, we compare the two methods from other viewpoints as well, like path tracking accuracy, computation time demands and stability.

References

[1] Kirgetov, V. I.: The Motion of Controlled Mechanical Systems with Prescribed Constraints (Servoconstraints). J. Appl. Math. Mech., 21, 433-466, 1967.

[2] Cunningham, D.; Asada, H. H.: The Winch-Bot: A Cable-Suspended, Under-Actuated Robot Utilizing Parametric Self-Excitation. IEEE International Conference on Robotics and Automation, Kobe, Japan, May 12-17, 2009.

[3] Kovács, L. L. and Bencsik, L.: Stability case study of the ACROBOTER underactuated service robot Theoretical and Applied Mechanics Letters 2(043004), 2012.

[4] Stépán, G.; et al.: A Ceiling Based Crawling, Hoisting and Swinging Service Robot Platform In P. Beyond Gray Droids: Domestic Robot Design for the 21st Century Workshop at HCI 2009.