DEXTEROUS TASK-PRIORITY BASED REDUNDANCY RESOLUTION FOR UNDERWATER MANIPULATOR SYSTEMS

Serdar Soylu, Bradley J. Buckham*, and Ron P. Podhorodeski
Department of Mechanical Engineering,
University of Victoria, P.O. Box 3055, Victoria, B.C., Canada, V8W 3P6
{serdar, bbuckham, podhoro}@me.uvic.ca
*Corresponding author: Fax: (250) 721-6035

Received June 2007, Accepted November 2007
No. 07-CSME-61, E.I.C. Accession 3030

ABSTRACT

The problem of redundancy resolution for underwater remote vehicle-manipulator systems (**URVM**) is addressed in the current work. In URVM applications, it is beneficial to have the underwater remote vehicle (**URV**) hold station using its thrusters while a human pilot operates the serial manipulator. This provides a stable platform for the manipulator and eases the pilot's job drastically when current and/or tether disturbances are present. However, when following this objective, the redundancy of the URVM as a whole is wasted; the four actively controlled motions of the URV are not used to improve the efficacy of the manipulator task. In fact, this standard operating procedure frequently puts the manipulator into near singular configurations. This is not desirable from the manipulator controller standpoint since near singular configurations result in undesirably high joint velocities and oscillations. In this work, a new heuristic approach based on the task-priority redundancy resolution scheme is applied to the URVM. The proposed approach provides a means to avoid singular configurations of the manipulator, and provides dexterous manipulation by using the URV's mobility in an optimal, coordinated manner. This scheme is particularly useful for remote systems where an a priori trajectory generator is not applicable. Numerical case studies are developed to demonstrate the effectiveness of the technique.

RÉSOLUTION REDONDANTE BASÉE SUR LA PRIORITÉ POUR LES SYSTÈMES DE MANIPULATEURS SOUS-MARINS POUR TÂCHES DEXTRES

RÉSUMÉ

Le problème de la résolution redondante pour les systèmes de véhicule sous-marin manipulateurs télécommandés (VSMT) est résolu dans ce travail. Dans les applications des VSMT, il est bénéfique que le véhicule sous-marin télécommandé (VST) maintienne sa position avec ses propulseurs tandis qu'un pilote humain actionne le manipulateur sériel. Cela fournit une plate-forme stable pour le manipulateur et facilite radicalement la tâche du pilote en présence de perturbations dues au courant et/ou au câble d'attache. Cependant, en poursuivant cet objectif, la redondance du VSMT est gâchée: les quatre déplacements activement commandés par le VST ne sont pas utilisés pour augmenter l'efficacité de la tâche du manipulateur. En fait, cette procédure habituelle d'opération place fréquemment le manipulateur en configurations quasi-singulières. Cela n'est pas souhaitable du point de vue de l'opérateur du manipulateur car lorsque le système est proche d'une configuration singulière, des vitesses articulaires élevées et des oscillations involontaires se produisent. Dans ce travail, une nouvelle approche heuristique basée sur le schéma de résolution redondante des priorités de tâches est appliquée au VSMT afin de concilier différents objectifs avec les degrés de liberté collectifs du VSMT. La méthode permet d'augmenter la dextérité de la composante manipulateur du VSMT, en utilisant les déplacements du VST pour améliorer le positionnement du manipulateur, tout en maintenant les déplacements du VST à un minimum. Des études de cas numériques sont développées pour présenter l'efficacité de la technique proposée.

1 INTRODUCTION

Underwater remote vehicles (**URVs**) equipped with robotic manipulators play an important role in a number of shallow and deep-water missions for marine science, oil and gas extraction, exploration and salvage [1]. In these applications, the URV is used as a mobile platform that delivers the robotic manipulator to a subsea work site. The motions of the URV and the manipulator are guided independently by a human pilot on a surface support vessel through a long slender tether that provides power and telemetry. For detailed surveys, the URV motion can be accomplished by an on-board URV controller. These controllers use URV state feedback provided by acoustic and inertial positioning systems [2], and dynamic models to intelligently control the conventional thrusters arranged on the URV chassis. Manipulator units are generally add-on technologies produced by independent manufacturers, and hence the manipulator most often has an independent control system. The desired manipulator joint motions are created using a teleoperated master-slave arm configuration. Driving the passive master-arm, the human pilot sets the desired end-effector position and orientation that is to be duplicated by the submerged slave arm, provided the URV can hold station.

However, during the URV manoeuvre, the pilot encounters enormous difficulties. The URV thrusters rely on momentum transfer to a fluid and have an inherent lag in their response to pilot inputs. Furthermore, URV's are bluff bodies designed for omnidirectional operation and have poor drag characteristics, which slow the response of the URV to the pilot's command. As such, the manipulator joints are relied upon for detailed interaction with the subsea environment, while the URV thrusters are generally used to try and counter tether and current disturbances. However, this URV functionality is compromised by the limited visual and navigational feedback available to the pilot, and the subsequent inability to sense disturbances being exerted by the tether and the current. Furthermore, when the movement is replicated by the slave arm, the inertial and hydrodynamic drag associated with the manipulator links create reactions at the manipulator-URV junction. The reaction loads act as disturbances to the URV motion which in turn disturbs the placement of the end-effector [1-4]. These factors make it very difficult for an individual to synchronize the thrusters and manipulator commands, and often two pilots will work together during deployment.

It is proposed that the URV and the manipulator motion be coordinated such that a consolidated controller amalgamates URV navigation data, manipulator state feedback, and a single pilot command to achieve a desired end-effector motion. The consolidated system is referred to as an underwater remote vehicle-manipulator (**URVM**). By coordinating the collective degrees of freedom of the URVM in response to a single pilot input, the operator would only be concerned with driving the end-effector as opposed to operating the end-effector and the URV separately. As a consequence of this collective mode of operation, the URVM efficacy would be greatly improved, and the scope of the detailed operations required by large-scale projects such as VENUS [5] and NEPTUNE [6], which involves setting up a cable-linked seafloor observatory, would become possible.

In addition to the manipulator revolute joints, the URV itself contributes four active degrees of freedom, including surge (forward), sway (lateral) and heave (vertical) translations, and a yawing rotation about the vertical axis. Also, the URV is a free-floating body, and thus moves in two additional rotations: pitch and roll. However, these motions are not controlled, but rather are mitigated by a strong buoyant restoring moment. Due to its kinematic redundancy, the URVM system admits an infinite number of joint-space solutions for a given end-effector position and orientation. This allows one to use the available redundancy to achieve additional objectives besides the given end-effector task using a redundancy resolution technique. The primary objective is to coordinate the URV and the manipulator motion for a given end-effector path, and eliminate the need for the pilot intervention in the URV motion. In addition to that, various optimization criteria (e.g., reduction of fuel consumption, increase of system

manipulability), can be fulfilled by using those degrees of freedom not needed in the satisfaction of the primary objective.

In URVM applications, there are additional constraints not present in most land-based redundant manipulators. These constraints are mostly due to the dynamics of the URV: it is not always desirable to use the URV surge, sway, heave, and yaw motions extensively in placing the end-effector for several reasons. Firstly, the small URV movements cause significant changes in the end-effector location since the small rotational URV motions are amplified by the link lengths. Secondly, the URV consumes relatively more energy than the manipulator for a given motion due to its larger inertia [3, 7, 8]. The consideration of the large inertia of the URV is important since the redundancy resolution scheme could ask the URV to move in a manner that is not possible. Thirdly, the control of the URV is relatively more difficult since its response time is much slower [3]. For these reasons, approximately 35 per cent of the total work done in a typical subsea task requires keeping the URV stationary [9].

The implementations of redundancy resolution methods to the URVM systems have been documented in only a few existing works [7]. The singularity robust task-priority redundancy resolution [10], which was originally proposed in [11] and [12], was shown to be useful for a URVM in [8] due to its multitask capabilities. In [8], the secondary task was set to be holding the vehicle stationary, and the manipulator singularity avoidance was realized by constraining joint motion ranges. However, this limits the available workspace of the manipulator. In [13], the kinematic redundancy is utilized to minimize the total hydrodynamic drag forces experienced by a URVM system in an effort to reduce the energy consumption. However, the different dynamic characteristics of the URV and the manipulator were not addressed. In [7], the singularity robust task-priority redundancy resolution is merged with a fuzzy technique to resolve the URV-manipulator coordination. It was shown that the fuzzy method provides a versatile tool to handle multiple secondary tasks. However, the proposed scheme does not provide a means to hold the vehicle motionless when the commanded end-effector location is within the reach of the manipulator. A screw theory method, combined with the Davies method to represent the differential motion, was implemented to solve the inverse kinematics of URVM systems in [14]. The minimization of the URV motion was realized by imposing kinematic constraints on velocity magnitudes. However, the singularity problem was not addressed in the same paper. The problem of eliminating unnecessary URV motions while avoiding manipulator singular configurations was addressed in [15], where a fuzzy hybrid system was proposed as a solution to this problem. However, the method proposed in [15] requires complex fuzzy rules for guiding the URV motion in a manner consistent with the predefined "hysteresis" behaviour.

Among the existing redundancy resolution schemes for the URVM systems, the task-priority approach is prominent. However, the direct implementation of the task-priority approach presented in prior works consistently forces the manipulator to work in, or near, degenerate configurations resulting in undesirably high joint velocities and oscillations. The solution to this problem requires either complex fuzzy rules or a priori trajectory knowledge [7, 8, 15]. Also, unlike a redundant land-based manipulator, the redundancy of the URVM can not be relied upon when the manipulator hits degeneracy: the response time of the URV motions may not meet that of the subsequent pilot end-effector command. In addition, if the URVM finishes a task in a singular configuration such as when the manipulator is stretched out, and the next task is to lift a heavy object, it may be impossible to lift up the material without exceeding the actuator torque limit, or the righting moment provided by the URV buoyancy module. Furthermore, when the manipulator is at a singular configuration and the URV is held fixed, arbitrary motion of the end-effector may not be possible due to the loss of degree(s) of freedom. To circumvent these potential problems, the URV mobility can be freed to provide only the slow motion of the manipulator base in an attempt to increase the dexterity of the manipulator, and move away from singular configurations.

In order to solve the problem of eliminating unnecessary URV motions while avoiding manipulator singular configurations in URVM systems, a new heuristic approach is presented. The proposed approach is a variation of the task-priority redundancy resolution. In this method, low priority secondary tasks

attempt to exploit slow URV motions and an increase in the system manipulability is sought when necessary. An on-line solution for the occurrence of algorithmic and kinematic singularities is used. To this end, a mobility index that is used to gauge proximity to singular configurations is used. The mobility index supervises the URV movement and its coordination with the end-effector movement. This scheme is particularly useful for remote systems where an a priori trajectory generator is not applicable.

2 THEORETICAL BACKGROUND

2.1 Forward Velocity Problem

The task-space velocity vector $\dot{\mathbf{x}} \in \mathbb{R}^m$ and joint-space velocity vector $\dot{\mathbf{q}} \in \mathbb{R}^n$ are related by:

$$\dot{\mathbf{x}} = \mathbf{J}\dot{\mathbf{q}} \tag{1}$$

where $\mathbf{J} \in \mathbb{R}^{m \times n}$ is the Jacobian matrix. For a kinematically redundant manipulator, $(n > m)$ and there are an infinity set of joint rates solution that can complete the desired end-effector motion, $\dot{\mathbf{x}}$. The existing solution procedures for Eq. (1) apply additional constraints in order to create a deterministic set of equations. The methods are distinguished by the influence of these constraint equations.

2.2 Pseudo Inverse Solution

The minimum-norm solution to the inverse kinematics problem associated with Eq. (1) that satisfies $\dot{\mathbf{x}} = \mathbf{J}\dot{\mathbf{q}}$ and minimizes $\|\dot{\mathbf{q}}\|_2 = \sqrt{\dot{\mathbf{q}}^T\dot{\mathbf{q}}}$ is given by:

$$\dot{\mathbf{q}} = \mathbf{J}^{\dagger}\dot{\mathbf{x}} \tag{2}$$

where $\mathbf{J}^{\dagger} = \mathbf{J}^T\left(\mathbf{J}\mathbf{J}^T\right)^{-1}$ is the right Moore-Penrose pseudoinverse [16]. However, this solution does not guarantee singularity avoidance.

2.3 Projected Gradient Method

A general solution to Eq. (1), that sacrifices the minimum norm properties, can be obtained by adding a null space solution to the minimum-norm solution [16]:

$$\dot{\mathbf{q}} = \mathbf{J}^{\dagger}\dot{\mathbf{x}} + \left(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J}\right)\dot{\mathbf{q}}_0 \tag{3}$$

where $\dot{\mathbf{q}}_0 \in \mathbb{R}^n$ is an arbitrary joint velocity vector. The term $\left(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J}\right)$ is called the projection operator, and it projects the vector $\dot{\mathbf{q}}_0$ onto the null space of the end effector Jacobian matrix. The resulting null-space velocities generate motions within the serial manipulator that do not produce any end-effector motion. Thus, these "internal" motions can be exploited to achieve additional objectives. Liegeois [17] proposed that the arbitrary vector $\dot{\mathbf{q}}_0$ be the gradient of a scalar objective (potential) function $h(\mathbf{q})$:

$$\dot{\mathbf{q}} = \mathbf{J}^{\dagger}\dot{\mathbf{x}} + \left(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J}\right)\left(\lambda \nabla h(\mathbf{q})\right) \tag{4}$$

where negative values of the scalar gain λ minimize $h(\mathbf{q})$, and positive values maximize this objective function. Equation (4) is called the projected gradient method.

2.4 Task Priority Redundancy Resolution

The task priority redundancy resolution technique attempts to divide a required task into subtasks according to the order of priority [11, 12]. To resolve conflicting directives from the multiple tasks, a hierarchy is established such that subtasks with lower priority are realized using extra degrees of freedom that are not taken by higher priority subtasks. When conflicts between tasks arise, the solution that is in favour of the higher priority task is realized.

The task priority redundancy resolution technique can be viewed as a variation of the projection gradient method. Instead of projecting the gradient of a scalar objective function through the projection

operator, lower priority tasks are sequentially projected onto the null space of higher priority tasks. For the sake of simplicity, consider a double-task case in which the primary task $\dot{\mathbf{x}}_p \in \mathbb{R}^{m_1}$ has high priority, and the secondary task $\dot{\mathbf{x}}_s \in \mathbb{R}^{m_2}$ has low priority.

$$\dot{\mathbf{x}}_p = \mathbf{J}_p \dot{\mathbf{q}} \tag{5}$$

$$\dot{\mathbf{x}}_s = \mathbf{J}_s \dot{\mathbf{q}} \tag{6}$$

where $\mathbf{J}_p \in \mathbb{R}^{m_1 \times n}$ and $\mathbf{J}_s \in \mathbb{R}^{m_2 \times n}$ are the primary-task Jacobian matrix and the secondary task Jacobian matrix, respectively. The general solution of Eq. (5) using the pseudoinverse is as follows:

$$\dot{\mathbf{q}} = \mathbf{J}_p \dot{\mathbf{x}}_p + \left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)\dot{\mathbf{q}}_0 \tag{7}$$

Substituting Eq. (7) into the secondary task forward velocity kinematics in Eq. (6) yields:

$$\mathbf{J}_s\left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)\dot{\mathbf{q}}_0 = \dot{\mathbf{x}}_s - \mathbf{J}_s\mathbf{J}_p^{\dagger}\dot{\mathbf{x}}_p \tag{8}$$

Then the unknown $\dot{\mathbf{q}}_0$ that minimizes $\left\|\dot{\mathbf{x}}_s - \mathbf{J}_s\dot{\mathbf{q}}\right\|_2$ is given by:

$$\dot{\mathbf{q}}_0 = \hat{\mathbf{J}}_s^{\dagger}\left(\dot{\mathbf{x}}_s - \mathbf{J}_s\mathbf{J}_p^{\dagger}\dot{\mathbf{x}}_p\right) \tag{9}$$

where $\hat{\mathbf{J}}_s = \left(\mathbf{J}_s\left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)\right)$. The joint space velocities are obtained by substituting Eq. (9) into Eq. (7). Exploiting the fact that the null-space projection operator is Hermitian and idempotent [11-12], $\dot{\mathbf{q}}$ is found as:

$$\dot{\mathbf{q}} = \mathbf{J}_p^{\dagger}\dot{\mathbf{x}}_p + \left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)^{\dagger}\hat{\mathbf{J}}_s^{\dagger}\left(\dot{\mathbf{x}}_s - \mathbf{J}_s\mathbf{J}_p^{\dagger}\dot{\mathbf{x}}_p\right) \tag{10}$$

However, Eq. (10) is vulnerable to the occurrence of kinematic and algorithmic singularities. A kinematic singularity corresponds to a loss in manipulator degree of freedom and a drop in the level of redundancy in the system. The primary Jacobian pseudoinverse is given by:

$$\mathbf{J}_p^{\dagger} = \sum_{i=1}^{r}\mathbf{v}_i\mathbf{u}_i^T/\sigma_i \tag{11}$$

where r is the rank of the primary Jacobian, and $\mathbf{v}_i$ and $\mathbf{u}_i$ are the right and left singular vectors of $\mathbf{J}_p$, respectively [16]. Hence, one can detect the occurrence of kinematic singularity by monitoring the corresponding singular value σ_r. Given that $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_r \geq 0$, at the singular configuration, σ_r becomes zero. This means that the joint-space velocities along $\mathbf{v}_r$ fall in the null-space of $\mathbf{J}_p$, and as a result the end-effector velocities along $\mathbf{u}_r$ become unrealizable [10]. Note that there could be cases where more than one singular value is zero. This situation corresponds to a further reduction in the range space of the primary Jacobian.

An algorithmic singularity occurs when the matrix $\hat{\mathbf{J}}_s = \left(\mathbf{J}_s\left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)\right)$ becomes rank deficient. At an algorithmic singularity, the null space of the primary task and the secondary task are linearly dependent, i.e., $N\left(\mathbf{J}_p\right) \cap N\left(\mathbf{J}_s\right) \neq 0$, reflecting the fact that the primary and secondary tasks can not be satisfied simultaneously. In this case, high joint velocities and oscillations occur [10]. In order to circumvent the algorithmic singularity problem, the components of the secondary task solution that conflict with the primary task solution must be removed.

2.5 Singularity Robust Task Priority Redundancy Solution

To alleviate the kinematic singularity problem in Eq. (10), the damped-least squares inverse given as $\mathbf{J}_p^* = \sum_{i=1}^{r}\left(\sigma_i/(\sigma_i^2+\eta^2)\right)\mathbf{v}_i\mathbf{u}_i^T$ is used in lieu of the pseudoinverse [18]. A non-null damping factor $\eta \in \mathbb{R}$ prevents the denominator from becoming zero. Thus, it provides continuity and good conditioning to the solution, but this improvement is obtained at the expense of an increased residual error.

To alleviate algorithmic singularities, the general solution for the primary task, Eq. (7), can be equalized to the minimum-norm solution of the secondary task, $\mathbf{J}_s^{\dagger}\dot{\mathbf{x}}_s$, to obtain a solution that solves both the primary and secondary task simultaneously [10]:

$$\mathbf{J}_s^{\dagger}\dot{\mathbf{x}}_s = \mathbf{J}_p^{\dagger}\dot{\mathbf{x}}_p + \left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)\dot{\mathbf{q}}_0 \tag{12}$$

Eq. (12) can be solved for $\dot{\mathbf{q}}_0$ yielding:

$$\dot{\mathbf{q}}_0 = \left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)^{\dagger}\left(\mathbf{J}_s^{\dagger}\dot{\mathbf{x}}_s - \mathbf{J}_p^{\dagger}\dot{\mathbf{x}}_p\right) \tag{13}$$

Since $\left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)^{\dagger} = \left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)$ and $\mathbf{J}_p^{\dagger} = \mathbf{J}_p^{\dagger}\mathbf{J}_p\mathbf{J}_p^{\dagger}$, Eq. (13) can be simplified to:

$$\dot{\mathbf{q}}_0 = \left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)\mathbf{J}_s^{\dagger}\dot{\mathbf{x}}_s \tag{14}$$

Substituting Eq. (14) into Eq. (7) and using the idempotence of $\left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)$ yields:

$$\dot{\mathbf{q}} = \mathbf{J}_p^{\dagger}\dot{\mathbf{x}}_p + \left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)\mathbf{J}_s^{\dagger}\dot{\mathbf{x}}_s \tag{15}$$

In Eq. (15) the minimum-norm solutions to the primary and secondary inverse kinematic problems, Eq. (5) and Eq. (6) respectively, are obtained using the pseudoinverse of the corresponding Jacobian. The joint velocities computed for the secondary task are then projected onto the null space of the primary task to remove components in conflict with the primary task. This provides robustness in the presence of algorithmic singularities. To incorporate the robustness to the kinematic singularities, $\mathbf{J}^*$ is used instead of $\mathbf{J}^{\dagger}$ in Eq. (15). However, in URVM applications, $\mathbf{J}_p^{\dagger}$ is used as it is, since the primary Jacobian of the URVM system will always exhibit full rank due to the mobility of the URV [7]. The extension of Eq. (15) to highly redundant systems performing more than two tasks is presented in [19].

The direct implementation of Eq. (15) is prone to a numerical drift occurring when the joint rates are integrated forward in time to obtain URV/manipulator position values. In order to avoid this problem, the closed-loop version of Eq. (15) can be employed [7]:

$$\dot{\mathbf{q}} = \mathbf{J}_p^{\dagger}\left(\dot{\mathbf{x}}_p + \mathbf{K}_p\mathbf{e}_p\right) + \left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)\mathbf{J}_s^{\dagger}\left(\dot{\mathbf{x}}_s + \mathbf{K}_s\mathbf{e}_s\right) \tag{16}$$

where $\mathbf{K}_p$ and $\mathbf{K}_s$ are user-defined positive definite matrix gains, and $\mathbf{e}_p$ and $\mathbf{e}_s$ are the numerical construction errors defined as $\mathbf{x}_{p,d} - \mathbf{x}_p$ and $\mathbf{x}_{s,d} - \mathbf{x}_p$ with d denoting the desired values.

3 DEXTEROUS TASK PRIORITY APPROACH

3.1 Overview

It was shown in [8] that the direct implementation of the singularity robust task priority approach might drive the manipulator component into its singular configurations when the secondary task is defined as station keeping of the URV; it fails to coordinate the URV and the manipulator motion in an effective manner. Therefore, prevention of manipulator singularities must be included in the framework of the redundancy resolution method. As such, the functionality of the robust-task priority approach in accommodating multiple kinematic tasks can be fully utilized. In the following sections, the method

proposed to circumvent the singularity problem will be presented. The method provides a means to coordinate the URV and the manipulator motion in such a way that the manipulator singularity is avoided.

3.2 Mobility of URVM Systems

There is a strong connection between the kinematic singularities of the sole manipulator and the algorithmic singularities of the URVM system for the current application. The algorithmic singularities occur when $\mathbf{J}_p$ and $\mathbf{J}_s$ have common linearly dependent columns meaning the primary and secondary tasks are in conflict [20]. Given that the secondary task is to keep the URV at the current location, any conflict indicates that the end-effector can not be placed at a desired location without using the URV's mobility. This happens when a loss in the degree of freedom of the manipulator, a kinematic singularity, is experienced during the URV station keeping. In an effort to monitor the kinematic singularity of the sole manipulator, it is proposed that the collective Jacobian matrix $\mathbf{J}_a = \begin{bmatrix} \mathbf{J}_p^T & \mathbf{J}_s^T \end{bmatrix}^T \in \mathbb{R}^{(m_1+m_2)\times n}$ be used. The rank deficiency of $\mathbf{J}_a$ indicates a conflict in the primary and the secondary task since $\mathbf{J}_p$ and $\mathbf{J}_s$ always have full rank. Consequently, the rank of $\mathbf{J}_a$ can be used as the indicator of the sole manipulator singularity.

In an attempt to track manipulator singularity, it is useful to use a parameter that quantifies the closeness to the manipulator singular configurations. The following index based on Yoshikawa's measure of manipulability [21] can be used for that purpose:

$$\rho = \sqrt{\det\left(\mathbf{J}_a \mathbf{J}_a^T\right)} \tag{17}$$

Differently from Yoshikawa's measure of manipulability where the end-effector Jacobian matrix is used, the collective matrix $\mathbf{J}_a$ is used in Eq. (17) instead. To distinguish this difference, Eq. (17) will be called mobility index in the subsequent sections. With this arrangement, Eq. (17) can be used not only to gauge the level of proximity of the manipulator to the singular configurations, but also to specify the extent to which the URV should move to prevent the manipulator from hitting its near singular configurations. Using the Singular Value Decomposition (**SVD**) of the Jacobian matrix, $\mathbf{J}_a = \mathbf{U}\sum\mathbf{V}^T$, it can be shown that Eq. (17) is merely the product of the singular values, $\rho = \prod_{i=1}^{m} |\sigma_i|$ [21]. Given that $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_n \geq 0$ always holds, the mobility index is lower and upper bounded by $\sigma_n^n \leq \rho \leq \sigma_1^n$; leading to $\rho/\sigma_1^n \leq 1$. As such, the mobility index can be defined as ρ/σ_1^n for cases in which the normalization of the mobility index is necessary. Since the mobility index is dependent on the physical size of the mechanism, this normalization would be necessary if the proposed approach is used on different systems for a comparison purpose. The mobility index becomes zero only when the collective Jacobian matrix is not full rank.

In order to monitor how the mobility index changes with respect to the joint vector, the following analytical formula can be used [22]:

$$\frac{\partial \rho}{\partial q_i} = \rho \cdot trace\left(\frac{\partial \mathbf{J}_a}{\partial q_i}\mathbf{J}_a^{\dagger}\right), \quad i = 1 \ldots n \tag{18}$$

3.3 Mobility Index Based Dexterous Redundancy Resolution

The following redundancy resolution is proposed with $\gamma = \left(1 - sign\left(d\rho/dt\right)\right)/2$:

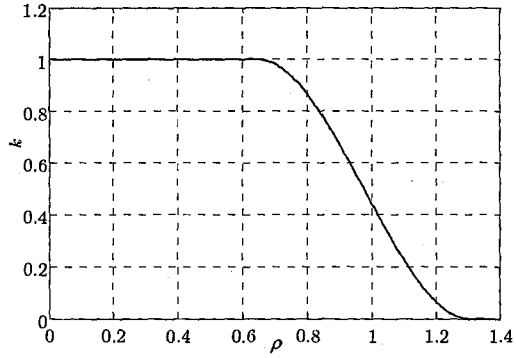
$$\dot{\mathbf{q}} = \mathbf{J}_{w,p}^{\dagger}\left(\dot{\mathbf{x}}_p + \mathbf{K}_p\mathbf{e}_p\right) + \left(\mathbf{I} - \mathbf{J}_{w,p}^{\dagger}\mathbf{J}_p\right)\left((1-\gamma k)\mathbf{J}_{w,s}^{\dagger}\left(\dot{\mathbf{x}}_s + \mathbf{K}_s\mathbf{e}_s\right) + \gamma k\left(\lambda\mathbf{W}^{-1}\frac{\partial\rho(\mathbf{J}_a)}{\partial\mathbf{q}}\right)\right) \tag{19}$$

which combines the gradient projection method given in Section 2.3 and the task-priority approach given in Section 2.5. The task-priority component is in charge of realizing minimal URV motion, whereas the gradient projection component is in charge of avoiding manipulator near singular configurations. In Eq. (19), the term $\mathbf{J}_w^{\dagger}$ is the weighted pseudoinverse that instantaneously solves the problem of minimizing $\dot{\mathbf{q}}^T\mathbf{W}\dot{\mathbf{q}}$ subject to $\dot{\mathbf{x}} = \mathbf{J}\dot{\mathbf{q}}$ in the form of $\dot{\mathbf{q}} = \mathbf{J}_w^{\dagger}\dot{\mathbf{x}}$:

$$\mathbf{J}_w^{\dagger} = \mathbf{W}^{-1}\mathbf{J}^T\left(\mathbf{J}\mathbf{W}^{-1}\mathbf{J}^T\right)^{-1} \tag{20}$$

with the inverse of the weighting matrix being defined as $\mathbf{W}^{-1} = diag(w_1, w_2, \ldots, w_n)$. This weighting matrix contains weight factors for each degree of freedom in a robotic system. The term λ is a positive constant that determines the convergence rate to the local minimum of the mobility index function. The limiting factors on the value of λ can be found in [23]. The term k is the sigmoid form shaping function [24] illustrated in Figure 1.

The success of the Gradient Projection Method relies on the fact that the projection operator $\left(\mathbf{I} - \mathbf{J}_p^{\dagger}\mathbf{J}_p\right)$ is semi-definite [25]. However, when the weighted pseudoinverse is used, the resulting matrix, $\left(\mathbf{I} - \mathbf{J}_{w,p}^{\dagger}\mathbf{J}_p\right)$, is no longer symmetric, and therefore is non-definite. For this reason, the gradient projection method must be modified. This can be done by postmultiplying the weighted null-space projection operator, $\left(\mathbf{I} - \mathbf{J}_{w,p}^{\dagger}\mathbf{J}_p\right)$, by the inverse of the positive-definite weighting matrix, $\mathbf{W}^{-1}$ [22]. The resulting matrix, $\left(\mathbf{I} - \mathbf{J}_{w,p}^{\dagger}\mathbf{J}_p\right)\mathbf{W}^{-1}$, is positive semi-definite. The proof can be found in [26]. The proof requires decomposing the positive definite weighting matrix as $\mathbf{W}^{-1} = \mathbf{U}^T\mathbf{U}$. Therefore, the weighting matrix must be a positive-definite matrix.



$$k = \begin{cases} 0 & \bar{\rho} \le \rho \\ 4\left(\dfrac{4\rho^3 - 9\rho^2\bar{\rho} + 6\bar{\rho}^2\rho - \bar{\rho}^3}{\bar{\rho}^3}\right), & \dfrac{\bar{\rho}}{2} \le \rho < \bar{\rho} \\ 1 & \rho < \dfrac{\bar{\rho}}{2} \end{cases}$$

Figure 1: Shaping function definition (right) and corresponding plot for $\bar{\rho} = 1.3$ (left).

An activation threshold value $\bar{\rho}$, determined based on the desired proximity from singular configurations, is set on the mobility index. As mentioned earlier, Eq. (19) can be thought of as the combination of the gradient projected method and the singularity robust task-priority approach. When activated, it works to move the manipulator away from the singular configurations. The task-priority component works as in Section 2.5, keeping the URV at the current location in an effort to provide a stable platform for the end-effector task.

In order to blend the task priority and projected gradient contributions, a shaping function is used to distribute the secondary task demand over the station-keeping and the singularity avoidance tasks. If the

singularity index is bigger than the designated threshold, k becomes zero, and Eq. (19) becomes equivalent to the task-priority approach of Section 2.4. In this case the URV could be still kept at rest while the manipulator is performing a task; hence, unnecessary URV motion is avoided. In cases where $\overline{\rho}/2 \le \rho < \overline{\rho}$, the shaping function distributes the secondary demand over the URV and the manipulator. When this happens, the URV is moved in proportion to the mobility index in an attempt to keep the manipulator away from a singular configuration. When the mobility index drops below half of the threshold, the emphasis is placed fully on the singularity avoidance. In this case, the URV-related secondary task is fully disregarded, and the available redundancy including the URV's mobility, is committed fully to the manipulator's singularity avoidance. However, the intervention of the projected gradient solution is not necessary when ρ is naturally increasing. As such, a discontinuous switching term, using the *sign* function defined as for $\dot{\rho} \le 0$, $sign(\dot{\rho}) = -1$ and for $\dot{\rho} > 0$, $sign(\dot{\rho}) = 1$, is added such that the singularity avoidance only intervenes in times of worsening manipulability.

In order to commit the full redundancy to the minimal URV motion criterion when the mobility index function is already an increasing function, the term γ must be included in the task-priority part of Eq. (19). If this term is not included, the full capacity of available redundancy can not be utilized by the system since it will be still weighted by the shaping function.

Equation (19) can be executed in real time since the primary and secondary Jacobians always exhibit full rank. As such, the time consuming SVD operation can be avoided when computing the weighted pseudoinverses. By exploiting this property, the analytical formula given in Eq. (20) can be used instead of the full SVD; leading to the suitability of Eq. (19) to real-time URVM applications.

4 SIMULATION

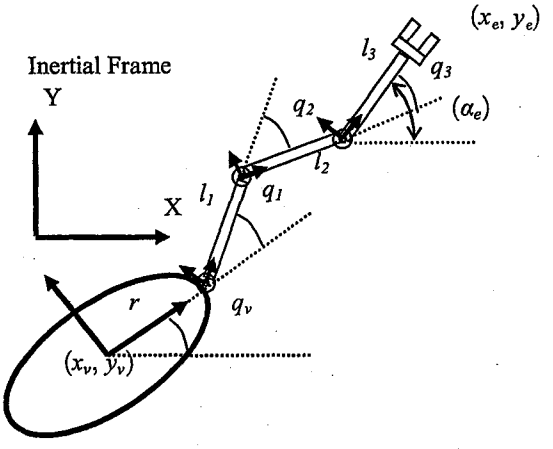


Figure 2: The kinematic chain of the planar underwater vehicle system with a 3 joint-DOF manipulator.

The URV system considered in this work is the Canadian Scientific Submersible Facility's ROPOS equipped with a 3-DOF manipulator as shown in Figure 2. For the sake of simplicity, only a planar end-effector task is considered. The state vector is given as $\mathbf{q} = [x_v \quad y_v \quad q_v \quad q_1 \quad q_2 \quad q_3]^T$, where x_v and y_v are the coordinates of the vehicle's center of mass expressed in the inertial frame, q_v is the yaw angle of the vehicle, and q_i is the joint position with respect to the body-fixed link coordinate system. The terms x_e and y_e are the coordinates of the end-effector, and α_e is the orientation of the end-effector. The distance from the center of mass of the vehicle to the first joint is $r = 1$ m. The distance from one link to another is denoted by l_i, and its values are 1m as well. The primary task is to make the end-effector

follow a predetermined trajectory. Therefore, the corresponding manipulation vector for the primary task is defined as $\mathbf{x}_p = [x_e \quad y_e]^T$ with $\mathbf{J}_p \in \mathbb{R}^{2\times 6}$. A series of waypoints were set for the end-effector and a continuous set of end-effector values were generated using a third order polynomial function with zero initial and final velocities. The system starts from the initial configuration of $\mathbf{q} = [0 \quad 0 \quad 0 \quad \pi/3 \quad -\pi/3 \quad \pi/3]^T$ m, rad. that corresponds to the end-effector position of $\mathbf{x}_p = [3, \quad 1.7321]^T$ m. The final end-effector location is $\mathbf{x}_p = [5, \quad 5.1962]^T$ m.

The secondary task variables are defined as $\mathbf{x}_s = [x_v \quad y_v \quad q_v]^T$ with $\mathbf{J}_s \in \mathbb{R}^{3\times 6}$. Note that this definition always leads to a full rank secondary task Jacobian matrix, $\mathbf{J}_s$. Provided that the primary task can be tracked using only the manipulator joints, it is desired to keep the URV location constant. To this end, the current URV position value is entered as the desired position values for the next step. This guarantees zero-URV motion as long as the desired end-effector location is within the current reach of the manipulator. In addition, this arrangement makes the URV follow the end-effector.

For the current implementation, the inverse of the weight matrix was chosen to be $\mathbf{W}^{-1} = diag(0.1 \quad 0.1 \quad \varepsilon \quad 1 \quad 1 \quad 1)$ with ε being a small positive number such as $\varepsilon \leq 10^{-8}$. Note that the third component of the weighting matrix corresponding to the yaw orientation of the URV is set to nearly zero. This forces the yawing motion of the URV to become zero. This arrangement makes the weighting matrix positive-definite. In the weighting matrix, bigger diagonal values require larger movements, whereas smaller values require smaller movements for the associated degree of freedom. In other words, it sets the motion preferences for each degree of freedom of the URVM system.

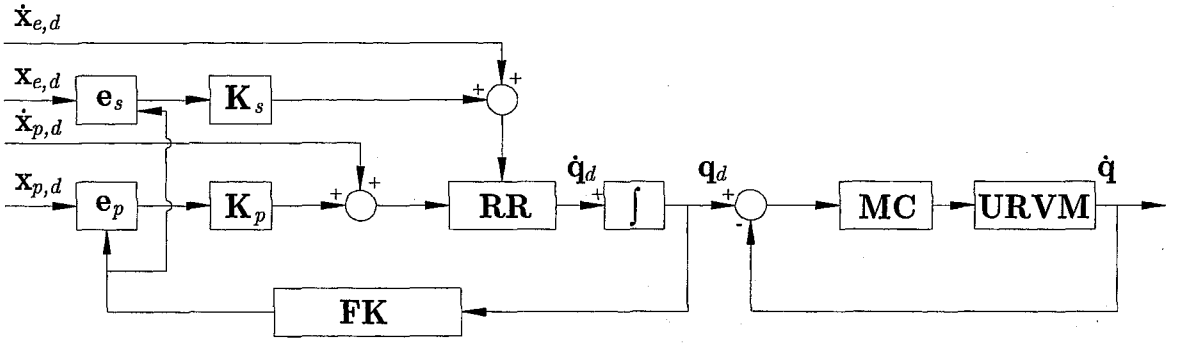


Figure 3: The simulation scheme consisting of the kinematic loop (left) and dynamic loop (right). In the figure, RR, MC, and FK stand for redundancy resolution, motion control and forward kinematics, respectively.

The time-domain simulation scheme that was used is illustrated in Figure 3. In the figure, the closed-loop redundancy resolution, Eq.(19), generates desired joint-space reference values for the dynamic control loop. The dynamic control loop computes the corresponding actuators forces, i.e. the URV thrusts and the manipulator torques, from the requested joint position and velocity values. For the current study, the computed-torque method of [27] is used.

The implementation of the computed torque method requires a dynamic model of the system given as:

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{D}\dot{\mathbf{q}} + \mathbf{G} = \tau \tag{21}$$

where $\mathbf{M} \in \mathbb{R}^{6\times 6}$ is the inertia matrix including the added mass effects, $\mathbf{C} \in \mathbb{R}^{6\times 6}$ is the matrix of centrifugal and Coriolis terms, $\mathbf{D} \in \mathbb{R}^{6\times 6}$ is the drag matrix, $\mathbf{G} \in \mathbb{R}^{6\times 1}$ is the vector of gravity and buoyancy forces and moments, and finally $\tau \in \mathbb{R}^{6\times 1}$ is the force vector acting on the URVM system. The dynamic parameters for ROPOS are taken from [28]. The computed-torque control law is given as:

$$\tau=\hat{\mathbf{M}}\left(\ddot{\mathbf{q}}_d+\mathbf{K}_v\left(\dot{\mathbf{q}}_d-\dot{\mathbf{q}}\right)+\mathbf{K}_c\left(\mathbf{q}_d-\mathbf{q}\right)\right)+\hat{\xi} \tag{22}$$

where $\hat{\mathbf{M}}$ and $\hat{\xi}=\hat{\mathbf{C}}\dot{\mathbf{q}}+\hat{\mathbf{D}}\dot{\mathbf{q}}+\hat{\mathbf{G}}$ are the estimates of the model parameters, and $\mathbf{K}_v$ and $\mathbf{K}_c$ are the positive gain matrices. Substituting Eq. (22) into Eq. (21) with the assumption of $\hat{\mathbf{M}}=\mathbf{M}$ and $\hat{\xi}=\xi$ yields:

$$\ddot{\mathbf{e}}+\mathbf{K}_v\dot{\mathbf{e}}+\mathbf{K}_c\mathbf{e}=\mathbf{0} \tag{23}$$

where "e" denotes errors defined as $\ddot{\mathbf{e}}=\ddot{\mathbf{q}}_d-\ddot{\mathbf{q}}$, $\dot{\mathbf{e}}=\dot{\mathbf{q}}_d-\dot{\mathbf{q}}$, and $\mathbf{e}=\mathbf{q}_d-\mathbf{q}$. Equation (23) guarantees the asymptotic reduction of the error.

The followed simulation scheme is illustrated in Figure 3. Equation (19) is implemented with $\lambda=10$, $\overline{\rho}=1.3$, $\mathbf{K}_p=[10\ \ 10]^T$ and $\mathbf{K}_s=[10\ \ 10\ \ 100]^T$, which are determined by trial and error. The simulation results are reported in Figure 4. The scheme produced joint positions and velocities that are realizable, as they show a smooth pattern, with no jerk. As can be seen from Figure 4, the primary task was successfully executed. The velocities of the state variables became null at the end of the simulation consistently with the desired initial and final velocities.

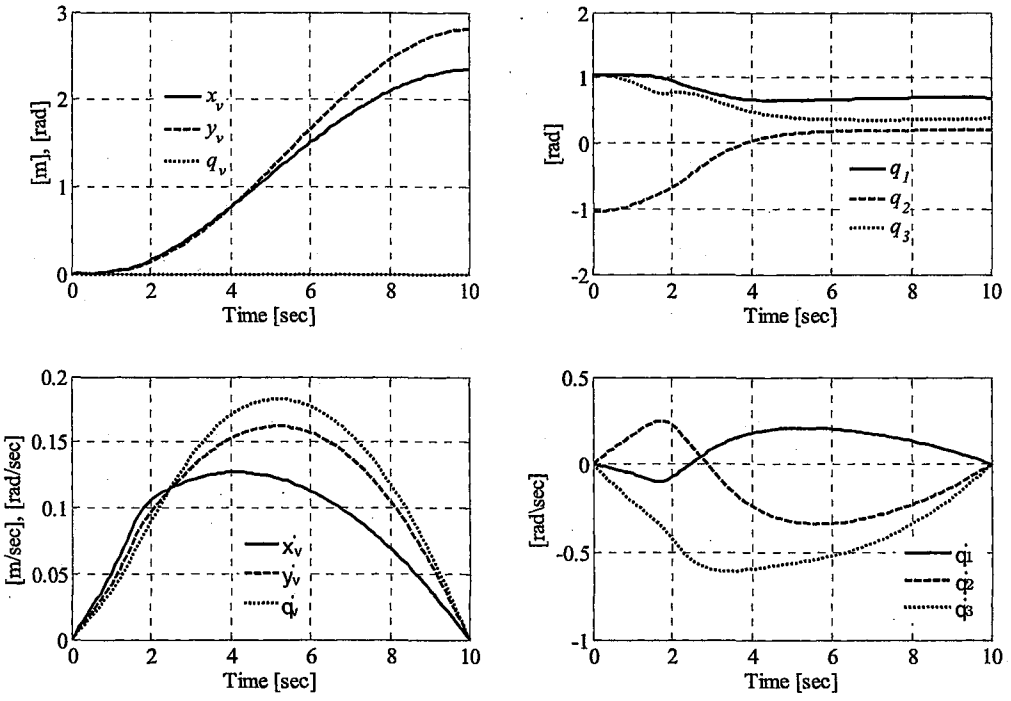


Figure 4: Time history of state variables for the URV (left figures) and the manipulator (right figures).

The robust task-priority approach has also been implemented with the same simulation parameters. The corresponding URVM manoeuvres produced by each method are illustrated in Figure 5. Both approaches successfully fulfilled the primary task. Despite the secondary task constraint, both methods caused the URV to move during the manoeuvre; it indicates that the trajectory goes out of the workspace of the sole manipulator. As can be seen from the left figure in Figure 5, the singularity robust task-priority approach drives the sole manipulator into a singular configuration, i.e., a stretched-out configuration for the current simulation case. On the other hand, the dexterous task-priority scheme does not force the manipulator component to hit its singular straight arm configuration. With the proposed approach, when the manipulator approaches a singular configuration, the system reconfigures itself in a dexterous

configuration using the URV's mobility with a minimal effort. The reconfiguration takes place while the URV provides a stable platform for the manipulator as much as possible; the URV moves along translational directions since it is unavoidable, but the rotational motion is prevented in agreement with the secondary task. Therefore, Eq. (19) provides a means to increase the dexterity of the manipulator by guiding the URV motion in an optimal, coordinated manner.

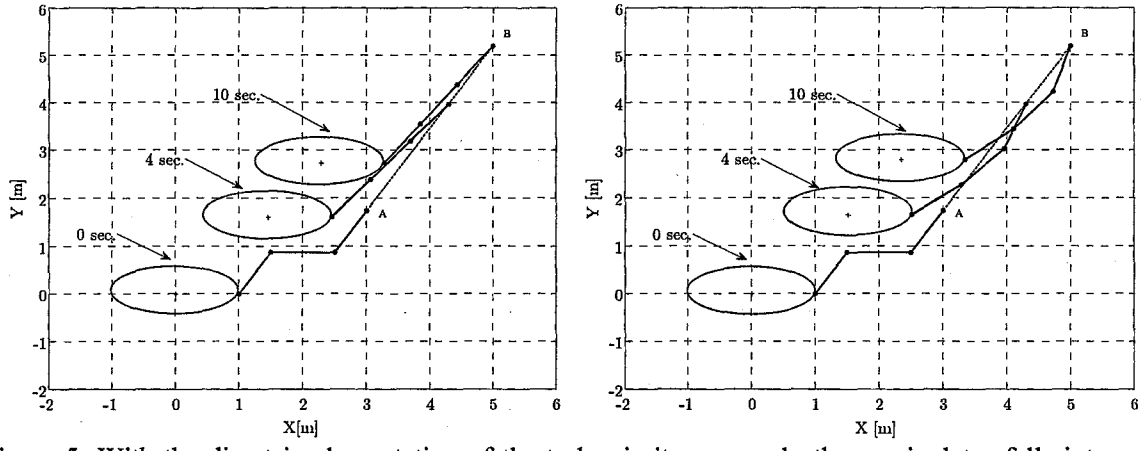


Figure 5: With the direct implementation of the task-priority approach, the manipulator falls into near singular configurations, i.e. a stretched-out configuration (shown on the left). The dexterous task-priority approach prevents the manipulator from falling into near singular configurations (shown on the right) by applying URV motions.

The left figure in Figure 6 reveals that the robust task-priority redundancy resolution, Eq. (16), leads to poor dexterity performance; it falls into near singularity at approximately 5 seconds. The mobility index never reaches the zero value due to the mobility of the URV, but the system dexterity is still very poor. The right figure in Figure 6 is obtained from the implementation of Eq. (19). As the figure reveals, the dexterity of the system is significantly improved. The lower bound $\overline{\rho} = 1.3$ determines the extent to which the system reconfigures itself in an effort to avoid manipulator singularities. This value can be adjusted on-line depending on the specific requirements of a given task.

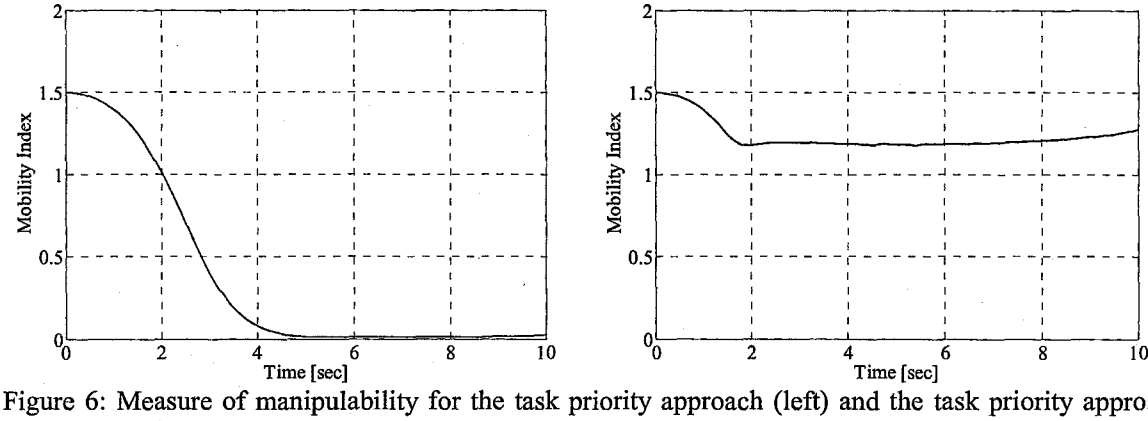


Figure 6: Measure of manipulability for the task priority approach (left) and the task priority approach augmented with manipulability monitoring.

5 DISCUSSION

The success of the model-based controller approach relies heavily upon the accuracy of the dynamic model of the URVM system. Given the nonlinear nature of the underwater environment, having a reliable dynamic model for a URVM system is hard to obtain. By virtue of this, other control strategies such as sliding-mode control, adaptive control, and neural-fuzzy control may be more effective. However, since the main focus of the current work is redundancy resolution, a simple control algorithm is implemented with the assumption of knowing the exact dynamics of the system and having a good response time. The inclusion of the control framework not only provides a complete redundancy resolution scheme, but also demonstrates the capability of the proposed method in generating dynamically realizable joint-space vectors for the URVM system.

The collective Jacobian can be decomposed as:

$$\mathbf{J}_a = \begin{bmatrix} \mathbf{J}_{u,p} & \mathbf{J}_{m,p} \\ \mathbf{J}_{u,s} & \mathbf{J}_{m,s} \end{bmatrix} \tag{24}$$

where $\mathbf{J}_{u,p} \in \mathbb{R}^{m_1 \times u}$ and $\mathbf{J}_{m,p} \in \mathbb{R}^{m_1 \times l}$ are the submatrices containing column vectors defining the contribution to the primary task velocities (end-effector velocities) made by the URV and the manipulator state velocities, respectively. Likewise, $\mathbf{J}_{u,s} \in \mathbb{R}^{m_2 \times u}$ and $\mathbf{J}_{m,s} \in \mathbb{R}^{m_2 \times l}$ are the submatrices containing column vectors defining the contribution to the secondary task velocities made by the URV and the manipulator state velocities with $u+l=n$, respectively. When the manipulator is at a singular configuration, the submatrix $\mathbf{J}_{m,p}$ loses rank, and so does the last l columns of $\mathbf{J}_a$, since $\mathbf{J}_{m,s}$ is a zero matrix, i.e., the manipulator joint velocities do not contribute to the URV velocities. Therefore, manipulator singularities can also be captured by observing the submatrix $\mathbf{J}_{m,p}$ of the primary task Jacobian; this implies that $\mathbf{J}_{m,p}$ can be used in lieu of $\mathbf{J}_a$ in Eq. (17). This choice is computationally more efficient, as it involves smaller size matrix computations. Note that $\mathbf{J}_a$ does not have homogeneous units. Therefore, when the mobility index is normalized according to ρ/σ_1^n, this normalization leads to different units for the index depending on the units associated to σ_1. This problem can be solved by using $\mathbf{J}_{m,p}$ instead of $\mathbf{J}_a$ since $\mathbf{J}_{m,p}$ has homogeneous units. In the current study, this is not necessary since the normalization is not used. As well, Eq. (17) is presented herein since it provides a versatile tool to quantitatively define the degree of conflict between general tasks provided the Jacobians of the defined tasks exhibit full rank. This piece of information has the potential to be used to coordinate different URVM tasks for different URVM applications.

6 CONCLUSION

This work has addressed the problem of kinematic redundancy in URVM systems. It has been shown that the robust task-priority approach is prone to manipulator singularities, when the secondary task is defined as station keeping of the URV. The singular configurations of the manipulator cause the same significant complications for the manipulator controller as land-based non-redundant cases. In this work, a new heuristic method has been proposed to remove these complications. The method provides a means to combine two different redundancy resolution techniques in a novel way. In order to gauge not only the singular configurations but also the need for the URV motion, a mobility index number is proposed that is derived from a modification of Yoshikawa's measure of manipulability. Using the mobility index, the URV motions are applied in an optimal, coordinated manner through a simple shaping function. This mobility index is a versatile tool that can be used to coordinate different URVM tasks. The proposed method removes the complexity associated with existing solutions to the singularity problem of the

URVM system. As well, the method eliminates unnecessary URV motions when the demanded end-effector location is within the reach of the manipulator, and hence provides energy efficient control for the URVM application. The method presented produces kinematically and dynamically realizable URV-manipulator state values that significantly improve the dexterity of the system, thus ensuring dexterous manipulation. This scheme makes the prevailing task-priority approach more suitable, and hence improves the coordinated control of the URVM systems.

ACKNOWLEDGEMENT

The authors wish to thank the Natural Sciences and Engineering Research Council (NSERC) of Canada for providing financial support for this research.

REFERENCES

[1] M. W. Dunnigan, and G. T. Russell, "Evaluation and reduction of the dynamic coupling between a manipulator and an underwater vehicle," *IEEE J. Oceanic Eng.*, vol. 23, no. 3, pp. 260-273, July 1998.

[2] G. Grenon, P. An, S. Smith, and A. Healey, "Enhancement of the inertial navigation system for the Morpheus autonomous underwater vehicles," *IEEE J. Oceanic Eng.*, vol. 26 no. 4, pp. 548-560, Oct. 2001.

[3] J. Kim and W. K. Chung, "Dynamic analysis and two-time scale control for underwater vehicle-manipulator systems," *Proc. 2003 IEEE/RSJ Int. Conf. on Intelligent Robot. and Syst., (IROS 2003)*, vol.1, pp. 577–582, 2003.

[4] S. Soylu, B. J. Buckham and R. P. Podhorodeski, "Using articulated-body algorithm within sliding mode control to compensate dynamic coupling in underwater manipulator systems," *CSME Trans.* vol. 29, no. 4, pp. 629-643, 2005.

[5] K. Kawaguchi, H. Momma, and R. Iwase, "VENUS PROJECT-submarine cable recovery system," *1998 Int. Symp. on Underwater Technology (Cat. No.98EX101)*, pp. 448-452, 1998.

[6] J. Delaney, P. Beauchamp, M. McNutt, C. Barnes, A. Chave, and J. Madden, "Project NEPTUNE: an interactive, regional cabled ocean observatory in the Northeast Pacific," *Oceans Conf. Record (IEEE)*, vol. 2, pp. 1038-1042, 2003.

[7] G. Antonelli, and S. Chiaverini, "Fuzzy redundancy resolution and motion coordination for underwater vehicle-manipulator systems, "*IEEE T. Fuzzy Syst.*, vol. 11, no. 1, pp. 109-120, 2003.

[8] G. Antonelli and S. Chiaverini, "Task priority redundancy resolution for underwater-manipulator systems," *1998 IEEE Int. Conf. Robotic. Autom.*, pp. 768-733, May 1998.

[9] K. R. Goheen and R. Jeffreys, "Multivariable self-tuning autopilots for autonomous and remotely operated vehicle, " *IEEE J. Oceanic Eng.*, vol. 15, no. 3, pp. 144-151, July 1990.

[10] S. Chiaverini, "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators, "*IEEE T. Robotic. Autom.*, vol. 13, pp. 398-410, June 1997.

[11] A. A. Maciejewski, and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *Int. J. Robot. Res.*, vol. 4, no.3, pp. 109-117, 1985.

[12] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task-priority based redundancy control of robot manipulators," *Int. J. Robot. Res.*, vol. 6, no. 2, pp. 3-15, 1987.

[13] N. Sarkar and T. K. Podder, "Motion coordination of underwater vehicle-manipulator systems subject to drag optimization," *Proc. 1999 IEEE Int. Conf. on Robotic. Autom.*, pp. 387-392, 1999.

[14] C. H. dos Santos, R. Guenther, D. Martins, and E. R. De Pieri, "Virtual kinematic chains to solve the underwater vehicle-manipulator systems redundancy," *J. of the Brazilian Society of Mechanical Sciences and Eng.*, vol. 28, no. 3, pp. 354-361, 2006.

[15] C. H. dos Santos, G. Bittencourt, and R. Guenther, "Motion coordination for underwater vehicle-manipulator systems using a fuzzy hybrid strategy," *Int. Conf. on Intell. Robot. and Syst.*, Beijing, China, pp. 3018-3023, Oct. 2006.
[16] D. S. Watkins, *Fundamentals of Matrix Computations*, 2nd Edition, John Wiley&Sons, 2002.
[17] A. Liegeois, "Automatic supervisory control of the configuration and behaviour of multibody mechanisms," *IEEE T. Syst. Man. Cyb.*, vol. SMC-7, pp. 868-871, 1977.
[18] Y. Nakamura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," *Trans. ASME J. Dynam. Syst., Meas., Control*, vol. 108, pp. 163-171, 1986.
[19] B. Siciliano and J. J. E. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," *Proc. Int. Conf. Adv. Robotic.*, Pisa, Italy, pp. 1211-1216, June 1991.
[20] H. Seraji and R. Colbaugh, "Singularity-robustness and task-prioritization in configuration control of redundant robots," *29th IEEE Conf. on Decision and Control*, pp. 3089-3095, 1990.
[21] T. Yoshikawa, "Manipulability of robotic mechanisms," *Int. J. Robot. Res*, vol. 4, no.2, pp.3-9, 1985.
[22] J. Park, "*Analysis and control of kinematically redundant manipulators: An approach based on kinematically decoupled joint space decomposition,*" PhD thesis, Pohang University of Science and Technology (POSTECH), 1999.
[23] L. Li, W. A. Gruver, Q. Zhang, and Z. Yang, "Kinematic control of redundant robots and the motion optimizability measure," *IEEE T. Syst. Man. Cyb.*, vol. 31, no. 1, pp. 155-160, 2001.
[24] G. Marani, J. Kim, J. Yuh, and W. K. Chung, "Algorithmic singularities avoidance in task-priority based controller for redundant manipulators," *2003 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems,* Las Vegas, pp.3570-3574, Oct. 2003.
[25] T. Yoshikawa, *Foundations of robotics: analysis and control*, MIT Press, 1990.
[26] B. Nemec, and L. Zlajpah, "Implementation of force control on redundant robots," *1998 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 1314–1319, 1998.
[27] J. J. Craig, *Introduction to Robotics*, 2nd Edition. Addison-Wesley, 1989.
[28] D. Steinke, "*Design and Simulation of a Kalman filter for ROV Navigation,*" Master's Thesis, University of Victoria, 2006.