Kinematic Control of Wheeled Mobile Manipulators

B. Bayle*, J.-Y. Fourquet**, F. Lamiroux* and M. Renaud*

* LAAS-CNRS, Toulouse, France, ** ENIT, Tarbes, France

Abstract

We propose a generic scheme to solve the kinematic control problem of wheeled mobile manipulators when the operational motion is imposed. We generalize the Additional Task Method to solve the control problem of these redundant nonholonomic systems.

1 Introduction

The theme of mobile manipulators kinematic control is now a quite mature topic. Several aspects have been dealt with for the last ten years. Many of these works are detailed and classified in [3]. Among the contributions we can distinguish several problems. To simplify, we can separate them into two main classes:

- the ones which solve a *path planning problem*. In that case, the desired solution is a *path*, generally in the configuration space, parameterized by its curvilinear abscissa;
- the ones which solve a *motion planning problem*. In that case, the desired solution is a *motion*, generally in the configuration space, parameterized by time. This requires a realizable control law for the system consistent with its motion constraints.

If we consider only pure motion tasks they can be specified:

- either by the evolution of the whole system: the configuration is imposed;
- or by the evolution of the end effector (grip, tool, from now on denoted by EE): the position and the orientation of the EE are then imposed.

Many tools of control theory have been applied to the generation of mobile manipulators motions, from pure computational methods of optimization [7, 26, 17] – in the case of path planning – to control laws with dynamics [11, 15, 14, 13, 8] in the case of motion planning. Up to now the problem of controlling systems subjected to rolling without slipping (*r.w.s.*) constraints (nonholonomic constraints) has given rise to less contributions. It remains a difficult problem. Thus a kinematic approach appears to be a first step, nevertheless sufficient in the case of indoor robotics, where the mobile manipulators have relatively slow motions.

From a kinematic point of view the problem of motion generation has been also studied by various authors [20, 10, 4]. The difficulty consists in the necessary coordination of the two kinematically different subsystems: the platform and the arm. Whereas the tasks of a mobile manipulator are often described by the EE evolution –

thus concerning manipulation aspects – it is compulsory to control the mobile platform adequately. Two methods are possible: i) to get a solution for the robotic arm joint values, an adequate platform position and orientation, and then to compute a control law to stabilize the system around the corresponding configuration [25]; ii) to compute directly the control inputs from the EE specified motion by an inverse of the adequate kinematic model [20, 10, 4].

In this article we consider the second approach: we synthesize the control inputs of the mobile manipulator to obtain the specified motion of the EE. The proposed scheme is as general as possible and consistent with on-line computation. Our control laws are based on the generic modelling proposed in [5], which we recall in section 2. From this description, we propose to extend Seraji Additional Tasks Method [21]. We thus integrate any number of additional user-defined constraints to the operational task. The different solutions to the problem of control of a redundant mobile manipulator are presented in section 3 and inspired by the work of Nakamura [16]. The main contribution of this paper, compared to previous papers on the same topic, is *to propose a generic approach to control a large class of mobile manipulators*. The second contribution is *to propose methods to express the additional tasks corresponding to real experimental constraints*. Thus the control designer can use purposely redundancy, particularly *to avoid obstacles*. We illustrate the Additional Task Method by a collision free simulation. This simulation is done in a 3D environment and uses an efficient collision detector.

2 Modelling

The modelling of robotic arms has been explored for more than twenty years, and the resulting models – kinematic or dynamic – are generic at least for open chains. This is not the case of the platforms as their structures are very numerous: wheeled platforms, free flying space systems, undersea vehicles, *etc.* If we only consider wheeled platforms, the number of different structures remains quite important, as some systems – outdoor platforms for instance – can be articulated or slip on the ground. To our knowledge, the only wheeled platforms which have led to systematic modelling are those moving on a planar ground, and subjected to *r.w.s.* constraints. They are described and modelled by Campion *et al.* in [6].

It results from this, that a systematic modelling of mobile

manipulators is also difficult. We present in [5] the extension of Campion's work to the case of wheeled mobile manipulators. In the following paragraph we recall the main results of this modelling.

2.1 Description, configuration, location

We consider the general case of a mobile manipulator built from a mobile platform equipped with an on-board robotic arm. This latter is composed of n_a mobile bodies articulated by n_a rotoid and/or prismatic joints. The robotic arm is described by its Denavit-Hartenberg modified parameters [12] which define the location of all its bodies, *i. e.* its whole geometry. The mobile platform has rigid wheels and moves on a planar surface. The number of such structures is limited [6]. The most widely used are the car-like platforms and platforms driven by two independent wheels.

Let $\mathcal{R} = (O, \vec{x}, \vec{y}, \vec{z})$ be any fixed frame. The *mobile manipulator configuration* is known when the position of all its points in $\mathcal{R}$ are known and thus defined by a vector $\mathbf{q}$ of n independent coordinates. The *location of the mobile manipulator* EE is given by m coordinates which define the position and the orientation of the EE in $\mathcal{R}$. We define the position of the EE in $\mathcal{R}$ by the Cartesian coordinates of its center and its orientation with the Euler classical angles.

Models. The usual models of manipulation systems have been modified to take into account the nature of the platform. It results that the models of mobile manipulators are compromises between the models of platforms and those of robotic arms. Particularly, as mobile manipulators are manipulation systems, we write models to describe the location ξ of the EE, which is important from an operational point of view. The *kinematic model* (KM) of a *mobile manipulator* sets the location of its EE as a function of the robotic arm configuration $\mathbf{q}_a$ and of the platform location: $\xi = f(\mathbf{q}_a, \xi_p)$.

At the velocity level the *instantaneous location kinematic model* (ILKM) of a *mobile manipulator* sets the derivative of its location as a function of a set of δ_m parameters of control, which form the *mobile manipulator control vector of mobility* η . These parameters are the control inputs of the system which have an influence on the EE velocity:

$$\dot{\xi} = \bar{J}\eta. \quad (1)$$

The rest of the control parameters of a mobile manipulator consist of the δ_{sp} velocities $\dot{\beta}_s(t)$ of the steering wheels – when they exist – of the platform about their orientation axis. The corresponding vector:

$$\zeta_p = \dot{\beta}_s \quad (2)$$

is called the *mobile platform control vector of steerability*. The *mobile manipulator control vector of manoeuvrability* is given by the set of all the control parameters, *i. e.* by $\mathbf{u} = [\eta^T \zeta_p^T]^T$.

Different features are associated to the control parameters of the system. The *platform degree of steerability* δ_{sp} is

the dimension of vector ζ_p . The *mobile manipulator degree of mobility* δ_m is the dimension of vector η . Finally the *mobile manipulator degree of manoeuvrability* δ_M is the dimension of vector $\mathbf{u}$.

To know the evolution of the mobile manipulator configuration we also define the *mobile manipulator instantaneous configuration kinematic model* (ICKM):

$$\dot{\mathbf{q}} = S\mathbf{u}. \quad (3)$$

It is fundamental to notice that, in general, the dimension m of operational space is less than the degree of mobility δ_m of the mobile manipulator. In this case we recall that the problem, mobile manipulator and task, is redundant. From now on, we assume that it is the case.

3 Kinematic Control

3.1 Problem position

The desired operational motion of a mobile manipulator is an application $(\xi^*(t) \mid t \in [t_0, t_f])$. The problem of control we deal with – often termed as inverse kinematics problem – is to find the control vector of manoeuvrability $(\mathbf{u}(t) \mid t \in [t_0, t_f])$ to achieve the desired operational motion. The corresponding evolution of the whole system is given by the actual generalized motion $(\mathbf{q}(t) \mid t \in [t_0, t_f])$. The aim of this paper is to provide a method to offer the most generic solution to this control problem, thus taking into account the generic modelling of mobile manipulators presented in [1, 5] and recalled in section 2.

In other words we want to propose a servoing scheme so that the actual location $\xi(t)$ of the EE corresponds to the desired location $\xi^*(t)$, *i. e.* to stabilize the error location $e(t) = \xi^*(t) - \xi(t)$. To solve this problem, we suppose that the following elements are known:

- the initial configuration $\mathbf{q}(t_0)$ and thus, in particular, $\mathbf{q}_a(t_0)$ and $\xi_p(t_0)$;
- the desired operational motion $\xi^*(t)$;
- the KM: $\xi(t) = f(\mathbf{q}_a(t), \xi_p(t))$;
- the ILKM: $\dot{\xi}(t) = \bar{J}(t)\eta(t)$;
- and the ICKM: $\dot{\mathbf{q}}(t) = S(t)\mathbf{u}(t)$.

From these data we compute $\xi(t_0) = f(\mathbf{q}_a(t_0), \xi_p(t_0))$. Thus there exist two types of problems:

- a motion following problem when $e(t_0) = \mathbf{0}$;
- a motion tracking problem when $e(t_0) \neq \mathbf{0}$.

We assume that the desired operational motion is achievable by the system, *i. e.* that the desired operational velocities can be obtained in accordance with the mobile manipulator kinematic bounds. An additional constraint is that we want to be able to produce the control inputs of the system at high frequency – typically around 200 Hz – to be consistent with data sampling.

In the case of robotic arms the State of the Art of kinematic control of robotic arms offers several solutions [18, 19] to our problem. Some of them have already been applied to a nonholonomic mobile manipulator built from

a nonholonomic platform with on board robotic arm [20]. We propose to give solutions consistent with a larger class of systems. We will only evoke *inversion methods* which are based on the computation of an ILKM inverse.

3.2 Additional Task Method

Simplest case: the mobile manipulability degree of steerability is equal to zero. In this case the mobile manipulator has no steering wheel since $\delta_{s_p} = 0$. We conclude that the degrees of mobility and of manoeuvrability are equal: $\delta_m = \delta_M$ and consequently $\boldsymbol{\eta} = \mathbf{u}$. The synthesis of the control vector of manoeuvrability $\mathbf{u}$ thus reduces to that of the control vector of mobility $\boldsymbol{\eta}$.

From now on, we simplify notation by hiding – except special cases – the dependence between features.

As the problem is redundant ($m < \delta_m$), we set m_{add} additional tasks in the form:

$$\dot{\boldsymbol{\xi}}_{add} = \bar{J}_{add}\boldsymbol{\eta}. \quad (4)$$

$\bar{J}_{add}$ is thus a matrix of dimension $m_{add} \times \delta_m$. Also we want to regulate $\boldsymbol{\xi}_{add}(t)$ to the velocities of the desired additional tasks $\dot{\boldsymbol{\xi}}_{add}^*(t)$. Then the resulting augmented linear system, obtained from equations (4) and (1) is:

$$\dot{\boldsymbol{\xi}}_t = \bar{J}_t\boldsymbol{\eta}, \quad (5)$$

with: $\bar{J}_t(\mathbf{q}) = \begin{bmatrix} \bar{J} \\ \bar{J}_{add} \end{bmatrix}$ and $\boldsymbol{\xi}_t = \begin{bmatrix} \boldsymbol{\xi} \\ \boldsymbol{\xi}_{add} \end{bmatrix}$.

The problem is now to regulate the actual value of $\boldsymbol{\xi}_t$ to $\boldsymbol{\xi}_t^* = [\boldsymbol{\xi}^{*T} \boldsymbol{\xi}_{add}^{*T}]^T$. Let $\mathbf{e}_t = [e^T e_{add}^T]^T = \boldsymbol{\xi}_t^* - \boldsymbol{\xi}_t$, with $\mathbf{e}_{add} = \boldsymbol{\xi}_{add}^* - \boldsymbol{\xi}_{add}$. Matrix $\bar{J}_t$ is of dimension $(m + m_{add}) \times \delta_m$. It is not necessarily square as it depends on the desired additional tasks that the user may define. Let $\rho = \text{rank}(\bar{J}_t)$, we now suppose – in order to simplify – that $\bar{J}_t$ is of full rank, that is $\rho = \min(m + m_{add}, \delta_m)$. We are going to express the control vector of mobility $\boldsymbol{\eta}$ by solving, exactly or not, the linear system $\dot{\boldsymbol{\xi}}_t^* = \bar{J}_t\boldsymbol{\eta}$ (see appendix A).

• **Case** $\rho = m + m_{add} \leq \delta_m$

Let $D_{\bar{J}_t}$ be any right inverse of $\bar{J}_t$ and $\boldsymbol{\eta}_{\text{Ker } \bar{J}_t}^*$ any element of $\bar{J}_t$ null space. We propose the solution:

$$\boldsymbol{\eta} = D_{\bar{J}_t}(\dot{\boldsymbol{\xi}}_t^* + W_t(\boldsymbol{\xi}_t^* - \boldsymbol{\xi}_t)) + \boldsymbol{\eta}_{\text{Ker } \bar{J}_t}^*, \quad (6)$$

where W_t is $(m + m_{add})$ -order definite positive matrix. The second term represents the *internal motion* of the mobile manipulator. The exponential stability of this control law is guaranteed as:

$$\dot{\boldsymbol{\xi}}_t = \bar{J}_t\boldsymbol{\eta} = \dot{\boldsymbol{\xi}}_t^* + W_t(\boldsymbol{\xi}_t^* - \boldsymbol{\xi}_t),$$

that is $\dot{\mathbf{e}}_t + W_t\mathbf{e}_t = 0$. The error $\mathbf{e}_t(t)$ exponentially decreases to zero, and so does $\mathbf{e}(t)$.

From theorem A.3 (see appendix A), the $(m + m_{add})$ -order matrix $\bar{J}_t\bar{J}_t^T$ is positive definite and its pseudoinverse $\bar{J}_t^+ = \bar{J}_t^T(\bar{J}_t\bar{J}_t^T)^{-1}$ is a right inverse of $\bar{J}_t$. We can

thus choose $D_{\bar{J}_t} = \bar{J}_t^+$. As:

$$\boldsymbol{\eta}_{\text{Ker } \bar{J}_t}^* = (I_{\delta_m} - \bar{J}_t^+\bar{J}_t)\mathbf{g}, \quad \forall \mathbf{g} \in \mathbf{R}^{\delta_m},$$

with I_{δ_m} the δ_m -order identity matrix, we can finally write:

$$\boldsymbol{\eta} = \bar{J}_t^+(\dot{\boldsymbol{\xi}}_t^* + W_t(\boldsymbol{\xi}_t^* - \boldsymbol{\xi}_t)) + (I_{\delta_m} - \bar{J}_t^+\bar{J}_t)\mathbf{g}, \quad \forall \mathbf{g} \in \mathbf{R}^{\delta_m}. \quad (7)$$

The choice of the additional tasks and of vector $\mathbf{g}$ is task-dependent. For instance, it can be used to optimize a criterion.

• **Case** $\rho = \delta_m \leq m + m_{add}$

Let $D_{\bar{J}}$ be any right inverse of $\bar{J}$ and $\boldsymbol{\eta}_{\text{Ker } \bar{J}}^*$ any element of $\bar{J}$ null space. We propose the solution:

$$\boldsymbol{\eta} = D_{\bar{J}}(\dot{\boldsymbol{\xi}}^* + W(\boldsymbol{\xi}^* - \boldsymbol{\xi})) + \boldsymbol{\eta}_{\text{Ker } \bar{J}}^*, \quad (8)$$

where W is m -order definite positive matrix. The second term represents the *internal motion*. The exponential stability of this control law is guaranteed as:

$$\dot{\boldsymbol{\xi}} = \bar{J}\boldsymbol{\eta} = \dot{\boldsymbol{\xi}} + W(\boldsymbol{\xi}^* - \boldsymbol{\xi}),$$

that is $\dot{\mathbf{e}} + W\mathbf{e} = 0$. The error $\mathbf{e}(t)$ exponentially decreases to zero.

From theorem A.2 (see appendix A), the δ_m -order matrix $\bar{J}_t^+\bar{J}_t$ is positive definite and its pseudoinverse $\bar{J}_t^+ = (\bar{J}_t^T\bar{J}_t)^{-1}\bar{J}_t^T$ is a left inverse of $\bar{J}_t$. We set:

$$L = (\bar{J}_t^T\bar{J}_t)^{-1}\bar{J}_t^T(\bar{J}(\bar{J}_t^T\bar{J}_t)^{-1}\bar{J}_t^T)^{-1}, \quad (9)$$

We notice that L is a right inverse of $\bar{J}$. We can thus choose $D_{\bar{J}} = L$. As:

$$\boldsymbol{\eta}_{\text{Ker } \bar{J}_t}^* = (I_{\delta_m} - L(\mathbf{q})\bar{J})\mathbf{g}, \quad \forall \mathbf{g} \in \mathbf{R}^{\delta_m},$$

we can finally write:

$$\boldsymbol{\eta} = L(\dot{\boldsymbol{\xi}}^* + W(\boldsymbol{\xi}^* - \boldsymbol{\xi})) + (I_{\delta_m} - L\bar{J})\mathbf{g}. \quad (10)$$

Remark: the additional tasks are taken into account thanks to L and possibly $\mathbf{g}$.

General case: the mobile manipulability degree of steerability is not equal to zero. In this case the mobile manipulator has δ_{s_p} steering wheels. The synthesis of the control vector of manoeuvrability $\mathbf{u}$ imposes to compute the control vector of mobility $\boldsymbol{\eta}$ and the control vector of steerability $\boldsymbol{\zeta}_p$. In spite of solutions proposed in [1] in the case of mobile manipulators built with a car-like platform, this problem remains open. Indeed, the orientation of the steering wheels can be controlled independently from the EE position. Nevertheless, their orientation is of great importance. If the position and the orientation of the platform is not fundamental to obtain the desired location of the EE the platform must of course remain in a certain proximity with the EE desired position. We are presently studying this aspect of the problem, to achieve a more general framework for mobile manipulators kinematic motion generation.

4 Writing constraints or additional tasks

We consider in this part only the previous simplest case for which $\delta_{s_p} = 0$ and in accordance $\eta = u$.

In the previous section, we have established control methods based on mobility control of mobile manipulators for which the operational motion is imposed. We have shown the advantages of these methods:

- to obtain a desired internal motion,
 - to solve imposed additional tasks,
- within the limits of redundancy and with respect to the task.

This section aims at establishing systematic methods that use redundancy in order to take into account joint limits, manipulability or obstacle avoidance, for instance. It is devoted to:

- determining a desired internal motion by a *gradient method*;
- determining additional tasks by the so-called *constraints method*.

4.1 Internal motions

- **Case** $\rho = m + m_{add} \leq \delta_m$: gradient method

We can use the problem redundancy to propose a coordination strategy for the internal motion. This can be done by a gradient descent method in which the potential function has its minimum value corresponding to the user requirements. Let $\mathcal{P}$ be a scalar function depending on the mobile manipulator configuration q . Using (2) we can write:

$$\dot{\mathcal{P}} = \nabla^T \mathcal{P} \dot{q} = \nabla^T \mathcal{P} S u = \nabla^T \mathcal{P} S \eta,$$

where $\nabla \mathcal{P}$ is the gradient of the function $\mathcal{P}$. If we consider only the internal motion, then:

$$\dot{\mathcal{P}} = \nabla^T \mathcal{P} S (I_{\delta_m} - \bar{J}^+ \bar{J}) g.$$

In order to decrease $\mathcal{P}$ we propose the choice:

$$g = -k (\nabla^T \mathcal{P} S (I_{\delta_m} - \bar{J}^+ \bar{J}))^T,$$

where k is a positive scalar. Indeed, with this choice it is obvious that $\dot{\mathcal{P}} \leq 0$. This approach has been applied with various functions for $\mathcal{P}$ in [2, 4], notably through the use of the mobile manipulator measure of manipulability [4].

- **Case** $\rho = \delta_m \leq m + m_{add}$: additional tasks error minimization

We choose $g = \bar{J}_t^+ \dot{\xi}_t^*$ in (10) such that the proposed solution was the solution – at least for a motion following problem – to the problem:

$$\min_{\eta \in \mathbb{R}^{\delta_m}} \frac{1}{2} \|\dot{\xi}_t^* - \bar{J}_t \eta\|^2,$$

with $\dot{\xi}_t^* - \bar{J}_t \eta = 0$.

Thus it allows to complete the task – the constraint – and minimize the error on the additional tasks. Indeed, those tasks are more than the degree of redundancy and thus cannot be achieved exactly.

4.2 Constraint method

Principle. The constraint method has been proposed by Faverjon and Tournassoud [9]. It allows to produce constraints taking into account obstacles from distance measures. Indeed, the minimization of a function hides a number of problems. The authors want to consider the anti-collision constraints and the task dependent constraints independently in order to avoid lockouts [23, page 99].

This method is based on distance measure between the nearest polyhedra in the workspace – denoted by P_1 and P_2 – and on its derivative. Let d be this distance and C_1 , C_2 the respective points of the polyhedra such that:

$$\|\overrightarrow{C_1 C_2}\| = d.$$

The coordinates of these points are respectively ξ_{C_1} and ξ_{C_2} in $\mathcal{R}$. The configurations of the polyhedra P_1 and P_2 are respectively q_1 and q_2 . Finally, let $\vec{n}$ be the unit vector of the line $(C_1 C_2)$, with components n in $\mathcal{R}$.

We have: $\overrightarrow{C_1 C_2} = d \vec{n}$. As:

$$\overrightarrow{C_1 C_2} \cdot \vec{n} = d \|\vec{n}\|^2 = d, \quad (11)$$

the distance d is such that:

$$d = (\xi_{C_2} - \xi_{C_1})^T n. \quad (12)$$

Equation (12) allows to compute the derivative of the distance, which is always defined. Let us present this calculation in the two main cases:

- 1- In the case C_2 is fixed and C_1 is a point of the mobile manipulator whose ILKM, written in C_1 is given by:

$$\dot{\xi}_{C_1} = \bar{J}_{C_1} \eta. \quad (13)$$

By derivating (11), we obtain:

$$\dot{d} = \dot{\overrightarrow{C_1 C_2}} \cdot \vec{n} + \overrightarrow{C_1 C_2} \cdot \dot{\vec{n}} = \dot{\overrightarrow{C_1 C_2}} \cdot \vec{n} + d \vec{n} \cdot \dot{\vec{n}}.$$

Since $\vec{n}$ is of constant norm, $\vec{n} \cdot \dot{\vec{n}} = 0$ and consequently:

$$\dot{d} = (\dot{\xi}_{C_2} - \dot{\xi}_{C_1})^T n = -\dot{\xi}_{C_1}^T n, \quad (14)$$

that is, from (13) and (14):

$$\dot{d} = -\eta^T \bar{J}_{C_1}^T n. \quad (15)$$

- 2- In the case both C_1 and C_2 are points of a unique mobile manipulator whose ILKM, written in C_1 and C_2 give respectively $\dot{\xi}_{C_1} = \bar{J}_{C_1} \eta$ and $\dot{\xi}_{C_2} = \bar{J}_{C_2} \eta$.

In a similar way we find:

$$\dot{d} = \eta^T (\bar{J}_{C_2}^T - \bar{J}_{C_1}^T) n, \quad (16)$$

In the different cases, equations (15) and (16) give the derivative of the distance. Of course, the calculations can be generalized when several obstacles are mobile and several mobile manipulators are present.

From the expression of the distance derivative $\dot{d}$, we have to express a constraint on the control inputs which allow to avoid collisions between P_1 and P_2 that is such that $d > 0, \forall t \geq 0$. We try to find this constraint in the form of an additional task. We previously established the relations linking the distance derivative to the control vector of mobility η . We now have to impose the evolution of the distance so that it remains positive. To that purpose we define the distance of influence d_i and the distance of security d_s , such that:

- d must always be greater than d_s ;
- if $d \geq d_i$ the evolution of the distance d is not controlled. To avoid collisions between the bodies, $\dot{d}(d_s)$ must be positive. Faverjon and Tournassoud [9] show that the choice:

$$\dot{d} \geq -\dot{d}_i \frac{d - d_s}{d_i - d_s}, \text{ for } d \leq d_i \quad (17)$$

ensures the anti-collision previous conditions.

Equation (17) corresponds to an inequality constraint on the control inputs of the mobile manipulator. The method is: we compute the control inputs without the anti-collision additional task. Then:

- if $\dot{d}$ satisfies (17), we don't add the constraint;
- otherwise we add the additional constraint in the form:

$$\dot{\xi}_a^* = \bar{J}_{add}\eta, \quad (18)$$

where $\dot{\xi}_a^* = \dot{d}$ and $\bar{J}_{add} = \mathbf{n}^T(\bar{J}_{C_2} - \bar{J}_{C_1})$.

4.3 Practical issues

The anti-collision constraint is of course a priority on a robotic system. satisfactory protections to avoid collisions with the environment and collisions between bodies of the robot itself.

We also notice that the collision avoidance constraint is not added when the mobile manipulator is not in the zone of influence of the obstacles or not in a configuration near auto-collision. So in an workspace with few obstacles, the system has a greater redundancy since this constraint doesn't exist anymore.

To write the collision avoidance constraint it is necessary to have an information on the distance between the bodies in the environment. Distance computation is not easy in a general framework. To characterize the position of the different polyhedra in the workspace and moreover to compute distances between them we need a dedicated tool. Particularly for its distance measure and its high performances we chose to use KCD collision detector [24].

A simulation of our control law has been developed with Move3D – LAAS software – [22]. Figure 1 represents the mobile manipulator of LAAS. It is built from a two independently driven wheel mobile platform and a six rotoid joint robotic arm. The desired motion corresponds to a

Cartesian straight line of the EE with adequate velocities – the EE orientation is not specified in this task. Figure 1 (a) represents the collision in motion of the mobile manipulator without taking into account the additional collision avoidance constraint. Figure 1 (b) shows the achievement of the motion with the use of the proposed method.

Figure 1: Additional task for obstacle avoidance with LAAS mobile manipulator

5 Conclusion

In this article, we propose a control scheme to solve on-line the problem of mobile manipulators control when the end effector motion is imposed. As the system is redundant we develop a method termed as the Additional Tasks Method to provide solutions whatever the number of additional user-defined tasks is. Indeed, the different solutions depend on the number of additional tasks compared to the degree of mobility of the mobile manipulator. We explain how to benefit from redundancy, thanks to the combined use of internal motions – gradient descent method – and additional tasks. In particular, we focus on the design of anti-collision additional tasks with the constraint method.

The case of mobile manipulator with platforms having steering wheels is evoked but not developed. It remains one of the main open problems associated to our study.

A Appendix

We recall some theorems and properties of linear systems to help the reader understand the proposed solutions.

Let A be a given $m \times n$ -dimensional matrix with rank r and let $\mathbf{y}$ be a given m -dimensional vector, we want to solve the linear system:

$$A\mathbf{x} = \mathbf{y}, \quad (19)$$

i. e. to find the unknown n -dimensional vector $\mathbf{x}$.

Theorem A.1 $A\mathbf{x} = \mathbf{y}$ has at least one solution $\mathbf{x}$ iff it is consistent, i. e. iff $\text{rank}[A \mid \mathbf{y}] = \text{rank } A$.

Theorem A.2 Let A be $m \times n$ with $m \geq n$, with rank $r = n$. Then, $A^T A$ is positive definite and thus has an inverse, and the pseudoinverse $A^+ = (A^T A)^{-1} A^T$ is a left inverse of A , since $A^+ A = I_n$.

Theorem A.3 Let A be $m \times n$ with $m \leq n$, with rank $r = m$. Then, AA^T is positive definite and thus has an inverse, and the pseudoinverse $A^+ = A^T (AA^T)^{-1}$ is right inverse of A , since $AA^+ = I_m$.

We define the following minimization problem:

$$\min_{x \in \mathbf{R}^n} \frac{1}{2} \|y - Ax\|^2. \quad (20)$$

If the minimal solution is zero, then x is such that $y = Ax$ and is an exact solution of the problem. A necessary and sufficient condition is the consistency of (19). If the minimal solution is not zero, then $y \neq Ax$ and y is a least square solution of (19). Then, system (19) is not compatible.

Theorem A.4 The general solution of (20) is:

$$x = A^+ y + (I_n - A^+ A)z, \quad (21)$$

where z is any vector of $\mathbf{R}^n$.

References

- [1] B. Bayle. *Modélisation et commande cinématiques des manipulateurs mobiles à roues*. PhD thesis, LAAS-CNRS, Toulouse University, 2001.
- [2] B. Bayle, J.-Y. Fourquet, and M. Renaud. A coordination strategy for mobile manipulation. In *IAS'2000*, pages 981–988, Venice, Italy, July 2000.
- [3] B. Bayle, J.-Y. Fourquet, and M. Renaud. Génération des mouvements des manipulateurs mobiles : Etat de l'art et perspectives. *Journal Européen des Systèmes Automatisés*, 35(6):809–845, 2001.
- [4] B. Bayle, J.-Y. Fourquet, and M. Renaud. Using manipulability with nonholonomic mobile manipulators. In *International Conference on Field and Service Robotics*, pages 343–348, Helsinki, Finland, June 2001.
- [5] B. Bayle, J.-Y. Fourquet, and M. Renaud. Kinematic modelling of wheeled mobile manipulators. In *Submitted to IROS'2002*, 2002.
- [6] G. Campion, G. Bastin, and B. D'Andréa-Novél. Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. *IEEE Transactions on Robotics and Automation*, 12(1):47–62, February 1996.
- [7] W. F. Carriker, P. K. Khosla, and B. H. Krogh. Path planning for mobile manipulators for multiple task execution. *IEEE Transactions on Robotics and Automation*, 7(3):403–408, June 1991.
- [8] R. Colbaugh. Adaptive stabilization of mobile manipulators. *Journal of Robotic Systems*, 15(9):511–523, 1998.
- [9] B. Faverjon and P. Tournassoud. A local based approach for path planning of manipulators with a high number of degrees of freedom. In *ICRA'87*, Raleigh, April 1987.
- [10] G. Foulon, J.-Y. Fourquet, and M. Renaud. Control of a rover-mounted manipulator. In *ISER'97*, pages 96–107, Barcelona, Spain, June 1997.
- [11] N. A. M. Hootsmans and S. Dubowsky. Large motion control of mobile manipulators including vehicle suspension characteristics. In *ICRA'91*, pages 2336–2341, Sacramento, USA, April 1991.
- [12] W. Khalil and J. Kleinfinger. A new geometric notation for open and closed loop robots. *ICRA'86*, pages 75–79, 1986.
- [13] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, A. Casal, and A. Baader. Force strategies for cooperative tasks in multiple mobile manipulation systems. In *International Symposium of Robotics Research*, Munich, Germany, October 1995.
- [14] K. Liu and F. Lewis. Control of mobile robot with onboard manipulator. In *ISRAM'92*, pages 539–546, Santa Fe, USA, November 1992.
- [15] W. Miksch and D. Schroeder. Performance based controller design for a mobile manipulator. In *ICRA'92*, pages 227–232, Nice, France, May 1992.
- [16] Y. Nakamura. *Advanced Robotics, Redundancy and Optimization*. Addison-Wesley Publishing, 1991.
- [17] F. G. Pin and J.-C. Culioli. Optimal positioning of combined mobile platform-manipulator systems for material handling tasks. *Journal of Intelligent and Robotic Systems*, 6:165–182, 1992.
- [18] L. Scialvico and B. Siciliano. *Modelling and Control of Robot Manipulators*. Springer-Verlag, 2000.
- [19] H. Seraji. Configuration control of redundant manipulators: Theory and implementation. *IEEE Transactions on Robotics and Automation*, 5(4):472–490, 1989.
- [20] H. Seraji. An on-line approach to coordinated mobility and manipulation. In *ICRA'93*, pages 28–35, Atlanta, USA, May 1993.
- [21] H. Seraji. A unified approach to motion control of mobile manipulators. *The International Journal of Robotics Research*, 17(2):107–118, February 1998.
- [22] T. Siméon, J.-P. Laumond, and F. Lamiroux. Move3d: a generic platform for path planning. In *IEEE International Symposium on Assembly and Task Planning*, pages 25–30, Fukuoka, Japon, May 2001.
- [23] P. Tournassoud. *Géométrie et intelligence artificielle pour les robots*. Traité des Nouvelles Technologies, Série Robotique. Hermès, 1988.
- [24] C. Van Geem and T. Siméon. Kcd: a collision detector for path planning in factory models. Technical Report 01073, LAAS-CNRS, Toulouse, France, March 2001.
- [25] Y. Yamamoto and X. Yun. Coordinating locomotion and manipulation of a mobile manipulator. *IEEE Transactions on Robotics and Automation*, 39(6):1326–1332, June 1994.
- [26] M. Zhao, N. Ansari, and E. S. H. Hou. Mobile manipulator path planning by a genetic algorithm. *Journal of Robotic Systems*, 11(3):143–153, 1994.