Pergamon

Control Eng. Practice, Vol. 4, No. 7 pp. 1009-1014, 1996

0967-0661/96 $15.00 + 0.00

PII:S0967-0661(96)00100-1

STATIC-STATE FEEDBACK LAWS FOR OUTPUT REGULATION OF NON-LINEAR SYSTEMS

R. Mahony*, I. Mareels*, G. Bastin and G. Campion****

**Department of Systems Engineering, R.S.I.S.E., A.N.U., ACT 0200, Australia (robert@alfred.anu.edu.au)*
***CESAME, Batiment EULER, 4-6, Avenue G. Lemaitre, B1348 Louvain la Neuve, Belgium*

(Received October 1995; in final form May 1996)

Abstract. In this paper static-state feedback laws are considered which address the problem of output stabilization of non-linear dynamic control systems without drift. It is shown that as long as a dynamic-state control law exists which input-output decouples a given system, it is possible to construct a *static*-state control which stabilizes the system's output. The design procedure sheds light on the inherent difficulties in designing internally stable control laws for non-linear systems. To demonstrate the theory, it is applied to the task of designing an output stabilizing feedback for a simple model of a mobile robot.

Key Words: Non-holonomic systems, output regulation, internal stability.

1. INTRODUCTION

When designing control laws for dynamic systems it is preferable to use the simplest, most direct design methodology available. In the case of smooth finite-dimensional non-linear dynamic control systems, arguably the most direct form of feedback law is "static-state feedback" where the control variable is computed as a time-invariant function of the state variable. If the system is smooth, then one would naturally like the controller also to be smooth. Unfortunately, when one considers the problem of designing a controller to asymptotically stabilize the state of a non-linear dynamic control system there is a fundamental limitation to the efficacy of such a class of controllers. In particular, Brockett (1983) showed that for many systems where the number of inputs is strictly less than the number of states, no smooth static feedback law exists that asymptotically stabilizes the state. To overcome this difficulty two approaches are commonly employed, firstly the use of time-varying control laws, see for example (Pomet,1992), and secondly the use of discontinuous static control laws, see for example (Canudas de Wit and Sordalen,1991; Kolmanovsky, *et al.*, 1994). A connection between these methods is presented in (Coron *et al.*, 1994). An important application for these control methodologies is the task of controlling mobile robots (d'Andréa-Novel, *et al.*, 1995).

This paper is concerned with non-linear dynamic control systems of the form

$$\begin{aligned} \frac{d}{dt}x &= f(x) + \sum_{i=1}^{m} g_i(x)u_i, \quad x(0) = x_0 \\ y &= h(x) \end{aligned} \qquad (1)$$

where $x \in \mathbf{R}^n$, the system is square $y, u \in \mathbf{R}^m$, and the functions $f, g_i : \mathbf{R}^n \to \mathbf{R}^n$ and $h : \mathbf{R}^n \to \mathbf{R}^m$ are analytic. The algorithm presented below may be applied to smooth systems; however, the restriction to analytic systems simplifies the presentation of the technical details. For typical models of mobile robots (d'Andréa-Novel, *et al.*, 1995) the drift term $f(x) \equiv 0$ is equivalent to zero. It is convenient in the following development, however, to derive the results for the general case. A related problem to the question of full-state stabilization, that of output stabilization, is investigated. For example, in the case of a mobile robot one may wish to "park" the robot at the origin but not care about the orientation of the robot's wheels. Thus, the output would be the Euclidean coordinates describing the position of the robot, while the orientation of the wheels, etc. would be additional states.

Since the systems of interest to this paper are square systems in which there are as many control inputs as outputs, the controller design is undertaken by utilising input-output decoupling algorithms available in the literature. In particular, the dynamic extension algorithm, outlined in Nijmeijer *et al.*, (1990, pp. 263-264), is used. In the case where the *input/output decoupling* matrix (or just *decoupling* matrix (Nijmeijer, *et al.*, 1990, Eq. (8.24))) of the system is full rank it is a trivial matter to assign linear stable dynamics to the output. Once a control law has been designed for the task of output stabilization, however, it is then necessary to consider the question of internal stability for the closed-loop system. A recent paper (Mahony, *et al.*, 1993) gave a result that characterises a set of initial conditions for which the internal stability of the closed-loop dynamics (given the trivial linearising control law) is guaranteed. This paper considers the case where the decoupling matrix is not full rank, and shows that it is possible to assign stable dynamics to outputs by utilising higher-order system dynamics using the dynamic extension algorithm. The specific task considered in the design procedure results in important modifications to the basic dynamic extension algorithm, and provides a way of studying the difficulties associated with designing internally stable control laws for non-linear systems. To provide an example, the methods outlined in the paper are applied to a simple dynamic model of a mobile robot.

2. STATIC OUTPUT-STABILIZING FEEDBACK LAWS FOR SQUARE SYSTEMS

This section presents an algorithm for computing a locally smooth static control law $u := u(x)$ which exponentially stabilizes the output of a system for which the decoupling matrix is rank degenerate. The algorithm is valid for those systems for which the dynamic extension algorithm (Nijmeijer, *et al.*, 1990, Section 8.2) can be used to design a *dynamic* feedback law to input-output decouple the system.

Consider the system Eq. (1), rewritten as follows,

$$\begin{aligned} \frac{d}{dt}x &= f(x) + g(x)u, \quad x(0) = x_0, \qquad (2) \\ y &= h(x). \end{aligned}$$

Here $g(x) = [g_1(x) \vdots \cdots \vdots g_m(x)]$ is the $\mathbf{R}^{n\times m}$ matrix made up of the input vectors. Let $\rho_1^1, \ldots, \rho_m^1$ be the characteristic numbers of the outputs $y = (y_1, \ldots, y_m)$ (Nijmeijer, *et al.*, 1990, p. 246). Thus one can write

$$\begin{pmatrix} y_1^{(\rho_1^1+1)} \\ \vdots \\ y_m^{(\rho_m^1+1)} \end{pmatrix} = E^1(x) + F^1(x)u$$

where $E^1(x) = (L_f^{\rho_1^1+1}h(x), \ldots, L_f^{\rho_m^1+1}h(x))^T$ and $F^1(x)$ is the decoupling matrix

$$F^1(x) = \begin{pmatrix} L_{g_1}L_f^{\rho_1^1}h_1(x) & \cdots & L_{g_m}L_f^{\rho_1^1}h_1(x) \\ \vdots & & \vdots \\ L_{g_1}L_f^{\rho_m^1}h_m(x) & \cdots & L_{g_m}L_f^{\rho_m^1}h_m(x) \end{pmatrix}$$

Since the system given by Eq. (1) is analytic, the rank of $F^1(x)$ is constant except on a set of measure zero in $\mathbf{R}^n$. Denote the generic rank of $F^1(x)$ (off the set of measure zero) by r_1. It is convenient to refer to the set of points at which the various rank conditions, required for the construction of the output stabilizing feedback law, do not hold as the set of *singular points* in $\mathbf{R}^n$. Thus, the set of singular points includes all points at which rank $F^1(x) \neq r_1$ as well as other points defined below. The set of singular points will always be of zero measure in $\mathbf{R}^n$.

Given a point $x \in \mathbf{R}^n$ which is not a singular point, reorder and relabel the output functions $h_1, \ldots, h_m$ to ensure that the first r_1 rows of $F^1(x)$ are linearly independent. Reorder the inputs u (and hence the columns of $F^1(x)$) to ensure that the first r_1 columns of $F^1(x)$ are also linearly independent. Partition the output into two parts: $(h_1, \ldots, h_{r_1})$, the outputs associated with the first r_1 linearly independent rows of $F^1(x)$, and $(h_{r_1+1}, \ldots, h_m)$, the remaining outputs. Correspondingly, partition the vector $E^1(x)$ into its first r_1 entries, $E_1^1(x)$, and its remaining entries, $E_2^1(x)$, and the matrix $F^1(x)$ into four submatrices where the upper left submatrix, $F_{11}^1(x) \in \mathbf{R}^{r_1\times r_1}$ is square. Thus one has

$$\begin{pmatrix} y_1^{(\rho_1^1+1)} \\ \vdots \\ y_m^{(\rho_m^1+1)} \end{pmatrix} = \begin{pmatrix} E_1^1(x) \\ E_2^1(x) \end{pmatrix} + \begin{pmatrix} F_{11}^1(x) & F_{12}^1(x) \\ F_{21}^1(x) & F_{22}^1(x) \end{pmatrix} u.$$

By construction $F_{11}^1(x)$ is a full-rank invertible matrix and the last $m - r_1$ columns of $F^1(x)$ are linearly dependent on the first r_1 columns of $F^1(x)$. The linear dependence can be computed from the dependence of $F_{12}^1(x)$ on $F_{11}^1(x)$. Consider the input transformation

$$u(x, u^1, \overline{u}^1) := \begin{pmatrix} -(F_{11}^1)^{-1}E_1^1 \\ 0 \end{pmatrix} + \begin{pmatrix} (F_{11}^1)^{-1} & -(F_{11}^1)^{-1}F_{12}^1 \\ 0 & I_{m-r_1} \end{pmatrix} \begin{pmatrix} u^1 \\ \overline{u}^1 \end{pmatrix}, \qquad (3)$$

where the new inputs are denoted $(u^1, \overline{u}^1) \in \mathbf{R}^{r_1} \times \mathbf{R}^{m-r_1}$. Applying this feedback transformation to Eq. (1) yields the output dynamics

$$\begin{pmatrix} y_1^{(\rho_1^1+1)} \\ \vdots \\ y_m^{(\rho_m^1+1)} \end{pmatrix} = \begin{pmatrix} 0 \\ E_2^1 - F_{21}^1(F_{11}^1)^{-1}E_1^1 \end{pmatrix} + \begin{pmatrix} I_{r_1} & 0 \\ F_{21}^1(F_{11}^1)^{-1} & 0 \end{pmatrix} \begin{pmatrix} u^1 \\ \overline{u}^1 \end{pmatrix}.$$

Thus, after applying Eq. (3), the first r_1 outputs are input-output decoupled from the new inputs $u^1 \in \mathbf{R}^{r_1}$, and fully decoupled from the remaining inputs $\overline{u}^1$.

It is at this point that the present development differs significantly from the standard dynamic extension algorithm. Since the aim is simply to stabilize the output, it is possible to specify the first r_1 inputs u^1 to stabilize the first r_1 outputs explicitly. Define the inputs $u^1 = (u_1^1, \dots, u_{r_1}^1)$ as follows

$$\begin{aligned} u_i^1(x) &= -\sum_{j=0}^{\rho_i^1} C_j^{\rho_i^1+1} \frac{d^j h_i}{dt^j}(x), \quad i = 1, \dots, r_1, \\ &= -\sum_{j=0}^{\rho_i^1} C_j^{\rho_i^1+1} L_f^j h_i(x), \end{aligned} \tag{4}$$

where $C_b^a = \frac{a!}{b!(a-b)!}$. Choosing $u_i^1(x)$ as given above ensures that $y_i(t)$ satisfies the linear homogeneous differential equation

$$\left(\frac{d}{dt} + 1\right)^{(\rho_i^1+1)} y_i(t) = 0,$$

which has the solution

$$y_i(t) = (c_i^0 + c_i^1 t + \cdots + c_i^{\rho_i^1} t^{\rho_i^1}) e^{-t},$$

for constants $c_i^0 = y_i(0)$,

$$c_i^p = \frac{1}{p!} \sum_{j=0}^{p-1} C_j^p \frac{d^j y_i}{dt^j}(0), \quad \text{for } p = 1, \dots, \rho_i^1. \tag{5}$$

Thus, one may define a new (partly closed-loop) system with inputs $u' \in \mathbf{R}^{m-r_1}$ and outputs $y' \in \mathbf{R}^{m-r_1}$,

$$\begin{aligned} \dot{x} = f(x) + g(x) \Bigg[&\begin{pmatrix} -(F_{11}^1)^{-1} E_1^1 \\ 0 \end{pmatrix} \\ &+ \begin{pmatrix} (F_{11}^1)^{-1} & -(F_{11}^1)^{-1} F_{12}^1 \\ 0 & I_{m-r_1} \end{pmatrix} \begin{pmatrix} u^1(x) \\ u' \end{pmatrix} \Bigg], \end{aligned} \tag{6}$$

$$\begin{pmatrix} y_1 \\ \vdots \\ y_{r_1} \end{pmatrix} = \begin{pmatrix} \sum_{j=0}^{\rho_1^1} c_1^j t^j e^{-t} \\ \vdots \\ \sum_{j=0}^{\rho_{r_1}^1} c_{r_1}^j t^j e^{-t} \end{pmatrix},$$

$$\begin{aligned} y' &= \begin{pmatrix} y_{r_1+1}^{(\rho_{r_1+1}^1+1)} \\ \vdots \\ y_m^{(\rho_m^1+1)} \end{pmatrix}, \\ &= E_2^1(x) - F_{21}^1(x)(F_{11}^1(x))^{-1} E_1^1(x) \\ &\quad + F_{21}^1(x)(F_{11}^1(x))^{-1} u^1(x). \end{aligned} \tag{7}$$

The new inputs $u' \in \mathbf{R}^{m-r_1}$ are just the inputs $\overline{u}^1$ that were not assigned in the first step of the algorithm, while the new outputs y' are the highest-order time derivatives (used until now) of the unstabilized outputs that were obtained in the previous step. Observe that the new inputs u' do not enter directly into the output equation (7).

Remark 1 It is most convenient to continue subscripting both remaining inputs and outputs as though they were parts of the full input and output vectors. Thus, the new input vector is $u' = (u_{r_1+1}, \dots, u_m)$, while the output vector is analogous. Recall that these labels do not necessarily correspond to the original labels, due to the relabelling and reordering procedures. □

To obtain a full control law it is simply a question of applying the same step to the newly defined system, Eqs (6) and (7). The new characteristic numbers are denoted ρ_i^2, $i = r_1 + 1, \dots, m$ and the matrices $E^2(x)$ and $F^2(x)$ are analogous to those derived above. Certainly, the new system is analytic except on the set of singular points (in particular those where $F^1(x)$ is rank deficient) and the generic rank of $F^2(x)$ is denoted r_2. One includes all the points at which $F^2(x)$ is rank deficient in the set of singular points and considers only points $x \in \mathbf{R}^n$ which do not lie in this set. An analogous input transformation to that used above will input-output decouple the first r_2 outputs of y', namely $(y_{r_1+1}, \dots, y_{r_1+r_2})$ from the first r_2 inputs of u', denoted $u^2 = (u_{r_1+1}, \dots, u_{r_1+r_2})$, as well as fully decoupling $(y_{r_1+1}, \dots, y_{r_1+r_2})$ from the remaining inputs $\overline{u}^2 = (u_{r_1+r_2+1}, \dots, u_m)$.

It is convenient at this stage to give the general form for the feedback used. Assume that one has assigned $k-1$ sets of inputs $u^1(x)$, $u^2(x), \dots, u^{k-1}(x)$ (and consequently $k-1$ sets of outputs. Let

$$\Upsilon_i^k = \rho_i^1 + \rho_i^2 + \cdots + \rho_i^k,$$

for $i = (r_1 + \cdots + r_{k-1} + 1), \dots, (r_1 + \cdots + r_k)$. Specify the k'th set of inputs as follows

$$u_i^k(x) = -\sum_{j=0}^{(\Upsilon_i^k)} C_j^{\Upsilon_i^k+1} k^{(\Upsilon_i^k+1-j)} \frac{d^j}{dt^j} h_i(x), \tag{8}$$

for $i = (r_1 + \cdots + r_{k-1} + 1), \dots, (r_1 + \cdots + r_k)$. Choosing $u_i^k(x)$ as given above ensures that $y_i(t)$ satisfies the linear O.D.E.

$$\left(\frac{d}{dt} + k\right)^{(\Upsilon_i^k+1)} y(t) = 0, \tag{9}$$

which has solution

$$y_i(t) = e^{-kt} \sum_{j=0}^{\Upsilon_i^k} c_i^j t^j, \tag{10}$$

for $i = (r_1 + \cdots + r_{k-1} + 1), \dots, (r_1 + \cdots + r_k)$, and for constants c_i^j given by Eq. (5).

As a consequence of this choice the k'th set of outputs converge to zero strictly faster (exponentially with rate $-k$) than the previous sets of outputs (at best exponential with rate $-(k-1)$). This choice is made in the hope that the later outputs, most prone to numerical ill-conditioning (due to repeated singular input transformations), converge before any ill-conditioning in the algorithm occurs.

To fully determine the inputs, one continues to apply the above procedure until all the inputs have been determined. Since the system has been assumed to be full rank (Nijmeijer, *et al.*, 1990, Theorem 8.19), it follows that each output can eventually be controlled by an input. Denote the total number of steps taken to assign every output as K. Then the evolution of the output of the system is given by

$$\begin{pmatrix} y_1(t) \\ \vdots \\ y_{r_1}(t) \\ y_{r_1+1}(t) \\ \vdots \\ y_m(t) \end{pmatrix} = \begin{pmatrix} \sum_{j=0}^{\rho_1^1} c_1^j t^j e^{-t} \\ \vdots \\ \sum_{j=0}^{\rho_{r_1}^1} c_{r_1}^j t^j e^{-t} \\ \sum_{j=0}^{\Upsilon_{r_1+1}^2} c_{\Upsilon_{r_1+1}^1}^j t^j e^{-2t} \\ \vdots \\ \sum_{j=0}^{\Upsilon_m^K} c_m^j t^j e^{-Kt} \end{pmatrix}$$

One of the consequences of focusing on the design of a practical control law using the algorithm outlined in this section is the structure of the set of singular points,

$$\mathcal{S} = \{x \in \mathbf{R}^n \mid \det F_{11}^k(x) = 0, k \in \{1, 2, \ldots, K\}\},$$

in the state space. On this set of points the above algorithm fails, and one would expect that the output laws themselves would become unbounded in the vicinity of $\mathcal{S}$. This issue is not only relevant for the algorithm outlined above, but it is a fundamental limitation for any algorithm that relies on the decomposition of input-output dependence used by the dynamic extension algorithm. Observe, however, that in the method outlined above the new singularities introduced at each step, following from the equation $\det F_{11}^k(x) = 0$, are explicitly known before the control law is constructed. In this manner one can design the control to achieve the control objective, as well as to avoid the singularities which would cause the system to become unstable internally. The choice of control law given earlier appears to achieve this aim effectively in the case studies completed to date. A further observation is that the set of singular points is dependent on the choices made during the algorithm, of which outputs to stabilize at each step and which inputs to use. This dependence is shown nicely by the example in Section 3.

3. A MOBILE ROBOT WITH A SINGLE CENTRED ORIENTABLE WHEEL

In this section an example of a kinematic model of a mobile robot (cf. Fig. 1) is considered. The robot state is described by the coordinates (x, y, θ, β), where (x, y, θ) describe the position and attitude of the robot and the additional coordinate, β, describes the orientation of the front driving wheel. The output regulation task considered is to design a controller that asymptotically stabilizes (x, y, θ) to the zero state. That is, a control that parks the robot at the origin, pointing straight along the x-axis.

Consider the robot shown in Fig. 1. The robot is a type (2, 1) robot using the classification scheme developed in (Campion, *et al.*, 1995). That is, it has two degrees of steering freedom and a single degree of freedom of motion. The steering freedom is provided by steering control on both the centred orientable wheel of the robot as well as one of the off-centre wheels. When steering input is applied to an off-centre wheel it translates into rotational motion of the robot frame. The single degree of freedom of motion is due to the rolling non-slipping motion of the centred orientable wheel at the reference point of the robot frame. The remaining off-centre wheel in Fig. 1 simply provides stability, and is free to rotate to accommodate the motion of the robot determined by the orientation of the other two wheels. The control inputs can be thought of as the drive velocity on the centred wheel, u_1, the steering control for the centre wheel, u_3, and the rotational velocity of the robot frame induced by steering the off-centre wheel, u_2. A kinematic model for the robot is given by (Campion, *et al.*, 1995)

$$\begin{aligned} \dot{x} &= u_1 \cos(\theta + \beta), \\ \dot{y} &= u_1 \sin(\theta + \beta), \\ \dot{\theta} &= u_2, \\ \dot{\beta} &= u_3. \end{aligned} \tag{11}$$

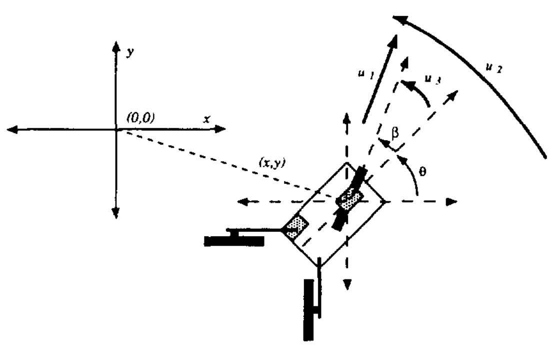

Fig: 1. A mobile robot of type (2,1) in the classification scheme presented in (Campion, *et al.*, 1995).

Remark 2 Observe that the steering input into the off-centred wheel will be related to the rotational velocity of the robot frame in a non-linear manner that depends on the drive velocity u_1. This paper is not concerned with details of this relationship. In addition, the model (Eq. (11)) is only a kinematic model of the robot. A dynamic model (involving forces and torque control rather than velocity control) contains singularities which are present since the rotational motion of the robot frame induced by steering the off-centre wheel is limited by the distance from the centre of the wheel to its pivot on the robot frame. □

Writing the kinematic model for the robot shown in Fig. 1 in the form of Eq. (2) one has

$$\frac{d}{dt}\begin{pmatrix} x \\ y \\ \theta \\ \beta \end{pmatrix} = \begin{pmatrix} \cos(\theta+\beta) & 0 & 0 \\ \sin(\theta+\beta) & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = g(x,y,\theta,\beta)u. \quad (12)$$

The output function considered is

$$h(x,y,\beta,\theta) = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \quad (13)$$

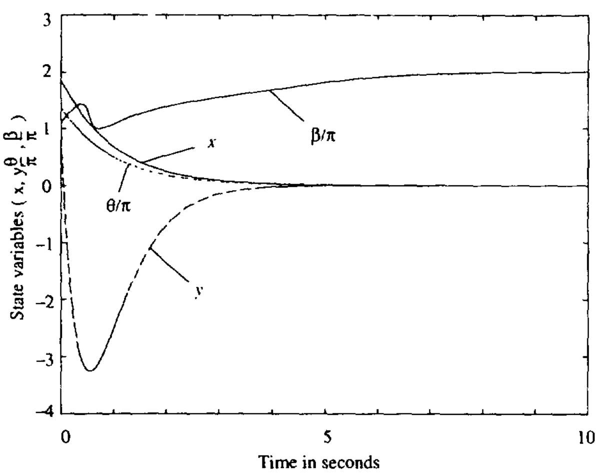

Fig: 2. Plot of the state $(x, y, \frac{\theta}{\pi}, \frac{\beta}{\pi})$ for a closed-loop simulation of the kinematic model of the mobile robot shown in Fig. 1, with static-state output stabilizing control law given by Eqs (14), (15) and (18).

Observe that the system, Eq's (12) and (13), cannot be input-output decoupled since

$$Dh(x,y,\theta,\beta)g(x,y,\theta,\beta) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}\begin{pmatrix} \cos(\theta+\beta) & 0 & 0 \\ \sin(\theta+\beta) & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, = \begin{pmatrix} \cos(\theta+\beta) & 0 & 0 \\ \sin(\theta+\beta) & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix},$$

is rank deficient. This matrix does have rank 2, however, and the algorithm outlined in Section 2 proceeds by using two control inputs u_1 and u_2 to regulate two output variables. The output variables that can be chosen are θ and either x or y. Consider the case where $h_1(x,y,\theta,\beta) = x$ and $h_3(x,y,\theta,\beta) = \theta$ are first regulated and then later $h_2(x,y,\theta,\beta) = y$ is regulated. The output dynamics are

$$\frac{d}{dt}h_1(x,y,\theta,\beta) = \dot{x} = u_1\cos(\theta+\beta),$$
$$\frac{d}{dt}h_3(x,y,\theta,\beta) = \dot{\theta} = u_2.$$

Following the algorithm presented in Section 2, the inputs u_1 and u_2 are chosen as

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = -\begin{pmatrix} \cos(\theta+\beta) & 0 \\ 0 & 1 \end{pmatrix}^{-1}\begin{pmatrix} x \\ \theta \end{pmatrix},$$

leading to the control laws

$$u_1 = \frac{-x}{\cos(\theta+\beta)}, \quad (14)$$
$$u_2 = -\theta. \quad (15)$$

Substituting these control laws into Eqs (12) and 13 leads to the new system

$$\frac{d}{dt}\begin{pmatrix} x \\ y \\ \theta \\ \beta \end{pmatrix} = \begin{pmatrix} -x \\ -x\tan(\theta+\beta) \\ -\theta \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}u_3 = f^{(1)}(x,y,\theta,\beta) + g^{(1)}u_3. \quad (16)$$

The new output function is

$$h'(x,y,\beta,\theta) = \dot{y} = Dh_2g(x,y,\beta,\theta) = -x\tan(\theta+\beta). \quad (17)$$

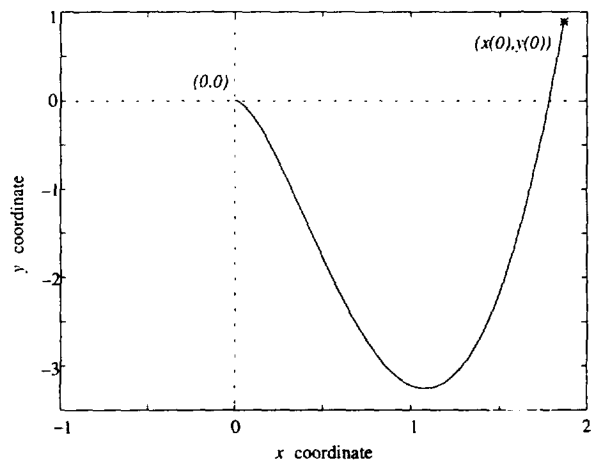

Fig: 3. Plot of the time evolution in Euclidean 2-space of a closed-loop simulation of the kinematic model of the mobile robot shown in Fig. 1, with static-state output stabilizing control law given by Eqs (14), (15) and (18).

The second iteration of the algorithm begins by computing the Lie-derivative, $L_{g^{(1)}}h'$, of the new output $h'(x,y,\beta,\theta)$ in direction $g^{(1)}(x,y,\beta,\theta)$. It is easily verified that

$$L_{g^{(1)}}h' = Dh'g^{(1)}(x,y,\beta,\theta), = -x(1+\tan^2(\theta+\beta)),$$

which is non-zero except when $x = 0$. Consequently, the time derivative of h' is

$$\frac{d}{dt}h'(x,y,\beta,\theta) = Dh'(f^{(1)} + g^{(1)}u_3) = x\tan(\theta+\beta) + x\theta(1+\tan^2(\theta+\beta)) - x(1+\tan^2(\theta+\beta))u_3.$$

Recall the general form (Eq. (9)) given for the desired linear stable output dynamics,

$$\ddot{y} = \frac{d}{dt}h'(x, y, \beta, \theta) = -4\dot{y} - 4y.$$

Equating these dynamics with those for h' while treating u_3 as an indeterminate allows one to solve for u_3 explicitly;

$$u_3 = \frac{4y + x\theta(1 + \tan^2(\theta + \beta) - 3x\tan(\theta + \beta)}{x(1 + \tan^2(\theta + \beta))}, \tag{18}$$

which is well defined except for states where $\cos(\theta + \beta) = 0$ or $x = 0$.

To demonstrate the performance of the closed-loop system, a simulation has been included (cf. Fig's 2-4). The initial condition for this example was

$$(x(0), y(0), \theta(0), \beta(0)) = (1.8620, 0.7931, 1.37930\pi, 1.1034\pi),$$

and the closed-loop system given by Eq. (11) along with Eqs (14), (15) and (18), was integrated over a period of ten seconds. Observe, in Fig. 2, that the convergence of x and θ/π is monotonic, while y is first subject to transient divergence from zero. However, it can be seen, on closer inspection of Fig. 2, that asymptotically y converges more quickly to zero than either x or θ. This can also be seen in Fig. 3, where the trajectory $(x(t), y(t))$ approaches the origin asymptotically along the x axis.

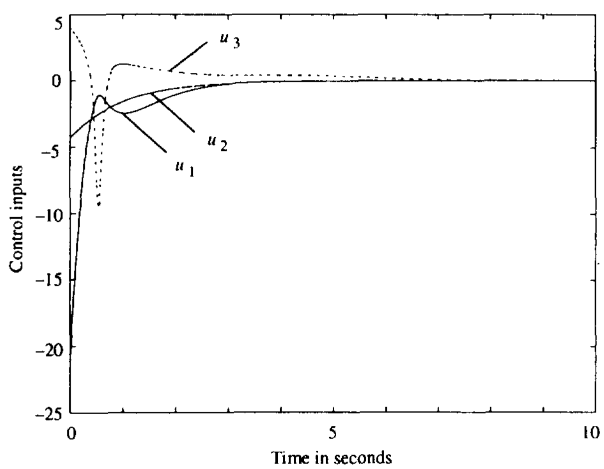

Fig: 4. Plot of the control inputs u_1, u_2 and u_3 generated by the static-state output stabilizing control law Eqs (14), (15) and (18).

The state β/π is not controlled to zero and has a final limiting value of $\beta(10) \approx 2\pi$. Looking at Fig. 4, there is a spike in the u_3 control action at around time $t = 0.7$. This corresponds to the time when the y trajectory has reached its most negative value. The spike in u_3 is connected with the change in relative rotation of the robot frame around the trajectory during the sharp turn (cf. the plot of β/π in Fig. 2).

Acknowledgements

The authors would like to acknowledge the funding of the Australian Commonwealth Government under the Cooperative Research Centres Program. The paper also presents research results of the Belgian Programme on Inter-University Poles of Attraction, initiated by the Belgian State, Prime Minister's Office for Science Technology and Culture. The scientific responsibility rests with its authors.

REFERENCES

Brockett, R. W. (1983), Asymptotic stability and feedback stabilisation, *in* R. Brockett, R. Millman and H. Sussman (eds), *Differential Geometric Control Theory*, Vol. 27 of *Progress in Mathematics*, Birkhauser, Boston, USA.

Campion, G., Bastin, G. and d'Andrea Novel, B. (1995), Structural properties and classification of kinematic and dynamic models of wheeled mobile robots, submitted for publication.

Canudas de Wit, C. and Sordalen, O. J. (1991), Exponential stabilization of mobile robots with nonholonomic constraints, *Proceedings of the 30th Conference on Decision and Control*, Brighton, England, pp. 692–697.

Coron, J.-M. and Rosier, L. (1994), A relation between continuous time-varying and discontinuous feedback stabilization, *Mathematics of Mathematical Systems, Estimation and Control* 4(1), 67–84.

d'Andréa-Novel, B., Bastin, G. and Campion, G. (1995), Control of nonholonomic wheeled mobile robots by state feedback linearization, International Journal of Robotic Systems.

Kolmanovsky, I. V., Reyhanoglu, M. and McClamroch, H. (1994), Discontinuous feedback stabilization of non-holonomic systems in power form, *Proceedings of the 33rd Conference on Decision and Control.*

Mahony, R. E., Mareels, I. M., Campion, G. and Bastin, G. (1993), Output regulation for systems linear in the input, *Conference on Mathematical Theory of Networks and Systems*, Regensburg, Germany.

Nijmeijer, H. and Van der Schaft, A. (1990), *Nonlinear Dynamical Control Systems*, Springer Verlag, New York, USA.

Pomet, J.-B. (1992), Explicit design of time-varying stabilizing control laws for a class of controllable systems without drift, *Systems and Control Letters* **18**, 147–158.