Proceedings of the 2006 IEEE International Conference on Robotics and Automation
Orlando, Florida - May 2006

Symbolic Control for Underactuated Differentially Flat Systems

Adriano Fagiolini, Luca Greco, Antonio Bicchi
Interdepartmental Research Center "E. Piaggio"
Faculty of Engineering, University of Pisa, Italy
Email: {a.fagiolini, bicchi}@ing.unipi.it,
luca.greco@dsea.unipi.it

Benedetto Piccoli, Alessia Marigo
C.N.R. - Istituto per le Applicazioni
del Calcolo "E. Picone", Rome, Italy
Email: {b.piccoli, marigo}@iac.rm.cnr.it

***Abstract*— In this paper we address the problem of generating input plans to steer complex dynamical systems in an obstacle-free environment. Plans considered admit a finite description length and are constructed by words on an alphabet of input symbols, which could be e.g. transmitted through a limited capacity channel to a remote system, where they can be decoded in suitable control actions.**

We show that, by suitable choice of the control encoding, finite plans can be efficiently built for a wide class of dynamical systems, computing arbitrarily close approximations of a desired equilibrium in polynomial time. Moreover, we illustrate by simulations the power of the proposed method, solving the steering problem for an example in the class of underactuated systems, which have attracted wide attention in the recent literature.

I. Introduction

The problem of steering complex physical plants requires the development of planning techniques capable of tackling both kinematic and dynamic constraints. Classical techniques usually try to face this problem by solving a kinematic motion planning problem and then looking for a trajectory and a control accounting for dynamic constraints. In recent years, several approaches inspired to the kinodynamic paradigm [1], [2], consisting in trying to solve these problems simultaneously, have been presented.

Moreover, dealing with physical systems and complex control frameworks, such as those based on hierachically abstracted levels of decision, usually involves additional issues related to limited communication and storage resources. Consider for instance the case where a robotic agent receives its motion plans from a remote high-level control center through a finite capacity communication channel. Or consider a scenario providing that plans are exchanged in a networked system of a large number of simple semi-autonomous agents, cooperating to achieve a common task, e.g. in a formation control operation or collaborative map building and object tracking.

In this vein, we address in this paper the problem of planning inputs to steer a controllable dynamical system of the type

$$\dot{x} = f(x, u),\ x \in X \subseteq \mathbb{R}^n, u \in U \subset \mathbb{R}^r \qquad (1)$$

between neighborhoods of given initial and final states. As a solution, we seek a *finite plan*, i.e. an input function which admits a finite description. We are interested in plans with short description length (measured in bits) and low computational complexity. Particular attention is given to plans among equilibrium states, regarded as nominal functional conditions.

Several important contributions have appeared addressing different instances of symbolic control problems, e.g. [3]–[5]. In particular, [6] shows that feedback can substantially reduce the specification complexity (i.e., the description length of the shortest admissible plan) to reach a certain goal state.

The main contribution of this paper is to show that, by suitable use of feedback, finite plans can be efficiently found for a wide class of systems. More precisely, we use a symbolic encoding ensuring that a *control language* is obtained whose action on the system has the desirable properties of additive groups, i.e. the actions of control words are invertible and commute. Furthermore, under the action of words in this language, the reachable set becomes a lattice. Finite–length plans to steer the system from any initial state to any final state within a given region can thus be computed in polynomial time. The contribution of this paper can be regarded as an extension of planning techniques in [7] (only applicable so far to driftless nonlinear systems in so-called "chained-form"), to a much wider class of systems, most notably systems with drift.

By virtue of feedback encoding, complex nonlinear systems — indeed, the same class of differentially flat systems [8] considered in [9] — can be transformed (at least locally) into a linear system. Planning for flat systems can then be achieved in a linear setting, hence projected back on the original systems by feedback decoding. This process is thoroughly illustrated in the paper by application to an interesting MIMO underactuated mechanical system: a simple model of a helicopter.

II. Symbolic control

Symbolic control is inherently related to the definition of elementary control events, or *quanta*, whose combination allows the specification of complex control actions. A finite or countable set $\mathcal{U}$ of control quanta can be *encoded* by associating its elements with symbols in a finite set $\Sigma = \{\sigma_1, \sigma_2, \ldots\}$. Furthermore, letters from the alphabet Σ can be employed to build words of arbitrary length. Let Σ^* be the set of such strings, including the empty one.

The analysis of the action of a generic Σ^* on the state space can be quite hard, and the structure of the reachable set under generic quantized controls can be very intricated (even for linear systems: see e.g. [7], [10], [11]). However, a suitable choice of the set $\mathcal{U}_\varepsilon$, namely a suitable control quantization, and of the control encoding can make symbolic control a powerful tool. This motivates the investigation of encodings achieving simple composition rules for the action of words in a sublanguage $\Omega \subset \Sigma^*$, so as to obtain that the global action of a command string is independent from the order of application of each control symbols in Ω. Under this hypothesis, a choice for Ω always exists such that the reachable set from any point in X under the concatenation of words in Ω can be described as a *lattice* which we assume henceforth. Hence, in suitable state and input coordinates, the system takes on the form

$$z^+ = z + \bar{H}\mu, \quad \bar{H} \in \mathbb{R}^{n\times n},\ \mu \in \mathbb{Z}^n. \tag{2}$$

Definition 1: A control system $\dot{x} = f(x,u)$ is *additively* (or *lattice*) *approachable* if, for every $\varepsilon > 0$, there exist a control quantization $\mathcal{U}_\varepsilon$ and an encoding $E^* : \Omega \mapsto \mathcal{U}_\varepsilon^*$, such that: i) actions of Ω commute and are invertible, and ii) for every $x_0, x_f \in X$, there exists x in the Ω–orbit of x_0 with $\|x - x_f\| < \varepsilon$.

Remark 1: The reachable set being a lattice under quantization does not imply additive approachability. For instance, consider the example used in [2] to illustrate kinodynamic planning methods [12]–[14]. This consists of a double integrator $\ddot{q} = u$ with piecewise constant encoding $\mathcal{U} = \{u_0 = 0, u_1 = 1, u_2 = -1\}$ on intervals of fixed length $T = 1$. The sampled system reads

$$\begin{bmatrix} q \\ \dot{q} \end{bmatrix}^+ = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + \begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix} u, \tag{3}$$

hence

$$\begin{aligned} q(N) &= q(0) + N\dot{q}(0) + \textstyle\sum_{i=1}^{N} \frac{2(N-i)+1}{2} u(i) \\ \dot{q}(N) &= \dot{q}(0) + \textstyle\sum_{i=1}^{N} u(i). \end{aligned}$$

The reachable set from $q(0) = \dot{q}(0) = 0$ is

$$R(\mathcal{U}, 0) = \left\{ \begin{bmatrix} q \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{bmatrix} \lambda,\ \lambda \in \mathbb{Z}^2 \right\}.$$

The quantization thus induces a lattice structure on the reachable set. The lattice mesh can be reduced to any desired ε resolution by scaling U or T. However, the actions of control quanta do not commute. Indeed, being $\phi^*(u^*)$ the solution of system (3) under the control sequence u^*, we have $\phi^*(u_1u_2) \neq \phi^*(u_2u_1)$ (for instance, $\phi^*(u_1u_2)(0,0) = (1,0)$, while $\phi^*(u_2u_1)(0,0) = (-1,0)$.

Motivations for planning on lattices are mainly due to the following theorem [15]:

Theorem 1: For an additively approachable system, a specification of control inputs steering the system from any initial state to any final state, can be given in polynomial time.

III. Feedback Encoding

A few examples of possible control encoding schemes of increasing generality were considered in [15]. Some of them are pictorially described in fig. 1. The most attractive one is the *feedback encoding* which consists in associating to each symbol a control input u that depends on the symbol itself, on the current state of the system, and on its structure. Feedback encoding was exploited for planning the trajectory of a car with n trailers ([16], [17]), whose kinematic model is locally feedback equivalent to chained form [18] and hence, additively approachable, thus demonstrating that finite plans to steer any nonlinear driftless system can be computed in polynomial time by theorem 1.

We now turn our attention to the much broader class of systems with drift, i.e. systems which possess an autonomous dynamics independent of applied inputs. More precisely, consider again system (1)

$$\dot{x} = f(x,u), \quad x \in X \subseteq \mathbb{R}^n,\ u \in U \subset \mathbb{R}^r$$

and the associate equilibrium equation $f(x,u) = 0$. Let the equilibrium set be $\mathcal{E} = \{x \in X | \exists u \in U, f(x,u) = 0\}$. We say that system (1) has drift if $\mathcal{E}$ has lower dimension than X.

Among systems with drift, linear systems are the simplest, yet their analysis encompasses the key features and difficulties of planning. Indeed, our strategy to attack the general case consists of reducing to planning for linear systems via feedback encoding. To achieve this, we introduce a further generalized encoder, i.e. the *nested feedback encoding* described in fig. 1-c. In this case, an inner continuous (possibly dynamic) feedback loop and an outer discrete-time loop – both embedded on the remote system – are used to achieve richer encoding of transmitted symbols. Additive approachability for linear systems, by discrete-time feedback encoding (see fig. 1-b), is proved in theorem 6 below. By using the nested feedback encoding, all feedback linearizable systems are hence additively approachable. Therefore, by resorting to dynamic feedback encoding ([8], [19], [20]), we can state the following theorem:

Theorem 2: Every differentially flat system is locally additively approachable.

IV. Linear Systems

In this section we consider linear systems of type

$$\dot{x} = Fx + Gu \tag{4}$$

with $x \in \mathbb{R}^n$, $u \in U = \mathbb{R}^r$ and rank $G = r$. We start by some preliminary results characterizing the equilibrium set $\mathcal{E}$.

A. Preliminaries

Let us recall from [15] the following lemmas:

Lemma 3: For a controllable linear system (4), dim $\mathcal{E} = r$.

Application to (4) of piecewise constant encoding of symbolic inputs (scheme a in fig.1) with durations $T_i = T$, $\forall i$, generates the discrete-time linear system

$$x^+ = Ax + Bu, \tag{5}$$

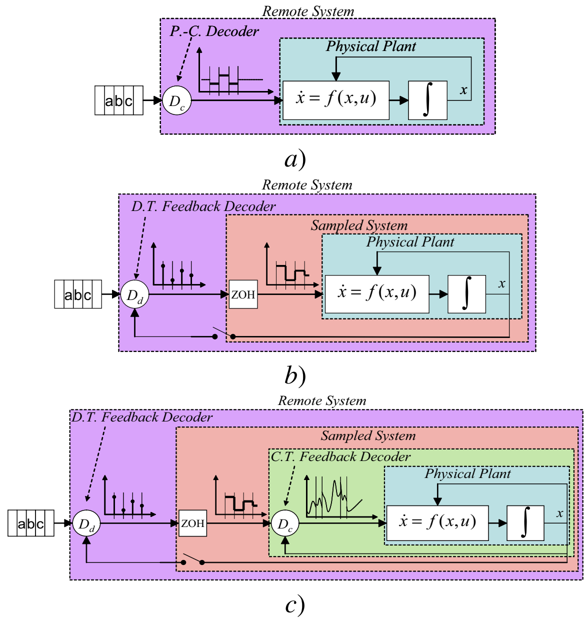


Fig. 1. Three examples of symbolic encoding of control. Symbols transmitted through the finite-capacity channel are represented by letters in the leftmost blocks. From the top: a) piecewise constant encoding; b) discrete-time feedback encoding; c) nested discrete-time continuous-time feedback encoding.

with

$$A = e^{FT}, \ B = \left(\int_0^T e^{(T-s)F} ds \right) G.$$

Lemma 4: The equilibrium manifold of a controllable linear continuous-time system is invariant under discrete-time feedback encoding, for almost all sampling times T.

A crucial observation concerning systems with drift is contained in the following lemma.

Lemma 5: For a linear system (4), it is impossible to steer the state among points in $\mathcal{E}$ while remaining in $\mathcal{E}$, except for special cases.

As we want $\mathcal{E}$ to be a lattice, the previous observations drive towards policies for *periodic* steering of systems among equilibria. In particular, we are interested in seeking symbolic input encodings that periodically achieve additive approachability on $\mathcal{E}$. Within this perspective, we recall the following theorem also from [15]:

Theorem 6: For a controllable linear discrete-time system $x^+ = Ax + Bu$, there exist an integer $\ell > 1$ and a linear feedback encoding

$$\begin{aligned} E: \quad & \Sigma \to \mathcal{U}, \\ & \sigma_i \mapsto Kx + w_i \end{aligned}$$

with constant $K \in \mathbb{R}^{n \times n}$ and $w_i \in \mathcal{W}$, $\mathcal{W} \subset \mathbb{R}^r$ a quantized control set, such that, for all subsequences of period ℓT extracted from $x(\cdot)$, the reachable set is a lattice of arbitrarily fine mesh. In other terms, for the undersampled system $z(k) = x(\tau + k\ell)$, $\tau, k \in \mathbb{N}$, it holds

$$z^+ = z + \bar{H}\mu, \ \bar{H} \in \mathbb{R}^{n \times n}, \ \mu \in \mathbb{Z}^n$$

and $\forall \varepsilon$ there exists a choice of a finite $\mathcal{W}$ such that $\|\bar{H}\| < \varepsilon$.

We recall preliminarily a result which can be derived directly from [7].

Lemma 7: The reachable set of the scalar discrete time linear system $\xi^+ = \xi + v$, $\xi \in \mathbb{R}$, $v \in \mathcal{W} \stackrel{def}{=} \gamma W$ with $\gamma > 0$ and $W = \{0, \pm w_1, \ldots, \pm w_m\}$, $w_i \in \mathbb{N}$ with at least two elements $w_i \ w_j$ coprime, is a lattice of mesh size γ.

Proof of Theorem 6: For the controllable pair (A, B), let S, V, and K_0 be matrices such that $(S^{-1}(A + BK_0)S$, $S^{-1}BV)$ is in Brunovsky form (see e.g. [21]). Denote with κ_i, $i = 1, \ldots, r$ the Kronecker control-invariant indices. In the new coordinates $\xi = S^{-1}x$ we have

$$\xi^+ = S^{-1}(A + BK_0)S\xi + S^{-1}BVv' = \widetilde{A}\xi + \widetilde{B}v'.$$

Let $v' = K_1\xi + v$, where:

- $v \in \mathcal{W} = \gamma_1 {}^1W \times \cdots \times \gamma_r {}^rW$, with ${}^kW = \{0, \pm{}^kw_1, \ldots, \pm{}^kw_{m_k}\}$, ${}^kw_j \in \mathbb{N}$ $k = 1, \ldots, r$, $j = 1, \ldots, m_k$, each ${}^kW$ including at least two coprime elements ${}^kw_i \ {}^kw_j$;
- $K_1 \in \mathbb{R}^{r \times n}$ such that its i–th row (denoted K_{1i}) contains all zeroes except for the element in the $(\kappa_{i-1} + 1)$–th column which is equal to one (recall that by definition $\kappa_0 = 0$).

Denoting with $(A_{\kappa_i}, B_{\kappa_i})$ the i-th sub-system in Brunovsky form, it can be easily observed that $(A_{\kappa_i} + B_{\kappa_i}K_{1i})^{\kappa_i} = I_{\kappa_i}$, the $\kappa_i \times \kappa_i$ identity matrix. Hence, if we let

$$\ell = \text{l.c.m. } \{\kappa_i : i = 1, ..., r\},$$

we get $\left[S^{-1}((A + BK_0)S + BVK_1)\right]^\ell = I_n$.

Let $\xi_i \in \mathbb{R}^{\kappa_i}$ denote the i–th component of the state vector relative to the pair $(A_{\kappa_i}, B_{\kappa_i})$. For any $\tau \in \mathbb{N}$ we have

$$\xi_i(\tau + \kappa_i) = \xi_i(\tau) + \begin{bmatrix} v_i(\tau) \\ \vdots \\ v_i(\tau + \kappa_i - 1) \end{bmatrix} \tag{6}$$

On the longer period of ℓT, we have

$$\begin{aligned} \xi_i(\tau + \ell) \quad &= \quad \xi_i(\tau) + \begin{bmatrix} \sum_{k=0}^{\frac{\ell}{\kappa_i}-1} v_i(\tau + k\kappa_i) \\ \vdots \\ \sum_{k=0}^{\frac{\ell}{\kappa_i}-1} v_i(\tau + \kappa_i - 1 + k\kappa_i) \end{bmatrix} \\ &\stackrel{def}{=} \quad \xi_i(\tau) + \bar{v}_i(\tau), \end{aligned}$$

hence

$$\xi(\tau + \ell) = \xi(\tau) + \begin{bmatrix} \bar{v}_1 \\ \vdots \\ \bar{v}_r \end{bmatrix} \stackrel{def}{=} \xi(\tau) + \bar{v}$$

or, in the initial coordinates,

$$x(\tau + \ell) = x(\tau) + S\bar{v}.$$

In conclusion, by the linear discrete–time feedback encoding

$$\begin{aligned} E: \quad & \Sigma \to \mathcal{U}, \\ & \sigma_i \mapsto (K_0 + VK_1S^{-1})x + Vv_i \end{aligned}$$

with $v_i \in \mathcal{W}$, for all ℓ-periodic subsequences $z(k) = x(\tau + k\ell)$, it holds

$$z^+ = z + S\Gamma\mu,\ \mu \in \mathbb{Z}^n$$

with

$$\Gamma = \text{diag}\left(\gamma_1 I_{\kappa_1}, \cdots, \gamma_r I_{\kappa_r}\right).$$

It is also clear that, for any ε, it is possible to choose Γ such that z can be driven in a finite number of steps (multiple of ℓ) to within an ε-neighborhood of any point in $\mathbb{R}^n$. ■

It is interesting to note that, for single-input systems, the encoding considered in theorem 6 is indeed optimal in terms of minimizing the periodicity by which the lattice is achievable. However, for multi-input systems, the period of $(\text{l.c.m.}_i\ \kappa_i)\,T$ used in theorem 6 can be reduced to a minimal of $(\max_i \kappa_i)\,T$. This can be achieved by the planning algorithm described below in section IV-B.

As it can be expected, the behavior of the system between such periodic samples is in general not specified, and may turn out to be unacceptable. Indeed, if a goal has to be reached, which is far from the origin, the intersample behavior may have a large-span erratic behavior. Nevertheless, the feedback encoding scheme allows to solve this problem while keeping the system's evolution arbitrarily close to the equilibrium manifold (see [15]).

Notice finally that, in Brunovsky coordinates, $\mathcal{E}$ has a particularly simple structure. Letting $\mathbf{1}_{\kappa_i} \in \mathbb{R}^{\kappa_i}$ denote a vector with all components equal to 1, we have that for each κ_i-dimensional subsystem, the equilibrium states are $\bar{\xi}_i = \alpha_i \mathbf{1}_{\kappa_i}$, $\alpha_i \in \mathbb{R}$, hence

$$\mathcal{E} = \left\{\bar{\xi} | \bar{\xi} = \text{diag}\left(\alpha_1 I_{\kappa_1}, \cdots, \alpha_r I_{\kappa_r}\right) \mathbf{1}_n\right\}$$

B. Planning algorithm

Based on the above results, we now provide explicitly an efficient method to steer from an arbitrary state $x \in \mathbb{R}^n$ to within an ε-neighborhood of a given goal state $x + \delta \in \mathbb{R}^n$ (x and δ not necessarily in $\mathcal{E}$).

1) Compute the desired displacement in Brunovsky coordinates $\Delta = S^{-1}\delta$, and let $\Delta_i \in \mathbb{R}^{\kappa_i},\ i = 1, \ldots, r$ denote the desired displacement for the i–th subsystem;
2) Compute the lattice mesh size in Brunovsky coordinates $\gamma_i = \frac{2\varepsilon}{\|\zeta_i\|}$, where

$$\left[\ \zeta_1 \quad \zeta_2 \cdots \quad \zeta_r\ \right] = S \begin{bmatrix} \mathbf{1}_{\kappa_1} & 0 & \cdots & 0 \\ 0 & \mathbf{1}_{\kappa_2} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \mathbf{1}_{\kappa_r} \end{bmatrix};$$

3) Find $\bar{\Delta}_i$, the nearest point to Δ_i on the lattice generated by $\gamma_i\,{}^iW$ and centered at $\xi_i = (S^{-1}x)_i$.
4) For each $i = 1, \ldots, r$, let the quantized control set be ${}^iW = \{0, \pm{}^iw_1, \ldots, \pm{}^iw_{m_i}\}$, ${}^iw_j \in \mathbb{N}$, and denote by ${}^iU$ the vector $[{}^iw_0\ {}^iw_1\ \cdots\ {}^iw_{m_i}]$, where ${}^iw_0 = 0$. Write

$$\bar{\Delta}_i = \gamma_i\,{}^iC\,{}^iU \tag{7}$$

where ${}^iC$ is a matrix in $\mathbb{Z}^{\kappa_i \times m_i+1}$ with components ${}^iC_{h,j+1} = {}^ic_{h,j},\ h = 1, \ldots, \kappa_i,\ j = 0, \ldots, m_i$. Each element ${}^ic_{h,j}$ of ${}^iC$ describes the number of times that the control ${}^iw_j$ has to be used to steer the h–th component of ξ_i.
5) Find integers ${}^ic_{h,j},\ h = 1, \ldots, \kappa_i,\ j = 1, \ldots, m_i$ solving the system of diophantine equations (7), and find the smallest integers ${}^ic_{h,0}$ such that, $\forall h$, $\sum_{j=0}^{m_i} |{}^ic_{h,j}| \stackrel{def}{=} N_i$. $N_i\kappa_i$ is thus a number of steps sufficient to steer the i–th subsystem;
6) (Optional) Among all solutions of (7), find the one which minimizes $\max_{h=1}^{\kappa_i} \sum_{j=1}^{m_i} |\,{}^ic_{h,j}| \stackrel{def}{=} \hat{N}_i$. Notice that $\hat{N}_i\kappa_i$ is the minimum length of a string of symbols in ${}^iW$ obtaining the goal. However, no polynomial-time algorithm is known for such optimization;
7) Let $N^\star_\kappa = \max_i N_i\kappa_i$, and $i^\star$ the corresponding index. Then, for all $i = 1, \ldots, r\ \ i \neq i^\star$, compute $\tilde{\Delta}_i = (\tilde{A}_i)^{-r_i}(\xi + \bar{\Delta}_i) - \xi$ with $r_i = N^\star_\kappa - N_i\kappa_i$. Repeat steps 4) and 5) with the new $\tilde{\Delta}_i$.

The explicit construction of a procedure to decode plan specifications ${}^iC$ into a string of control inputs ${}^iV$ for the i–th channel is finally described in Matlab-like code:

```
C=^iC;
^iV=[];
while (C ~= 0)
  for h=1:κ_i,
    j=1;
    while C(h,j) == 0, j=j+1; end
    ^iV = cat (^iV, sign(C(h,j))*^iw_j);
    C(h,j) = C(h,j)-sign(C(h,j));
  end
end
```

V. Simulations

A. Helicopter

In this section, we illustrate the power of the nested feedback encoding of fig. 1-c, by solving the steering problem for an example in the class of underactuated, differentially flat mechanical systems.

Consider a simplified model for the helicopter depicted in fig. 2. At first approximation, we can look at the helicopter as a rigid body, directly actuated by the thrust T of the main rotor and the torque τ of the tail rotor. Referring to fig. 2, denote with (x, y, z) the position of its center of mass and with (ϕ, θ, ψ) its orientation with respect to the x, y, and z axes. With this choice, the dynamic model takes on the form

$$\begin{cases} M\,\ddot{x} = T\,S_\theta\,, \\ M\,\ddot{y} = -T\,C_\theta\,S_\phi\,, \\ M\,\ddot{z} = T\,C_\theta\,C_\phi - M\,g\,, \\ J\,\ddot{\psi} = \tau\,C_\theta\,C_\phi\,, \end{cases}$$

where M is the mass of the helicopter, J is its inertia about the z axis, and g is the gravity acceleration. It is worth noting that we have no direct control over the angles ϕ (roll) and θ (pitch), but only through aileron and fore-aft cyclic control respectively (see [9]). Nonetheless, in the remainder of this

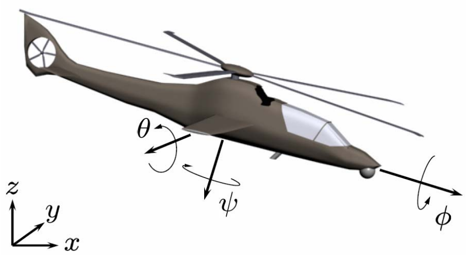


Fig. 2. Simplified model of a helicopter. The system is actuated by the thrust T of the main rotor and the torque τ of the tail rotor. Angles ϕ (roll) and θ (pitch) are indirectly established by means of cyclic control of the aileron and the fore-aft respectively, and therefore can represent two additional inputs.

section, we will consider them as real control inputs for the sake of simplicity.

The helicopter's model is dynamically feedback equivalent to a linear system, as we demonstrate in the following. Take as system outputs $y_1 = x$, $y_2 = y$, $y_3 = z$, $y_4 = \psi$. Differentiating twice these outputs, yields

$$\begin{cases} \ddot{y}_1 = \frac{T\, S_\theta}{M}\,, \\ \ddot{y}_2 = -\frac{T\, C_\theta\, S_\phi}{M}\,, \\ \ddot{y}_3 = \frac{T\, C_\theta\, C_\phi}{M} - g\,, \\ \ddot{y}_4 = \frac{\tau\, C_\theta\, C_\phi}{J}\,, \end{cases} \tag{8}$$

where the inputs are nonlinearly coupled. As a first step, we add one integrator on each input channel, and extend the system state by defining the auxiliary variables $\dot{T} = u_1, \dot{\theta} = u_2, \dot{\phi} = u_3$, and $\dot{\tau} = u_4$. Then, we differentiate once more the equations in (8). Thus, we obtain:

$$\begin{bmatrix} x^{(3)} \\ y^{(3)} \\ z^{(3)} \\ \psi^{(3)} \end{bmatrix} = \begin{bmatrix} \frac{S_\theta}{M} & \frac{T\, C_\theta}{M} & 0 & 0 \\ -\frac{C_\theta\, S_\phi}{M} & \frac{T\, S_\theta\, S_\phi}{M} & -\frac{T\, C_\theta\, C_\phi}{M} & 0 \\ \frac{C_\theta\, C_\phi}{M} & -\frac{T\, S_\theta C_\phi}{M} & -\frac{T\, C_\theta\, S_\phi}{M} & 0 \\ 0 & -\frac{\tau\, S_\theta\, C_\phi}{J} & -\frac{\tau\, C_\theta\, S_\phi}{J} & \frac{C_\theta\, C_\phi}{J} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} =$$

$$= D(T, \theta, \phi, \tau)\, u\,.$$

Under the hypothesis that $T \neq 0$, $\theta \neq \pm\frac{\pi}{2}$, $\phi \neq \pm\frac{\pi}{2}$, and $\theta \neq -\arcsin\left(\frac{C_\phi^2}{S_\phi}\right)$, matrix D is nonsingular, and the system can be exactly linearized by the choice $u = D^{-1}v'$. With respect to the new input v', the system's dynamics is indeed composed of four chains of integrators, i.e. $y_1^{(3)} = v_1'$, $y_2^{(3)} = v_2'$, $y_3^{(3)} = v_3'$, $y_4^{(3)} = v_4'$. Due to its linearity, the system can be steered by means of the method described in section (IV-B). First of all, observe that its equilibrium manifold is

$$\mathcal{E} = \{x \in \mathbb{R}^{12} | x = (\alpha_x, 0, 0, \alpha_y, 0, 0, \alpha_z, 0, 0, \alpha_\psi, 0, 0)\}\,.$$

Now, apply the discrete-time feeedback encoding of fig. 1-c with unit sampling time $t = 1s$, and compute matrices S, V, and K as in theorem 6. In the new coordinates, the equilirium manifold is given by

$$\mathcal{E} = \{(\beta_x \mathbf{1}_3, \beta_y \mathbf{1}_3, \beta_z \mathbf{1}_3, \beta_\psi \mathbf{1}_3)\}\,,$$

where $\alpha_x = \zeta_1\, \beta_x$, $\alpha_y = \zeta_2\, \beta_y$, $\alpha_z = \zeta_3\, \beta_z$, $\alpha_\psi = \zeta_4\, \beta_\psi$, and $\zeta_1 = \zeta_2 = \zeta_3 = \zeta_4 = \zeta = 1$.

When building a lattice for the system, it is reasonable to ask for a tolerance ε_1 on the x, y, and z coordinates which are measured in meters, and a different one, ε_2, on the ψ variable which is instead measured in radiants. Take e.g. as numerical values $\varepsilon_1 = 1$m and $\varepsilon_2 = 0.01$rad, hence it holds $\gamma_1 = \gamma_2 = \gamma_3 = 2$ and $\gamma_4 = 0.02$. Assume that all points in a hypercube of sizes 32m×32m×32m×0.32rad have to be reached, using control sets with $m = 4$. From [15], we know that the optimal choice is ${}^1W = {}^2W = {}^3W = {}^4W = \{0, \pm 3, \pm 6, \pm 7, \pm 8\}$ and every point can be reached within $N = 2$ steps. Similarly to the previous example, the actual execution of the plan takes $n = 3$ times N sampling instants, because of the maximum dimension of the blocks in the Brunovsky form.

As for the helicopter's motion, the following task is specified: lift up of a relative altitude of 6m from the actual position, rotate of an angle $\psi = \pi/4$ rad while hovering, travel horizontally of a relative displacement (20m, 20m), and finally go down to the initial altitude. Plans for steering the system according to the task are computed as in theorem 6. Such plans and the corresponding state evolution are reported in fig. 3. In fig. 4 the helicopter's trajectory and shots of the helicopter's position and attitude are finally shown.

VI. CONCLUSIONS

In this paper, we have described methods for steering complex dynamical systems by signals with finite-length descriptions. Systems tractable by symbolic control under encoding include all controllable linear systems, nilpotent driftless nonlinear systems and (dynamically) feedback-linearizable systems. Many other open problems remain open in order to fully exploit the potential of symbolic control. A limitation of our current approach is that we assume that a flat, linearizing output to be available, as well as state measurements. Connections to state observers in planning are unexplored at this stage. Future work will first investigate application of these methods to non–differentially flat systems, and will enquire in the additional possibilities which might be offered by using different feedback laws at different time instants. Finally, effective finite planning in presence of obstacles has not been considered yet and would need thorough investigation.

VII. ACKNOWLEDGMENTS

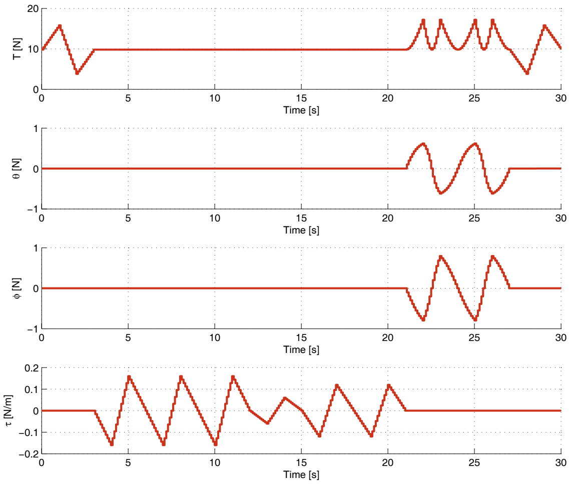
This work was partially supported by EC through the Network of Excellence contract IST-2004-511368 "HYCON - HYbrid CONtrol: Taming Heterogeneity and Complexity of Networked Embedded Systems", and Integrated Project contract IST-2004-004536 "RUNES - Reconfigurable Ubiquitous Networked Embedded Systems".

REFERENCES

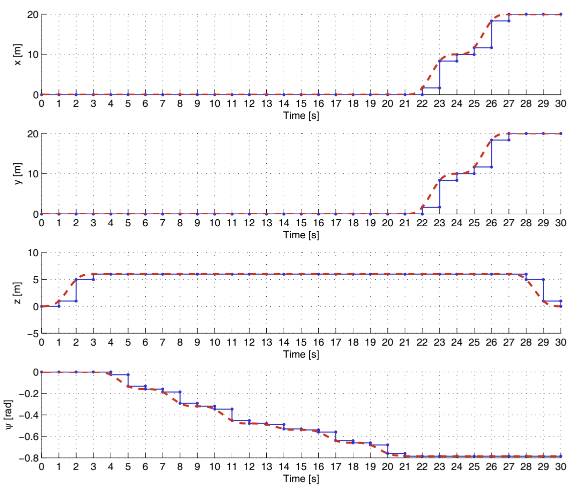
[1] S. M. LaValle and P. Konkimalla, "Algorithms for computing numerical optimal feedback motion strategies," (in) press.

[2] S. M. LaValle, *Planning Algorithms*. Cambridge University Press (also available at http://msl.cs.uiuc.edu/planning/), 2005, to be published.

[3] V. Manikonda, P. S. Krishnaprasad, and J. Hendler, "A motion description language and hybrid architecture for motion planning with nonholonomic robots," in *Proc. Int. Conf. on Robotics and Automation*, 1995.



a)



b)

Fig. 3. Planned motion for the helicopter (a) and corresponding symbolic inputs (b). The helicopter has to accomplish the following task: lift up of a relative altitude of 6m from the actual position, rotate of an angle $\psi = \pi/4$ rad while hovering, travel horizontally of a relative displacement $(20\text{m}, 20\text{m})$, and go down to the original altitude.

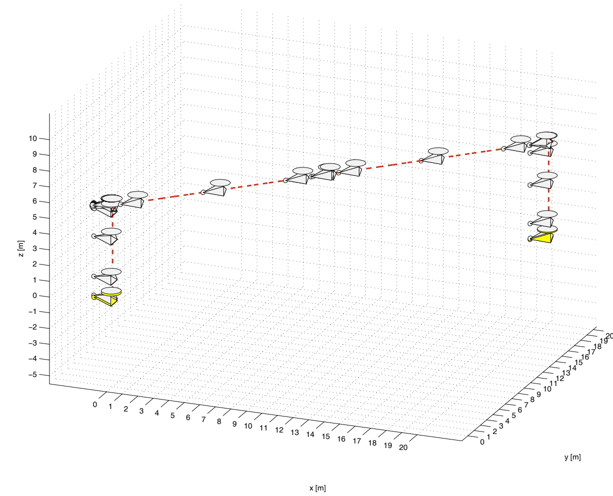


Fig. 4. Helicopter's trajectory along with some shots showing the helicopter's position and attitude.

[4] M. Egerstedt, "Motion description languages for multimodal control in robotics," in *Control Problems in Robotics*, ser. STAR, A. Bicchi, H. Christensen, and D. Prattichizzo, Eds. Springer-Verlag, 2003, no. 4, pp. 75–89.
[5] E. Frazzoli, M. A. Dahleh, and E. Feron, "Maneuver-based motion planning for nonlinear systems with symmetries," *IEEE Trans. on Robotics*, 2005.
[6] M. Egerstedt and R. W. Brockett, "Feedback can reduce the specification complexity of motor programs," *IEEE Trans. on Automatic Control*, vol. 48, no. 2, pp. 213–223, 2003.
[7] A. Bicchi, A. Marigo, and B. Piccoli, "On the reachability of quantized control systems," *IEEE Trans. on Automatic Control*, vol. 47, no. 4, pp. 546–563, April 2002.
[8] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, "Flatness and defect of nonlinear systems: Introductory theory and examples," *Int. J. of Control*, vol. 61, pp. 1327–1361, 1995.
[9] M. J. van Nieuwstadt and R. M. Murray, "Real time trajectory generation for differentially flat systems," *Int. J. Robust & Nonlinear Control*, vol. 8, no. 11, pp. 995–1020, 1998.
[10] Y. Anzai, "A note on reachability of discrete–time quantized control systems," *IEEE Trans. on Automatic Control*, vol. 19, no. 5, pp. 575–577, 1974.
[11] Y. Chitour and B. Piccoli, "Controllability for discrete systems with a finite control set," *Math. Control Signals Systems*, vol. 14, no. 2, pp. 173–193, 2001.
[12] G. Heinzinger, P. Jacobs, J. Canny, and B. Paden, "Time-optimal trajectories for a robot manipulator: a provably good approximation algorithm," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 1990, pp. 150–156.
[13] B. R. Donald and P. G. Xavier, "Provably good approximation algorithms for optimal kinodynamic planning for cartesian robots and open chain manipulators," *Algorithmica*, vol. 14, no. 6, pp. 480–530, 1995.
[14] S. M. LaValle and J. Kuffner, "Randomized kinodynamic planning," *Int. Jour. of Robotics Research*, vol. 20, no. 5, pp. 378–400, 2001.
[15] A. Bicchi, A. Marigo, and B. Piccoli, "Feedback encoding for efficient symbolic control of dynamical systems," *submitted to IEEE Trans. on Automatic Control*, 2005.
[16] D. Tilbury, R. Murray, and S. S. Sastry, "Trajectory generation for the n trailer problem using goursat normal form," *IEEE Trans. on Automatic Control*, vol. 40, no. 5, pp. 802–819, 1995.
[17] P. Rouchon, M. Fliess, J. Lévine, and P. Martin, "Flatness, motion planning, and trailer systems," in *Proc. IEEE Int. Conf. on Decision and Control*, 1993, pp. 2700–2705.
[18] O. Sordalen, "Conversion of the kinematics of a car with n trailers into a chained form," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 1993, pp. 382–387.
[19] B. Charlet, J. Levine, and R. Marino, "Sufficient conditions for dynamic state feedback linearization," *SIAM J. Control*, vol. 29, no. 1, pp. 38–57, 1991.
[20] E. Aranda-Bricaire, C. H. Moog, and J.-B. Pomet, "A linear algebraic framework for dynamic feedback linearization," *IEEE Trans. on Automatic Control*, vol. 40, no. 1, pp. 127–132, 1995.
[21] E. Sontag, "Control of systems without drift via generic loops," *IEEE Trans. on Automatic Control*, vol. 40, no. 7, pp. 1210–1219, 1995.