A NEW APPROACH FOR COLLISION AVOIDANCE OF MANIPULATORS OPERATING IN UNSTRUCTURED AND TIME-VARYING ENVIRONMENTS

Carlos Rodrigues Rocha, cticarlo@furg.br
Universidade Federal de Santa Catarina - Departamento de Engenharia Mecânica

Henrique Simas, hsimas@univali.br
Universidade do Vale do Itajaí - Centro de Ensino São José
Universidade Federal de Santa Catarina - Departamento de Engenharia Mecânica

Daniel Martins, daniel@emc.ufsc.br
Altamir Dias, altamir@emc.ufsc.br
Universidade Federal de Santa Catarina - Departamento de Engenharia Mecânica

Abstract. *This paper presents a new approach for collision avoidance of manipulators. When the workspace is completely mapped and the obstacles are known, it is possible to define the task considering the presence of obstacles or their proximity can be monitored simply by the inverse kinematics resolution of the complete system. However, when the workspace is poorly mapped/unstructured and/or time-varying, it is necessary to rely on sensor information to monitor proximity of obstacles and, consequently, define a suitable collision avoidance strategy. This kind of environment is increasingly common in recent robotic applications, and the use of kinematic redundant manipulators is considered, because of increased dexterity. There are being developed different strategies to address this problem, and the literature presents different kinds of sensors and their use to detect the proximity of an obstacle. These strategies use the more traditional pseudo-inverse methods to solve the inverse kinematics of the redundant kinematic chain. In this work, it will be shown the kinematic constraints method as an alternative to solve the inverse kinematics. This method is based on screw theory, the Kirchhoff-Davies method and virtual chains. Among other advantages over the pseudo-inverse methods, the kinematic constraints method is dimensional consistent and conserves movement. An example is used to illustrate the method and its main characteristics.*

Keywords: *Collision Avoidance, Redundancy, Screw Theory, Kinematic Constraints Method, Unstructured Workspace*

1. INTRODUCTION

Manipulators usually work in structured, fully-mapped environments. However, as new robot applications arise, this reality changes from a *safe* operational environment to poorly mapped or even unstructured one. The environment can also be *time-varying*, where obstacles move through it. This situation is illustrated in several literature examples, like the human/robot interaction problems, where safety issues assume especial importance (Bicchi et al., 2008). The ability to autonomously operate in these environments is a key feature to intelligent robotic systems (Wang et al., 2007).

Collision avoidance strategies are necessary to guarantee properly and safe operation. In some situations, which could be named *uncertain environments*, these strategies must rely on sensor information, since it is not possible to completely model the workspace. They must also have additional conditions especified, in order to modify the originally planned task, as new obstacles are detected and need to be avoided during its execution. These kind of problem has motivated several researches, which have used redundant manipulators to explore their extra degrees-of-freedom to perform collision avoidance while complying with the task defined. In these works, redundant inverse kinematics is solved by use of traditional methods, which are reviewed in (Chiaverini et al., 2008).

The *kinematic constraints method* is an alternative to solve the redundant inverse kinematics, which have advantages over those traditional methods. This method is based on screw theory, the Kirchhoff-Davies method and the virtual chains concept (Campos, 2004; Santos et al., 2006). It is already being used to explore redundancy in cases where workspace is structured and collision avoidance is necessary (Simas et al., 2008; Fontan, 2007). The extension of this method for use in uncertain environments is the contribution ofthis work.

The paper is organised as follows. In Section 2, the problem of operating in uncertain environments is discussed, and some approaches used in literature are presented. Section 3 consists of a short review of the kinematic tools used in this work. An example is presented in Section 4. Session 5 presents the final conclusions.

2. MANIPULATORS OPERATING IN UNCERTAIN ENVIRONMENTS

Many researches have presented work about structured, fully-modelled environments, where collision possibilities can be mapped and monitored. Cheung and Lumelsky call this approach *the piano movers problem* (Cheung and Lumelsky, 1989). In most general case, however, *a priori* information about the environment must be assumed as incomplete, and it could not be considered time-invariant. In this situation, sensors must provide the necessary information during task execution. This information could refer to complete or partial environment, and its processing is time-consuming, which limits the amount of information that could be used.

Information provided by sensors relative to collision avoidance must be sufficient to detect obstacles on the fly to provide a deviance strategy. Although there are some work using external vision systems, the use of sensors attached to the manipulator structure is the most common approach. They must cover not only the end-effector and tools, but also the full manipulator body. Several author present different approaches and sensor types to solve obstacle proximity detection (Wang et al., 2007; Cheung and Lumelsky, 1989; Iwata et al., 2001; Um and Hung, 2006). Lumelsky discusses the requisites and uses of a sensitive skin, and its importance to intelligent robots operating in non-structured environments (Lumelsky, 1987).

Most research consider that with full body coverage by sensors (which are usually infrared), a *sensitive aura* over the manipulator structure is obtained, and the space inside it can be considered obstacle-free. As the manipulator moves, the aura moves with it. When an obstacle *touches* the aura, information provided by sensors is enough to determine its position relative to one or more sensors. With some kinds of sensor, even partial shape of the obstacle can be computed. Since the sensors are attached to the links, it is possible to express this position in any desired reference system, after some coordinate transformation. So, obstacle position relative to any part of the manipulator is considered known, and collision avoidance strategy can be executed.

A good way to have control over these system is to have the manipulator operating in two distinct states. The first one checks the free movement or *monitoring state*, when there is no obstacle detected, and the originally planned trajectory is followed. The second state is the *collision avoidance state*, where a strategy modifies locally the original planned trajectory. The manipulator switches between states depending on obstacle detection. This monitoring state can yield discontinuities in the movement, with undesired effects over joints and actuators (Fontan, 2007).

Obstacle avoidance implies additional constraints over the task that is performed by manipulator. In this case, *kinematic redundancy* approach is a necessary feature (Lee and Buss, 2007; Chiaverini et al., 2008). In a simpler way, redundancy occurs when the mobility of the kinematic chain is greater than the necessary to execute a movement. Kinematic redundancy is properly defined as the difference between the degrees-of-control and the connectivity between two links of a kinematic chain (Martins and Carboni, 2008). It can be noted that kinematic redundancy (from now on redundancy, for short) can depend on the task to be executed. Although redundancy increases dexterity, allowing to deal with additional constraints and other complementary goals, its inverse kinematics has infinite solutions, which introduces a problem widely studied. There are several methods proposed in literature, such as pseudo-inverse Jacobian, extended Jacobian, task-priority resolution, etc. (Chiaverini et al., 2008), which can be time-consuming, and may present dimensional inconsistencies. Usually they do not have the conservative movement property (Simas, 2008; Fontan, 2007; Santos, 2006).

The kinematic constraints method overcomes this flaws, and usually is computationally less demanding. The next section presents the method and its fundamentals in a short way. It is recommended that the reader consults the references for more information.

3. KINEMATIC TOOLS

This section shortly presents the methods used to solve the manipulators kinematics and the theoretical foundations on which they are based. The *kinematic constraints method* and the error-controlled numerical integration algorithm are based on *screw theory*, in *Kirchhoff-Davies method* and the *virtual kinematic chains*.

3.1 Screw Theory

Screw Theory is a tool used in static and kinematic analysis of rigid bodies and mechanisms. Its origins date back to Mozzi(1763) and Chasles(1830) studies, and it were systematised by Ball in 1900. Later, Hunt, Phillips, Roth e Tsai, among others, employed the theory to the study of mechanisms (Ceccarelli, 2000; Dai, 2006; Hunt, 2000).

A *screw* is a geometric entity which represents both rotational and translational quantities. It is composed by an *axis*, on which both quantities are defined, and a scalar *pitch*, which relates translation and rotation (Hunt, 2000). Screw theory associates physical meaning to a purely geometric entity, by its use to express velocities (angular and linear ones) as *twists*, and forces/torques as *wrenches* (Dai, 2006).

Screws are usually expressed in Plücker coordinates. As shown in Figure 1, vector $\mathbf{S}$ and its moment $\mathbf{s}_0 \times \mathbf{S}$ define the axis around which rotation occurs. Translation parallel to the axis is equal to $h\mathbf{S}$, and it is added to the moment. Rotation magnitude $\|\mathbf{S}\|$ is related to the translation magnitude by pitch h. With the unitary $\mathbf{s}$ of $\mathbf{S}$, a purely geometric *normalised screw* is obtained. The original screw is obtained multiplying the normalised screw by the magnitude of $\|\mathbf{S}\|$.

This development leads to

$$\$ = \left[\begin{array}{c}\mathbf{S} \\ \hdashline \mathbf{s}_0 \times \mathbf{S} + h\mathbf{S}\end{array}\right] = \left[\begin{array}{c}\mathbf{s} \\ \hdashline \mathbf{s}_0 \times \mathbf{s} + h\mathbf{s}\end{array}\right] \Psi = \hat{\$}\Psi \tag{1}$$

where $\$$ is the screw, $\hat{\$}$ is the normalised screw and Ψ is the magnitude. It must be noted that the six Plücker coordinates do not compose a vector, and that $\mathbf{S}$ and its moment are orthogonal (Hunt, 2000).

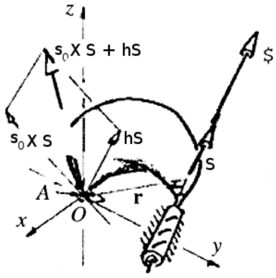


Figure 1: Geometric definition of a screw (Hunt, 2000)

There are two particular cases: the pure rotation implies $h = 0$, and its normalised screw is defined as $\hat{\$} = [\mathbf{s}; \mathbf{s}_0 \times \mathbf{s}]^{\mathrm{T}}$. In special case of pure translation, it is assumed that $h = \infty$, and the normalised screw is defined as $\hat{\$} = [\mathbf{0}; \mathbf{s}]^{\mathrm{T}}$.

3.1.1 The Successive Screw Displacements Method

According to Chasles Theorem, general displacement of a rigid body in space can be expressed by a rotation around an axis and a translation parallel to it (Simas, 2008; Dai, 2006). This concept leads to a method to describe the *pose* of a kinematic chain, similar to the Denavit-Hartenberg notation.

Expression of the pose of a rigid body in space is derived using *Rodrigues parameters*. Consider a point $\mathbf{P}$ moving from position $\mathbf{P}_1$ to position $\mathbf{P}_2$, as depicted in Figure 2a. The normalised screw, the rotation angle around screw axis θ and the translation t parallel to the axis define the homogeneous transformation matrix A, in Eq. 2, where $c\cdot$ and $s\cdot$ mean $\cos(\cdot)$ and $\sin(\cdot)$, respectively. (Tsai, 1999).

$$\mathbf{A} = \begin{bmatrix} \mathbf{R}(\theta) & \mathbf{p}(t) \\ 0 & 1 \end{bmatrix} \tag{2}$$

$$\mathbf{R}(\theta) = \begin{bmatrix} c\theta + s_x^2(1 - c\theta) & s_y s_x(1 - c\theta) - s_z s\theta & s_z s_x(1 - c\theta) - s_y s\theta \\ s_y s_x(1 - c\theta) - s_z s\theta & c\theta + s_y^2(1 - c\theta) & s_y s_z(1 - c\theta) - s_x s\theta \\ s_z s_x(1 - c\theta) - s_y s\theta & s_y s_z(1 - c\theta) - s_x s\theta & c\theta + s_z^2(1 - c\theta) \end{bmatrix} \tag{3}$$

$$\mathbf{p}(t) = t\mathbf{s} + [\mathbf{I} - \mathbf{R}(\theta)]\,\mathbf{s}_0 \tag{4}$$

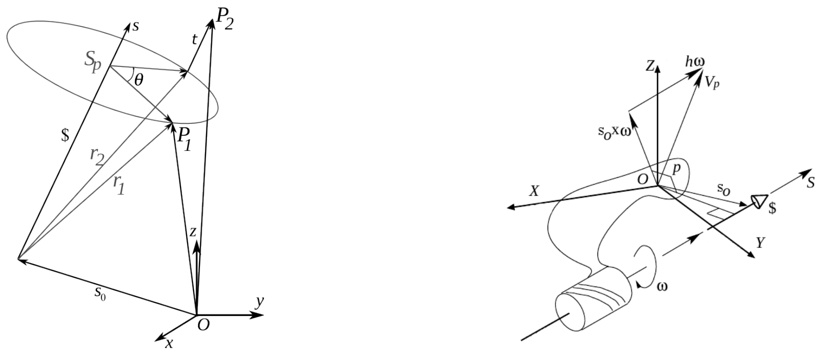
In a kinematic chain, the pose of a link e relative to a link b is defined by *successive screw displacements* made by the joints in the subchain between b and e. The overall displacement is obtained by premultiplication of the homogeneous transformation matrices ${}^{i-1}\mathbf{A}_i$, which defines the displacement of link i relative to link $i - 1$:

$${}^{b}\mathbf{A}_e = {}^{b}\mathbf{A}_1 {}^{1}\mathbf{A}_2 \cdots {}^{n-1}\mathbf{A}_n {}^{n}\mathbf{A}_e \tag{5}$$

Screw representation is always relative to a referential coordinate system, which can be chosen to obtain simplified representation or in order to study a particular feature of the problem (Simas, 2008). Screw parameters are determined relative to a *reference configuration* of the chain.

3.1.2 Screw-based Differential Kinematics

Mozzi's theorem states that the instantaneous movement of a rigid body can be decomposed into a differential rotation ω around an axis and a differential translation σ parallel to it. This screw movement is called *twist* (Hunt, 2000). As depicted in Figure 2b, a twist is defined by $\$ = (\boldsymbol{\omega}; \mathbf{v}_p)^{\mathrm{T}}$, where $\boldsymbol{\omega}$ is the angular velocity of the body and $\mathbf{v}_p$ is the linear velocity of a point instantaneously coincident with origin $\mathbf{O}$ which moves with the body. $\mathbf{v}_p$ has a component normal to the twist ($\mathbf{s}_0 \times \boldsymbol{\omega}$) and a component parallel to it ($\boldsymbol{\sigma} = h\boldsymbol{\omega}$). The twist also can be expressed as a normalised screw multiplied by a magnitude, as $\$ = \hat{\$}\dot{q}$, where $\dot{q}$ represent the magnitude of the velocity. For a pure rotation it is written as $\dot{q} = \|\boldsymbol{\omega}\| = \omega$. In special case of a pure translation it is expressed by $\dot{q} = \|\mathbf{v}_p\| = v_p$.



(a) (b)

Figure 2: Screw representation: (a)Screw displacement; (b)Twist components.

In a kinematic chain, the instantaneous movement of a link e relative to a link b is found out by the sum of the twists of the joints between the two links considered:

$$\$_e = \begin{bmatrix} \boldsymbol{\omega}_e \\ \mathbf{v}_{p_e} \end{bmatrix} = \sum_{i=b+1}^{e} \hat{\$}_i \dot{q}_i = \mathbf{J}\dot{\mathbf{q}} \tag{6}$$

where twist i is the velocity of link i relative to link $i-1$, $\mathbf{J}$ is the *Jacobian* which relates link e velocity to joints velocities and $\dot{\mathbf{q}} = [\dot{q}_{b+1} \cdots \dot{q}_e]^{\mathrm{T}}$ is a vector of twist magnitudes (Tsai, 1999; Hunt, 2000; Campos, 2004).

All twists must be defined relative to a common reference. Eventually, it could be necessary to work with a different reference system. In this case, a transformation between the coordinates systems must be made. The ${}^i\mathbf{T}_j$ matrix defined in Equation 7 represents this transformation, from the j referential to the i referential. In this Equation, $\mathcal{S}\left({}^i\mathbf{p}_j\right)$ is the antisymmetric matrix from the position vector between the two origins, expressed in i reference (Tsai, 1999).

$${}^i\mathbf{T}_j = \begin{bmatrix} {}^i\mathbf{R}_j & 0 \\ \mathcal{S}\left({}^i\mathbf{p}_j\right) {}^i\mathbf{R}_j & {}^i\mathbf{R}_j \end{bmatrix} \tag{7}$$

Twists can be determined using the homogeneous transformations obtained by the successive screw displacements (Simas, 2008). To do so, before determine the twist of a joint i, vectors $\mathbf{s}_0$ and $\mathbf{s}$ are transformed by the ${}^b\mathbf{A}_i$ matrix. Another way to obtain such twists is to compute them on reference position, each one according to a reference system particular to each joint, and applying the transformation ${}^b\mathbf{T}_i$ to them. Santos uses this method in (Santos, 2006).

3.2 The Kirchhoff-Davies Method

The *Kirchhoff-Davies* method is an adaptation of Kirchhoff's circuit laws for use on closed kinematic chains, in order to define its differential kinematics (Campos, 2004). Davies establishes that "the algebraic sum of relative velocities of kinematic pairs along any closed kinematic chain is zero" (Davies, 1981):

$$\sum_{i=1}^{n} \$_i = \sum_{i=1}^{n} \hat{\$}_i \dot{q}_i = 0 \tag{8}$$

The *constraint equation* (8) links the joint velocities of the chain (Simas, 2008). They can be split in two sets, denominated *primary* and *secondary* joints. This process allows to express velocities of some joints (usually, the secondary ones) as functions of known velocities of other joints (usually, the primary ones).

The constraint equation may be difficult to obtain, depending on the complexity of the kinematic chain. Graph theory can be used to systematise and to simplify this process.The use of graphs in kinematic chains and mechanisms analysis is presented in detail in (Tsai, 2000). Relevant information obtained from graph analysis are the gross degrees-of-freedom number F_b, the number of independent loops l and the incidence matrix B.

So, Equation 8 can be rewritten as $\mathbf{N}\dot{\mathbf{q}} = 0$, where $\mathbf{N}$ is the *network matrix*, defined as

$$\mathbf{N} = \begin{bmatrix} \mathbf{DB}_1 \\ \vdots \\ \mathbf{DB}_l \end{bmatrix} \tag{9}$$

Diagonal matrices $\mathbf{B}_c$ have elements from line c of $\mathbf{B}$ as their principal diagonal, and D is a matrix which columns are the direct screws of the chain.

The constraint equation is composed by λl constraints to the closed chain, where λ is the order of the screw system. The number of independent variables, or its *mobility*, is equal to $F_N = F_b - \lambda l$, which is also the number of actuated (primary) joints.

The passive (secondary) joint velocities are computed by rearranging the constraint equation, in order to split it in their primary and secondary elements:

$$\left[\begin{array}{c:c} \mathbf{N}_p & \mathbf{N}_s \end{array}\right] \left[\begin{array}{c} \dot{\mathbf{q}}_p \\ \hdashline \dot{\mathbf{q}}_s \end{array}\right] = 0 \tag{10}$$

$\dot{\mathbf{q}}_s$ can be now isolated, resulting in Equation 11. Now the secondary joint velocities can be evaluated if $\mathbf{N}_s$ is invertible. Otherwise, the system is singular (Campos, 2004).

$$\dot{\mathbf{q}}_s = -\mathbf{N}_s^{-1}\mathbf{N}_p\dot{\mathbf{q}}_p \tag{11}$$

3.3 Virtual Kinematic Chains

The virtual kinematic chain concept was introduced in Campos research (Campos, 2004). It can be taken as a tool either to monitor the behaviour of a real kinematic chain or to impose movements to it (Campos, 2004; Campos et al., 2005). In monitoring action, information about link displacements is obtained, which can be used to avoid obstacles, joint limits or singularities. When movement is enforced, the virtual chains can be used to specify tasks where constraints to the movement of particular links of the real chain are imposed. In this case, they can be a tool to explore the additional mobility provided by redundancy.

On an open chain, typical in serial manipulators, a virtual chain is introduced to close it. This allows the use of Kirchhoff-Davies method to solve the differential kinematics of the resultant chain. On a closed chain, virtual chains add loops to the existing circuit, allowing movement analysis and constraints specification (Campos et al., 2005).

Virtual chains are formed by links and joints, like a real chain, and must have the following properties: a) they must be open chains; b) the normalised screws related to their joints are linearly independent; and c) they do not modify the mobility of the real chain. As a consequence, mobility of a virtual chain is equal to the order of the screw system λ.

PPR and RPR chains are commonly used in planar chains ($\lambda = 3$), while PPPS chains are commonly inserted in spatial case ($\lambda = 6$). These chains are schematically represented in Figure 3.

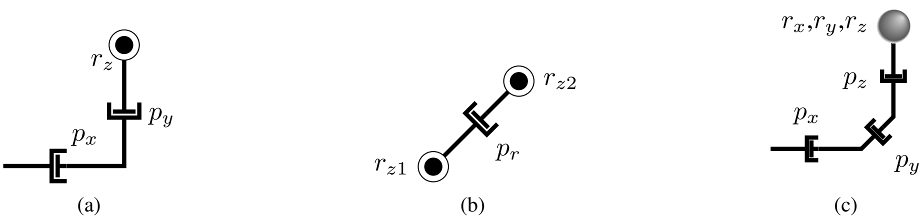


Figure 3: Commonly used virtual chains: (a)PPR chain; (b)RPR chain; (c)PPPS chain.

3.4 Inverse Kinematics Resolution

The joints behaviour are described by using all methods described above. The *kinematic constraints method* is used to obtain joints velocities, and the joints positions are computed by an integration algorithm method. Santos introduced the kinematic constraints method and use virtual chains connected to a real chain in order either to close it or to directly connect particular links of the chain, again, either to impose or to monitor the real chain movement (Santos, 2006). The virtual joints that impose or constraint movement are considered primary, while the real and the monitored virtual ones are considered secondary. The Kirchhoff-Davies method is used subsequently to obtain the velocities of these joints.

It is possible to define an invertible square $\mathbf{N}_s$ matrix by properly choosing the actuated joints in Eq. 11. The remaining joints, of course, must have their movement defined. In case of redundancy, additional constraints can be specified, to explore the increased dexterity feature. If it is not possible to define $\mathbf{N}_s$ as square, the pseudo-inverse operator can be used. This methodology, when applied to redundant chains, is equivalent to the extended Jacobian method (Simas, 2008).

Traditional methods used to solve redundancy like pseudo-inverse resolution, task-priority or even extended Jacobian present dimensional inconsistencies, and the movement, in general, is not conservative (Chiaverini et al., 2008). The kinematic constraints method does not have these disadvantages, as showed by several authors (Santos, 2006; Fontan, 2007; Simas, 2008).

In this procedure it is necessary to use integration methods to compute joint position. Numerical methods are usually employed to this end, although the Jacobian matrix is typically complex. These methods are approximation-based methods, prone to numerical errors, causing *drift* in open chains and the opening of the closed ones, during computations. Simas deals with this problem in his studies and proposes to add *error virtual chains* in the kinematic chain to overcome the problem of drift or opening occurrence (Simas, 2008; Simas et al., 2009; Guenther et al., 2008; Fontan, 2007). Again, PPR and PPPS virtual chains are usually used to this end, because they decouple position and orientation errors, and they do not introduce singularities to the resulting chain (Fontan, 2007). The network matrix $\mathbf{N}$, in this case, has to be partitioned considering the error chains:

$$\mathbf{N}_s\dot{\mathbf{q}}_s + \mathbf{N}_p\dot{\mathbf{q}}_p + \mathbf{N}_e\dot{\mathbf{q}}_e = 0 \tag{12}$$

$\dot{\mathbf{q}}_s$ (11) is modified by adding pose error, resulting in Eq. 13, whose stability was proved in (Simas, 2008).

$$\dot{\mathbf{q}}_s = -\mathbf{N}_s^{-1}\left(\mathbf{N}_p\dot{\mathbf{q}}_p - \mathbf{N}_e\mathbf{K}_e\mathbf{q}_e\right) \tag{13}$$

Pose error is determined by use of the successive screw displacements method, observing that the overall homogeneous transformation of a link relative to itself, in a closed chain, is an identity matrix. This leads to the error matrix

$$\mathbf{E} = \left\{\mathbf{A}_{p_1}\mathbf{A}_{p_2}\cdots\mathbf{A}_{p_{np}}\mathbf{A}_{s_1}\mathbf{A}_{s_2}\cdots\mathbf{A}_{s_{ns}}\right\}^{-1} = \left\{\prod_{i=1}^{n_p}\mathbf{A}_{p_i}\prod_{j=1}^{n_s}\mathbf{A}_{s_j}\right\}^{-1} = \begin{bmatrix}\mathbf{R}_e & \mathbf{p}_e\\ \mathbf{0} & 1\end{bmatrix} \tag{14}$$

This solution is effective even with simple numerical integration methods, such as Euler method. For an interval Δt and known velocities in t_{k-1} instant, joint positions can be calculated as

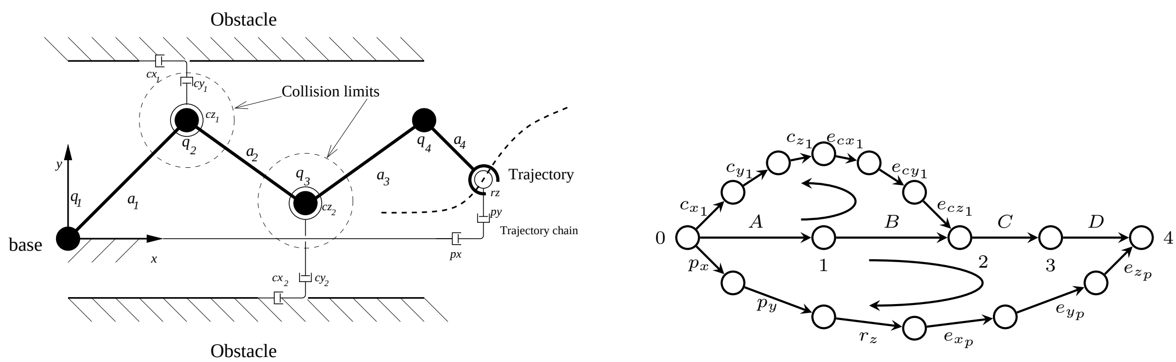
$$\mathbf{q}_s(t_k) = \mathbf{q}_s(t_{k-1}) - \mathbf{N}_s^{-1}(t_{k-1})\left(\mathbf{N}_p(t_{k-1})\dot{\mathbf{q}}_p - \mathbf{N}_e(t_{k-1})\mathbf{K}_e\mathbf{q}_e\right)\Delta t \tag{15}$$

Besides its asymptotic stability, it could be desired to minimise error along all movement. To do so, an inner integration loop can be used, where the desired primary joints positions are fixed, and the iterations occurs until the error is between admissible tolerances (Simas et al., 2009). As $\dot{\mathbf{q}}_p = 0$, Eq. 13 simplifies to

$$\dot{\mathbf{q}}_s = \mathbf{N}_s^{-1}\mathbf{N}_e\mathbf{K}_e\mathbf{q}_e \tag{16}$$

4. EXAMPLE

A numerical simulation of a RRRR planar manipulator is presented to illustrate and validate the present theory written in this paper. The manipulator structure is depicted in Figure 4a. A PPR virtual chain is used to impose end-effector movement, and there are two other PPR chains to monitor the proximity of known obstacles. Error virtual chains are not shown, in order to simplify the drawing. The graph representing the system is plotted in Figure 4b. It presents the manipulator, the virtual chain that imposes movement to the end-effector and the virtual chain that imposes constraints to link 2, in order to avoid the obstacle. Each circuit in the graph has an error virtual chain represented. This example was already explored by Simas and Fontan, in a structured, completely mapped constrained environment. They used virtual chains to monitor and to impose the collision avoidance strategy.



(a) (b)

Figure 4: RRRR planar manipulator

To simulate a time-varying environment, it will be considered that the upper wall moved vertically, in a way the robot model cannot predict. In that case, monitoring using a virtual chain is not an option. The use of sensor information replaces the virtual chain monitor, and during the monitoring state, the system is modelled as a single-loop one, disregarding the virtual chains relative to obstacle avoidance. p_x, p_y and r_z joints are chosen as primary, while joints 1, 2, 3 and 4 are secondary.

When sensors detect proximity of an obstacle, system model is changed to one that have an additional loop concerning with virtual actuation between the obstacle and the link which is near to it (link 2 is near the upper wall, and link 3 is near the lower wall). In this collision avoidance state, p_x, p_y, r_z and c_y joints are chosen as primary. Joints 1, 2, 3 and 4 remain as secondary. As soon as sensors do not detect proximity of an obstacle, the state returns to monitoring state, and the model changes back to the single-loop one. Figure 5 illustrates the task performed, where the end-effector has to follow an elliptical path twice.

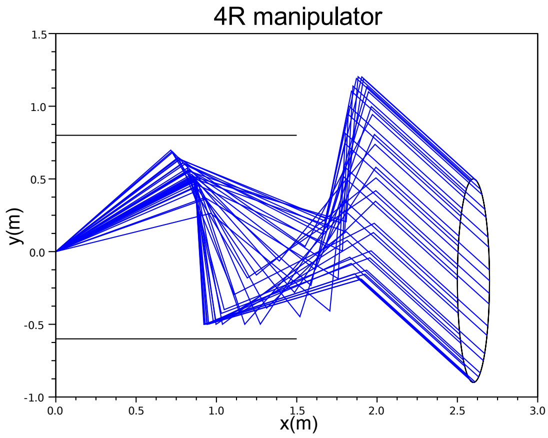
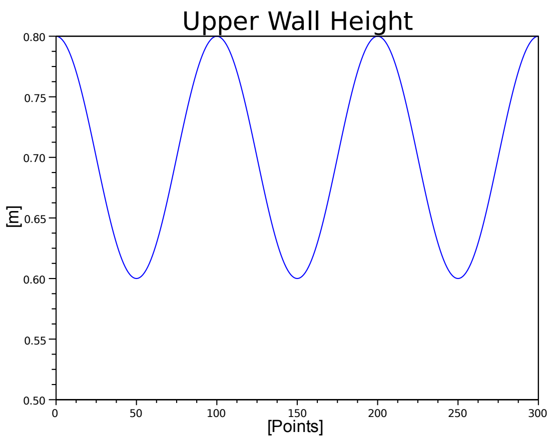
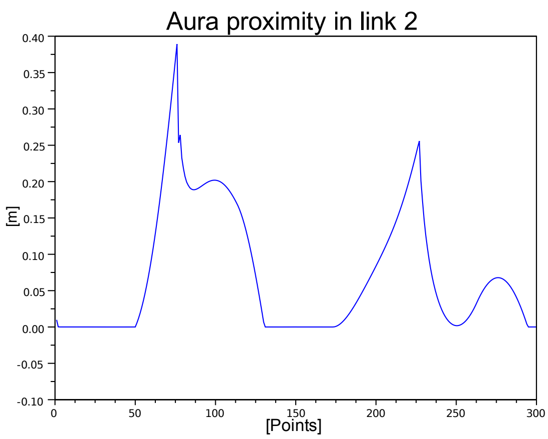


Figure 5: RRRR manipulator tracking an elliptical path

The upper wall vertical moves in a sinusoidal way, performing three cycles during the simulation, as it is shown in Figure 6a. The distance variation between the wall and the aura in link 2 is plotted in Figure 6b.



(a)



(b)

Figure 6: (a) Upper wall movement; (b) aura proximity

Real joint positions are depicted in Figure 7. It can be observed that the moving obstacle difficult movement conservation. It can be also noted that model switching produces discontinuities on the movement. Fontan and Santos have proposed some strategies to reduce this effects in case of known environments (Fontan, 2007; Santos, 2006). This problem can be more relevant in uncertain environments, because of the continuous and unpredictable switchings, and this is a topic which needs further study.

This example considered the avoidance of only one obstacle. The example shows two collision possibilities, but only one is handled at any time. Simulations concerning two or more simultaneous obstacles and considering their shapes will be explored in future work.

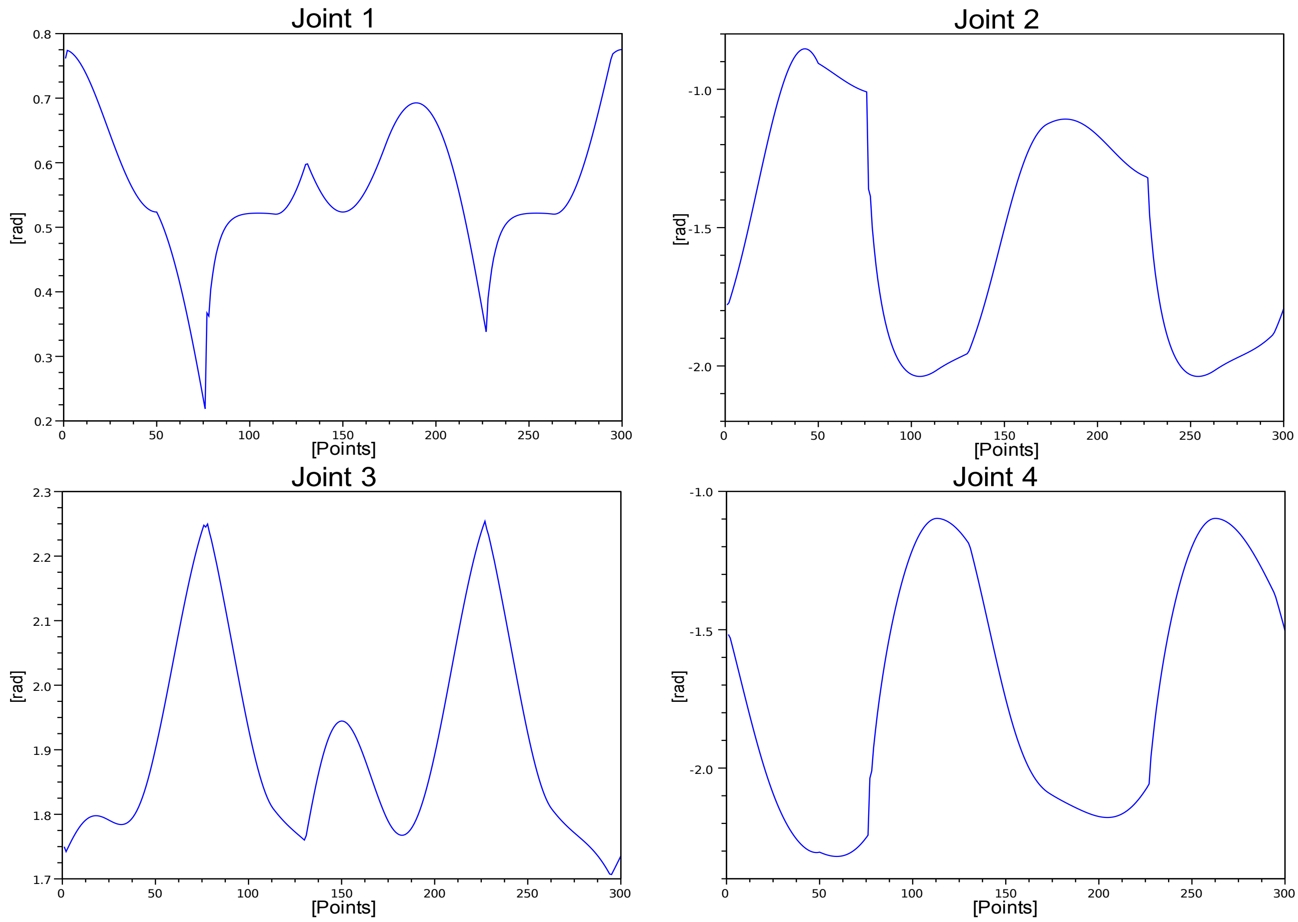


Figure 7: 4R planar manipulator joints positions

5. CONCLUSIONS

This work presented a new approach to solve inverse kinematics of manipulators operating in uncertain environments, which are poorly-mapped/unstructured and usually time-varying. This kind of environment is increasingly common in new manipulator applications, and motivates several researches, which rely on sensor information and the use of redundancy. To solve inverse kinematics, this work proposed an alternative to the traditional method of pseudo-inverse resolution and its derivatives. The fundamentals of the kinematic constraints method were shortly reproduced, and an example of its use was presented, where it was shown the effectiveness of this method. This extension of the method is an innovation in relation to previous work on obstacle avoidance in unstructured environments, and is the contribution of this paper. Further work will be needed, in order to reduce descontinuities caused by model switching, to address the multiple obstacle problem and to evaluate the performance of the method in experimental implementation. Future work will present this results.

6. ACKNOWLEDGEMENTS

This work was partially supported by *Fundação Coordenação de Aperfeiçoamento de Pessoal de Nível Superior* (CAPES), Brazil.

7. REFERENCES

Bicchi, A., Peshkin, M., and Colgate, J. (2008). *Safety for Human-Robot Interaction*, chapter 43, pages 1335–1348. Springer Tracts in Advanced Robotics. Springer Berlin Heidelberg, Heidelberg.

Campos, A. (2004). *Cinemática Diferencial de Manipuladores Empregando Cadeias Virtuais*. PhD thesis, Universidade Federal de Santa Catarina.

Campos, A., Guenther, R., and Martins, D. (2005). Differential kinematics of serial manipulators using virtual chains. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 27(4):345–356.

Ceccarelli, M. (2000). Screw axis defined by giulio mozzi in 1763 and early studies on helicoidal motion. *Mechanism and Machine Theory*, 35(6):761–770.

Cheung, E. and Lumelsky, V. (1989). Proximity sensing in robot manipulator motion planning: system and implementation issues. *IEEE Transactions on Robotics and Automation*, 5(6):740–751.

Chiaverini, S., Oriolo, G., and Walker, I. (2008). *Kinematically Redundant Manipulators*, cap. 11, pages 245–268. Springer Berlin Heidelberg, Heidelberg.

Dai, J. (2006). An historical review of the theoretical development of rigid body displacements from rodrigues parameters to the finite twist. *Mechanism and Machine Theory*, 41(1):41–52.

Davies, T. H. (1981). Kirchhoff's circulation law applied to multi-loop kinematic chains. *Mechanism and Machine Theory*, 16(3):171–183.

Fontan, D. (2007). Implementação da cinemática inversa de robôs redundantes operando em ambientes confinados no projeto roboturb. Dissertação de mestrado, Departamento de Engenharia Mecânica - UFSC, Florianópolis.

Guenther, R., Simas, H., da Cruz, D., and Martins, D. (2008). A new integration method for differential inverse kinematics of closed-chain robots. In *ACBM Symposium Series In Mechatronics*, volume 3 of *ACBM Symposium Series*, pages 225–235. ACBM, Rio de Janeiro, Brasil.

Hunt, K. (2000). Don't cross-thread the screw. In *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*, pages 1–37, Cambridge, United Kingdom. University of Cambridge - Trinity College, Cambridge University Press.

Iwata, H., Hoshino, H., Morita, T., and Sugano, S. (2001). Force detectable surface covers for humanoid robots. In *Proc. IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, volume 2, pages 1205–1210. IEEE / ASME.

Lee, K.-K. and Buss, M. (2007). Obstacle avoidance for redundant robots using jacobian transpose method. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS 2007*, pages 3509–3514. IEEE.

Lumelsky, V. (1987). Effect of kinematics on motion planning for planar robot arms moving amidst unknown obstacles. *IEEE Journal of Robotics and Automation*, 3(3):207–223.

Martins, D. and Carboni, A. P. (2008). Variety and connectivity in kinematic chains. *Mechanism and Machine Theory*, 43(10):1236–1252.

Santos, C. H. F. (2006). *Movimento Coordenado de Sistemas Veículo-Manipulador Submarinos Utilizando Técnicas de Inteligência Artificial e Sistemas Híbridos*. PhD thesis, Universidade Federal de Santa Catarina, Florianópolis.

Santos, C. H. F. d., Guenther, R., Martins, D., and De Pieri, E. R. (2006). Virtual kinematic chains to solve the underwater vehicle-manipulator systems redundancy. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 28:354 – 361.

Simas, H. (2008). *Planejamento de Trajetórias e Evitamento de Colisão em Tarefas de Manipuladores Redundantes Operando em Ambientes Confinados*. PhD thesis, Departamento de Engenharia Mecânica - UFSC, Florianópolis.

Simas, H., da Cruz, D., Guenther, R., and Martins, D. (2008). A collision avoidance method using assur virtual chains. In *ACBM Symposium Series In Mechatronics*, volume 3 of *ACBM Symposium Series*, pages pp 316–325. ACBM, Rio de Janeiro, Brasil.

Simas, H., Guenther, R., da Cruz, D. F. M., and Martins, D. (2009). A new method to solve robot inverse kinematics using assur virtual chains. *Robotica*, Forthcoming:1–10.

Tsai, L.-W. (1999). *Robot Analysis: The Mechanics of Serial and Parallel Manipulators*. Wiley-Interscience, New York.

Tsai, L.-W. (2000). *Mechanism Design: Enumeration of Kinematic Structures According to Function*. CRC-Press, Boca Raton, Florida.

Um, D. and Hung, W. (2006). A novel infrared proximity array sensor for 3d visual sensing: Modeling and applications. In *Proc. IEEE Conference on Robotics, Automation and Mechatronics*, pages 1–6. IEEE.

Wang, S., Bao, J., and Fu, Y. (2007). Real-time motion planning for robot manipulators in unknown environments using infrared sensors. *Robotica*, 25(02):201–211.

8. Responsibility notice

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