ABCM Symposium Series in Mechatronics - Vol. 3 - pp.316-325

A COLLISION AVOIDANCE METHOD USING ASSUR VIRTUAL CHAINS

Henrique Simas, hsimas@univali.br
Universidade do Vale do Itajaí - Centro de Ensino São José
Curso de Engenharia de Computação
Rodovia SC 407, km 4 - Sertão do Imaruim
88122 000 - São José, SC. Brasil

Daniel Fontan Maia da Cruz, danielfontan@yahoo.com.br
Raul Guenther, guenther@emc.ufsc.br
Daniel Martins, daniel@emc.ufsc.br
Universidade Federal de Santa Catarina
Departamento de Engenharia Mecânica
Laboratório de Robótica
Campus Universitário - Trindade
88040-900 – Florianópolis, SC. Brasil

Abstract. *In hydroelectric power plants, the rotor blades are eroded by the cavitation process. The erosion results in craters that are usually recovered by a manual welding process. The ROBOTURB project developed an automatized system, where a robot is used for recovery the rotor blade surfaces by welding process. The robot workspace is constrained by freeform surfaces and the collision imminency is constant. For this reason, the robot is redundant, making possible the collisions avoidance, keeping the end-effector trajectory tracking. In recent years, the collision avoidance problem has been dealt with algorithms based on artificial intelligence. This article considers a methodology for implementation a deterministic collision avoidance algorithm. The Solution of this proposal is based on the method of kinematic constraint and on the use of Assur virtual chains. The method consists in the obstacle identification and in the definition of a free collison workspace area. Thus, it is used a Assur virtual chain to detect and avoid the collision by evaluating the distance between a point at the robot and the obstacle. The Assur virtual chain is responsible for maintaining the robot inside the free collision workspace. This process is carried out for each point of the desired trajectory.*

Keywords: *Differential kinematics, Assur virtual chains, Screw theory, Collision avoidance, Redundant robots*

1. INTRODUCTION

For many reasons, such as safety and economics, it is desirable to develop robots able to perform more complex tasks with minimal or no human intervention. A kind of these complex task is that in which the robot should avoid collisions autonomously.

An example of this practical situation occurs in the Roboturb project, which aims to develop an automatized system for the repair of regions eroded by cavitation in the rotors of hydraulic turbines. The use of this system intends to eliminate the unhealthy work and to improve the welding quality in the repair. The central element of the system is a robot to operate in the confined space of the canal of the rotor, measuring the eroded region and performing the recovery welding. To reach this goal, the robot movements should be precise and the manipulator links cannot collide with the surfaces of the blades, i.e., the robot workspace is constrained. To make it possible to avoid collisions while the tasks are executed we designed a 7-DOF redundant robot (Guenther et al., 2000). This work was initially developed to solve the collision avoiding related to that project.

Numerous methods have been proposed to achieve collision-free path planning, where the mainly application is for mobile robots. However, in the case of manipulator arms, the structure of the robot adds to the complexity of the solution (Soucy and Payeur, 2005).

The numerous approaches for collision avoidance available on the literature are based basically on the potential fields' method, fuzzy logic or neural networks.

Potential fields method is based on defining an attractive field, which directs the robot towards its objective, and a repulsive field that pushes the robot away from obstacles. Many researches have been developed using this approach (see (Muller, 2004), (Caselli *et al*, 2002) and (Piaggio and Sgorbissa, 1999)). However, in general these approaches do not offer a solution that can be used with generic robot architectures (Soucy, 2005).

The methods based on fuzzy logic and neural networks are more complex methods, which are normally combined with potential fields concepts, as can be seen in ((Soucy and Payeur, 2005), (Soucy, 2005) and (Tian and Mao, 2002)). These approaches are considered as non-deterministic methodologies where it is not possible to proof the stability.

The main contribution of this paper is to introduce a new method to avoid collision based on the use of kinematic constraints (Simas, 2005) and Assur virtual chains (Campos et al., 2005).

To present the method we shortly reproduce three fundamental kinematic tools used in the paper: the theory of differential kinematics based on the screw representation; the Davies Method; and the Assur virtual chain concept (see (Davies, 1981) and (Bonilla, 2004) for more details). In the sequence, we present the proposed method and an application example based on Roboturb project robot.

2. FUNDAMENTAL KINEMATIC TOOLS

Our approach is based on the screw representation of differential kinematics, on the Davies method, and on the Assur virtual chain concept, which are shortly presented in this section.

2.1. Screw representation of differential kinematics

The Mozzi theorem states that the general spatial differential motion of a rigid body consists of a differential rotation about, and a differential rotation along a axis named instantaneous screw axis (see Ceccarelli (2000)). In this way the velocities of the points of a rigid body with respect to an inertial reference frame *O-xyz* may be represented by a differential rotation ω about the instantaneous screw axis and a simultaneously differential translation τ about this axis. The complete movement of the rigid body, combining rotation and translation, is called screw movement or twist and is here denoted by $\$$. Figure 1 shows a body "twisting" around the instantaneous screw axis. The ratio of the linear velocity and the angular velocity is called pitch of the screw $h = \|\tau\| / \|\omega\|$.

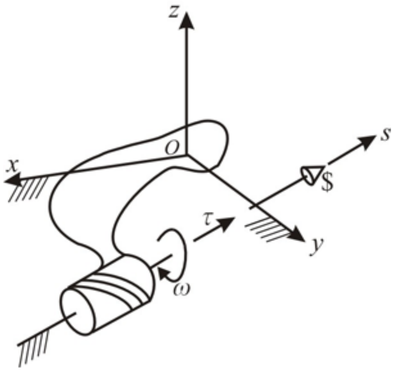

Figure 1. Screw movement or twist.

The twist may be expressed by a pair of vectors, i.e. $\$ = \left[\omega^T ; V_p^T\right]^T$, where ω represents the angular velocity of the body with respect to the inertial frame, and V_p represents the linear velocity of a point *P* attached to the body which is instantaneously coincident with the origin *O* of the reference frame. A twist may be decomposed into its amplitude and its corresponding normalized screw. The twist amplitude, denoted as $\dot{q}$ in this work, is either the magnitude of the angular velocity of the body, $\|\omega\|$, if the kinematic pair is rotative or helical, or the magnitude of the linear velocity, $\|V_p\|$, if the kinematic pair is prismatic. The normalized screw, $\hat{\$}$, is a twist in which the magnitude is factored out, i.e.

$$\$ = \hat{\$}\dot{q} \tag{1}$$

The twist given in Eq. (1) expresses the general spatial differential motion movement (velocity) of a rigid body with respect to an inertial reference frame *O-xyz*. The twist could also represent the movement between two adjacent links of a kinematic chain. In this case, the twist $\$_i$ represents the movement of link *i* with respect to link $(i-1)$.

2.2. Davies method

Davies method is a systematic way to relate the joint velocities in closed kinematic chains. Davies (Davies, 1981, 2000) derives a solution to the differential kinematics of closed kinematic chains from the Kirchhoff circulation law for electrical circuits. The resulting Kirchhoff-Davies circulation law states that "The algebraic sum of relative velocities of kinematic pairs along any closed kinematic chain is zero" (Davies, 1981).

We use this law to obtain the relationship among the velocities of a closed kinematic chain as in Campos et al. (2005) and in Santos et al. (2006). In this way, considering that the velocity of a link with respect to itself is null, the circulation law could be expressed as

$$\sum_{i=1}^{n} \$_i = 0 \tag{2}$$

where 0 is a vector which dimension corresponds to the dimension of the twist $\$_i$.

According to the above introduced normalized screw definition this equation may be rewritten as

$$\sum_{i=1}^{n} \hat{\$}_i \dot{q}_i = 0 \tag{3}$$

where $\hat{\$}_i$ and $\dot{q}_i$ represent the normalized screw and the magnitude of twist $\$_i$, respectively.

Equation (3) is the constraint equation which, in general could be written as

$$N\dot{q} = 0 \tag{4}$$

where $N = \lfloor \hat{\$}_1 \quad \hat{\$}_2 \quad ... \quad \hat{\$}_n \rfloor$ is the network matrix containing the normalized screws which signs depend on the screw definition in the circuit orientation, and $\dot{q} = [\dot{q}_1 \quad \dot{q}_2 \quad ... \quad \dot{q}_n]$ is the magnitude vector.

In case the kinematic chain has more than one closed kinematic chain, say *k* closed kinematic chains, the Kirchhoff-Davies circulation law is applied to each one of the chain closed circuits. In this case the network matrix has *k* rows of screws, each one corresponding to a closed circuit.

A closed kinematic chain as actuated joints, here named primary joints, and passive joints, here named secondary joints. The constraint equation, Eq. (4), allows calculate the secondary joint velocities as functions of the primary joint velocities. To this end the constraint equation is rearranged highlighting the primary and secondary joint velocities. So, Eq. (4) can be written as follows:

$$\lfloor N_p \quad \vdots \quad N_s \rfloor \begin{bmatrix} \dot{q}_p \\ ... \\ \dot{q}_s \end{bmatrix} = 0 \tag{5}$$

where N_p and N_s are the primary and secondary network matrices, respectively, and $\dot{q}_p$ and $\dot{q}_s$ are the corresponding primary and secondary magnitude vectors, respectively.

So, the secondary magnitude vector is given by:

$$\dot{q}_s = -N_s^{-1} N_p \dot{q}_p \tag{6}$$

2.3. The Assur virtual kinematics chain concept

The Assur virtual kinematic chain concept, virtual chain for short, is essentially a tool to obtain information about the movement of a kinematic chain or to impose movements on a kinematic chain (Campos et al. (2005)).

This concept was first introduced by Campos (2004) which defines the virtual chain as a kinematic chain composed of links (virtual links) and joints (virtual joints) satisfying the following three properties: a) the virtual chain is open; b) it has joints whose normalized screws are linearly independent; and c) it does not change the mobility of the real kinematic chain, in other words, it is an Assur group (Baranov, 1985).

The two useful virtual chains to describe movements in bi-dimensional space are the *PPR* and the *RPR* chain.

2.4.1 The orthogonal PPR Assur virtual chain

The PPR virtual chain is composed by two virtual links (*C1*, *C2*) connected by two prismatic joints, whose movements are in the *x* and *y* orthogonal directions, and a rotational joint, whose the movement is in the *z* directions, see Fig.2. The prismatic joints are called *px* and *py*, and the rotative joint is called *rz*.

The first prismatic joint (*px*) and the rotative joint (*rz*) are attached to the chain to be analyzed (real chain). Joint *px* connects the link *R1* with virtual link *C1*, joint *py* connects virtual link *C1* with virtual link *C2*, and joint *rz* connect the virtual link *C3* with real link *R2* (see Fig.2).

Let the twist $\$_{px}$ represent the movement of link *C1* in relation to link *R1*, twist $\$_{py}$ represent the movement of link *C2* in relation to link *C1*, twist $\$_{rz}$ represent the movement of link *R2* in relation to link *C2*. Therefore, the movement of link *R2* in relation to real link *R1* may be expressed by $\$_{px} + \$_{py} + \$_{rz}$.

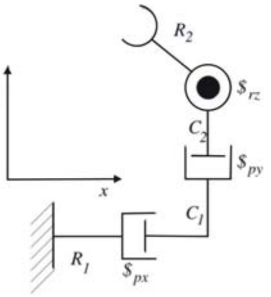

Figure 2. PPR Assur virtual chain.

Consider the C-reference system (*C-system*) attached to the virtual link *C2* at the *rz* joint. Therefore, there is no rotation between the *C-system* and the *B-system* (attached to the inertial base), and the *rz* joints are aligned with *z* axes. So, the normalized screws corresponding to the virtual joints represented in the *C-system* are

$$^{C}\hat{\$}_{rz}=\begin{bmatrix}1\\0\\0\end{bmatrix};\,^{C}\hat{\$}_{px}=\begin{bmatrix}0\\1\\0\end{bmatrix};\;^{C}\hat{\$}_{py}=\begin{bmatrix}0\\0\\1\end{bmatrix} \tag{7}$$

It may be observed that the orthogonal PPR Assur virtual chain represents the movements in a planar Cartesian system.

2.4.2 The orthogonal RPR Assur virtual chain

The RPR virtual chain is composed by one virtual links (*C1,C2*) connected by two rotatives joints, whose movements are in the *z* directions, and a prismatic joint, whose the movement is in the radial direction, see Fig.3. The prismatic joints is called *pr*, and the rotatives joints are called rz_1 and rz_2.

The two rotatives joints (rz_1 and rz_2) and the prismatic joint (*pr*) are attached to the chain to be analyzed (real chain). Joint rz_1 connect the link *R1* with virtual link *C1*, joint *pr* connects virtual link *C1* with virtual link *C2*, and joint rz_2 connect the virtual link *C3* with real link *R2* (see Fig.3).

Let the twist $\$_{rz1}$ represent the movement of link *C1* in relation to link *R1*, twist $\$_{pr}$ represent the movement of link *C2* in relation to link *C1*, twist $\$_{rz}$ represent the movement of link *R2* in relation to link *C2*. Therefore, the movement of link *R2* in relation to real link *R1* may be expressed by $\$_{rz1}+\$_{pr}+\$_{rz2}$.

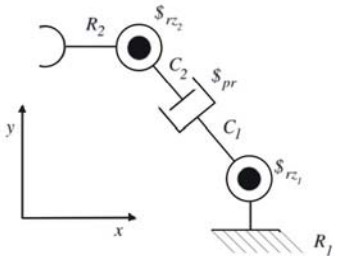

Figure 3. RPR Assur virtual chain.

Consider the C-reference system (*C-system*) attached to the virtual link *C2* at the rz_2 joint. Therefore, there is a rotation between the *C-system* and the *B-system* (attached to the inertial base). So, the normalized screws corresponding to the virtual joints represented in the *C-system* are

$$^{C}\hat{\$}_{rz1}=\begin{bmatrix}1\\0\\q_{pr}\end{bmatrix};\,^{C}\hat{\$}_{pr}=\begin{bmatrix}0\\1\\0\end{bmatrix};\;^{C}\hat{\$}_{rz2}=\begin{bmatrix}1\\0\\0\end{bmatrix} \tag{8}$$

where the q_{pr} is the *pr* joint displacement .

It may be observed that the orthogonal PPR Assur virtual chain represents the movements in a planar Cartesian system.

Other Assur virtual chains can be found in Campos (2004) and Campos et al. (2005).

3. PROPOSED METHOD

The proposed method consists in introduce an Assur virtual chain between the collision point of the manipulator and the obstacle, which will be used to monitor the obstacle approximation and to impose movement in order to avoid collision. To this end we define two different stages: *monitoring* and *avoidance*.

In monitoring stage we intend to identify when the position of a joint (or joints) of the virtual chain is near to a limit given by the obstacle position. This joint position is obtained by the integration of differential kinematic where this virtual joint is obtained as a secondary joint. When the position of the virtual joint indicates the proximity of the obstacle the stage is changed to the avoidance stage.

In the avoidance stage the joint used to monitor the collision possibility (as a secondary joint) is taken as a primary joint in order to impose a movement that avoids the collision. This changes the closed kinematic chain.

To present the method, a 2D avoiding collision application is considered in the sequence.

A 2D avoiding collision application

Let a planar robot *P3R*, that represent a planar version of the robot of ROBOTURB project, composed by 5 links and four joins. The links are numerated from *1*, at the base, to *5* on end-effector; and the joins are identified from *A* to *D*, see Fig.4. Let an obstacle defined by a point in the workspace as in Fig.4.

The *A* joint is a prismatic joint and it movement is represented by the screw $\$_A$. The joins *B*, *C* and *D* are rotational joints and its movements are represented by the screws $\$_B$, $\$_C$, e $\$_D$, respectively.

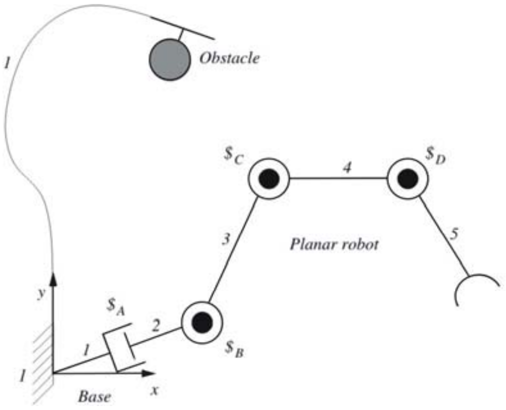

Figure 4. Planar ROBOTURB

Inverse differential kinematics without obstacle avoidance

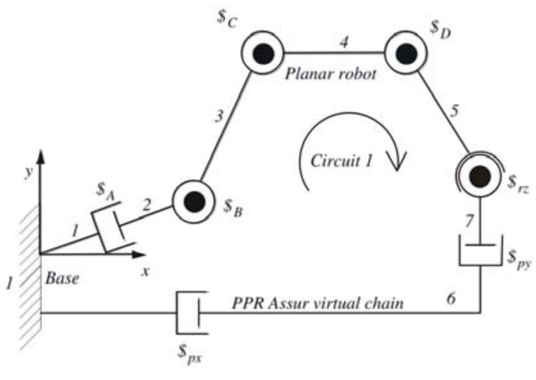

Figure 5. Closed kinematic chain

To solve the inverse differential kinematics we introduce a *PPR* Assur virtual chain to close the kinematic chain (Fig.5) as in Campos et al. (2005), and apply the Davies method to determine the kinematic constraint equation (Eq. (4)),

$$N_p = -\left[\$_{px} \quad \$_{py} \quad \$_{rz}\right] \;\; ; \;\; \dot{q}_p = \left[\dot{q}_{px} \quad \dot{q}_{py} \quad \dot{q}_{rz}\right]^T \tag{9}$$

$$N_s = \left[\$_A \quad \$_B \quad \$_C \quad \$_D\right] \;\; ; \;\; \dot{q}_s = \left[\dot{q}_A \quad \dot{q}_B \quad \dot{q}_C \quad \dot{q}_D\right]^T \tag{10}$$

To solve the differential kinematics the secondary matrix has to be inverted (see Eq. (6)). In Eq. (10) it has more columns than rows, which characterizes a redundant manipulator. So, a method to solve redundant manipulators has to be used. In this example we use the method of kinematic constrains introduced in Santos et al. (2006). According the method of kinematic constraints we impose the movement of joint A and, consequently, take it as a primary joint instead secondary as before.

So, we obtain a new kinematic constraint equation given by:

$$\overline{N}_p = -\left[\$_{px} \quad \$_{py} \quad \$_{rz} \quad \$_A\right] \;\; ; \;\; \dot{\overline{q}}_p = \left[\dot{q}_{px} \quad \dot{q}_{py} \quad \dot{q}_{rz} \quad \dot{q}_A\right]^T \tag{11}$$

$$\overline{N}_s = \left[\$_B \quad \$_C \quad \$_D\right] \;\; ; \;\; \dot{\overline{q}}_s = \left[\dot{q}_B \quad \dot{q}_C \quad \dot{q}_D\right]^T \tag{12}$$

Now, even the matrix $\overline{N}_s$ is invertible, the secondary velocity vector $\dot{\overline{q}}_s$ could be calculated using Eq. (6), Eq. (11) and Eq. (12) and giving the primary trajectory to be executed.

Inverse differential kinematics with obstacle avoidance

To describe the method proposed in this paper it is considered that we intend to avoid the collision of the point corresponding to the link 3 with the obstacle depicted in the Fig. 4.

To this end, the first step is to introduce an Assur virtual chain connecting the manipulator's collision point (in link 3) and the obstacle. As could be observed in Fig.6, the obstacle is considered as an extension of link 1. We chose a *RPR* virtual chain, described in section 2, because it allows to monitor or to impose the limit to the radial distance directly. As we wish to avoid the obstacle collision during a simultaneous trajectory execution, we should consider a virtual chain attached to the end effector, introduced to this end as before. The resulting kinematic chain is shown in Fig.6.

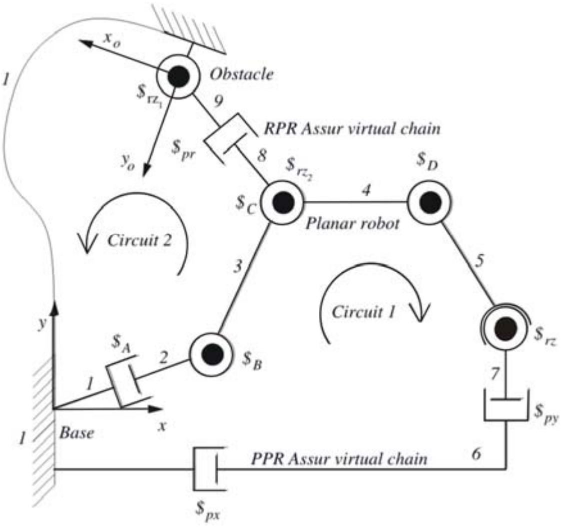

Figure 6. Planar robot PRRR connected with Assur virtual chains PPR and RPR (to collision avoidance)

The kinematic chain shown in Fig.6 has two closed circuits. The robot chain and the PPR virtual chain used to define the end-effector desired movement compose the circuit 1. The partial robot chain until link 3 and the RPR virtual chain form the circuit 2, taken into account that the obstacle point is an extension of link 1.

Applying the Davies method we determine the kinematic constraint equation (Eq. (4)) where:

$$N = \begin{bmatrix} \hat{\$}_A & \hat{\$}_B & \hat{\$}_C & \hat{\$}_D & 0 & 0 & 0 & -\hat{\$}_{rz} & -\hat{\$}_{px} & -\hat{\$}_{py} \\ \hat{\$}_A & \hat{\$}_B & 0 & 0 & -\hat{\$}_{rz1} & -\hat{\$}_{pr} & -\hat{\$}_{rz2} & 0 & 0 & 0 \end{bmatrix}_{6\times 10} \tag{13}$$

$$\dot{q} = \begin{bmatrix} \dot{q}_A & \dot{q}_B & \dot{q}_C & \dot{q}_D & \dot{q}_{rz1} & \dot{q}_{pr} & \dot{q}_{rz2} & \dot{q}_{rz} & \dot{q}_{px} & \dot{q}_{py} \end{bmatrix}^T \tag{14}$$

Now, in monitoring stage, we choose joints *px*, *py*, *rz* and *A* as the primary ones as before. So, the resulting secondary joint velocity vector is calculated using Eq. (6), where:

$$\left[N_p\right]_m = \begin{bmatrix} \hat{\$}_A & -\hat{\$}_{px} & -\hat{\$}_{py} & -\hat{\$}_{rz} \\ \hat{\$}_A & 0 & 0 & 0 \end{bmatrix} \;\; ; \;\; \left[\dot{q}_p\right]_m = \begin{bmatrix} \dot{q}_A & \dot{q}_{px} & \dot{q}_{py} & \dot{q}_{rz} \end{bmatrix}^T \tag{15}$$

$$\left[N_p\right]_m = \begin{bmatrix} \hat{\$}_B & \hat{\$}_C & \hat{\$}_D & 0 & 0 & 0 \\ \hat{\$}_B & 0 & 0 & \hat{\$}_{rz1} & \hat{\$}_{pr} & \hat{\$}_{rz2} \end{bmatrix} \;\; ; \;\; \left[\dot{q}_s\right]_m = \begin{bmatrix} \dot{q}_B & \dot{q}_C & \dot{q}_D & \dot{q}_{rz1} & \dot{q}_{pr} & \dot{q}_{rz2} \end{bmatrix}^T \tag{16}$$

Note that the secondary matrix is square and even it is invertible the secondary vector $\left[\dot{q}_s\right]_m$ could be calculated using Eq. (6). So, the prismatic virtual joint velocity could be obtained. By integrating $\dot{q}_{pr}$ we obtain the prismatic joint position q_{pr}, which is compared with a given limit, until it reaches this limit. Then the avoidance stage begins.

In the avoidance stage, we choose the radial virtual joint *pr* as a primary one, in order to use it to impose the collision avoiding movement. At the same time, to maintain the resulting secondary matrix square, the robot joint *A* velocity is taken as a secondary one. So, joints *px*, *py*, *rz* and *pr* are the primary ones in the avoidance stage. Consequently, the resulting secondary joint vector is calculated using Eq. (6), in which:

$$\left[N_p\right]_a = \begin{bmatrix} \hat{\$}_{pr} & -\hat{\$}_{px} & -\hat{\$}_{py} & -\hat{\$}_{rz} \\ 0 & 0 & 0 & 0 \end{bmatrix} \;\; ; \;\; \left[\dot{q}_p\right]_a = \begin{bmatrix} \dot{q}_{pr} & \dot{q}_{px} & \dot{q}_{py} & \dot{q}_{rz} \end{bmatrix}^T \tag{17}$$

$$\left[N_p\right]_a = \begin{bmatrix} \hat{\$}_B & \hat{\$}_C & \hat{\$}_D & 0 & \hat{\$}_A & 0 \\ \hat{\$}_B & 0 & 0 & \hat{\$}_{rz1} & \hat{\$}_A & \hat{\$}_{rz2} \end{bmatrix} \;\; ; \;\; \left[\dot{q}_s\right]_a = \begin{bmatrix} \dot{q}_B & \dot{q}_C & \dot{q}_D & \dot{q}_{rz1} & \dot{q}_A & \dot{q}_{rz2} \end{bmatrix}^T \tag{18}$$

The secondary matrix $\left[N_s\right]_a$ is square and even it is invertible the secondary vector $\left[\dot{q}_s\right]_a$ could be calculated using Eq. (6). So, the whole robot joints movement, i.e., the joint velocity magnitudes $\dot{q}_A$, $\dot{q}_B$, $\dot{q}_C$ and $\dot{q}_D$, is determined satisfying the trajectory to be executed and the collision avoiding requirements.

4. NUMERICAL SIMULATION

To ilustrate the method we present a numerical simulation of the robot *P3R* working in a constraint environment defined by a punctual obstacle.

4.1 Simulation description

Consider the planar robot P3R in Fig. 7 whose link lengths are link$_3$ = link$_4$ = 2.0*m* and link$_5$ = 1.0*m*. The length of link$_2$ is the displacement of joint *A*. The axis of prismatic joint *A* is rotated in the plane *xy* by an angle equal to $\theta/4\,rad$ about *z* axis. The initial posture is $q_0 = \begin{bmatrix} 1.36m & 0.92rad & -1.33rad & -0.64rad \end{bmatrix}$, corresponding to the end-effector location: $p = \begin{bmatrix} 4.5 & 0.9 \end{bmatrix}^T$ m, $\phi = \left[-\pi/3\right]$rad (see Fig.7).

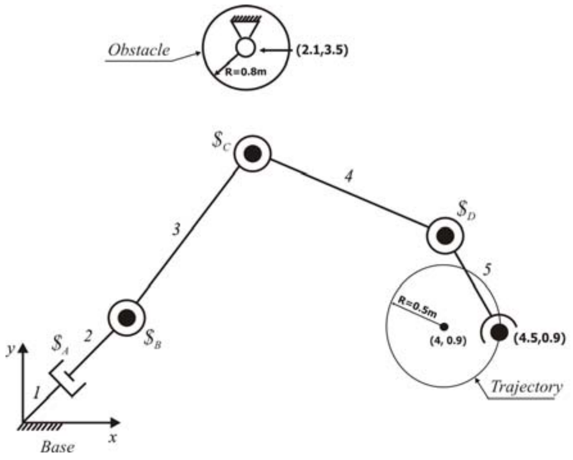

Figure 7. Planar robot PRRR – Numerical simulation

The motion trajectory is defined by a circular path with *0.5m* radius and the center point on *(4.5,0.9)m* (Fig.7) specified by:

$$p_d(t) = \begin{bmatrix} 4 + 0.5\cos\left(\frac{\pi t}{2}\right) \\ 0.9 + 0.6\sin\left(\frac{\pi t}{2}\right) \end{bmatrix} \;;\; \phi_d(t) = \left[-\frac{\pi}{3} - 0.4\sin\left(\frac{\pi t}{2}\right) \right] \tag{19}$$

The collision region is defined by a circle path with *0.8m* radius and the center point on *(2.1,3.5)m*; and the integration is performed in discrete time by resorting to numerical techniques, as depicted in (Guenther, 2007).

4.2 Graphical results

Considering the aspects presented on section 4.1, Fig. 8 and Fig. 9 shows the results of inverse kinematic solution of robot *P3R*, based on the method proposed.

On the monitoring stage (Fig. 8 and Fig. 9), the solution is obtained using the Eq. (15) and Eq. (16), and on the avoidance stage, the solution is obtained using the Eq. (17) and Eq. (18).

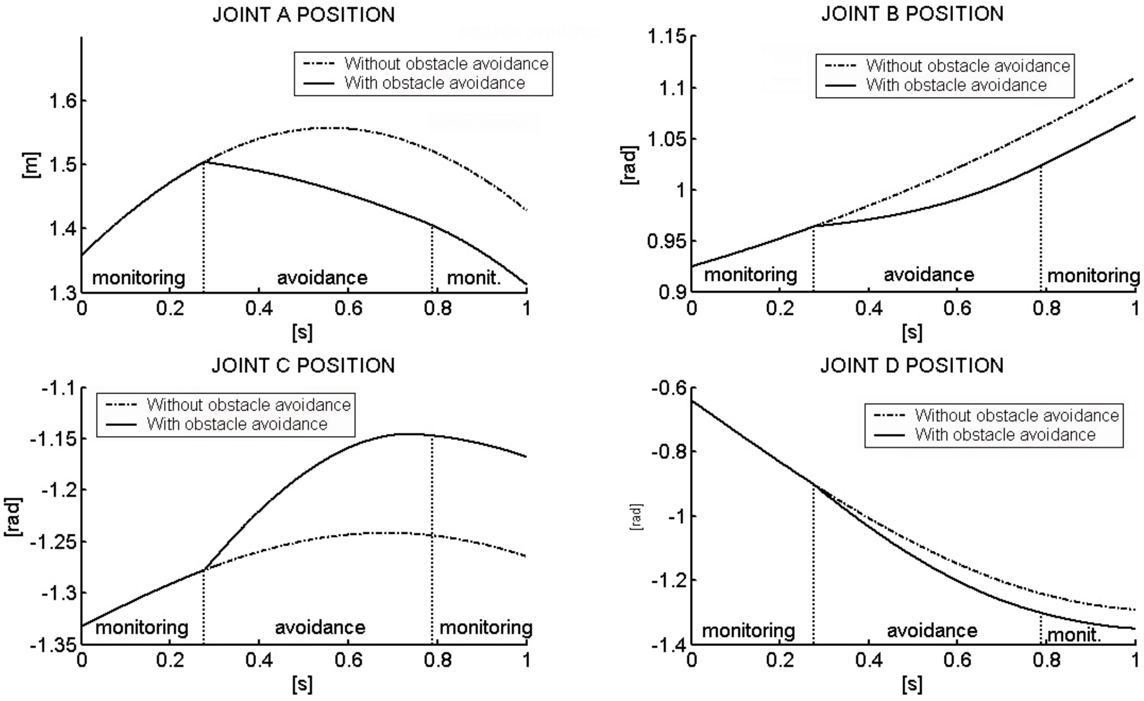

Figure 8. Inverse kinematic solution of *P3R* robot – Real joints.

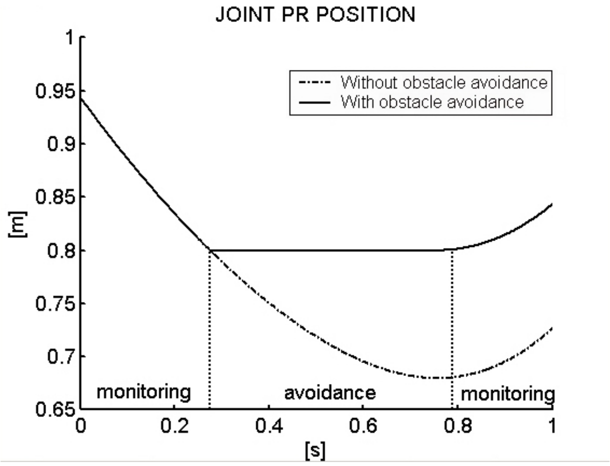

Figure 9. Inverse kinematic solution of *P3R* robot – Joint P_r

5. CONCLUSION

This paper presents a new methodology to solve the avoiding collision problem in robot manipulators. The proposed methodology is based on the method of kinematic constraint, of the screw theory, of Kirchhoff-Davies circulation law, and on the use of Assur virtual chains.

The collision avoidance methodology was applied on a *PRRR* redundant manipulator, that's the planar version of a 7 DOF manipulator, called ROBOTURB.

The performed simulation clearly demonstrates the usefulness of the proposed methodology.

6. ACKNOWLEDGEMENTS

This work was supported by by 'Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPQ)' Brazil.

7. BIBLIOGRAPHY

Bottema, O. O., Roth, B., 1979, "Theoretical kinematics", New York: North-Holand Pub. Co., pp. 533-544. ISBN 0444851240.

Campos, A,"Cinemática Diferencial de Manipuladores Empregando Cadeias Virtuais". 157 p. Tese(Doutorado em Engenharia Mecânica) — Universidade Federal de Santa Catarina, Florianópolis, Março 2004.

Campos, A., Guenther, R. and Martins, D., 2005, "Differential kinematics of serial manipulators using virtual chains", J. Brazilian Soc. Mechanical Sciences & Engineering, Vol. 27, No. 4, pp. 345-356.

Caselli S., Reggiani, M. and Sbravati, R., "Parallel path planning with multiple evasion strategies". In: .Washington, DC: Proceedings of the IEEE Internacional Conference on Robotics and Automation, 2002. p. 260–266.

Ceccarelli, M., 2000, "Screw axis defined by Giulio Mozzi in 1763 and early studies on helicoidal motion", Mechanism and Machine Theory, Vol. 35, No. 6, pp. 761-770.

Davidson, J. K., Hunt, K. H., 2004, "Robots and screw theory: applications of kinematics and statics to robotic", New York: Oxford University Press Inc. ISBN 0-19-856245-4.

Davies, T. H. ,"Kirchhoff's circulation law applied to multi-loop kinematic chain". Mechanism and Machine Theory, v. 16, p. 171–173, 1981.

Guenther, R., Simas, H. and Pieri, E. R. ,"Concepção cinemática de um manipulador para volumes de trabalho restrito", In: Congresso Nacional de Engenharia Mecânica, 2000, Natal/RN. 2000.

Guenther, R., Cruz, D. F. M., Martins, D., Simas, H., 2007, "An integration method for differential inverse kinematic of closed chain robots",

Muller, A. ,"Collision avoiding continuation method for the inverse kinematics of redundant manipulators". In: .New Orleans, LA: Proceedings of the IEEE Internacional Conference on Robotics & Automation, 2004. v. 2, p.1593–1598.

Piaggio, M. and Sgorbissa, A. ,"Ai-cart: an algorithm to incrementally calculate artificial potencial fields in real-time". In: . Monterey, CA: Proceedings of the IEEE International Conference on Robotics and Automation, 1999.p. 238–243.

Santos, Carlos H. F. dos, Guenther, Raul, Martins, Daniel et al, 2006 “Virtual kinematic chains to solve the underwater vehicle-manipulator systems redundancy”, J. Brazilian Soc. Mechanical Sciences & Engineering, Vol. 28, No. 3, pp. 354-361.

Soucy, M., “Manipulator Path Planning with Multi-Resolution Potencial Fields and Fuzzy Logic Control”. Dissertação (Mestrado), University of Ottawa, March 2005.

Soucy M. and Payeur, P. ,“Flexible fuzzy logic control for collision-free manipulator operation”. In: . Niagara Falls, Canada: Mechatronics and Automation, 2005 IEEE International Conference, 2005. v. 2, p. 723–728.

Tian, L. and Mao, Z. ,“Fuzzy neuro controller for a two-link rigid-flexible manipulator system”. In: Singapore: Proceedings of the International Conference on Neural Information Processing, 2002. p. 1867–1871.

Tsai, L., “Robot analysis: the mechanics of serial and parallel manipulators”, John-Wiley and Sons, New York, 1999.

8. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.