SMOOTH TRANSITIONS IN TRAJECTORY PROFILES FOR REDUNDANT ROBOTS PERFORMING SECONDARY TASKS

Henrique Simas, simas@emc.ufsc.br
Daniel Fontan Maia da Cruz, danielfontan@yahoo.com.br
Daniel Martins, daniel@emc.ufsc.br
Departamento de Engenharia Mecânica - Universidade Federal de Santa Catarina

Abstract. *The kinematics model for redundant robot can be based in a classical solution using Denavit-Hartanberg or, more recently, using the screw theory. The model based on screw theory associated with Davies method for redundant robot results in set of solutions which is always dimensional consistent; the same cannot be said from solutions bases on Moore-Penrose pseudoinverse Jacobians. More recently, some works have shown the use of virtual chains to construct a strategy of trajectory generation for redundant robots satisfying an additional task: avoidance collision. However the proposed solution has problems for high joint velocities movements. This disadvantage is characterized by a discontinuity on velocities and aceleration funtions when the robot is under imminence of a collision. This paper proposes an algorithm to smooth the transitions of joint velocities and accelerations. The proposed method is based on polynomial functions. The proposed solution is verified in a* $P3R$ *redundant manipulator.*
Keywords: *Redundant robot, Avoidance collision , Screw theory, Davies Method, Smooth transaction*

1. INTRODUCTION

A challenge in robots application in an industrial environment is the development of robots able to adapt to the workplace performing tasks with minimal or no human intervention.

Robot autonomy requires a capability to perform secondary tasks. For instance, a robot has to weld on a piece but also need to avoid collision with an obstacle in its work volume without compromising the welding procedure (Simas *et al.*, 2004). For a robot to perform one or more secondary tasks, it is necessary to have additional degrees of mobility or, in other words, the robot must be redundant.

Redundant robots are classically defined as robots that have more degrees of freedom (DOF) than the task requires. If a task requires a number r of movements and the robot permits n possible movements, the redundancy is given by $n - r$, ie, the robot has $n - r$ degrees of redundancy. (Simoni *et al.*, 2009)

The literature provides several methodologies for solving direct and inverse kinematics for redundant robots (Siciliano *et al.*, 2009). These methodologies are mainly based on the pseudoinverse matrices and null space algorithms. These methodologies have limitations as metric problems and in the control over the resulting postures for the robot (Campos *et al.*, 2009).

Some works present numerical solutions for the inverse kinematics of robots based on the method of kinematic constraint, on the Davies method, on the screw theory and on the uses of Assur virtual kinematic chains (Simas *et al.*, 2009). These works introduced the concept of error virtual chain for parallel mechanisms. This method was applied in practice on the Roboturb manipulator (Simas *et al.*, 2004) and has proved its effectiveness (Simas *et al.*, 2009). However, there was a limitation: discontinuities in the positions and velocities when abrupt changes occur in the strategy of trajectories.

The previous method introduces secondary tasks that are activated under certain conditions of operation. For a change in the activities in the task operations, discontinuities may occur in position and velocity profiles in the joint space. For the example here discussed, if a redundant robot, when approaching of an obstacle, the trajectory generator changes the strategy of movements of the robot, contouring the obstacle and thus avoiding the collision. The changes of operations implies changing the structure of the Jacobian matrix (Simas *et al.*, 2009). The new structure of the Jacobian changes the directions of velocities on joint space. This abrupt change of velocities characterizes these discontinuities. That solution proposed in (Simas *et al.*, 2009) is therefore only suitable for processes with low velocities of the end-effector, such as most current welding and painting processes. By using the same strategy in tasks that may require relatively higher velocities and accelerations, failures and damage may occur in the robot's mechanical systems. In this case a new form of generation trajectories should be developed to avoid such discontinuities.

This paper presents a method for smooth transition in velocities in inverse kinematics algorithms for redundant robots. The proposed methodology operates in offline mode dealing with the discontinuities in transitions that occur with the change of strategy for trajectory generation. The transients are then smoothed through the use of parameterized curves. This solution allows to impose positions, velocities and accelerations at the beginning and at the end of each transitional section. This paper presents the theoretical aspects of the development of methodology and to validate the proposal, a experiment was developed for a $P3R$ redundant manipulator.

2. Inverse kinematics for redundant robots operating in confined environments

In robotics, several methods are described in the literature dealing with obstacle avoidance. Most of them are applied to mobile robots. In the case of manipulators, the kinematic structure of the robot increases the complexity of the solution (Soucy and Payeur, 2005). Among the main methods used are highlighted as follows: methods based on potential fields, methods based on fuzzy logic, and methods based on neural networks.

The methods based on potential fields have been most studied in the past decade due to its relative simplicity and efficiency (Soucy and Payeur, 2005). Potential fields, used in robotics, generally consist of fields of attraction, that direct the robot to its goal, and repulsion fields, which the robot away from obstacles. The solution approaches that use these concepts are very varied, as can be seen in (Muller, 2004), (Barraquand and Latombe, 1990), (Caselli and Sbravati, 2002), (Chang, 1996), (Laliberte and Gosselin, 1994) and (Piaggio, 1999). Generally these approaches do not offer a solution for generic robot architecture (Soucy and Payeur, 2005), moreover, the characteristic non-deterministic of this method does not guarantee the kinematic control problems (such as singularities).

Methods based on fuzzy logic and neural networks are more complex and usually are combined with the use of the concepts of potential fields, as can be seen in (Soucy, 2005) (Nedungadi, 1992) and (Tian and Mao, 2002). In (Soucy and Payeur, 2005) is presented an approach to path planning of manipulators with collision avoidance where potential fields are combined with discrete fuzzy logic. This work used mainly two phases: global path planning, which determines the trajectory of the end effector, and a phase of local planning, which determines an optimal configuration of the rest of the manipulator arm to avoid a collision.

A systematic solution to the problem of collision avoidance is presented by (Simas, 2008). This method has limitations and provided the basis for the solution proposed here in this paper. The next section presents this method pointing its limitations and the theoretical approaches for its development.

3. Collision avoidance by switching the differential kinematic model

Recently a method based on the screw theory produced an suitable solution for the treatment of collision for redundant robots operating in confined environments (Simas, 2008). The same approaches are also used for the proposed smoothing method in this paper. The proposed method combines Davies method with Assur virtual chains and graph notation. In following we present summarized the screw theory, the Davies method, the Assur virtual chains and graph notation.

3.1 Screw theory

The general spatial differential movement of a rigid body consists of a differential rotation about, and a differential translation along an axis named the instantaneous screw axis. The complete movement of the rigid body, combining rotation and translation, is called screw movement or twist and is here denoted by $\$$. The ratio of the linear velocity to the angular velocity is called the pitch of the screw denoted as h.

The twist may be expressed by a pair of vectors $\$ = \left[\omega^T; V_p^T\right]^T$, 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 magnitude and its corresponding normalized screw. The twist magnitude $\dot{q}$ is either the magnitude of the angular velocity of the body, $\|\omega\|$, if the kinematic pair is rotative ($h = 0$) or helical, or the magnitude of the linear velocity, $\|V_p\|$, if the kinematic pair is prismatic ($h \rightarrow \infty$). The normalized screw $\hat{\$}$ is a twist of unitary magnitude, i.e.

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

The normalized screw coordinates $\hat{\$}$ is written as:

$$\hat{\$} = \left[\begin{array}{c} s_i \\ s_{oi} \times s_i + h s_i \end{array} \right] \tag{2}$$

where $s_i = \left[s_{i_x}, s_{i_y}, s_{i_z}\right]$ denotes an unit vector along the direction of the screw axis, and vector s_{oi} denotes the position vector of a point lying on the screw axis.

Thus, the twist in Eq. (2) expresses the general spatial differential movement (velocity) of a rigid body relative to an inertial reference frame $O - xyz$. The twist can also represents the movement between two adjacent links of a kinematic chain. In this case, twist $\$_i$ represents the movement of link i relative to link $(i - 1)$.

More details of the screw theory and its applications can be found in the following works: (Hunt, 2000), (Davies, 1981).

3.2 Davies method

Davies method is a systematic way to relate the joint velocities in closed kinematic chains. Davies (Campos *et al.*, 2009) derived a solution to the differential kinematics of closed kinematic chains from 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" (Campos *et al.*, 2009). This method is used to obtain the relationship between the velocities of a closed kinematic chain. Since the velocity of a link with respect to itself is null, the circulation law can be expressed as:

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

where n denotes the number of screw of the closed kinematic chain and 0 is a vector of dimension equal to twist $\$_i$ dimension.

According to the normalized screw definition introduced above, Eq. (3) may be rewritten as

$$\sum_{0}^{n} \hat{\$}_i \dot{q}_i = 0 \tag{4}$$

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

Equation (4) is the constraint equation which, in general can be written as

$$N\dot{q} = 0 \tag{5}$$

where $N = [\hat{\$}_1 \quad \hat{\$}_2 \quad \cdots \quad \hat{\$}_n]$ is the network matrix containing the normalized screws, the signs screws dependent on the definition of the circuit orientation (as will be presented later) (Campos *et al.*, 2009), and $\dot{q} = [\dot{q}_1 \quad \dot{q}_2 \quad \cdots \quad \dot{q}_n]$ is the magnitude vector.

A closed kinematic chain has actuated joints, here named primary joints, and passive joints, named secondary joints. The constraint equation, Eq. (5), allows the calculation of 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 and Eq. (5) is rewritten as follows:

$$\left[\begin{array}{ccc} N_p & \vdots & N_s \end{array} \right] \left[\begin{array}{c} \dot{q}_p \\ \cdots \\ \dot{q}_s \end{array} \right] = 0 \tag{6}$$

where N_p and N_s are the primary and secondary network matrices, respectively, and $\dot{q}_p$ and $\dot{q}_p$ are the corresponding primary and secondary magnitude vectors, respectively. Equation (6) can be rewritten as

$$N_p\dot{q}_p + N_s\dot{q}_s = 0 \tag{7}$$

The secondary joint position can be computed by integrating Eq. (7) as follows:

$$q_s(t) - q_s(0) = \int_0^t \dot{q}_s dt = -\int_0^t N_s^{-1} N_p \dot{q}_p dt \tag{8}$$

3.3 Assur virtual chains

The Assur virtual kinematic chain concept, virtual chain for short, is essentially a tool to obtain information on the movement of a kinematic chain or to impose movements on a kinematic chain (Campos *et al.*, 2009).

This concept was first introduced by (Campos *et al.*, 2009), which defines the virtual chain as a kinematic chain composed of links (virtual links) and joints (virtual joints) which possesses three properties: a) the virtual chain is open; b) it has joints whose normalized screws are linearly independent; c) it does not change the mobility of the real kinematic chain.

From the c) property, the virtual chain proposed by (Campos *et al.*, 2009) is in fact an Assur group, i.e. a kinematic subchain with null mobility that when connected to another kinematic chain preserves mobility (Artobolevskii, 1970-75).

For example, to represent the movements in the Cartesian system the $3P3R$ virtual chain is used. This chain is composed of three orthogonal prismatic joints (in the x, y, and z directions), and a spherical wrist, which can be decomposed in three rotational joints (in the x, y, and z directions).

3.4 The direct graph notation

Consider a kinematic pair composed of two links E_i and E_{i+1}. This kinematic pair has its relative velocity defined by a screw ${}^{R}\$_j$ (joint j) relative to a reference frame R. Joint j represents the relative movement of the link E_i with respect to the link E_{i+1}. This relation can be represented by a graph, as shown Fig. 1(a), where the vertices represent links and the arcs represent joints. The relative movement is also indicated by the arcs directions. Figure 1(a), for instance, link E_{i+1} moves relative to link E_i via joint j.

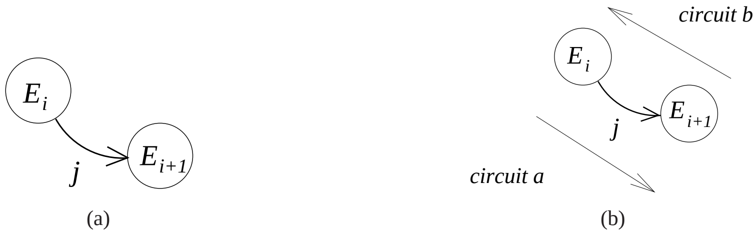

Figure 1. (a) Movement of link E_i relative to link E_{i+1}; (b) Relation between joint j and the circuits a and b

Considering the following example, where joint j is part of two closed chains. For each closed chain the circuit direction is defined (Campos *et al.*, 2009). Figure 1(b) illustrates this case. In a direct mechanism graph, if the joint has the same direction as the circuit, the twist associated with the joint has a positive sign in the circuit equation (see Eq. (3)), and a negative sign if the joint has the opposite direction to the circuit. In the example, twist ${}^{R}\$_j$, associated with joint j will have a positive sign in circuit a equation and a negative sign in circuit b equation.

4. Modeling of the proposed switching differential kinematic model

To develop the method, we initially mapped geometrically physical limits into robot workspace, which it could come to collide. In the robot is identified a part, region, or physical element that has the major possibility to collide with any of those mapped areas.

The method of collision avoidance by the switching differential model introduces two Assur virtual chains:

- Virtual chain attached on end-effector, to impose a trajectory to the end-effector by successive positioning of the virtual joints.
- Virtual chain attached on any part of the robot kinematic chain, to monitor and impose (when under imminence of collision) its position in relation to the obstacle of the workspace.

The method operates from the construction of two differential models. The first model (*model 1*) deals with the computing the velocities of the joints of the robot when there is no possibility of collision. The second model (*model 2*) deals with the computing the velocities of the joints of the robot when it is in a kinematic configuration such that there is a possibility of collision.

4.1 The model 1

In model 1, the collision avoidance virtual chain only monitors the distance between the robot and its area (or point) of possible collision. In this case the joints that composes the collision avoidance virtual chain and the joints of the robot are characterized as secondary joints. In this model, only the set joints of the trajectory virtual chain are primary.

In the case of redundant robots, additional strategies should be used. An example is chosen a secondary joint of the robot and impose on it a movement, so this joint was once secondary becomes primary. (Simas, 2008).

Once imposed on the positions of the primary joints, the positions of secondary joints are computed from solving the integral presented on Eq. (8) by numerical methods. In the simulations to be presented in this paper, we use the method of the error Assur virtual chains (Simas *et al.*, 2009).

4.2 The model 2

The model 2 works when the robot enters a particular condition that indicates a possible collision, for example, when the distance between a surface into robot workspace and a point of interest in the robot is smaller than a given value.

In this model, according to the operational dimension of space, are actuated one or more joints of the collision avoidance virtual chain with objective of removing the robot from collision.

Similar to model 1, the joints of the trajectory virtual chain are primary together with the actuated joints of the collision avoidance virtual chain. The joints of the robot and the joints not actuated of the collision avoidance virtual chain comprise the set of secondary joints.

Using Eq. (7) are computed the velocities of secondary joint and using Eq. (8) (by numerical method) are computed the position of the secondary joints.

4.3 Operational aspects of the method of switching differential models - Abrupt switching method

The abrupt switching method works by switching between the differential models. For each desired position and orientation for the end-effector, according to the trajectory, the condition of collision is checked. If the condition of collision is negative, that is, the robot is not under imminent collision, we use a model 1 to compute the robot joint velocities. If the condition of collision is true, that is, the robot is under imminent collision at some point in workspace, we use the model 2 to compute the robot joint velocities.

In each model are used together sets of different primary and secondary joints, which differentiates N_p and N_s matrices for each model (see Eq. (7)). The consequence of this strategy, is that the velocities computed for the joints in a transition from model 1 to model 2, and vice-versa, are generally discontinuous.

The discontinuity in the velocities of joints results in abrupt movements of the robot structure. Depending on the magnitude of velocities variations (resulting acceleration), failures can occur in mechanical elements, such as pulleys, gears and belts.

This method is useful in applications where operating velocities are relatively low, eg in the welding process.

But for higher velocities the method does not apply. For higher velocities it is necessary smoothing these transients. This was the motivation for the method proposed in this paper. The next section presents the new method.

5. Method for smoothing of transitions of differential kinematic models

To solve the problem of abrupt switching between systems and avoid discontinuities, was developed a method for smoothing these discontinuity.

Firstly it is defined intermediate steps to control the movement, ie, steps that perform the smooth transition of the operation monitoring to the collision avoidance operation and vice-versa.

Each step of the motion can be defined as a condition of movement of the kinematic chain. To have a smooth transition between conditions, either in approximation or the move away the obstacle, it is necessary to define four different movement conditions. The four conditions of motion are:

- Collision monitoring;
- Input transition;
- Avoid collision;
- Output transition.

The condition *Collision monitoring* represents the moment where there is no sufficient approximation of the obstacle to cause a collision. This condition represents the solution of the system as if do not exist obstacle.

During the *Collision monitoring* condition is necessary to make a prediction of a collision, checking if there is a possibility of collision if the movement continues its natural course, in other words, the prediction of collision detect whether certain part of the robot is moving on direction of the obstacle, on a collision course.

Once we set the distance limit between a part of the robot and the obstacle is possible, by anticipating the collision, detect when the part of the robot body beyond this distance limit.

When this limit is exceeded it is necessary to take some action to avoid collision. In the method of switching differential models, a set of joints of the collision avoidance virtual chain were locked preventing the movement of the robot toward the obstacle. This action causes the motion discontinuities.

To smooth the discontinuity, the condition *Input transition* makes the smooth transition between the operation of monitoring operation and avoid collision. The input transition imposes a smooth movement for all joints of the collision avoidance virtual chain, in such a way that does not occur discontinuity on its velocities and accelerations.

The condition must spend enough time for smoothing the movement and does not generate accelerations and velocities on the actuators that exceed predetermined limits.

Defining the moment of collision detection as t_i and the time interval during the transition t_{trans} as input, we can use a polynomial interpolation between the instants t_i and $t_i + t_{trans}$ to construct a trajectory to remove robot from collision condition. The polynomial interpolator must ensure the following specification:

- Initial positions, velocities and accelerations, at instant t_i, of the joints of the collision avoidance virtual chain to be actuated;

- Final positions, velocities and accelerations, at instant $t_i + t_{trans}$, of the joints of the collision avoidance virtual kinematic chain to be actuated: $q(t_{trans}) = q_{lim}$, $\dot{q}(t_{trans}) =$ and $\ddot{q}(t_{trans}) = 0$, where q_{lim} is the joint position where the robot keeps on limit distance from obstacle.

So, for each joint actuated we have six parameters to be solved: initial and final positions, velocities and accelerations. In this sense, a 5^{th} order polynomial provides these specifications at the beginning and end of the input transition.

At the end of the input transition condition we pass to *Avoid collision* condition. In this condition, the magnitude of the joints of the collision avoidance virtual chain remains the same as the end of the transition (q_{lim}).

This action prevents the robot entries into the region of collision. At each step of solution of the inverse kinematics, it is checked if the robot configuration is setting outside the limit of the region of collision. This characterizes the instant detection of obstacle clearance.

The condition *Output transition* makes the smooth transition between the movements of the robot on avoid collision condition and the monitoring condition with aim to avoid discontinuities in joints velocities and accelerations of the robot.

The smooth movement of the joints are obtained through a polynomial interpolation, where the position, velocity and acceleration are used as boundary conditions of the polynomial. In this case, we can not use a 5^{th} order polynomial because the final position of the joints of the collision avoidance virtual chain in the output transition condition is undetermined. In this case we use a 4^{th} order polynomial, where we can provide five boundary conditions, which are: the position, velocity and acceleration at the instant of exit from the obstacles and the velocity and acceleration at the end of the output transition.

As the input transition, output transition condition should be long enough for the smoothing of movements does not generate accelerations on the actuators that exceed a predetermined maximum. Defining the moment of detection of exit from obstacles as t_i and the time interval during the transition t_{trans} as input, the polynomial interpolation should occur between the instants t_i and $t_i + t_{trans}$.

At the end of the output transition condition we return to collision monitoring condition and the cycle begins again.

Mode details of this method can be found in (Cruz, 2007).

6. Differential kinematic model using $P3R$ planar redundant robot

To validate the method of smoothing transition of kinematic models was used a $P3R$ planar redundant robot (Simas, 2008). This robot represent a planar form of the Roboturb prototype (Simas *et al.*, 2004) .

The $P3R$ planar robot is a redundant robot with 4-DOF composed by a prismatic joint, perpendicular to the z axis of the reference coordinate system (coordinate system of the base), and three rotative joints with rotation axis parallel to this same z axis. The $P3R$ robot will execute a task avoiding the collision with a obstacle inside its workspace.

In the differential kinematic model, was added a virtual chain (PRR) to control the end-effector and generate the trajectories and a virtual chain(RPR) for avoid collisions.

The avoidance collision virtual chain is connected in the third body of the $P3R$ robot, the link 2, in opposite positition of the joint B, in the region called *elbow*. This body can collide with obstacle during operation. The base of the virtual chain is fixed on the collision point in such way that the prismatic joint (joint pr) always is aligned between the obstacle and robot's elbow. When the elbow is closer to the collision point, or entering into a collision region, the joint pr is activated driving the elbow away from the collision limits. Figure 2 depicts the complete kinematic chain including the virtual chains.

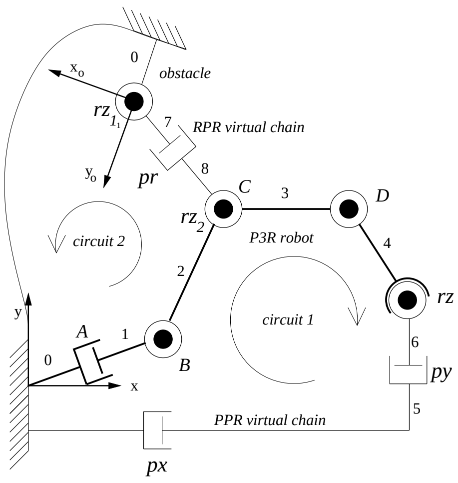

Figure 2. Complete kinematic chain

where to $P3R$: A is the prismatic joint; B, C and D are the revolute joints; 0 indicate the link of the base and 1, 2, 3 and 4 are robot links; to trajectory virtual chain: px, py and rz are the joint and the links 5 and 6 its links; and to collision avoidance virtual chain: rz_1, pr and rz_2 are the joints of chain and links 7 and 8 its links. Circuit 1 and 2 define the orientation of each closed loop chain in accordance with the Davies method.

In agree with the kinematic presented on Fig. 2 we have to constraint equation the Eq (9):

$$N = \begin{bmatrix} \hat{\$}_A & \hat{\$}_B & \hat{\$}_C & \hat{\$}_D & 0 & 0 & 0 & -\hat{\$}_{rz} & -\hat{\$}_{px} & -\hat{\$}_{py} \\ \hat{\$}_A & \hat{\$}_B & 0 & 0 & -\hat{\$}_{rz_1} & -\hat{\$}_{pr} & -\hat{\$}_{rz_2} & 0 & 0 & 0 \end{bmatrix}$$
$$\dot{q} = \begin{bmatrix} \dot{q}_A & \dot{q}_B & \dot{q}_C & \dot{q}_D & \dot{q}_{rz_1} & \dot{q}_{pr} & \dot{q}_{rz_2} & \dot{q}_{rz} & \dot{q}_{px} & \dot{q}_{py} \end{bmatrix} \tag{9}$$

where $\hat{\$}_A$, $\hat{\$}_B$, $\hat{\$}_C$ and $\hat{\$}_D$ are the screws of the $P3R$ robot; $\hat{\$}_{rz_1}$, $\hat{\$}_{pr}$ and $\hat{\$}_{rz_2}$ are the screws of the collision avoidance virtual chain; $\hat{\$}_{rz}$, $\hat{\$}_{px}$ and $\hat{\$}_{py}$ are the screws of the trajectory virtual chain; $\dot{q}_A$, $\dot{q}_B$, $\dot{q}_C$ and $\dot{q}_D$ are magnitudes of the velocities of the $P3R$ robot; $\dot{q}_{rz_1}$, $\dot{q}_{pr}$ and $\dot{q}_{rz_2}$ are magnitudes of the velocities of the collision virtual chain; $\dot{q}_{px}$, $\dot{q}_{py}$ and $\dot{q}_{rz}$ are magnitudes of the velocities of the trajectory virtual chain.

From the constraint equation are extracted the network primary (N_{p1}) and secondary (N_{s1}) matrices for model 1 and the network primary (N_{p2}) and secondary (N_{s2}) matrices for model 2.

Equation (10) shows the network matrices for model 1, or the differential kinematic model when there is no collision imminence.

$$N_{p1} = \begin{bmatrix} \hat{\$}_A & -\hat{\$}_{rz} & -\hat{\$}_{px} & -\hat{\$}_{py} \\ \hat{\$}_A & 0 & 0 & 0 \end{bmatrix} \qquad N_{s1} = \begin{bmatrix} \hat{\$}_B & \hat{\$}_C & \hat{\$}_D & 0 & 0 & 0 \\ \hat{\$}_B & 0 & 0 & -\hat{\$}_{rz_1} & -\hat{\$}_{pr} & -\hat{\$}_{rz_2} \end{bmatrix} \tag{10}$$

Note that in model 1, we choose impose a velocity to prismatic joint of the $P3R$ robot, and for that reason the screw of the joint A is considered primary.

Equation (11) shows the network matrices for model 2, or the differential kinematic model when the robot is under imminent collision.

$$N_{p2} = \begin{bmatrix} -\hat{\$}_{rz} & -\hat{\$}_{px} & -\hat{\$}_{py} & 0 \\ 0 & 0 & 0 & -\hat{\$}_{pr} \end{bmatrix} \qquad N_{s2} = \begin{bmatrix} \hat{\$}_A & \hat{\$}_B & \hat{\$}_C & \hat{\$}_D & 0 & 0 \\ \hat{\$}_A & \hat{\$}_B & 0 & 0 & -\hat{\$}_{rz_1} & -\hat{\$}_{rz_2} \end{bmatrix} \tag{11}$$

Under imminent collision, the joint virtual pr is activated to push the elbow of the robot P3R out of bounds the region of collision. For this reason the screw of this joint is shown in the matrix N_{p2}.

7. Final results

The method was validated in a experimental application which the $P3R$ prototype held a circular trajectory.

To this simulation we use the following specifications:

- $P3R$ redundant robot
  - Length of the links 1, 2 = 2 m;
  - Length of the link 3 = 1 m;
  - Displacement of the joint A : m
  - Displacement of the joint B, C and D : $radians$
- Desired trajetory: *semi-circle*
  - Position:

$$p_d(t) = \begin{bmatrix} 4 + 0.5cos\left(\frac{\pi}{2}t\right) \\ 0.9 + 0.6sin\left(\frac{\pi}{t}t\right) \end{bmatrix} m \qquad \phi_d = \left(\frac{\pi}{3} - 0.4sin\left(\frac{\pi}{2}t\right)\right) rads \tag{12}$$

  - Number of loops = 0.5 (for best viewing results)
  - Experiment time = 1 sec
- Obstacle:
  - Center position: coordinates (2.1, 3.5) m
  - Radius : 0.8 m

For purposes of comparison, was also implemented the method of switching of models (abrupt switching) discussed above in this paper. The objective is to evaluate the differences in the profiles of trajectories of the joints and performance of the robot during task execution.

The figures in following depict, and comparing, the results of the joints position, velocity and acceleration for the $P3R$ robot. The resultant profiles are highlighted to the abrupt switching of models and the smooth switching model proposed in this paper.

Figure 3 depicts the profiles of positions of the joints when applied the abrupt and smooth switching method.

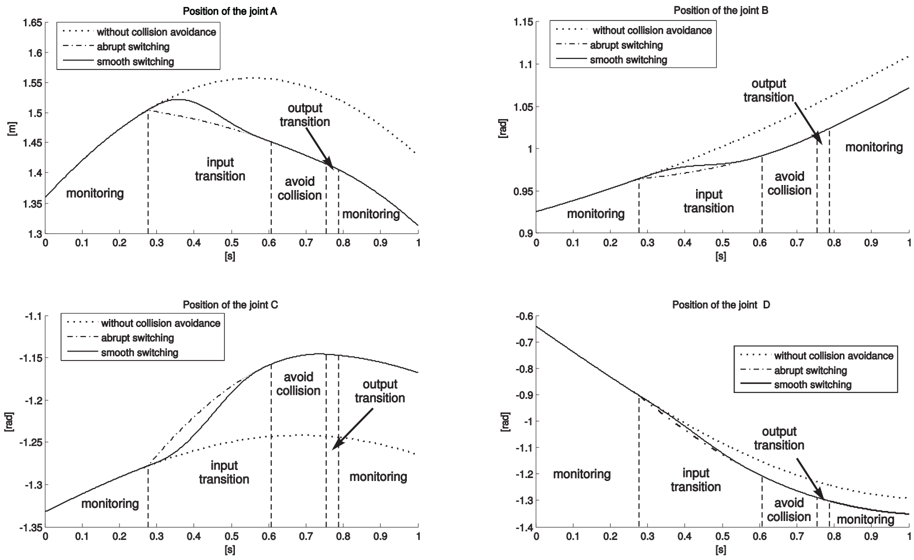

Figure 3. Position of the joints A, B, C and D

It can be seen in the Fig. 3, which in the beginning of the operation the robot is in the monitoring condition from instant 0 until time 0.276 sec. Following the robot enters in the input condition and remains in this condition between the times 0.276 sec and 0.609 sec. The condition for avoidance of collision occurs between the times 0.609 sec and 0.755 sec. Soon after, the robot leaves the collision by output condition between the times 0.755 sec and 0.788 sec, so it comes back in monitoring condition until the end of the experiment. These times are indicated in next figures

Figure 4 depict the profiles of velocities of the joints when applied the abrupt and smooth switching method.

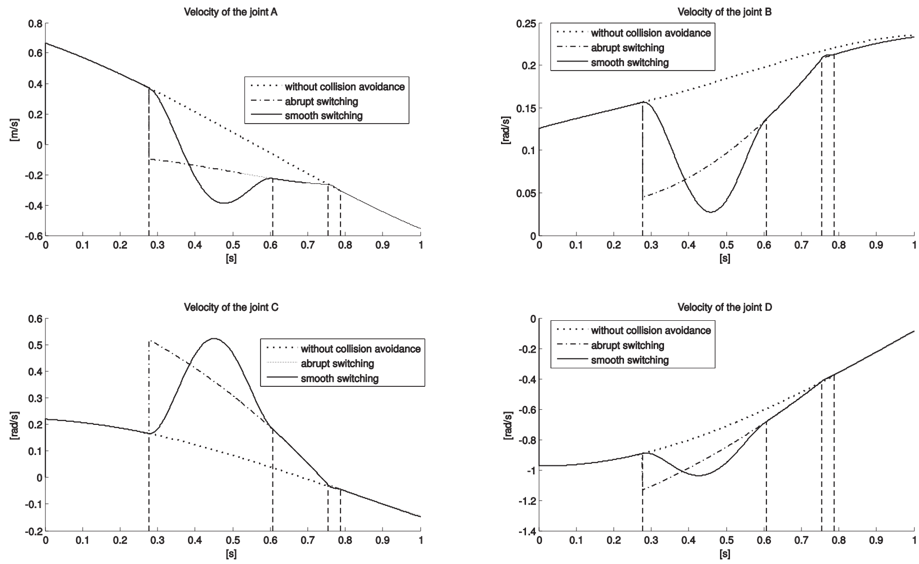

Figure 4. Velocities of the joints A, B, C and D

Figure 5 depict the profiles of accelerations of the joints when applied the abrupt and smooth switching method.

Figure 6 shows the profile of the position, velocity and acceleration of the joint pr of the collision virtual chain. The results are presented only for abrupt and smoothing switching, highlighting the segments of the position curve where it was applied the 5^{th} order polynomial interpolation. Note that the joint pr leads and hold the position of the $P3R$ elbow with distace of $0.8\ m$ from the obstacle.

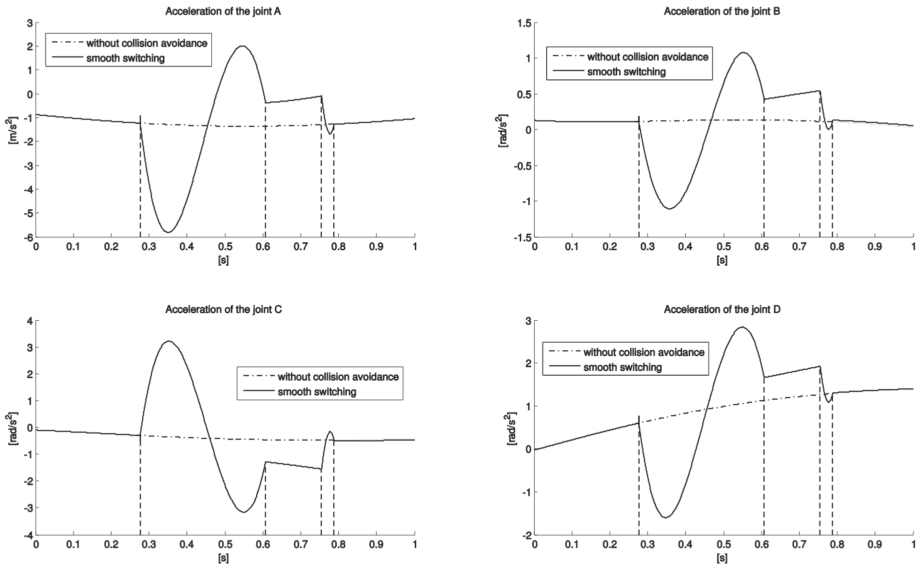

Figure 5. Accelerations of the joints A, B, C and D

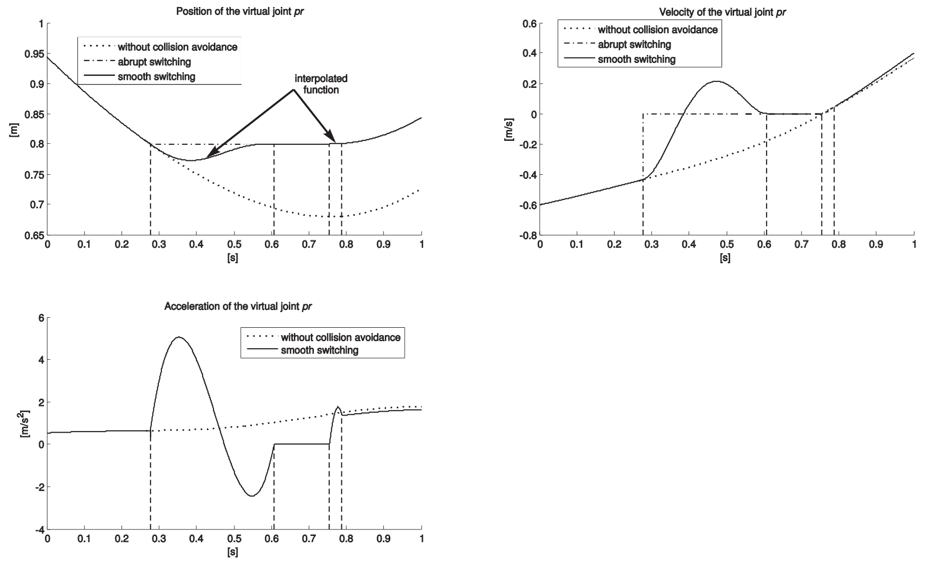

Figure 6. Resultant position, velocity and acceleration profiles for the virtual joint pr

With respect to Fig. 6, it is important to note that was not shown the profile of the acceleration with abrupt switching for collision avoidance. The fact is that, at time 0.279 sec, the acceleration assumes a peak with a value 422.95 m/s^2 (At this point a discontinuity occurs in velocity of the joint pr) and graphically the resultant profile does not contribute to an effective comparison of accelerations.

The results of the inverse kinematics obtained in a experimental test comply with the expectations, where the transition motions was smoothed during the collision avoidance.

The results demonstrated the efficiency and applicability of the proposed methodology, making it an interesting tool for project of trajectory generators for robots that operate in confined environments.

8. Conclusion

This work presented a new method of collision avoidance for redundant robots that allowing the robot to avoid obstacles without causing discontinuities in the magnitudes of its joints. Furthermore, the methodology is also based on the use of the screw theory in the Davies method, the use of Assur virtual chains and the method of kinematic constraints. The methodology uses a systematic method of solving inverse kinematics that can be applied both serial or parallel robots.

The proposed collision avoidance methodology allows to act in the robot control system in two different ways: with the robot out of a collision condition and with the robot in imminent collision. The methodology smooths switching between the motion control systems, preventing discontinuities in the magnitudes of the velocities and accelerations of the joints of the robot during the obstacle avoidance, without change the performance of the main task.

The method was applied experimentally in a $P3R$ redundant robot, which were obtained relevant results. The deviation from the obstacle was conducted smoothly without generating discontinuities in the magnitudes of the joints.

It is understood that in a real application, applying the abrupt switching method, vibrations could occurs in the structure of the manipulator, during the collision avoidance. These vibrations would be reduced if used smooth switching switching. The results indicate the importance of smoothing of the discontinuities and the application of the methodology in practice. The next step will be apply the method of smoothing discontinuities in a real spatial robot.

9. REFERENCES

Artobolevskii, I.I., 1970-75. *Mechanisms in modern engineering design : a handbook for engineers, designers and inventors*. Mir.

Barraquand, J. and Latombe, J., 1990. "A monte-carlo algorithm for path planning with many degrees of freedom". In *Proceedings of the IEEE International Conference on Robotics and Automation*. Cincinnati, OH, Vol. 3, pp. 1712–1717.

Campos, A., Guenther, R. and Martins, D., 2009. "Differential kinematics of parallel manipulators using Assur virtual chains". *Proceedings of the Institution of Mechanical Engineers. Part C, Journal of Mechanical Engineering Science*, Vol. 223, pp. 1697–1711.

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

Chang, H., 1996. "A new technique to handle local minimum for imperfect potencial field based motion planning". In *Proceedings of the IEEE International Conference on Robotics and Automation*. Minneapolis, MN, pp. 108–112.

Cruz, D.F.M., 2007. *Implementação da cinemática inversa de robôs redundantes operando em ambientes confinados no projeto Roboturb*. Dissertação de (mestrado), Universidade Federal de Santa Catarina.

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

Hunt, K.H., 2000. "Don't cross-thread the screw!" In *Proceeding of A Symposium Commemorating the Legacy, Works, and Life of Sir Robert Stawell Ball Upon the 100th Anniversary of A Treatise on the Theory of Screws*. University of Cambridge.

Laliberte, T. and Gosselin, C., 1994. "Efficient algorithms for the trajectory planning of redundant manipulators with obstacle avoidance". In *Proceedings of the IEEE International Conference on Robotics and Automation*. San Diego, CA, pp. 2044–2049.

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

Nedungadi, A., 1992. "A fuzzy robot controller - hardware implementation". In *Proceedings of the IEEE International Conference on Fuzzy Systems*. San Diego, USA, pp. 1325–1331.

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

Siciliano, B., Sciavicco, L., Villani, L. and Oriolo, G., 2009. *Modeling and control of robot manipulator*. Springer.

Simas, H., 2008. *Planejamento de Trajetórias e Evitamento de Colisão em Tarefas de Manipuladores Redundantes Operando em Ambientes Confinados*. Tese de doutorado, Universidade Federal de Santa Catarina.

Simas, H., Guenther, R., Cruz, D.F.M. and Martins, D., 2009. "A new method to solve robot inverse kinematics using Assur virtual chains". *Robotica, (Cambridge)*. doi:10.1017/S0263574709005426.

Simas, H., Martins, D. and Guenther, R., 2004. "Redundant manipulator to operate in confined spaces". In *Proceedings VI Induscon- Conferência Internacional de Aplicações Industriais*. Joinville-Brazil.

Simoni, R., Carboni, A.P. and Martins, D., 2009. "Enumeration of kinematic chains and mechanisms. proceedings of the institution of mechanical engineers. part c". *Journal of Mechanical Engineering Science*, Vol. 223, pp. 1017–1024.

Soucy, M., 2005. *Manipulator Path Planning with Multi-Resolution Potencial Fields and Fuzzy Logic Control*. Dissetação (mestrado), University of Ottawa.

Soucy, M. and Payeur, P., 2005. "Flexible fuzzy logic control for collision-free manipulator operation". In *Proceedings of 2005 IEEE International Conference*. Niagara Falls, Canada, Vol. 2, pp. 723–728.

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

10. Responsibility notice

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