On Optimal Steering Of Quantized Input Systems

Stefania Pancanti, Lucia Pallottino, Antonio Bicchi

***Abstract*—In this paper we consider the problem of optimal control (specifically, minimum-time steering) for systems with quantized inputs. In particular, we propose a new approach to the solution of the optimal control problem for an important class of nonlinear systems, i.e. chained–form systems. By exploiting results on the structure of the reachability set of these systems under quantized control, the optimal solution is determined solving an integer linear programming problem. Our algorithm represents an improvement with respect to classical approaches in terms of exactness, as it does not resort to any a priori state-space discretization. Although the computational complexity of the problem in our formulation is still formally exponential, it lends itself to application of Branch and Bound techniques, which substantially cuts down computations in many cases, as it has been experimentally observed.**

I. Introduction

In this paper we consider the problem of optimally steering discrete-time dynamic systems of the form

$$x^+ = g(u,x),\ x \in \mathbb{R}^n,\ u \in \mathcal{U} \subset \mathbb{R}^m \tag{1}$$

where the input set, U, is quantized, i.e. finite or with values on regular lattices in $\mathbb{R}^m$.

While, discrete state space (X), discrete set of input symbols (U) and discrete set of output symbols (Y) characterize an automata, X, U, Y continuous are characteristic of a differentiable discrete time control system. Quantized control systems (QCS) are characterized by a continuous state space X and by U and Y that are discrete.

Often different physical phenomena and technological constraints give rise to QCS. Examples of QCS are given by problems such as the stability of an inverted pendulum on a cart with a stepping motor or the rolling of an object with polyhedral shape. Since all stepper motors have finite range and resolution and control signals are computer generated, quantization effects may result in practice from the use of digital actuators or sensors (encoders).

Even when the input is available in continuous form, quantizing the input can be a good alternative to approaches based on the discretization of the whole state space such as, e.g., in dynamic programming.

In literature, quantization has been occasionally considered as an approximation-induced disturbance to be rejected ([1], [2]). On the other hand, quantization might also be introduced on purpose to provide algorithms to solve optimal control problems: a different opinion on quantization has been taken more recently ([3], [4], [5], [6]). Quantization is a deterministic, memoryless nonlinear phenomenon that may affect inherent properties of the system in very specific ways. Furthermore, the study of quantization can be performed directly. This approach is particularly meaningful when quantization is rough, or when it is introduced on purpose in order to reduce the technological complexity of the control systems, as e.g. in mass-produced embedded systems or in distributed control systems.

In this paper, we will argue that this is the case at least for an important, albeit particular class of nonlinear systems, i.e. nonholonomic systems in chained form ([7]). In particular, the problem of optimal control for discrete-time chained-form systems in an unconstrained state space with quantized inputs has been considered. Our approach to the steering problem is based on the theory of quantized control systems, and exploits results reported in [9] and with more details in [10] about the lattice structure of the reachable set for this class of systems. In particular, conditions have been obtained under which the reachability set is a lattice, and for such lattice a complete description can be obtained by a finitely computable algorithm. The algorithm described in [9] offers a polynomial time, computationally very effective steering method for the system based on standard integer programming techniques.

To solve the optimal control steering problem, which turns out to be an integer linear programming problem, tools from graph theory are adopted. Although standard techniques cannot be applied directly, we propose a solution algorithm to solve the optimal steering problem for quantized chained form systems, which is shown to converge to the optimum.

The paper is organized as follows. We begin by introducing some basic definitions and ideas that will be necessary in the work. In Section (II) some properties of discrete chained-form systems related to reachability are explained. Generators of reachable space are described in section III and in section III-C the algorithm to find optimal transits is reported.

The optimal control problem is formulated in section 11, while in section V the solution algorithm is described.

Finally, obtained algorithm is applied to solve the steering problem for an n-trailers system,.results are reported in section VI.

II. Problem formulation

We consider a particular class of nonlinear systems, specifically two-inputs driftless nonholonomic systems. A system is said to be driftless if all configurations are equilibrium under zero control. Upon coordinate changes and state feedback, two-inputs driftless nonholonomic systems can be written in a so–called chained–form. Such form has been introduced by [7] as a canonical form for some continous-time, driftless nonholonomic systems and can be described by the ordinary differential equation

$$\begin{array}{rcl} \dot{x}_1 & = & u_1, \\ \dot{x}_2 & = & u_2, \\ \dot{x}_3 & = & x_2 u_1, \\ \vdots & = & \vdots \\ \dot{x}_n & = & x_{n-1} u_1. \end{array} \quad (2)$$

In the automatic control literature, chained-form systems have been widely used for modelling and controlling systems that range from wheeled vehicles with an arbitrary number of trailers, to satellites ([11], [12], [13], [14], [15], [16], [17]). While many steering methods for chained form systems have been provided in the literature, optimal control for these systems is still an open problem.

Consider the case where inputs to the system, rather than being allowed to change continuously in time, are bound to switch among a finite set of different levels at given switching times, which are multiples of a given time interval. Assuming such sampling interval to be of unit length, a discrete time model of chained-form systems can be easily obtained from (2) by integration as

$$\begin{array}{rcl} x_1^+ & = & x_1 + u_1, \\ x_2^+ & = & x_2 + u_2, \\ x_3^+ & = & x_3 + x_2 u_1 + \frac{1}{2} u_1 u_2, \\ \vdots & = & \vdots \\ x_n^+ & = & x_n + \sum_{j=1}^{n-2} x_{n-j} \frac{u_1^j}{j!} + u_1^{n-2} u_2 \frac{1}{(n-1)!}. \end{array} \quad (3)$$

We will assume that inputs $u = (u_1, u_2)$ can take values within a state-independent set of input symbols U, which is symmetric (i.e., if $u \in U$, then also $\bar{u} = -u \in U$). The set Ω of admissible control words (i.e. strings of admissible input symbols) is endowed with a composition law given by concatenation of strings. Because of the symmetry of U, every element $\omega \in \Omega$ has an inverse $\omega^{-1} \in \Omega$, simply defined as $(u_1 u_2 \cdots u_m)^{-1} = -u_m \cdots - u_2 - u_1, \pm u_i \in U, \forall i$. Let us denote by $\mathcal{A} : \Omega \times \mathcal{X} \to \mathcal{X}$ the state transition map for system (3), i.e. the map that associates to every initial state and every admissible input word, the corresponding end point reached by the state. In general one can write, $\forall \omega \in \Omega$

$$\mathcal{A}(\omega, x) = x + \Delta(\omega) + A(\omega, x), \quad (4)$$

where, by some simple, if tedious, calculations it can be shown (see Lemma 2 and Theorem 10 in [10]) that, chosen $\omega = v\ -v$, we have:

1) the mixed term $A(\omega, x)$ is zero;
2) the $\Delta(\omega)$ term is zero.

Hence, $\mathcal{A}(\omega^{-1}, \mathcal{A}(\omega, x)) = x$, i.e. system (3) is invertible.

In the state manifold of chained-form systems (2, 3) it is customary to distinguish a *base* subsystem, consisting of the first two state variables (x_1, x_2), and a *fiber* susbsytem with coordinates $(x_3, \ldots, x_n)$. Observe that the restriction of chained-form systems to the base variables is linear, and indeed trivial to control. On the other hand, the difficulty in controlling fiber variables increases with the dimension of the state space. A typical example of such situation is in parking maneuvers of tractor-trailer systems, where base variables are associated with the steering tractor, and fiber variables correspond to the configurations of the trailers (see section VI).

Accordingly, the reachability problem for discrete-time chained form systems can be decoupled in the analysis of reachability of the base space, and of the fiber space $\mathbb{R}^{n-2}$ associated with a reachable base point $(\overline{x}_1, \overline{x}_2)$. On the base space system (3) has the simple form

$$x^+ = x + u,\ x \in \mathbb{R}^2, u \in U. \quad (5)$$

For such linear driftless systems, the analysis of the reachable set has been characterized as follows ([9]):

Theorem 1: A necessary condition for the reachable set from the origin R_0 to be dense in $\mathcal{R}^n$ is that U contains $n+1$ controls of which n are linearly independent. If $u_1, \ldots, u_n \in U$ are linearly independent and there exist n irrational negative numbers $\alpha_1, \ldots, \alpha_n$ such that $v_i = \alpha_i u_i \in U$ for every $i = 1, \ldots, n$ then R_0 is dense. If there exists $m \leq n$ vectors v_i such that $\forall u \in U$, there exists m integers $a_i, \ldots, a_m$ such that $u = a_i v_i$, then R_0 is discrete. In particular, it is a lattice.

Observe that the reachable set R_x from a generic point x is obtained by translation of R_0. Therefore, if the control set U is quantized, symmetric and rational (as it almost always is in cases of interest, and as we assume in the rest of this paper), the reachable set is a lattice.

Fixed a base point $(\overline{x}_1, \overline{x}_2)$, consider the subgroup $\tilde{\Omega} \subset \Omega$ of control words that take the base variables back to their initial configuration. In particular, these are sequences of inputs such that the sum of the first and the second components are zero, so that the quantity $A(\omega, x)$ and the first and the second components of $\Delta(\omega)$ are zero, namely from Lemma 2 in [10]:

$$\begin{array}{l} \forall \tilde{\omega} \in \tilde{\Omega} \text{ and } \forall x,\ A(\tilde{\omega}, x) = 0; \\ \Delta_1(\tilde{\omega}) = 0; \\ \Delta_2(\tilde{\omega}) = 0. \end{array}$$

The effect of such subgroup on the fiber subsystem can be described by

$$z^+ = z + v, z = (x_3, x_4, \cdots, x_n) \in \mathbb{R}^{n-2}, v \in \tilde{U} \quad (6)$$

where $\tilde{U} = \{\Delta^f(\omega), \omega \in \tilde{\Omega}\}$ and where $\Delta^f(\omega)$ denotes the $(n-2)$-dimensional projection of Δ on the fiber space. Clearly, $\tilde{U}$ is itself symmetric: indeed if $\omega \in \tilde{\Omega}$ then also $\omega^{-1} \in \tilde{\Omega}$ and $\Delta^f(\omega^{-1}) = -\Delta^f(\omega)$. The action of the subgroup $\tilde{\Omega}$ on the fiber is additive (namely, $\mathcal{A}(\tilde{\omega}_1, \mathcal{A}(\tilde{\omega}_2, x)) = \mathcal{A}(\tilde{\omega}_1, x) + \mathcal{A}(\tilde{\omega}_2, x), \forall \tilde{\omega}_1, \tilde{\omega}_2 \in \tilde{\Omega}$), and the structure of the reachable set in the fiber is the same over every (reachable) base point.

For the set $\tilde{U}$ of all control inputs that can be applied to the fiber dynamics (6), corresponding to the set of input words $\tilde{\Omega}$ that drive base variables back to their initial values, the following result holds ([9]):

Theorem 2: Let the control set U be quantized, symmetric and rational. Then, all elements $\Delta^f(\tilde{\omega}) \in \tilde{U}$ can be written as integer combinations of a finite set of generators Δ_i^f, uniquely determined from U. Each generator is a rational vector in $\mathbb{Q}^{n-2}$, corresponding to a control word $\tilde{\omega}_i \in \tilde{\Omega}$ in the original alphabet U.

As a consequence, with reference to system (5), we can conclude that if the controls set U is rational and quantized, the reachability structure of a chained form discrete-time system is completely described by a lattice in the state space (the cartesian product of the base and fiber lattices). Such lattice structure, which plays a central role in our approach in solving the optimal steering problem, can be described completely by a finite number of generators, whose evaluation can be done in polynomial time with respect to the state space dimension and the number of control symbols in U ([10]). In next sections the computation of generators is described in details.

III. Generators and transits

In order to compute generators we need several definitions and lemmas that can be found in details in [10] and are reported here for reader convienence. First of all, let consider a function Σ defined on the set of input words Ω and that counts the number of symbol that appear in a word taking into account signs, for each positive symbol in the control set $U \in \mathbb{Q}^n$. Since U is symmetric its cardinality is even, for example $2c$, then the function Σ takes value in $\mathbb{Z}^c$. Furthermore, let N_W be an integer value matrix such that $WN_W = 0$ and such that G.C.D. of element of each column is 1, for each column. Examples on the construction of matrix N_W will be shown in next sections.

Let c be the number of positive symbol in U, the subgroup $\tilde{\Omega}$ can be described also through Σ and N_W as follow:

$$\tilde{\Omega} = \{\omega \in \Omega | \Sigma(\omega) = (N_W \alpha), \alpha \in (\mathbb{N} \cup \{0\})^{c-2}\}$$

Furthermore, if we define

$$\mathcal{L} = \{\omega \in \Omega | \Sigma(\omega) = \pm(N_W)_j, \omega \text{ of minimal lenght}\},$$

where $(N_W)_j$ is the j-th column of N_W, we have that the set $C = \{\omega\tilde{\omega}\omega^{-1};\, \omega \in \Omega,\, \tilde{\omega} \in \mathcal{L}\}$ is a set of generators for $\tilde{\Omega}$ but it is not finite.

Let Δ^f be the projection of Δ on the fiber space, for all $\omega \in \Omega$ and $\tilde{\omega} \in \mathcal{L}$, we have $\Delta^f(\omega\tilde{\omega}\omega^{-1}) = G(\omega)\Delta^f(\tilde{\omega})$, where $G(\omega) = exp(-J_0\sigma(\omega))$ and J_0 is the transpose of a Jordan block associated with zero eigenvalue of dimension $n-2$.

Finally, consider $u_i \in U \subset \mathbb{Q}^n$ (for each i) with $u_{i,k} = \frac{p_{i,k}}{q_{i,k}}$ where $p_{i,k}$ and $q_{i,k}$ are integer and coprime for $k = 1, ..., n$. Let $d_{i,k}$, p, q integer and p, q coprime such that $\frac{p_{i,k}}{q_{i,k}} = d_{i,k}\frac{p}{q}$ $\forall i = 1, ..., c$ and $k = 1, ..., n$. Hence, there exist some $\alpha_i \in \mathbb{Z}$ such that $\sigma(\omega) = \sum_{i=1}^{c} \alpha_i u_{i,1} = \frac{p}{q}\sum_{i=1}^{c} \alpha_i d_{i,1}$. Let $k(\omega)$ be a map from the Ω group to the $\mathbb{Z}$ space such that $k(\omega) = \sum_{i=1}^{c} \alpha_i d_{i,1}$, hence $\sigma(\omega) = \frac{p}{q}k(\omega)$.

It is possible to conclude that chosen $\hat{\omega}_i \in \Omega$ such that $k(\hat{\omega}_i) = i$, the set

$$\mathcal{B} = \{G(\hat{\omega}_0)\Delta^f(\tilde{\omega}), ..., G(\hat{\omega}_{n-3})\Delta^f(\tilde{\omega}), \tilde{\omega} \in \mathcal{L}\},$$

is finite and generate the action of the group Ω on the fiber with integer combinations [10].

In the following, words $\hat{\omega}_i$ will be referred to as *transits*. On a two dimensional lattice, the transit u and the word $\omega = v\,u\,-v\,-u \in \mathcal{L}$ (figure 1, left) give the generator $u\,v\,u\,-v\,-u\,-u$ represented in figure 1 (right).

Let $\mathcal{B}_{\text{base}} = \{b_i \in \Omega | \Delta^f(b_i) \in \mathcal{B}\}$, then b_i, called *generators*, can be written as $b_i = \hat{\omega}_i\tilde{\omega}\hat{\omega}_i^{-1}$ for some transit $\hat{\omega}_i$ and $\tilde{\omega} \in \mathcal{L}$.

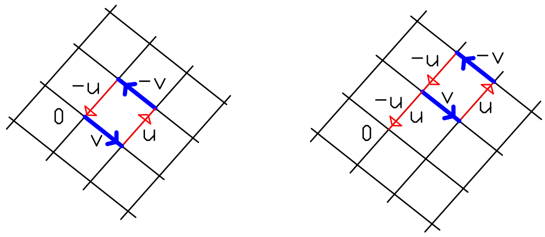

Fig. 1. Left: representation of $\omega = v\,u\,-v\,-u \in \Omega$, it takes back the base variables (form a cycle on the base lattice), right: an example of composition of ω with transit u, it is a non minimal cycle.

In following sections we introduce two examples for which transits and generators computation is described: the car-like system and the one-trailer system. For higher dimensional cases, the computation of transit is not simple but can be obtained as the result of an algorithm described in next sections.

A. Lattice generators for a car-like system

Consider first a car-like model:

$$\begin{cases} \dot{x} = v\cos(\theta), \\ \dot{y} = v\sin(\theta), \\ \dot{\theta} = \omega, \end{cases}$$

where v and ω are the linear and angular velocities respectively.

The chained form corresponding model is:

$$\begin{cases} x_1^+ = x_1 + u_1, \\ x_2^+ = x_2 + u_2, \\ x_3^+ = x_3 + x_2u_1 + \frac{1}{2}u_1u_2, \end{cases}$$

where x_1, x_2, x_3 and u_1, u_2 are given by the input transformations in [11]:

$$\begin{cases} x_1 = x, \\ x_2 = \tan(\theta), \\ x_3 = y, \end{cases} \qquad \begin{cases} u_1 = v\cos(\theta), \\ u_2 = \frac{\omega}{\cos^2(\theta)}, \end{cases}$$

Variables x_1, x_2 are the base variables, and since we are interested in cycles on the base space, at the final time T we have:

$$\begin{cases} x_1(0) = x_1(T), \\ x_2(0) = x_2(T), \end{cases}$$

and then initial and final values of x and $\tan(\theta)$ are the same. In the car-like case we choose the control input to be $u = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \in U = \{\pm s,\, \pm r,\, \pm t\}$, where

$$s = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, r = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, t = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Hence, with s we denote a straight line motion, with r a rotation "on the spot" and with t a turn. Let U be the matrix of the control inputs:

$$U = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

we have:

$$UN_W = 0 \Rightarrow N_W \propto \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}.$$

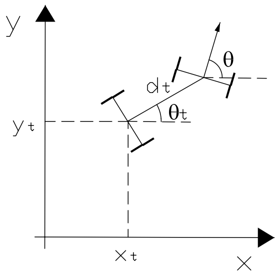

Fig. 2. State variables $(x_t,\, y_t,\, \theta_t,\, \theta)$ for the one-trailer system.

It is sufficient (Lemma 4 in [10]) to consider control inputs of minimal lenght and then given the control input $w = s, r, -t$, it is sufficient to consider all words consisting in all possible permutations of $s,\ r,\ -t$ and of $-s,\ -r,\ t$ (for a total of 12 control inputs). No transits has been considered since in this case we have $n = 3$ and then

$$\mathcal{B} = \{G(\hat{\omega}_0)\Delta^f(\tilde{\omega}), \tilde{\omega} \in \mathcal{L}\},$$

but $G(\hat{\omega}_0)$ is the identity matrix. Transits appear then in cases with $n > 3$ such as in next example in which a one-trailer system is considered.

B. Lattice generators for a one-trailer system

Consider a one-trailer system, represented by the model:

$$\begin{cases} \dot{x}_t = v_t \cos(\theta_t), \\ \dot{y}_t = v_t \sin(\theta_t), \\ \dot{\theta}_t = \frac{v}{d_t}\sin(\theta - \theta_t), \\ \dot{\theta} = \omega, \end{cases}$$

where $(x_t,\, y_t)$ is the position of the trailer, θ_t is the heading angle of the trailer, $v_t = \cos(\theta - \theta_t)$ is the tangential velocity of the trailer and d_t is the distance from the wheels of the trailer to the wheels of the car, (see figre 2).

In chained form, the model becomes:

$$\begin{cases} x_1^+ = x_1 + u_1, \\ x_2^+ = x_2 + u_2, \\ x_3^+ = x_3 + x_2 u_1 + \frac{1}{2}u_1 u_2, \\ x_4^+ = x_4 + x_3 u_1 + x_2 \frac{u_1^2}{2} + \frac{1}{6}u_1^2 u_2, \end{cases}$$

where $x_1,\, x_2,\, x_3,\, x_4$ and $u_1,\, u_2$ are given by the input transformations in [11]:

$$\begin{cases} x_1 = x, \\ x_2 = \frac{\tan(\theta_t - \theta)}{d_t \cos^3(\theta_t)}, \\ x_3 = \tan(\theta_t), \\ x_4 = y, \end{cases}$$

$$\begin{cases} u_1 = v\cos(\theta_t)\cos(\theta - \theta_t), \\ u_2 = \frac{\omega}{d_t \cos^3(\theta_t)\cos^2(\theta - \theta_t)} + \tau\cos(\theta_t)\cos(\theta - \theta_t)v, \end{cases}$$

where $\tau = \frac{\tan(\theta_t - \theta)\cos(\theta_t) + 3\sin(\theta_t)\sin^2(\theta_t - \theta)}{d_t \cos^3(\theta_t)}$, it is necessary that $\theta_t \neq \pm\frac{\pi}{2}$.

Let $u = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \in U = \{\pm s,\, \pm r,\, \pm t\}$, where

$$s = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, r = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, t = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

In this case, s denotes a straight line motion only when $\theta_t = \theta$, r still denotes a rotation "on the spot" and t a turn.

Computing N_W, we have that also in this case there are 12 control inputs $\Delta^f(\tilde{\omega})$ to be considered in the generators set, in particular they are the same of the generators obtained in the car-like example. In the one-trailer case, we have $n = 4$ and then we have transits since

$$\mathcal{B} = \{\Delta^f(\tilde{\omega}),\, G(\hat{\omega}_1)\Delta^f(\tilde{\omega}), \tilde{\omega} \in \mathcal{L}\},$$

where $\hat{\omega}_1$ is such that $\sigma(\hat{\omega}_1) = 1$, for example $\hat{\omega}_1 = s$. Concluding there are other 12 generators ($G(\hat{\omega}_1)\Delta^f(\tilde{\omega})$) that complete a set of 24 generators. The choice $\hat{\omega}_1 = s$ has been possible since the dimension of the problem is still small; when the dimension grows it is not easy to compute the tranists. In next section we describe an algorithm to compute optimal transits.

C. Algorithm for optimization on transits

In this section we are interested in optimizing transits that will be used in an optimal control problem decribed in next section. As shown above, transits occur when the steering problem for a chained-form system has a configuration space of at least dimension four. In this case it is necessary to take into account cyclic generators with transit. With respect to cyclic generators (elements of $\tilde{\Omega}$), transits cause a translation on the lattice structure of cyclic generators (see figure 1).

The following algorithm guarantees the choice of transits at lowest cost (in this formulation the problem is solved in minimal time but more general weights-problems can be solved equivalently).

Suppose that: *the G.C.D. between at least two first components of symbols in U is one*. This condition is strictly related to the existence of $\hat{\omega}_1$ such that $k(\hat{\omega}_1) = 1$, and it is sufficient to allow correctness of the following algorithm:

Step i: for i from 1 to $n - 3$

Solve

$$\begin{array}{ll} \min & \hat{\Sigma}(\omega) \\ \text{s. t.} & \begin{cases} k(\omega) & = & i \\ \omega & \in & \Omega \end{cases} \end{array} \quad (7)$$

where the function $\hat{\Sigma}(\omega) : \Omega \mapsto \mathbb{N}$ counts the number of symbols in the word ω without taking into account signs. Let $\hat{\omega}_i$ be the optimal solution founded at step i.

Since the optimization problems (7) are linear and have infinite dimension, they are $\mathcal{NP}$-complete. The $\mathcal{NP}$-completeness can be solved rewriting the problem 7 as follow:

$$\begin{array}{ll} \min & \|x\|_1 \\ \text{s. t.} & \begin{cases} Fx & = & i\frac{p}{q} \\ x_i & \in & \mathbb{N} \end{cases} \end{array} \quad (8)$$

where $|U| = m$ and the matrix $F \in \mathbb{Q}^{1\times 2m}$ is composed of the first component of each control input in U and the component x_i of the vector x counts how many times the control to which F_i belongs is considered in the solution. Such optimization problems can be easly solved by an integer optimization commercial package such as CPLEX [18].

As we have explained at the beginning of this section, the transits are employed to construct cyclic generators in relation to the dimension of the configuration space but it is important to underline that the choice of the transit $\hat{\omega}_i$ is independent of the construction of the increment matrix $\Delta^f(\omega)$ of the steering problem. Indeed, in order to determine the control generators we use only the fact that a transit verifies the condition $k(\omega) = i$, for particular i, without using an explicit word ω, [9].

IV. Steering on lattices: the optimal control problem

Consider the system (3) with a quantized, rational and symmetric control set U. Let $H \in \mathbb{Q}^{n-2\times l}$ denote the matrix whose columnns are the l generators contribution on the fiber $\Delta_i^f = \Delta(\tilde{\omega}_i), i = 1, \cdots, l$. Without loss of generality, up to rescaling the fiber state space, we may take H to be an integer matrix. The *steering problem* on the fiber, i.e. the problem of finding a control sequence that takes the system (3) from an initial q_{start} to a desired q_{goal}, consists hence in solving a linear system of the form:

$$Hx = (q_{goal} - q_{start}). \tag{9}$$

Integer solutions $x \in \mathbb{Z}^l$ of this equation exist if an only if the initial and goal points differ by a vector belonging to the fiber lattice, which we will assume henceforth (in other cases, integer truncations of a real solution x will provide approximated steering to the goal, within a tolerance dictated by the lattice mesh).

Any solution $x = (x_1, \cdots, x_l) \in \mathbb{Z}^l$ of system (9) gives a sequence of cyclic control inputs that includes x_i instances of the words $\tilde{\omega}_i$. There are of course infinitely many possible solutions x, each corresponding to a combinatoric number of different possible sequences of control words $\tilde{\omega}_i$.

Optimal steering strategies among solutions of (9) will be considered introducing a cost p_i associated to the control symbol $u_i \in U$. The corresponding cost for a word $\omega = (u_1, u_2, \cdots, u_N), u_i \in U$ is defined as $C(\omega) = \|Px\|$, where x_i stands for the number of appearances of the symbol u_i in ω (with negative sign if $-u_i$ appears), and $P = \mathrm{diag}(p_i)$.

A constrained minimization problem can be considered at this point, i.e.

$$\begin{array}{ll} \min_x & \|Px\| \\ \text{s. t.} & \left\{ \begin{array}{l} Hx = x_{goal} - x_{start} \\ x \in \mathbb{Z}^m \end{array} \right. \end{array} \tag{10}$$

leading to a linear integer program if a one-norm is considered, while using a two-norm would result in an integer quadratic program. Efficient algorithms do exist for both these problems: however, unfortunately, such formulation does not reflect the reality of our optimal control problem.

Indeed, in combining control words by concatenation cancellations of symbols may occur. To obtain the sum of two control actions $\Delta(\tilde{\omega}_i), \Delta(\tilde{\omega}_j)$ on the fiber, corresponding to control words $\tilde{\omega}_i, \tilde{\omega}_j$ whose costs are $C_i = C(\tilde{\omega}_i)$ and $C_j = C(\tilde{\omega}_j)$, respectively, the sum $C_i + C_j$ is only an upper bound to the actual cost of the corresponding control. Indeed, cancellations of one or more trailing symbols in $\tilde{\omega}_i$ with an equal number of symbols leading in $\tilde{\omega}_j$ is possible. We will denote by $\hat{C}(\tilde{\omega}_i, \tilde{\omega}_j)$ the actual cost of the word pair $(\tilde{\omega}_i, \tilde{\omega}_j)$.

For example, if $\tilde{\omega}_i = u_1u_2u_3u_4$ and $\tilde{\omega}_j = -u_4u_5 - u_2 - u_1$, one has (in a minimum time problem) $C_i = 4$ and $C_j = 4$. However, the concatenation of $\tilde{\omega}_i$ with $\tilde{\omega}_j$ leads, by cancellations, to the control word $u_1u_2u_3u_5 - u_2 - u_1$, so that $\hat{C}(\tilde{\omega}_i, \tilde{\omega}_j) = 6 < 8$. Obviously, cancellations are crucial in minimizing unnecessary maneuvers in the steering problem, and motivate the following reformulation of the optimal control problem.

Consider an oriented graph $G_0 = (N_0, A_0)$ with a set N_0 of $l + 2$ nodes, l of which are associated with the contributions $\Delta(\omega_i)$ on the fiber given by generators $b_i = \tilde{\omega}_i$, and where a start node S and a goal node F are additionally considered. In the arc set A_0 of G_0, all arcs connecting the start and goal nodes S, F with all other nodes are included, i.e. $(S, i) \in A_0, i = 1, \ldots, l$ and $(i, F) \in A_0, i = 1, \ldots, l$. An arc (i, j) is included in A_0 only if $\tilde{\omega}_i$ and $\tilde{\omega}_j$ are not the inverse of each other. In particular, for every node $i \neq S, F$, the arc (i, i) is included in A_0.

To the arc $(i, j) \in A_0$ we associate the cost $\hat{C}_{i,j} = \hat{C}(\tilde{\omega}_i, \tilde{\omega}_j) - C(\tilde{\omega}_i) \geq 0$ of the control sequence $(\tilde{\omega}_i, \tilde{\omega}_j)$ (so that the cost of path (S, i), (i, j) on the graph is $C(\tilde{\omega}_i, \tilde{\omega}_j)$), taking into account all possible cancellations. Notice that in general $\hat{C}_{i,j}$ is not equal to $\hat{C}_{j,i}$ (in the example above, for instance, $\hat{C}_{i,j} = 6 - 4 = 2$ while $\hat{C}_{j,i} = 4 - 4 = 0$). In figure 3 an example of graph is represented. A refinement step is necessary to finalize the graph construction for pairs $(\tilde{\omega}_i, \tilde{\omega}_j)$ where the number of cancellations is larger than the half-length of the shortest of the two words. Indeed, in this case it may happen that the cost of a triplet $(\tilde{\omega}_i, \tilde{\omega}_j, \tilde{\omega}_k)$ is underestimated by $\hat{C}_{ij} + \hat{C}_{jk}$. For example, if $\tilde{\omega}_i = u\,v - u\,v\,u$, $\tilde{\omega}_j = -u - v\,u\,v$ and $\tilde{\omega}_k = -v - u - u$, we have $\tilde{\omega}_i\tilde{\omega}_j = u\,v\,v$ ($\hat{C}_{ij} = 3 - 5 = -2$) and $\tilde{\omega}_j\tilde{\omega}_k = -u - v - u$ ($\hat{C}_{jk} = 3 - 4 = -1$) while the triplet is $\omega_i\omega_j\omega_k = u\,v - u - u$ whose cost is $4 - 5 = -1$ whereas on the graph the path (S, i), (i, j), (j, k) would cost $5 - 2 - 1 = 2$. To avoid this problem, we remove in the graph the arc (i, j) corresponding to such pairs, and add a new node associated to $\Delta(\tilde{\omega}_i) + \Delta(\tilde{\omega}_j)$ with cost $\hat{C}_{ij}$. These new nodes are connected to all other nodes by arcs whose cost is evaluated

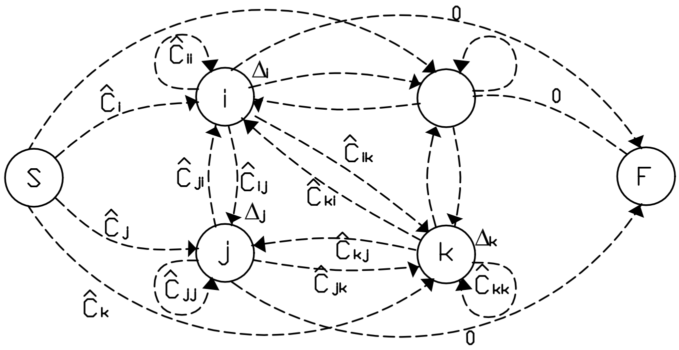

Fig. 3. Graph associated with generators of the fiber displacements.

as usual, with the exception of arcs corresponding again to cancellations of more than the half-length of either words, which are not considered in the new graph.

On the graph G_0, all possible combinations of the generating control words $\tilde{\omega}_i$ are represented by connected paths from S to F. The optimal control problem on the fiber space can hence be formulated as follows:

Given the oriented graph G_0, determine the minimum-cost path from S to F with the constraint that the sum of all Δ_i of visited nodes equals the desired fiber displacement $z_{goal} - z_{start}$.

Thus, the optimal control problem can be regarded as a minimum-cost path search on a graph, with a constraint on the sum of "tokens" collected at each visited node. Notice that G_0 contains cyclic arcs of type (i, i), allowing to collect an arbitrary integer number of the corresponding token $\Delta(\tilde{\omega}_i)$. The search problem is an $\mathcal{NP}$-complete linear integer programming problem ([19][20]), and differs substantially from standard shortest path searches on a graph because of the constraint and of the presence of cycles (cyclic paths are obviously never considered in unconstrained path searches). The following section proposes a correct and complete algorithm to solve this optimal control problem.

V. A SOLUTION ALGORITHM

The non-standard nature of the optimization problem described above is such that even rather general solution techniques, as e.g. branch and bound, and commercial software tools for integer programming, cannot be used directly to solve the problem. We propose a procedure for the solution of this problem which basically consists of solving a sequence of problems of increasing complexity.

Consider first that an upper limit U on the optimal control cost can be easily obtained by evaluating the cost U_0 of *any* solution of the integer linear system (9) – for instance, a solution to problem (10), in the following will be referred to as *starting solution*.

At the first stage of the proposed algorithm, a new graph $G_1 = (N_1, A_1)$ is built by setting $N_1 = N_0$ and by removing all cyclic arcs from A_0, namely $A_1 = A_0 \setminus \{(i,i), \forall i\}$. Let now formalize the optimization problem obtained with the formulation given in previous section. Consider the incidence matrix $E \in \mathbb{R}^{s \times t}$ associated with the graph G_1: given an order to the elements of set A_1 (cardinality t) and of set N_1 (cardinality s), the element $E_{ij} = -1$ if the $i - th$ node is the first node of arc j, $E_{ij} = 1$ if the $i - th$ node is the second node of arc j, $E_{ij} = 0$ otherwise. Let $x \in \mathbb{R}^t$ be the vector variables taking values in $\{0, 1\}^t$ and representing the ordered arcs of the graph. Let $q \in \mathbb{R}^s$ such that $q_S = -1$, $q_F = 1$ and $q_i = 0$ for $i \neq S, F$. Finally, let $C^T \in \mathbb{R}^t$ be the vector in wich the cost of the arcs are reported, the optimization problem is then

$$\begin{array}{ll} \min & Cx \\ s.t. & \left\{ \begin{array}{l} Ex = q \\ \tilde{H}x = d \\ x \in \{0, 1\}^t \end{array} \right. \end{array} \qquad (11)$$

where the set of costraints $\tilde{H}x = d$ (in the following will be referred to as set of *token constraints*) represents the costraints given in (9) where $\tilde{H} \in \mathbb{R}^{n-2 \times t}$ and the column $\tilde{H}_j$ is associated to the arc $j = (i, k)$ and represent the "token" payed at node k that is $\Delta(b_k)$ (where b_k is the generator associated with node k). The vector D represent the total displacement we intend to achieve on the fiber.

A branch-and-bound algorithm is applied to search minimum cost, token-constrained paths on G_1. Within such branch-and-bound subprocedure, the token constraint is relaxed, hence a number of classical minimum cost path search problems are obtained (solvable by the Dijkstra algorithm [21]) in each of which an arc is forced to be ($x_i = 1$) or not ($x_i = 0$) in the optimal solution. If the forced condition $x_i = 1$ or $x_i = 0$ brings to a shortest path of cost larger that U then the relative branch is cut and not further explored. Otherwise, another arc is forced to be or not in the optimal solution. If all branch are cut then no solution with cost less than U has been found. Otherwise, an optimal solution is found with cost $U_1 < U$. This solution is the shortest path from node S to F but in order to be an admissible solution of problem (11) it has to verify the token constraint. In this case the upper bound U on the optimal cost is updated, $U = U_1$.

At the $i + 1$–th step of the algorithm, a graph $G_{i+1} = (N_{i+1}, A_{i+1})$ is built such that $N_{i+1} = N_i + N_0 \setminus \{S, F\}$, and A_{i+1} contains all connecting arcs between different nodes in N_{i+1} (without cyclic arcs). In other words, each node j with a cycle arc is splitted into two nodes (see figure 4) so that at step i, path with i cycles can be considered. A branch-and-bound algorithm is used again to find the constrained minimum cost U_{i+1}, and the upper bound is updated if $U_{i+1} < U$ and if the solution verifies the token constraint.

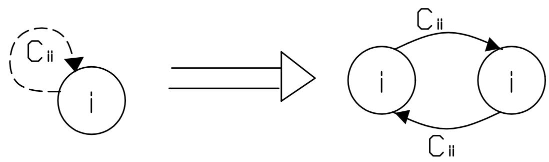

Fig. 4. The node i with a cycle arc is splitted into two nodes and two arcs.

A stopping condition for the procedure can be provided as follows. A lower bound on the optimal control cost solution L is initially set equal to the cheapest cost $L_0 = C_i$ of arcs of type (S, i) in the G_1 graph, since the cost of arc (i, F) is zero. At each step, the lower bound is updated as $L = L_{i+1} = L_i + \hat{C}_c$, where $\hat{C}_c$ denotes the minimum cost of a closed cycle in the graph G_1. The value of $\hat{C}_c$ is determined once and for all at the beginning of the procedure, by solving a standard (unconstrained) minimum-cost path problem on G_1.

The overall procedure is stopped whenever $L \geq U$.

Theorem 3: The solution algorithm is correct and complete.

Proof: Because initial and goal configurations are assumed to belong to the lattice, the optimum exists. Also, because the action on the fiber of the whole group $\tilde{\Omega}$ of control inputs that correspond to the desired final value of the base variables, is generated by the finite set of generators $\Delta(\tilde{\omega}_i), i = 1, \ldots, m$, and this set is (implicitly, but completely) searched by the branch-and-bound algorithm at successive stages of the algorithm, the algorithm is correct. On the other hand, the two sequences $\{L_i\}_{i \geq 0}$ and $\{U_i\}_{i \geq 0}$ are strictly uniformly increasing and non-increasing, respectively, and at any stage it holds $L_i \leq U_i$. Hence the algorithm stops in a finite number of stages, all of which consist of an implicit search on a finite graph, i.e. of a finite number of operations. ■

The proposed algorithm has exponentially increasing complexity with the number of generators, as it uses a number of instances of a branch and bound procedure: this is hardly a surprise, as we are after all dealing with a nontrivial optimal control problem. However, performance can be improved by providing good initial estimates of the upper bound U_0. Some preprocessing of generators to facilitate the algorithm convergence can also help, and work is currently ongoing in this direction. The next section will provide some numerical examples of application of the proposed algorithm.

VI. Application to n-trailer steering

As mentioned in the introduction, among the nonlinear systems which can be converted in chained form (2), wheeled vehicles represent a particularly interesting class.

The kinematic model of a tractor with n trailers is given by

$$\begin{array}{rcl} \dot{x} & = & \cos\theta_n v_n \\ \dot{y} & = & \sin\theta_n v_n \\ \dot{\theta}_n & = & \frac{1}{d_n}\sin(\theta_{n-1}-\theta_n)v_{n-1} \\ & \vdots & \\ \dot{\theta}_i & = & \frac{1}{d_i}\sin(\theta_{i-1}-\theta_i)v_{i-1} \quad i=1,\ldots,n \\ & \vdots & \\ \dot{\theta}_1 & = & \frac{1}{d_1}\sin(\theta_0-\theta_1)v_0 \\ \dot{\theta}_0 & = & \omega \end{array} \tag{12}$$

where (x, y) is the absolute position of the center of the axle between the two wheels of the rear-most trailer; θ_i is the orientation angle of trailer i with respect to the x-axis, with $i \in \{1, \ldots, n\}$; θ_0 is the orientation angle of the tractor axle with respect to the x-axis; d_i is the distance from the center of trailer i to the center of trailer $i-1$, $i \in \{2, \ldots, n\}$; d_1 is the the distance from the wheels of trailer 1 to the wheels of the tractor. The two inputs of the systems are v_0 and ω, the tangential velocity of the car and the angular velocity of the tractor respectively. The tangential velocity of a trailer i, v_i, is given by

$$v_i = \cos(\theta_{i-1}-\theta_i)v_{i-1} = \prod_{j=1}^{i}\cos(\theta_{j-1}-\theta_j)v_0,$$

where $i \in \{1, \ldots, n\}$. Incidentally, this model is identical to the model of a four-wheeled car pulling $n-1$ trailers, provided $\theta_0 - \theta_1$ denotes the angle of the front wheels relative to the orientation θ_1 of the rear axle of the four-wheeled car.

Sørdalen in [11] has shown, by a constructive method, that system (12) can be converted in chained form. We consider here the application of our proposed optimal quantized control algorithm to the approximate determination of an optimal continuous control for system (12). This implies introducing time and control quantizations, and applying the computed solutions as piece-wise constant inputs to system (12). The quantized control set we consider is comprised of three inputs,

$$U = \left\{ \pm \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \pm \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \pm \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right\},$$

corresponding respectively to straight motions, rotations about the axle center, and arcs of a circumference by the tractor.

We report the results of optimization runs for a tractor with 1 and 2 trailers. Minimum time controls are determined in each case, and a non-optimal trajectory is reported for comparison.

For the one-trailer problem, the graph G_0 has 24 nodes, see section (III-B). We have considered the problem with initial and goal configurations given by

$$x_{ic} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad x_{fc} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

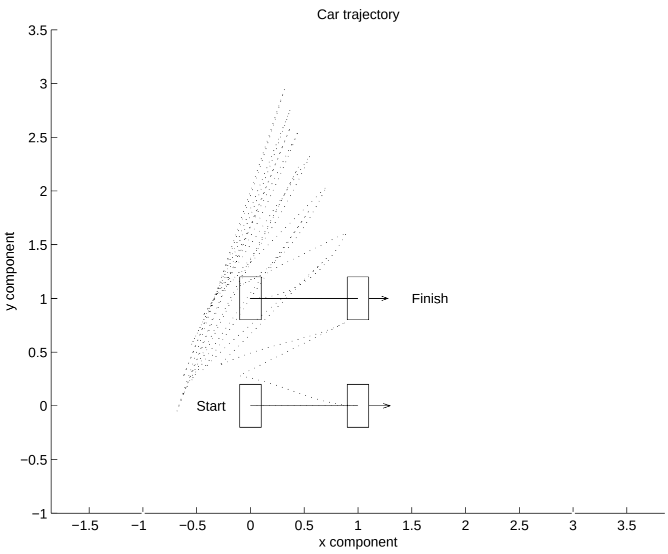

Fig. 5. Non optimal trajectory for the one-trailer. The minimal time cost is 36.

For the two-trailers problem, the graph G_0 has 36 nodes. We have solved the problem with initial and goal configurations given by

$$x_{ic} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad x_{fc} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

These example show that the proposed solution is applicable to problems in dimension up to 5. Larger dimensions of the state space are computationally expensive at this stage, although we expect that refinements on the choice of the generators could lead to an increase of the tractable dimension by one or two.

VII. Conclusions

In this paper we have studied the optimal steering problem for chained-form systems by quantized control inputs. The

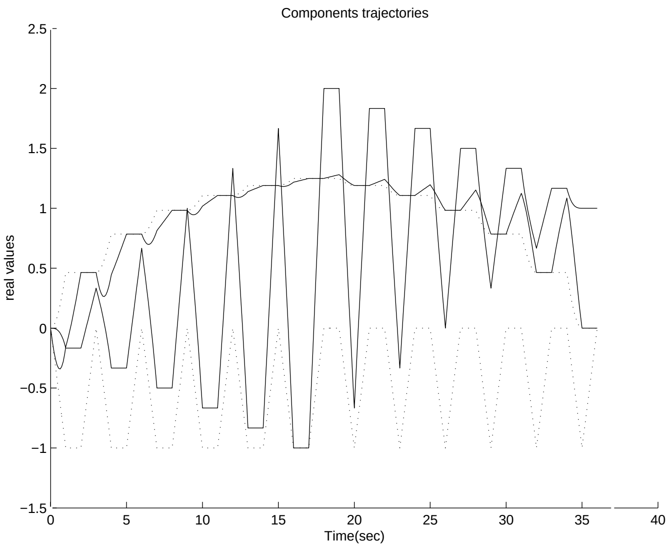

Fig. 6. The trajectories of the base variables (dashed) and fiber variables (continuos) fro the non-optimal one-trailer solution.

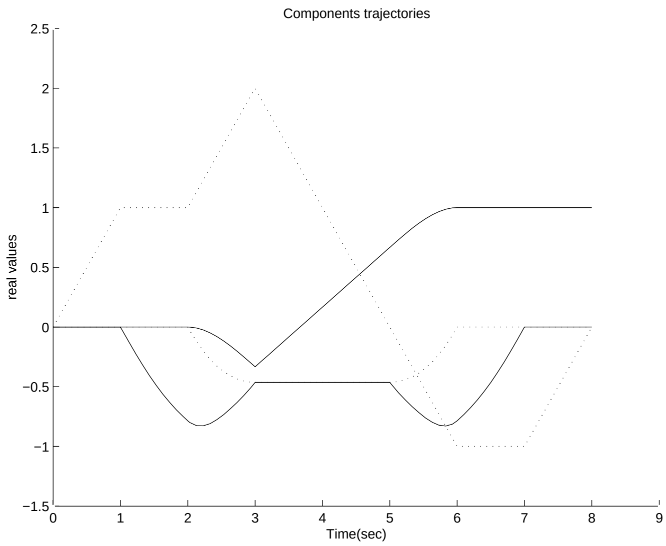

Fig. 8. The trajectories of the base variables (dashed) and fiber variables (continuos) for the optimal one-trailer solution

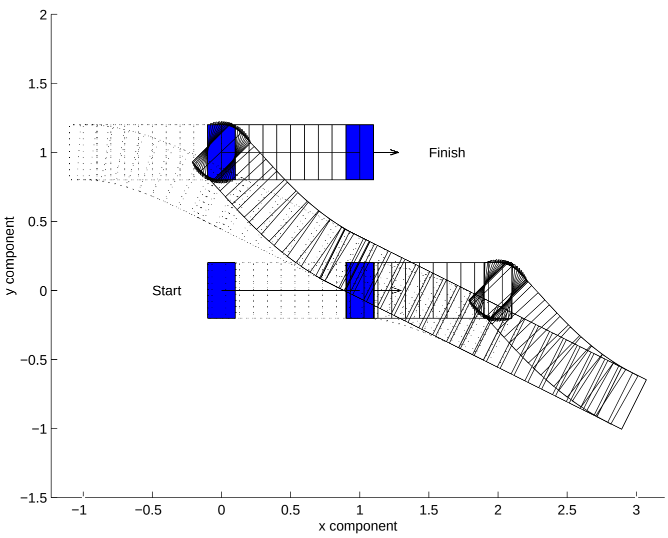

Fig. 7. Optimal trajectory for the one-trailer problem. The minimal time cost is 8.

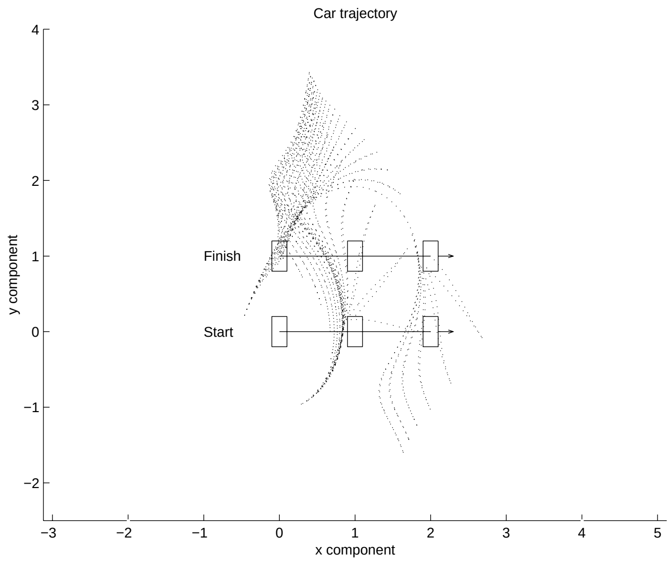

Fig. 9. Non optimal trajectory for the two-trailers. The minimal time cost is 108.

reachable set for these systems class under quantized rational inputs is a lattice and this structure can be used to solve the optimal steering problem.

We have formalized the steering problem on lattices as an integer linear programming problem that cannot be solved directly by standard integer programming techniques. Generators of reachable space have been obtained. A complete and correct solution algorithm has been proposed.

We have proposed to apply this optimal control on lattices to solve the steering problem for continuos systems that can be converted in chained form: in particular, discretizing the time, we consider a quantized control, and solve the steering task on the lattice. The optimal control strategy obtained is then applied to determine piecewise-constant sub-optimal controls in continuos time. Applications give rather satisfactory results.

REFERENCES

[1] J. E. Bertram, "The effect of quantization in sampled feedback systems," *Trans. AIEE Appl. Ind.*, vol. 77, pp. 177–181, Sept. 1958.

[2] J. B. Slaughter, "Quantization errors in digital control systems," *IEEE Trans. Autom. Control*, vol. 9, pp. 70–74, Sept. 1964.

[3] D. F. Delchamps, "Extracting state information from a quantized output record," *Systems and Control Letters*, vol. 13, pp. 365–371, 1989.

[4] D. F. Delchamps, "Stabilizing a linear system with quantized state feedback," *IEEE Trans. Autom. Control*, vol. 35, no. 8, pp. 916–926, 1990.

[5] R.W. Brockett, "Asymptotic stability and feedback stabilization," in *Differential Geometric Control Theory*, Millmann Brockett and Sussmann, Eds., pp. 181–191. Birkhauser, Boston, U.S., 1983.

[6] N. Elia and S. K. Mitter, "Quantization of linear systems," in *Proc. 38th Conf. Decision & Control*. IEEE, 1999, pp. 3428–3433.

[7] S. S. Sastry R. M. Murray, "Nonholonomic motion planning: Steering using sinusoids," *IEEE Trans. on Automatic Control*, vol. 38, pp. 700–716, 1993.

[8] A. Marigo and A. Bicchi, "Steering driftless nonholonomic systems by control quanta," in *Proc. IEEE Int. Conf. on Decision and Control*, 1998.

[9] A. Marigo, B. Piccoli, and A. Bicchi, "Reachability analysis for a class

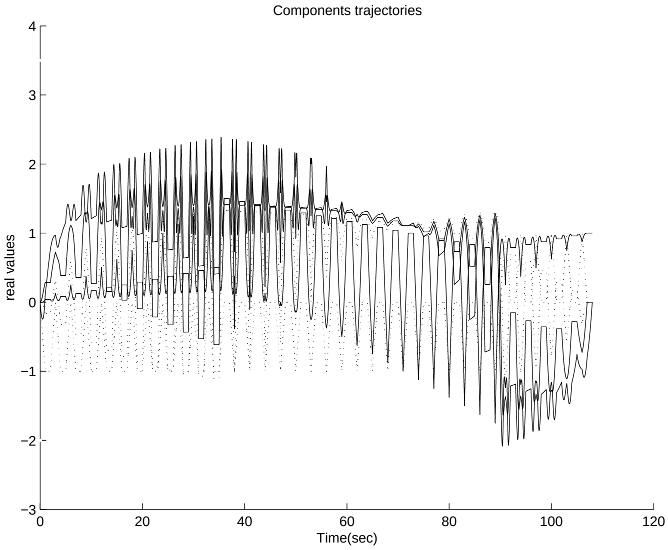

Fig. 10. The trajectories of the base variables (dashed) and fiber variables (continuos) for the nonoptimal two-trailers solution.

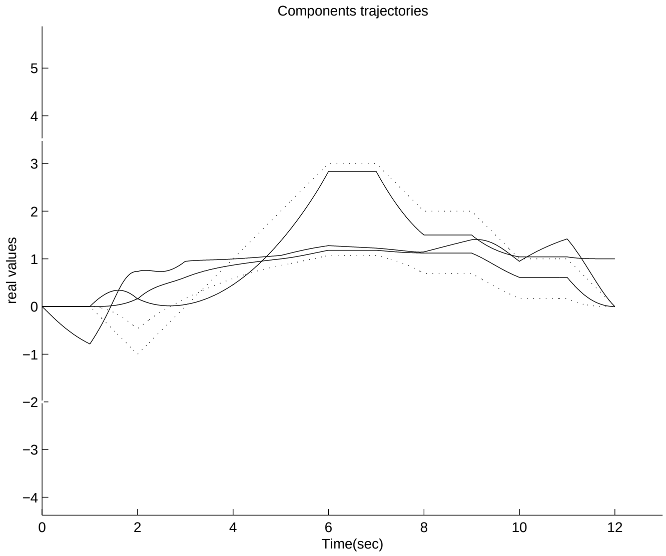

Fig. 12. The trajectories of the base variables (dashed) and fiber variables (continuos) for the optimal two-trailers solution.

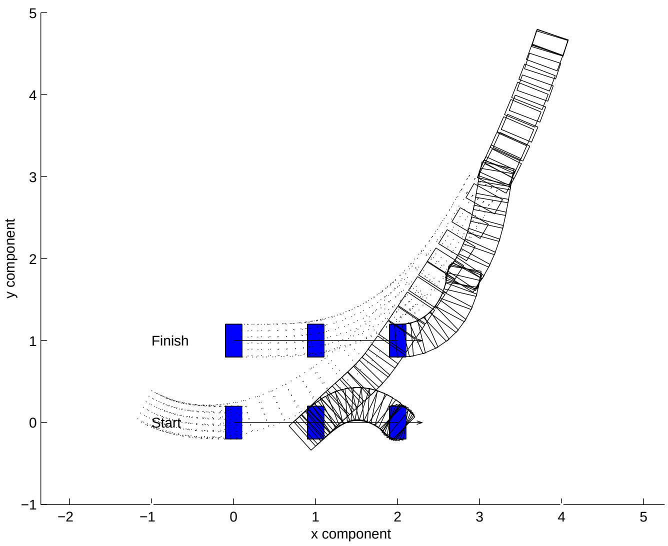

Fig. 11. Optimal trajectory for the two-trailers. The minimal time cost is 12.

of quantized control systems," in *Proc. IEEE Int. Conf. on Decision and Control*, 2000, pp. 3963–3968.

[10] A. Bicchi, A. Marigo, and B. Piccoli, "On the recahability of quantized control systems," *IEEE Trans. on Automatic Control*, May 2002, in press.

[11] O.J. 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.

[12] O.J. Sordalen and O. Egeland, "Exponential stabilization of nonholonomic chained systems," *IEEE Trans. on Automatic Control*, vol. 40, no. 1, pp. 35–49, 1994.

[13] R.M. Murray, "Nilpotent bases for a class of non-integrable distributions with applications to trajectory generation for nonholonomic systems," *Math. Control Signals Systems*, vol. 7, pp. 58–75, 1994.

[14] E. Sontag, "Control of systems without drift via generic loops," *IEEE Trans. on Automatic Control*, vol. 40, no. 7, pp. 1210–1219, 1995.

[15] C. Samson, "Control of chained systems, application to path following and time varying point stabilization of mobile robots," *IEEE Trans. on Automatic Control*, vol. 40, no. 1, pp. 64–67, 1995.

[16] I. Kolmanovsky and N.H. McClamroch, "Developments in nonholonomic control problems," *IEEE Control Systems*, pp. 20–36, December 1995.

[17] S. Sekhavat and J. P. Laumond, "Topological properties for collision free nonholonomic motion planning: the case of sinusoidal inputs for chained form systems," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 5, pp. 671–680, October 1998.

[18] "Ilog cplex user-s guide," Tech. Rep., 1999.

[19] A. Schrijver, *Theory of Linear and Integer Programming*, Wiley Interscience Publ., 1986.

[20] L. A. Wolsey, *Integer Programming*, Wiley Interscience Publ., 1998.

[21] Eric V. Denardo, *Dynamic Programming: Models and Applications*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey,, 1982.