

Fast Obstacle Evaluation of the Configuration Space for a Construction Towercrane

B. Curto V. Moreno R. Theron J. Blanco
 Dept. Computers and Automation
 U. of Salamanca
 Salamanca, 37008, Spain
 e-mail: {control,roberto,jblanco}@abedul.usal.es

Abstract

In this paper it is shown a practical method which lets evaluate the full set of crane configurations that produce collisions between the construction environment and the crane and its load. The method is based on a sound mathematical formalism so the feasibility of the results is guaranteed. Moreover, a suitable application of the formalism is performed in this work by means of choosing a suitable coordinate functions, in the workspace, as well as in the configuration space. It leads to the fact that the obstacle representation in the crane configuration space could be seen as a convolution of two functions that describe the crane-robot and the obstacles respectively. Additionally, the computational load is independent of the shape and number of obstacles and of the robot shape. The algorithmic tool that has been used is FFT (Fast Fourier Transform), since it eases the parallel implementation of the resulting algorithms reducing significantly the computational load, so the computational time is highly optimized.

1 Introduction

A towercrane is an invaluable tool in the construction field as it performs the manipulation of heavy loads, doing this in a little amount of time, if we compare it with the effort and waste of time that workers would have to make. These characteristics contribute to improve work conditions and quality of buildings and to decrease costs. For that reason concerted efforts have been made with regard to design and manufacture of towercranes with better mechanical properties. These efforts have been traditionally focused into two objectives: to increase both the lifting capacity and the accessibility of the load to be manipulated. Nevertheless, few efforts have been centred in automating the operations of transport of materials, notwithstanding the advantages that this would report. In this sense it is necessary to emphasise the security, in terms of damages to the constructed elements, and the optimisation of the transport operations.

From the point of view of Robotic, a towercrane can be associated with a robotic manipulator with a cylindrical cinematic structure, with a control of the teleoperator type and making operations of pick-and-place. The evolution of Robotic in other scopes and tasks has

provided to manipulation more powerful strategies of control, incorporating a greater degree of intelligence. In this context the advances in the planning and generation of sequences of free collision movements are specially remarkable. This problem can be solved either in the workspace or in the configuration space (C-space), where the position and the orientation of the robot are characterised with only one point. In it each coordinate represents a degree of freedom in the position or the orientation of the robot. This idea of reducing the robot to a point in an appropriate space was introduced by Udupa [6], although the term *space of the configurations* had still not been used. Later, Lozano-Perez [10] adopted the notion of *space of the configurations* of the Mechanics and popularised it in the path planning. Lozano-Perez [11] proposes to divide this task in two stages:

1. to calculate the set of safe configurations, denominated *findspace*,
2. to calculate a sequence of configurations from the set of safe ones, denominated *findpath*.

The work that appears in this article takes care fundamentally of the first stage. This approach, applied to the problem of transporting heavy objects with a towercrane, would suppose to find the set of positions and orientations of the towercrane that produced collisions with the present objects and constructions in the workspace. The results generated in this stage would be used later by a planner, which would calculate the sequence of movements to go from the origin and to the goal without collisions with the hook nor with the cable.

Most of methods that explicitly evaluate C-space [12] [13] [14] [2] consider either a polygonal or polyhedral mobile object in a workspace, either R^2 or R^3 , in the presence of polygonal or polyhedral obstacles. But only a few of them [13] [14] [12] are concerned with the study of articulated manipulators. In any case, the computational time, associated to the obstacle projection from the workspace to the C-space, depends on the number of vertices of each obstacle and also of the robot shape.

In [2] the obstacle representation in the C-space was carried out by means of the convolution of an

obstacle and a mobile robot. This idea is implicitly considered in [11], where for a mobile robot A , a obstacle in the C-space due to B (C-obstacle) is defined as $CO_A(B) \equiv \{x \in Cspace_A / (A)_x \cap B \neq \emptyset\}$. In [7] is stated that this operation can be seen as the convolution of A with B . Later, [8] shows, through the Convolution Theorem, that this calculation is independent of the complexity and shape of the robot and the obstacles. This idea is the basis of the work presented in this paper, that proposes a mathematical formalism for the representation of the obstacles in C-space by means of convolution of two different functions: one describing the robot A and the other describing the obstacles, if a proper coordinate system is chosen. This formalism can be established for articulated manipulators with two and three degrees of freedom in a workspace R^2 or R^3 .

The algorithms for path planning that have been developed in the last years, propose to work in discrete C-spaces [1]. More concisely, a bitmap is used to represent the obstacles in C-space, reducing drastically the computation time to detect collisions. The derivation of the C-space bitmap can be done in two different ways: either by discretizing the C-space once it has been calculated analytically [9], or by direct computation of a discrete representation of the C-obstacle. Kavraki [8] proposes the derivation of the C-space bitmap as the convolution of the obstacle bitmap and the robot bitmap in the workspace. In order to carry out this convolution efficiently, Fast Fourier Transform is used. This technique provides bitmaps with higher accuracy and it is independent of the obstacles number and shape, and the robot shape [8]. It only depends on the discretization resolution.

But in [8] only 2D mobile robots are studied. In [3] [4] the proposed method is applied to 2D and 3D mobile robot leading to the same conclusions as in [8], and to a planar revolute manipulator. In both cases, it is necessary to choose a proper coordinate system to work with, considering the particular robot mechanical structure and relating it with the particular degrees of freedom. In this way the existence of the convolution between the robot and the obstacles, over one or more coordinates, is guaranteed. In this paper it will be shown that the formalism can be applied to towercranes with quite successful results. So, a fast procedure can be designed to obtain the set of free-collision configurations for the crane. And this way the method will be applicable to changing environments, which are common situations at the construction world.

The remainder of this paper is organised as follows. In section 2, the mentioned mathematical formalism is proposed, where a function that allow to know whether the robot at a given configuration collides (or not) with the obstacles previously is defined. In the section 3, the problem is stated in order to describe the mechanical structure of the robot and to propose the workspace and C-space parameterizations. Next, section 4 shows that by choosing an adequate coordinate system for the robot, the calculation for the

mentioned function is equivalent to the convolution of the robot and the workspace over certain variables. In section 5, a new algorithm for the C-space calculation, by using FFT, is proposed and implemented for the crane. Finally, in section 6 and 7 the main results and conclusions are presented.

2 Proposed Mathematical Formalism

In this section a new C-obstacle evaluation is proposed as the integral of two functions product: one that represents the robot and the other one that represents the obstacles in the workspace.

Let W and C be the workspace and configuration space (C-space). The first function A that represents the robot at the configuration $q \in C$ for each point $x \in W$ is given by:

Definition 1 Let $A(q)$ be the subset of W that represents the robot at the configuration q . The function $A : C \times W \rightarrow R$ is defined by:

$$A(q, x) = \begin{cases} 1 & \text{if } x \in A(q) \\ 0 & \text{if } x \notin A(q) \end{cases} \quad (1)$$

The second function B is associated to the obstacles in the workspace and it is given by the following:

Definition 2 Let B be the subset of W formed by the obstacles. The function $B : W \rightarrow R$ is defined by:

$$B(x) = \begin{cases} 1 & \text{if } x \in B \\ 0 & \text{if } x \notin B \end{cases} \quad (2)$$

By considering the previous definitions a new function is proposed to compute the C-obstacles:

Definition 3 Let $CB : C \rightarrow R$ be the function defined by

$$CB(q) = \int A(q, x)B(x)dx \quad \forall q \in C \quad (3)$$

The region CB_f is defined as:

$$CB_f = \{q \in C / CB(q) > 0\}$$

As is stated at [5] this definition is equivalent to the proposed by [9].

Theorem 1 Let $CB = \{q \in C / A(q) \cap B \neq \emptyset\}$ the definition of C-obstacles region that is given by [9]. Then it is verified that:

$$CB = CB_f$$

From this result, it is straightforward that $q \in C_{free} \Leftrightarrow CB(q) = 0$, where the C_{free} is the set of free-collision configurations.

So, in order to know whether the robot A at a given configuration q collides with the obstacles or not it is necessary to evaluate $CB(q)$ and, furthermore, the value of $A(q, x)$. Next, the formalism is going to be applied to a construction towercrane and it is going to be shown the appropriate choice of frame and coordinate functions that lets get a fast procedure to compute this function.

3 Problem Statement

The towercrane's structure can be modeled as a kinematic chain constituted by three rigid bodies (A_0 , A_1 , A_2) with three degrees of freedom. The first *dof* is associated with the rotation (θ_1) of the jib (A_1) around the base A_0 . The second one is the displacement (d_2) along the jib and the last one is related to the load vertical movement (d_3). Both, the cable and the load, constitute the third element (A_2) and it depends on the two displacements (d_2 and d_3). The base does not have movement capabilities so it is no considered for a collision check. In the figure 1 the robot elements and the manipulator *dofs* can be seen.

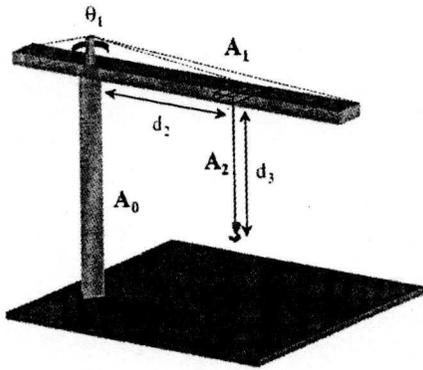


Figure 1: Crane mechanical structure

As it is described in [5], the main goal is to choose a set of proper frames and the suitable coordinate functions. To be exact, a fixed frame (F_W) for the workspace and three moving frames (F_{A_1} , F_{A_2} and F_{A_3}) for the links have been considered and they are shown in figure 2.

A configuration for the robot is a specification for the position and orientation of the frames (F_{A_1} , F_{A_2} and F_{A_3}) in relation to F_W . So, a configuration will be parameterized by the 3-tuple $(\theta_1, d_2, d_3) \in [0, 2\pi] \times [0, L] \times [0, H]$, where L is the length of jib and H is the height of the crane.

As it will be shown, a very suitable coordinate functions choice for the workspace W is the cylindrical one (r, φ, z) . In this way, the variations of the *dof* θ_1 are related with the variations of coordinate φ and the same relation is found with the pair d_3 and z . With this relationships, and using the formalism proposed at [5], the evaluation of the function $CB(q) = CB(\theta_1, d_2, d_3)$ can be performed applying the convolution theorem and as a result the computation time is reduced drastically. In the next section the procedure will be shown in order to obtain the expression to be computed.

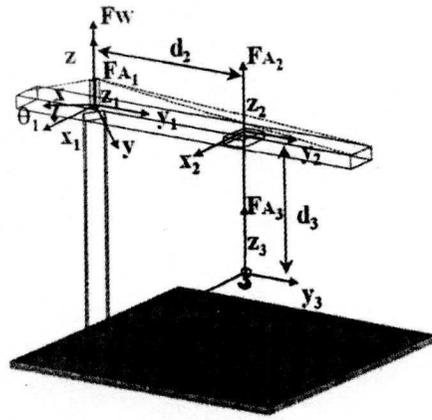


Figure 2: Suitable frames for the towercrane

4 Formalism Application

The function $B : W \rightarrow R$, with the previous considerations, will be given by:

$$B(r, \varphi, z) = \begin{cases} 1 & \text{if } (r, \varphi, z) \in B \\ 0 & \text{if } (r, \varphi, z) \notin B \end{cases}$$

By considering the mechanical manipulator structure, the function $A(\theta_1, d_2, d_3, r, \varphi, z)$ can be decomposed into two terms as follows:

$$A(\theta_1, d_2, d_3, r, \varphi, z) = A_1(\theta_1, r, \varphi, z) + A_2(\theta_1, d_2, d_3, r, \varphi, z) \quad (4)$$

In this way, the expression to evaluate, in order to obtain the set of configurations that produces collisions with obstacles, is:

$$CB(\theta_1, d_2, d_3) = \int A(\theta_1, d_2, d_3, r, \varphi, z) B(r, \varphi, z) dr d\varphi dz$$

The main problem is the evaluation of the function A for each value of the configuration because it produces a very high computational load. Following the formalism proposed at [5], the functional dependencies for the A function can be reduced due to relationships between the coordinates (φ, z) and the *dofs* (θ_1, d_3) . So, it can be stated that:

$$A_1(\theta_1, r, \varphi, z) = A_1(0, r, \varphi - \theta_1, z) \quad (5)$$

$$A_2(\theta_1, d_2, d_3, r, \varphi, z) = A_2(0, d_2, 0, r, \varphi - \theta_1, z - d_3)$$

In the following it will be denoted:

$$A_{1(0)}(r, \varphi - \theta_1) \delta(z) = A_1(0, r, \varphi - \theta_1, z)$$

$$A_{2(0, d_2, 0)}(r, \varphi - \theta_1, z - d_3) = A_2(0, d_2, 0, r, \varphi - \theta_1, z - d_3)$$

So, using (4 and 5), $CB(q)$ can be calculated by:

$$CB(\theta_1, d_2, d_3) = CB_1(\theta_1) + CB_2(\theta_1, d_2, d_3)$$

where

$$CB_1 = \int A_{1(0)}(r, \varphi - \theta_1) B(r, \varphi, 0) dr d\varphi \quad (6)$$

$$CB_2 = \int A_{2(0,d_2,0)}(r, \varphi - \theta_1, z - d_3) B(r, \varphi, z) dr d\varphi dz$$

If the two functions convolution definition is considered the two previous expressions (6) can be written as:

$$CB_1(\theta_1) = \int (\bar{A}_{1(0)} * B)_{\varphi}(r, \theta_1, 0) dr \quad (7)$$

$$CB_2(\theta_1, d_2, d_3) = \int (\bar{A}_{2(0,d_2,0)} * B)_{(\varphi,z)}(r, \theta_1, d_3) dr$$

where

$$\bar{A}_{1(0)}(r, \theta_1 - \varphi) = A_{1(0)}(r, \varphi - \theta_1)$$

$$\bar{A}_{2(0,d_2,0)}(r, \theta_1 - \varphi, d_3 - z) = A_{2(0,d_2,0)}(r, \varphi - \theta_1, z - d_3)$$

The expression (7) is fundamental at the C-obstacle computation. First, it has been reduced the computational load because it is only necessary to evaluate the function $A_{1(0)}(r, \varphi)$ (i.e at the configuration $\theta_1 = 0$) and the set of functions $A_{2(0,d_2,0)}(r, \varphi, z)$ (i.e for each value of d_2). Secondly, the convolution theorem can be applied in order to perform the C-obstacle computation. So, the Fourier transform of the function CB is:

$$\mathcal{F}[CB] = \int \mathcal{F}[\bar{A}_{1(0)}(r, \theta_1)]_{\varphi} \mathcal{F}[B(r, \theta_1, 0)]_{\varphi} dr + \quad (8)$$

$$\int \mathcal{F}[\bar{A}_{2(0,d_2,0)}(r, \theta_1, d_3)]_{(\varphi,z)} \mathcal{F}[B(r, \theta_1, d_3)]_{(\varphi,z)} dr$$

In the next section an algorithm to perform this calculation will be described.

5 Algorithm

In this algorithm a set of bitmaps to represent the crane and the objects in the workspace are used. A main characteristic of the representation is the resolution N . If the convolution is used, the running time will be $O(N^d)$, being N the discretization resolution and d the number of convolution variables. Another choice is to apply the convolution theorem and the FFT algorithm, so the running time will be $O(N^d \log N)$. In order to implement this algorithm, the functions have to be periodical but, at both instances, the involved bitmaps are no cyclic. Kavraki [8] shows how to solve this problem: placing a bit "1" at every workspace contour elements. So, it is ensured that robot can not "wrap around" the W limits.

Taking into account the expression (8), next, it is presented the algorithm to compute the C-space

Construct $W(r, \varphi, z)$ using cylindrical coordinates

Construct $A_{1(0)}(r, \varphi)$ using polar coordinates

$P_1 T = 0$

For each radius

 Compute the one-dimensional $\mathcal{F}[W(r, \varphi, 0)]$

 Compute the one-dimensional $\mathcal{F}[A_{1(0)}(r, \varphi)]$

 Let $P_1 = \mathcal{F}[W] \cdot \mathcal{F}[A_{1(0)}]$

 Accumulate at $P_1 T$

Let $IP_1 T = \mathcal{F}^{-1}[P_1 T]$ (the inverse 1D FFT)

For each d_2

 Construct $A_{2(0,d_2,0)}(r, \varphi, z)$

$P_2 T = 0$

 For each radius

 Compute the two-dimensional $\mathcal{F}[W(r, \varphi, z)]$

 Compute the two-dimensional $\mathcal{F}[A_{2(0,d_2,0)}]$

 Compute $P_1 = \mathcal{F}[W] \cdot \mathcal{F}[A_{2(0,d_2,0)}]$

 Accumulate at $P_2 T$

 Let $IP_2 T = \mathcal{F}^{-1}[P_2 T]$ (the inverse 2D FFT)

$IP_2 T = IP_2 T + IP_1 T$

 Let $CB(\theta_1, d_2, d_3) = 1$ iff $|IP_2 T| > 0$

The discretization is performed with the same resolution N , at both spaces W and C . Note that two-dimensional FFT is used, so the algorithm running time is $O(N^4 \log N)$.

6 Results

A Silicon Graphics Power Challenge XL platform, with four processors MIPS R8000, has been used to implement the algorithm. As well, it makes use of a mathematical paralleled library.

To validate the algorithm a typical construction workspace with a building was considered. It can be seen in the figure 3 as well as the cylindrical representation of the same workspace is shown in figure 4, which is necessary for the C-space evaluation. The towercrane manipulator is the same that appears in the figure 1 and its dimensions were 32m high, a 60m jib and a 3m radius cylindrical hook-load. The dimensions of the workspace used were 100m \times 100m where the towercrane was located in the middle.

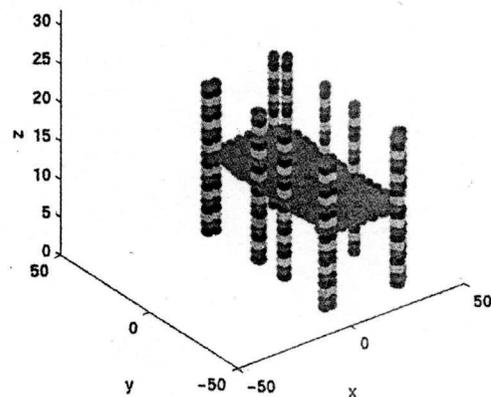


Figure 3: A construction workspace (cartesian coordinates)

In figure 5, the algorithm output is shown. Note that the pillars become thicker according to hook dimensions and the configurations under the first floor

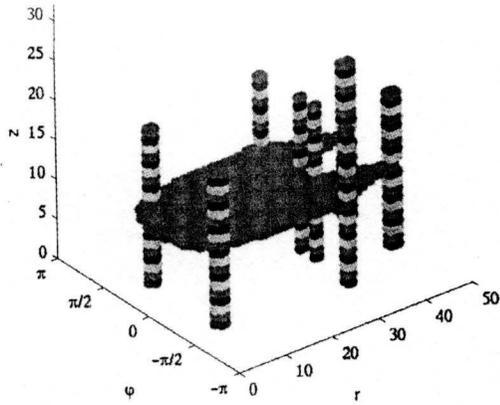


Figure 4: A construction workspace (cylindrical coordinates)

produce collisions with the cable. Also, a free-collision path is shown for a pick-and-place operation. Considering 64 resolution bitmaps, the running times are around one minute, which seems to be a suitable result for the construction environment. Nevertheless, the results could be optimized by means of the algorithm parallelization or/and the utilization of the *ad hoc* hardware.

7 Conclusions

In this paper, a general mathematical formalism for C-space computation has been used in order to compute the set of configurations that produces collisions between a towercrane and its environment. The proposed expression for C-space evaluation has been simplified due to a correct choice of coordinate functions, that is, cylindrical. Consequently, crane C-space could be seen as a convolution of two functions that describe the crane-robot and the obstacles, respectively.

By applying the convolution theorem the FFT algorithms can be used, therefore, the parallel implementation of the resulting algorithms was achieved. And as a result the computational time was highly optimized. Since the inputs to the algorithm are bitmaps, its running time is independent of the shape and number of obstacles and of the robot shape. These features bring the advantage to detect collisions between the crane and the construction elements in a changing environment.

Finally, the resulting C-spaces can be efficiently used as inputs to path planning procedures or to perform security tasks in the construction environment, such as alarm activation or security stops, when a collision is detected.

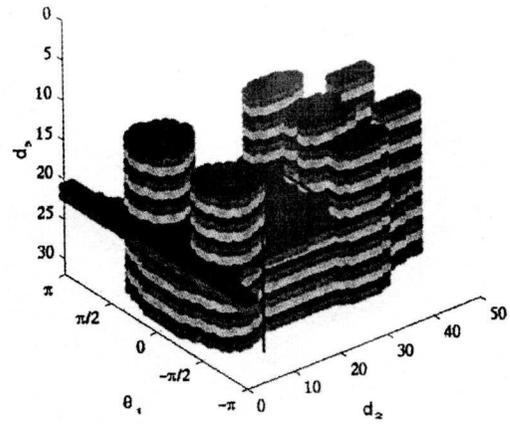


Figure 5: A construction workspace (cylindrical coordinates)

Acknowledgments

The authors gratefully acknowledge the support of the Castilla y León Council through the project C02/197.

References

- [1] J. Barraquand and J. C. Latombe, "Robot Motion planning: a distributed representation approach," *Int. J. of Robotics Research*, Vol. 10, No. 6, 1991.
- [2] R. C. Brost, "Computing metric and topological properties of configuration-obstacles," *Proc. of the IEEE Conf. on Rob. and Autom.*, pp. 170-176, 1989.
- [3] B. Curto, V. Moreno, "Mathematical formalism for the fast evaluation of Configuration Space," *Proc. of the IEEE Int. Symposium on Comput. Intel. in Rob. and Autom.*, 1997.
- [4] B. Curto, P. Vega, V. Moreno, "Fast procedure to obstacle representation in the Configuration Space for mobile robots," *Proc. of the IFAC Workshop on Int. Comp. for Vehicles*, 1998.
- [5] B. Curto, "Mathematical formalism for obstacle representation in the Configuration Space for robots," *PhD dissertation, Univ. Salamanca*, 1998.
- [6] S. Udupa, "Collision Detection and avoidance in controlled manipulators," *PhD dissertation, Cal. Inst. Techn.*, 1977.
- [7] L. Guibas, L. Ramshaw and J. Stolfi, "A kinetic framework for computational geometry," *Proc. of the IEEE Conf. on Rob. and Autom.*, 1985.
- [8] L. E. Kavraki, "Computation of configuration-space obstacles using the Fast Fourier Transform,"

IEEE Tr. on Rob. and Autom., Vol. 11, No. 3, 1995.

- [9] J. C. Latombe, *Robot motion planning*, Kluwer Academic Publishers, Boston, MA, 1991.
- [10] T. Lozano-Pérez, "Automatic planning of manipulator transfer movements," *IEEE Tr. on Systems, Man and Cyber.*, Vol. 11, No. 10, 1981.
- [11] T. Lozano-Pérez, "Spatial planning: a configuration space approach," *IEEE Tr. on Comp.*, 1983.
- [12] T. Lozano-Pérez, "A simple motion-planning algorithm for general robot manipulators," *IEEE Journal of Rob. and Autom.*, Vol. 3, No. 3, 1987.
- [13] A. A. Maciejewski and J. J. Fox, "Path planning and the topology of configuration space," *IEEE Tr. on Rob. and Autom.*, Vol. 9, No. 4, 1993.
- [14] W. Newman and M. Branicky, "Real-time Configuration space transforms for obstacle avoidance," *The Int. J. of Robotics Res.*, Vol. 10, No. 6, 1991.