

# Stereographic projection for industrial manipulator tasks: Theory and experiments

Magnus Bjerkgeng, Kristin Y. Pettersen and Erik Kyrkjebø

**Abstract**—In this paper, we present a solution and experimental results for real-time control of manipulation tasks considering the alignment of a robots end effector relative to some reference. The developed controller is applicable to industrial manipulators and is based on the pseudoinverse redundancy resolution method. The application considered is the employment of two industrial robots in an offshore remote inspection and maintenance system. A leader robot is controlled freely from onshore, and a follower robot uses an attached camera to provide the onshore operator with live video feed of the ongoing operation. Robot manipulators constitute flexible camera platforms, compared to e.g. simple pan/tilt units, for monitoring offshore operations. We develop a controller for the follower robot such that automatic camera tracking is achieved using pseudoinverse redundancy resolution control. A minimal task space parametrization relying on stereographic projection is constructed which achieves relative end effector alignment tracking without introducing representational or algorithmic singularities. It is shown that singular configurations will only in special cases affect the closed loop behavior. The controller is applicable to tasks such as spray painting or polishing on curved surfaces. The control approach is experimentally validated on two Kuka KR-16 industrial robot manipulators.

## I. INTRODUCTION

Offshore oil and gas platforms constitute harsh working conditions and are extremely costly to build, maintain and operate. Environment, Health and Safety (EHS) issues can be improved and costs can be significantly reduced by remote control of offshore operations from onshore. One of the key issues to performing such operations is to provide onshore operators with a sufficient overview of the processes they will control offshore. To this end, robot manipulators can be used for automatic live-video monitoring of operations performed by onshore operators. The control of this task, and the development and analysis of a task description fitting this application is the topic of this paper.

The view direction of a camera is controlled using two degrees of freedom (DOF) where pan/tilt is a common choice of variables. A standard 6 DOF industrial manipulator is employed in this paper and is thus redundant with respect to the camera task. Hence, the robot is considered to be “task redundant” with respect to camera control using the definition given in [1]. The control of redundant robots has been studied thoroughly in the literature. Redundancy

M. Bjerkgeng and K.Y. Pettersen are with Department of Engineering Cybernetics, Norwegian University of Science and Technology, NO-7491 Trondheim, Norway. {Magnus.Bjerkgeng, Kristin.Y.Pettersen}@itk.ntnu.no

E. Kyrkjebø is with SINTEF ICT Applied Cybernetics PO Box 4760 Sluppen, NO-7465 Trondheim, Norway. {Erik.Kyrkjebø}@sintef.no

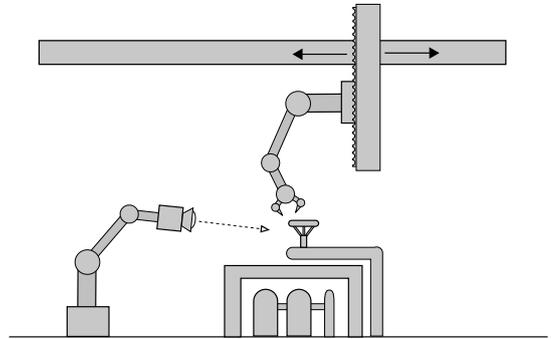


Fig. 1: An illustration of the robot monitoring system. The leader robot is performing a task while the follower robot is monitoring the operation.

is attractive since the extra degrees of freedom, which do not affect the task performance, may be efficiently used to increase performance with respect to additional criteria such as joint range limits [2], singularity avoidance [3], obstacle avoidance [4] and compliance [5].

The most common strategy to resolve redundancy in real-time applications includes the use of pseudoinverse control and was introduced in [6]. The pseudoinverse is used to resolve redundancy through local optimization of some objective function either at the acceleration level [7] or at the velocity level [8]. A review of velocity level redundancy resolution can be found in [9].

Most previous works consider 3 DOF planar robots or 7 DOF robots as examples of redundant control problems. It is then natural to assume that the reference is given as a full or partial end-effector position and/or orientation trajectory. The controllers are thus developed under the assumption that the control objective is given explicitly as a linear relationship in the task space. For a general output feedback task however, no such reference trajectory is available. Care has to be taken in constructing the task parametrization for tasks which are nontrivial functions of the end effector coordinates as this choice directly impacts the closed loop dynamics.

The camera tracking problem may be solved using *visual servoing* techniques, [10], where image information is used directly in the feedback loop. Joint velocity control is achieved using a camera dependent image to joint velocity mapping. We stress that our approach, while being similar to, does not use visual servoing. The video feed is presented to the operator, and is not used explicitly in the feedback loop. Our approach is feasible given that a robot is under

surveillance, and it allows for greater design freedom since the task velocity mapping is a choice rather than being camera dependent.

The main contributions of this paper is the introduction of a task parametrization based on stereographic projection, which allows for task tracking using pseudoinverse redundancy resolution. We will in this paper refer to this task as the *camera task* given the intuition of the application considered. This representation has clear advantages over previously proposed methods such as pan-tilt [11], isometric projection [12] or the visual servoing technique [10] especially with respect global stability and singularities. A thorough investigation of the singularities and the kinematics of the stereographic projection is carried out.

The differences between a task redundant and a fully redundant control problem is highlighted, especially considering the time dependency of the task representation. The proposed control law is applicable to real-time control of redundant or non-redundant industrial robot manipulators, either using direct torque control, or through direct joint velocity/increment input. Furthermore, experiments which validate the proposed control system, are carried out using two KUKA-KR16 industrial robots.

The paper is organized as follows. Section II presents the kinematics of an industrial manipulator and an overview of the pseudoinverse redundancy resolution scheme. Section III contains a review of different task parameterizations previously proposed as well as some qualitative analysis on the closed loop behaviour. The possibility of singularities is investigated in Section VI. Experimental results are provided in Section VIII, and conclusions and suggestions to further work are given in Section IX.

## II. BACKGROUND

In this section we review the kinematics of a robot manipulator and present a short overview of the pseudoinverse control strategy for redundancy resolution.

### A. Kinematics of a serial manipulator

In this section we present the kinematics of a robot manipulator. Consider an  $n$ -link serial robot manipulator with joint angles  $\mathbf{q} \in \mathbb{R}^n$ . The forward kinematics is the mapping from the joint angles to the position and orientation of the end effector [13]

$$\mathbf{T}(\mathbf{q}) = \begin{bmatrix} \mathbf{R}(\mathbf{q}) & \mathbf{x}(\mathbf{q}) \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}, \quad (1)$$

where  $\mathbf{R}(\mathbf{q}) = [e_1, e_2, e_3] \in SO(3)$  is a rotation matrix composed of unit vectors  $e_i$  describing the orientation of the end-effector, and  $\mathbf{x} \in \mathbb{R}^3$  denotes its position. We will without loss of generality assume that the forward kinematics is constructed such that  $\mathbf{x}$  is the camera lens position and  $e_3$  points in the view direction of the camera. The manipulator Jacobian  $\mathbf{J} \in \mathbb{R}^{6 \times n}$  maps the joint velocities  $\dot{\mathbf{q}}$  to the linear velocity  $\dot{\mathbf{x}}$ , and angular velocity  $\boldsymbol{\omega}$ , of the end-effector:

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \boldsymbol{\omega} \end{bmatrix} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \quad (2)$$

### B. The pseudoinverse redundancy resolution method

We present in this section a short overview of the pseudoinverse redundancy resolution method. A more thorough overview may be found in [9]. Consider a task implicitly represented by the function

$$\Psi(\mathbf{q}, t) \in \mathbb{R}^m \quad (3)$$

with  $m < n$ . The objective is to stabilize the zero set of  $\Psi$ . If it is possible to derive a differential relationship

$$\dot{\Psi} = \mathbf{J}_t(\mathbf{q}, t)\dot{\mathbf{q}} + \mathbf{P}(\mathbf{q}, t) \quad (4)$$

then the zero set of  $\Psi$  may be stabilized using either velocity level redundancy resolution control or at the acceleration level. The time dependent term  $\mathbf{P}$  is often omitted either since it is unknown or assumed small. We will only consider the velocity level control problem since we are dealing with industrial manipulators.

### C. Velocity level redundancy resolution

Velocity level control is the most common strategy in the control of industrial manipulators since direct torque input is typically unavailable. An interface where angle increments or joint velocity commands is passed to an opaque low level servo controller is however available. The control task is then simplified to generating desired joint position increments and/or velocities. The velocity level redundancy resolution scheme is derived by considering the least squares solution to (4) for  $\dot{\mathbf{q}}$  which is given by

$$\dot{\mathbf{q}} = \mathbf{J}^+(\dot{\Psi} - \mathbf{P}(\mathbf{q}, t)), \quad (5)$$

which minimizes  $\dot{\mathbf{q}}^T \dot{\mathbf{q}}$ . The  $n \times n$  identity matrix is denoted  $\mathbf{I}_{n \times n}$  and  $\mathbf{J}^+ \in \mathbb{R}^{n \times m}$  is the Moore-Penrose pseudoinverse of  $\mathbf{J}_t$  given by

$$\mathbf{J}^+ = \mathbf{J}_t^T (\mathbf{J}_t \mathbf{J}_t^T)^{-1}, \quad (6)$$

which satisfies  $\mathbf{J}_t \mathbf{J}^+ = \mathbf{I}_{m \times m}$  provided that  $\mathbf{J}_t$  has full rank. A more general solution to (4) is to given by the addition of an arbitrary vector  $\mathbf{F} \in \mathbb{R}^n$  onto the nullspace of the task Jacobian

$$\dot{\mathbf{q}} = \mathbf{J}^+(\dot{\Psi} - \mathbf{P}(\mathbf{q}, t)) + (\mathbf{I}_{n \times n} - \mathbf{J}^+ \mathbf{J}_t^T) \mathbf{F}. \quad (7)$$

Closing the loop by setting  $\dot{\Psi} = -\mathbf{K}_p \Psi$  for some  $\mathbf{K}_p > 0$  in (7) results in the joint velocity control law

$$\dot{\mathbf{q}} = \mathbf{J}^+(-\mathbf{K}_p \Psi - \mathbf{P}(\mathbf{q}, t)) + (\mathbf{I}_{n \times n} - \mathbf{J}^+ \mathbf{J}_t^T) \mathbf{F}. \quad (8)$$

This is called the pseudoinverse redundancy resolution method since  $\Psi = \mathbf{0}$  is stabilized using the joint velocity (8). We observe this either by premultiplying (8) by  $\mathbf{J}_t$  or via the Lyapunov method using  $V = \frac{1}{2} \Psi^T \Psi$ . The system (8) has  $\Psi = \mathbf{0}$  as a globally exponentially stable equilibrium point for all positive definite  $\mathbf{K}_p$  assuming that  $\text{rank}\{\mathbf{J}_t\} = m$ .

The vector field  $\mathbf{F}$  may be used to optimize solutions of (8) with respect to sub-tasks without affecting the stability of  $\Psi = \mathbf{0}$ . Let a scalar function  $U(t, \mathbf{q})$  denote how "good", e.g. with respect to the sub-task, a solution to (8) is, and denote  $\mathbf{F} \in \mathbb{R}^n$  as the steepest descent direction  $\mathbf{F}(\mathbf{q}) = -(\nabla U(t, \mathbf{q}))^T$ , then solutions to (8) will minimize

$U$  in the task nullspace. This technique is called the *Gradient projection method* [9]. Motions induced by  $F$  are called *self motion* or *nullspace motion*, as they span the nullspace of  $\mathbf{J}_t$ .

### III. THE CAMERA TASK PARAMETRIZATION

We will in this section consider different ways to represent the camera tracking task, and provide a comparison. Some subtle differences between constructing the camera tracking task in different coordinate systems are discussed in section V.

The minimal requirement we have to impose for camera tracking is that the robot points its camera at a point described by the trajectory  $\mathbf{p}(t) \in \mathbb{R}^3$ . We consider additional requirements such as a desired viewing angle or viewing distance as less critical, and will not include these directly in the task space. A sufficient and necessary condition for assigning the correct view direction is that the following vector function is zero

$$\Psi_0(\mathbf{q}) = \mathbf{e}_3 - \frac{\mathbf{p} - \mathbf{x}}{\|\mathbf{p} - \mathbf{x}\|}. \quad (9)$$

Writing (9) in the end effector coordinates gives

$$\Psi_0(\mathbf{q}) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} - \mathbf{R}^T \frac{\mathbf{p} - \mathbf{x}}{\|\mathbf{p} - \mathbf{x}\|} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} - \mathbf{y}. \quad (10)$$

However, directing a camera towards a point requires only two degrees of freedom, pan-tilt being a possible parametrization. The task space parametrization (9) is hence over-parametrized. One can check that the Jacobian of (9) loses rank when  $\Psi_0 = 0$ , resulting in a degenerate task space when the camera is close to alignment. We need to express our task space by a minimal set of equations in order to overcome this problem. In the following sections some common strategies for constructing this minimal representation are considered.

#### A. Perspective projection

The perspective projection is often used in visual servoing since it maps a pinhole camera view onto a plane which represents an image on a screen, [10]. If we express the point  $\mathbf{p}$  in the end effector coordinates, then the perspective transformation which maps the point  $\mathbf{p}$  to a screen is given by

$$\Psi_v = \begin{bmatrix} \frac{y_1}{y_3} & \frac{y_2}{y_3} \end{bmatrix}^T. \quad (11)$$

This is not a suitable choice for global camera control, however, as points lying in the camera plane are mapped to infinity. This means that if the point  $\mathbf{p}$  at any time is behind the camera, then we cannot continuously control the camera to point at  $\mathbf{p}$  without passing through an undefined point. This represents the inability of controlling the velocity of points which are not on the screen using visual servoing.

#### B. Euler angles

Euler angles or spherical coordinates may be used as an explicit pan/tilt parametrization. Expressing  $\Psi$  in the inertial coordinate system results in a parametrization in the form

$$\Psi_{\text{Euler}} = \begin{bmatrix} \text{atan}_2(e_{3x}, e_{3y}) - \text{atan}_2(p_1 - x_1, p_2 - x_2) \\ e_{3z} - (p_3 - x_3) \end{bmatrix}. \quad (12)$$

It can be shown that  $\Psi_{\text{Euler}}$  is zero if and only if the camera is pointing at  $\mathbf{p}$ . However, the Jacobian of (12) is not defined when  $e = [0, 0, 1]^T$ , so the camera cannot look straight up. This can be relieved by expressing the Euler angles in the end effector coordinate system, as is done in [14]. We cannot fully rid us of this representation singularity however. Another issue with the Euler angles is their non-uniqueness, as one vector may be rotated to another it two ways. Therefore care must be taken in implementation to pick the *shortest* rotation, to avoid unnecessarily large transients.

#### C. Orthographic projection

The Orthographic projection is used in [12] as a task/constraint parametrization in a polishing ball hybrid force motion problem. Our camera tracking problem is analogous to the problem of polishing a moving ball with arbitrary radius. The projection is given in end effector coordinates by

$$\Psi_e = \begin{bmatrix} \mathbf{e}_1^T(\mathbf{x} - \mathbf{p}) \\ \mathbf{e}_2^T(\mathbf{x} - \mathbf{p}) \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}. \quad (13)$$

We observe that when the camera is pointed in the correct direction, then  $(\mathbf{x} - \mathbf{p})$  is parallel to  $\mathbf{e}_3$ , and perpendicular to  $\mathbf{e}_1$  and  $\mathbf{e}_2$ . The Jacobian of  $\Psi_e$  is everywhere rank 2. The null set of  $\Psi_e$  includes the point directly behind the camera, and this will appear as a locally stable equilibrium in (8). A polishing robot using a controller based on (13) will not know whether it is polishing the inside of a sphere or the outside. Therefore we can, using the orthographic projection, have cases where the camera settles  $180^\circ$  in the wrong direction.

#### D. Stereographic projection

The last camera task parametrization which we will consider, is the stereographic projection which is a general version of the perspective projection. There is a choice of defining the projection point and the projection plane when the projection is constructed. This allows for some design freedom. The stereographic projection has also been considered for motion planing in [15]. We may construct the projection in the end effector coordinate system by identifying a point on the sphere  $\|\mathbf{y}\| = 1$  with a point on the  $y_1y_2$ -plane tangential to the sphere at  $\mathbf{y} = [0, 0, 1]^T$  by a line through  $\mathbf{y} = [0, 0, -1]$ , see Fig 2. The projection is with this choice given by

$$\Psi = \begin{bmatrix} \frac{2y_1}{y_3+1} & \frac{2y_2}{y_3+1} \end{bmatrix}^T \quad (14)$$

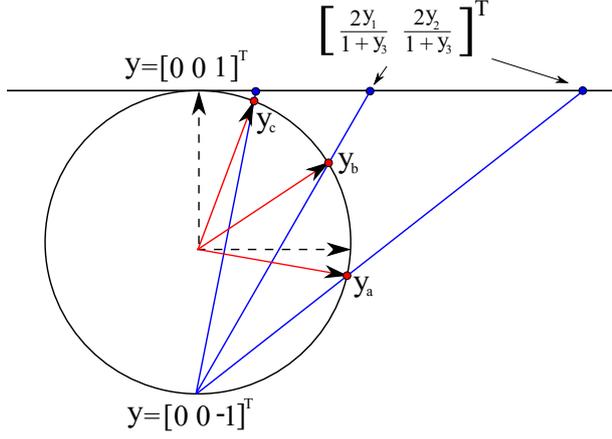


Fig. 2: The Stereographic projection.

The desired camera orientation  $\mathbf{y} = [0, 0, 1]^T$  is mapped to the projected point  $\Psi = [0, 0]^T$  and the antipodal point  $\mathbf{y} = [0, 0, -1]$  is mapped to infinity. The zero set of  $\Psi$  is with this projection point the same as (9) such that global camera alignment is assured using the system (8). The stereographic projection is a diffeomorphism excluding the projection point  $\mathbf{y} = [0, 0, -1]$  where it is not defined, and hence the Jacobian  $\frac{\partial \Psi}{\partial \mathbf{y}}$  has rank 2 for all other points. The Jacobian of the projection is derived in Appendix I, and is given by

$$\mathbf{J}_t = \frac{\partial \Psi}{\partial \mathbf{q}} = \frac{\partial \Psi}{\partial \mathbf{y}} \frac{\partial \mathbf{y}}{\partial \mathbf{q}}, \quad (15)$$

where

$$\frac{\partial \Psi}{\partial \mathbf{y}} = 2 \begin{bmatrix} 1/(y_3 + 1) & 0 & -y_1/(y_3 + 1)^2 \\ 0 & 1/(y_3 + 1) & -y_2/(y_3 + 1)^2 \end{bmatrix} \quad (16)$$

$$\mathbf{J}_t = \frac{\partial \Psi}{\partial \mathbf{y}} \left[ \frac{1}{\|\mathbf{p} - \mathbf{x}\|} \mathbf{S}(\mathbf{y})^2 \mathbf{R}^T \quad \mathbf{S}(\mathbf{y}) \mathbf{R}^T \right] \mathbf{J} \quad (17)$$

where  $\mathbf{S} : \mathbb{R}^3 \mapsto \mathbb{R}^{3 \times 3}$  is the skew-symmetric operator given by

$$\mathbf{S}(\mathbf{y}) = \begin{bmatrix} 0 & -y_3 & y_2 \\ y_3 & 0 & -y_1 \\ -y_2 & y_1 & 0 \end{bmatrix}, \quad (18)$$

and the time dependent term  $\mathbf{P}$  is given by

$$\mathbf{P}(\mathbf{q}, \mathbf{p}) = -\frac{\partial \Psi}{\partial \mathbf{y}} \mathbf{S}(\mathbf{y})^2 \mathbf{R}^T \frac{\dot{\mathbf{p}}}{\|\mathbf{p} - \mathbf{x}\|}. \quad (19)$$

These matrices together with  $\Psi$  is what we need in order to construct the pseudoinverse redundancy resolution control scheme. Some intuition may be gained by noting that the linear velocity part of the manipulator Jacobian is scaled by the inverse of the view distance  $\|\mathbf{p} - \mathbf{x}\|$ . This reflects the fact that if a point is far away, then translating the camera will result in a small view angle error reduction. This is a familiar fact to anyone who has looked at a distant point while moving. If  $\mathbf{p}$  is infinitely far away, then no finite translation will change the view direction error.

#### IV. PROPERTIES OF THE TASK JACOBIAN

In this section we will shortly state a useful property of the task Jacobian  $\mathbf{J}_t$ . If we define the matrix  $\mathbf{E}$  as

$$\mathbf{E} = \frac{\partial \Psi}{\partial \mathbf{y}} \left[ \mathbf{S}(\mathbf{y})^2 \frac{1}{\|\mathbf{p} - \mathbf{x}\|} \quad \mathbf{S}(\mathbf{y}) \right], \quad (20)$$

then  $\mathbf{E}$  has the form

$$\mathbf{E} = \frac{2}{y_3 + 1} \begin{bmatrix} \mathbf{v}_1^T \frac{1}{\|\mathbf{p} - \mathbf{x}\|} & \mathbf{v}_2^T \\ \mathbf{v}_2^T \frac{1}{\|\mathbf{p} - \mathbf{x}\|} & -\mathbf{v}_1^T \end{bmatrix} \quad (21)$$

where  $\mathbf{v}_i \in \mathbb{R}^3$  and  $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{y}\}$  form an orthonormal basis of  $\mathbb{R}^3$  for all points except  $y_3 = -1$ . It is apparent from (21) that  $\mathbf{E}$  has rank 2 for all points except for the projection point. This result is easily verifiable by substituting (41) into (21).

#### V. THE CHOICE OF COORDINATES

In this section we consider the subtleties which arise when the stereographic projection is constructed in different coordinate systems. The two coordinate systems which are natural to consider are the inertial coordinate system with the standard orthonormal basis, and the end effector coordinate system with the basis  $[e_1, e_2, e_3] = \mathbf{R}$  relative to the inertial system. The view direction  $\mathbf{y}$  may be expressed in the inertial coordinate system using the rotation  $\mathbf{y}_n = \mathbf{R}\mathbf{y}$ . The stereographic projection of the view direction error may be constructed in the inertial coordinate system by projecting the two vectors of (9) individually and taking their difference

$$\Psi_n = 2 \left[ \frac{e_{31}}{e_{33} + 1} \quad \frac{e_{32}}{e_{33} + 1} \right]^T - 2 \left[ \frac{y_1^n}{y_3^n + \|\mathbf{y}^n\|} \quad \frac{y_2^n}{y_3^n + \|\mathbf{y}^n\|} \right]^T. \quad (22)$$

The function  $\Psi_n$  is not defined when the end effector is pointing straight down, and when the view point  $\mathbf{p}$  is directly below the end effector. Expressing the projection in the end effector coordinates, (14), leads to only one prohibited point located  $180^\circ$  away from the reference. Using (14) it is hence possible to command any view direction. The projection  $\Psi_n$  is hence inferior to (14) with respect to singular points.

A case may be made for the choice of  $\Psi_n$ , however, as it possesses a useful property not shared by (14). Consider an n-link robot where the last joint revolves the end effector around its link axis, e.g. a spherical wrist configuration. The controllers (8) based on these two representations exhibit different behaviors apart from the undefined points. The robot with a controller based on (22) behaves as expected, assuming that no singularities occur. The robot directs its camera towards the desired point and follows it. The camera of the robot using (14) will also direct its camera towards the desired point, but will in addition roll the camera around its axis, i.e.  $q_n(t) \neq 0$ . The question is why this controller generates motions which do not affect the stability of the view direction error? We can verify that the norm of the error does not decrease by changing  $q_n$ ,

$$\frac{\partial \|\Psi_n\|^2}{\partial q_n} = 0, \quad \frac{\partial \|\Psi\|^2}{\partial q_n} = 0. \quad (23)$$

This is expected as rotating a camera around its view axis does not change the view axis, and hence not the view error. This does not imply, however, that the error does not depend upon  $q_n$ , and we can check that

$$\frac{\partial \Psi_n}{\partial q_n} = 0, \quad \frac{\partial \Psi}{\partial q_n} \neq 0, \quad (24)$$

since  $\mathbf{y}$  depends on  $q_n$  through  $\mathbf{R}(q)$  while  $\mathbf{y}_n$  does not. This is a problem since we would like to avoid singularities and keep the camera image steady. There are ways to fix this problem. The simplest may be to remove  $q_n$  from our kinematics entirely, such that no control inputs for  $q_n$  are generated. Other solutions are to use the weighted pseudoinverse, [16], to force  $\dot{q}_n$  to zero, or to disregard nonzero  $q_n$ . It may be however, considering this phenomenon, that other unnecessary motions  $\mathbf{q}(t) = \mathbf{q}_a(t) + \mathbf{q}_b(t)$  are generated using (8) such that  $\frac{d}{dt} \|\Psi\|(\mathbf{q}_b) = 0$  with  $\mathbf{q}_b \neq \mathbf{0}$ .

#### A. Stereographic projection for other tasks

We will shortly present some applications for which the stereographic projection is useful. The closed loop system (8) stabilizes a time and configuration dependent direction for the end effector, which is useful for several application. For a spray painting or polishing operation on a curved surface, one may construct the task parametrization as

$$\Psi_{\text{spray}}(\mathbf{q}) = \Psi(\mathbf{y}_{\text{spray}}(\mathbf{q})) \quad (25)$$

where  $\mathbf{y}_{\text{spray}}(\mathbf{q})$  is the negative normal vector to the shape in question. The robot will under (8) align the end effector normal to this surface, and nullspace control may be used to control the distance from the object and motions parallel to the surface.

### VI. SINGULARITIES OF THE TASK JACOBIAN

In this section we will study the nature of the singularities of the task Jacobian. Insight into the character of these singular points is valuable as they represent a breakdown of the closed loop dynamics. We will prove that there may exist a singular configuration and a unique view point  $\mathbf{p}$  given a singular configuration such that the task Jacobian is rank deficient.

#### A. Existence and uniqueness of singularities

We will in this section study the conditions under which the task Jacobian is rank deficient. If  $\mathbf{J}$  has full rank, then  $\mathbf{J}_t$  has full rank since  $\mathbf{J}_t = \mathbf{E}\mathbf{N}\mathbf{J}$  and  $\mathbf{E}\mathbf{N}$  has rank 2. No representational singularities are hence introduced. However if  $\mathbf{J}$  does not have full rank, there may exist a case where  $\mathbf{J}_t$  has rank 1. A specific combination of a view point  $\mathbf{p}$  and a robot configuration  $\mathbf{q}$  is needed in order for the pseudoinverse to be singular as the task Jacobian  $\mathbf{J}_t$  depends upon both  $\mathbf{q}$  and  $\mathbf{p}$ . The result in this section is summarized in the following.

##### Proposition 1:

Assume that  $n \geq 5$ . If  $\text{rank}\{\mathbf{J}(\mathbf{q}_s)\} = 5$  and

$\text{rank}\{\mathbf{J}_t(\mathbf{q}_s, \mathbf{p}_s)\} = 1$ , then the view point  $\mathbf{p}_s$  is uniquely given by  $\mathbf{q}_s$  as

$$\mathbf{p}_s = \mathbf{x}(\mathbf{q}_s) + \frac{\boldsymbol{\nu}_1(\mathbf{q}_s) \times \boldsymbol{\nu}_2(\mathbf{q}_s)}{\|\boldsymbol{\nu}_1(\mathbf{q}_s)\|^2} \quad \text{and} \quad \boldsymbol{\nu}_1 \perp \boldsymbol{\nu}_2 \quad (26)$$

where  $\boldsymbol{\nu}^T = [\boldsymbol{\nu}_1^T, \boldsymbol{\nu}_2^T]$  is the basis vector of  $\text{Null}\{\mathbf{J}^T\}$  with  $\boldsymbol{\nu}_1, \boldsymbol{\nu}_2 \neq \mathbf{0}$ .

The proof of proposition 1 is found in Appendix II.

The result on uniqueness of singularities is a attractive in that singular configurations are *rare* in the task space. The implication in proposition 1 does not go both ways, i.e. if  $\mathbf{J}$  is singular with its left nullspace on the required form, and  $\mathbf{p}$  is given by (26), then it is not given that  $\mathbf{J}_t$  is singular. It follows trivially from proposition 1 that singularities are isolated with respect to  $\mathbf{p}$  if  $\mathbf{J}_t$  is rank deficient and  $\text{rank}\{\mathbf{J}\} = 5$ . That is, there exists no  $\mathbf{p} + d\mathbf{r}$  in a neighborhood of  $\mathbf{p}$  such that  $\mathbf{J}_t(\mathbf{p} + d\mathbf{r})$  is rank deficient. It is also apparent by considering the contrapositive of Proposition 1 that if the left nullspace of  $\mathbf{J}$  does not have the required form, or if the view point is not equal to (26), then  $\mathbf{J}_t$  cannot be singular.

### VII. DYNAMICS NEAR THE PROJECTION POINT

We will in this section analyze the joint angle dynamics give by (8) when the view error is close to  $180^\circ$ . Consider the joint velocity controller with  $\mathbf{F} = \mathbf{0}$  given by

$$\dot{\mathbf{q}} = \mathbf{J}_t^+(-\mathbf{K}_p \Psi) = \mathbf{J}_R^T \mathbf{E}_k^T (\mathbf{E}_k \mathbf{J}_R \mathbf{J}_R^T \mathbf{E}_k^T)^{-1} (-\mathbf{K}_p \Psi). \quad (27)$$

Our objective is to determine how  $\dot{\mathbf{q}}$  behaves when  $y_3$  tends to  $-1$ . We know from (17) that  $\mathbf{E}_k$  is not defined at  $y_3 = -1$ , and we also note that the limit of  $\Psi$  when  $y_3 \rightarrow -1$  is not defined. One could in light of this guess that the dynamics close to this point is badly behaved. This is, however, not the case, as we shall see by some algebraic manipulation. Consider the following change of variables

$$\mathbf{E}_k = \frac{1}{y_3 + 1} \tilde{\mathbf{E}}_k, \quad \Psi = \frac{1}{y_3 + 1} \tilde{\Psi}, \quad (28)$$

which factors out  $(y_3 + 1)^{-1}$ . The system (27) with these matrices is given by

$$\dot{\mathbf{q}} = \frac{1}{(y_3 + 1)^2} \mathbf{J}_R^T \tilde{\mathbf{E}}_k^T \left[ \frac{1}{(y_3 + 1)^2} \tilde{\mathbf{E}}_k \mathbf{J}_R \mathbf{J}_R^T \tilde{\mathbf{E}}_k^T \right]^{-1} (-\mathbf{K}_p \tilde{\Psi}). \quad (29)$$

Observe that the terms  $(y_3 + 1)^{-2}$  cancel out such that (27) has the identical form in these new variables

$$\dot{\mathbf{q}} = \mathbf{J}_R^T \tilde{\mathbf{E}}_k^T (\tilde{\mathbf{E}}_k \mathbf{J}_R \mathbf{J}_R^T \tilde{\mathbf{E}}_k^T)^{-1} (-\mathbf{K}_p \tilde{\Psi}). \quad (30)$$

Using  $\tilde{\Psi}$  and  $\tilde{\mathbf{E}}_k$  in place of  $\Psi$  and  $\mathbf{E}_k$  is appealing since we gain calculation speed and numerical stability, especially close to the projection point. The vector  $\tilde{\Psi} = [y_1, y_2]^T$  tends to zero as  $y_3 \rightarrow -1$  since  $\|y\| = 1$ . The matrix  $\tilde{\mathbf{E}}_k$ , however, does not have a limit in the classical sense as may be seen by taking the limit of  $\tilde{\mathbf{E}}_k$  with  $\mathbf{y}$  expressed in spherical coordinates  $(\phi, \theta)$ . The vector  $\tilde{\mathbf{E}}_{k1}$  is given by

$$\tilde{\mathbf{E}}_{k1}^T = \begin{bmatrix} \frac{y_1^2}{y_3+1} - 1 & \frac{y_1 y_2}{y_3+1} & y_1 & \frac{y_1 y_2}{y_3+1} & \frac{y_2^2}{y_3+1} - 1 & y_2 \end{bmatrix}. \quad (31)$$

And its limiting value  $\lim_{y_3 \rightarrow -1} \tilde{\mathbf{E}}_k = \tilde{\mathbf{E}}_{\text{lim}}$  is shown to be dependent upon  $\theta$

$$\tilde{\mathbf{E}}_{\text{lim}} = \begin{bmatrix} \cos(2\theta) & \sin(2\theta) & 0 & \sin(2\theta) & -\cos(2\theta) & 0 \\ \sin(2\theta) & -\cos(2\theta) & 0 & -\cos(2\theta) & -\sin(2\theta) & 0 \end{bmatrix}. \quad (32)$$

The velocity  $\dot{\mathbf{q}}$  will, however, tend to zero as  $\tilde{\Psi}$  tends to zero even though  $\tilde{\mathbf{E}}_k$  depends upon our angle of approach towards or away from the projection point. It is apparent that the joint velocities close to the point  $y_3 = -1$  are not degenerate, and joint velocities will not become unbounded assuming that  $\mathbf{J}_t \mathbf{J}_t^T$  is nonsingular. The situation is rather the opposite as joint velocities tend to zero close to the projection point.

## VIII. EXPERIMENTAL RESULTS



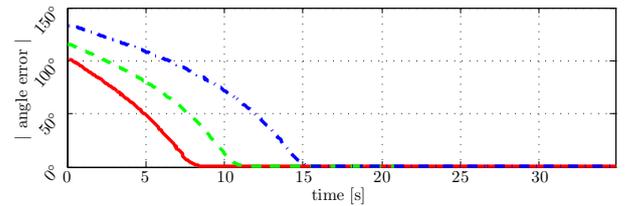
Fig. 3: The lab setup where the floor mounter robot is equipped with a stereoscopic camera.

In this section the experimental setup is described and data from the experiments carried out is presented.

### A. Experimental setup

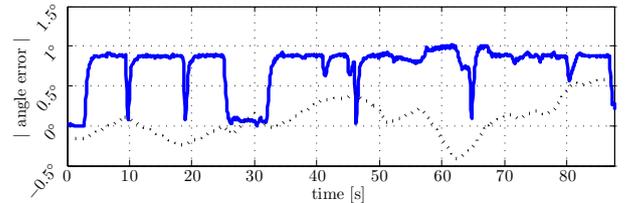
The experimental setup consists of a floor mounted robot (FR) and a gantry mounted robot (GR) shown in Fig. 3. An operator can perform various joystick-controlled operations with the GR such as using a gripper. The FR holds a stereo vision camera and is given joint angle increments in real time by numerically integrating (8). Only position measurements are available, and it is assumed that  $P$  is negligible. The desired view point  $p$  is set to the end effector of the GR. This provides the operator with a clear view of the operation performed with the GR. Further details regarding the experiment, laboratory setup and the computer system used is found in [17].

Two separate experiments were performed and these are presented and analyzed in order to highlight different aspects of the closed loop robot controller performance. The numerical integration of (8) was implemented in Matlab (R2009b), and was ported to C using the embedded Matlab subset in order to minimize its runtime and overhead.

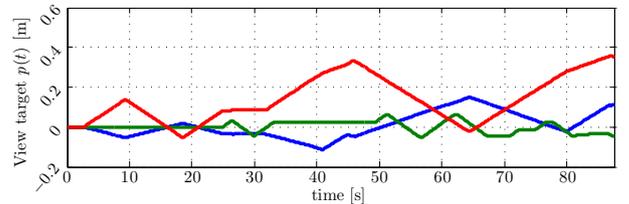


(a) The absolute angle error  $\text{acos}(e_3^T(\mathbf{x} - \mathbf{p})/\|\mathbf{x} - \mathbf{p}\|)$  during three separate transient approaches

Fig. 4: Data from three transient approach experiments where the focus point is fixed and the initial error is large.



(a) The absolute angle error  $\text{acos}(e_3^T(\mathbf{x} - \mathbf{p})/\|\mathbf{x} - \mathbf{p}\|)$  during tracking. A scaled norm of  $\mathbf{p}$  is shown dotted to indicate the movement of the focus point.



(b) The leader end effector position  $\mathbf{p}(t)$  initialized at zero during tracking.

Fig. 5: Data from a camera tracking experiment where the initial error is zero.

### B. Approach transient

The first experiment shows the transient of the FR following an initial condition with a large error. Three separate experiments were conducted. The leader robot is static throughout the run. The camera angle error is shown in Fig. 4. A graceful transient is achieved even when the FR is initially facing the wrong direction.

### C. Camera tracking

This experiment shows the tracking performance as the leader robot is moving around. The focus point  $\mathbf{p}$  is set to be the end effector position of the leader. The initial error is small. The leader robot is freely controlled on-line with inputs given by an Xbox gamepad such that only delayed position measurements of the leader can be used. Plots of the experiment data is shown in Fig. 5. The camera angle error is within  $\pm 1^\circ$ . The error is due to the unknown velocity of the leader which appears as the disturbance term  $\mathbf{P}$  in the closed loop system (4).

## IX. CONCLUSIONS AND FUTURE WORK

### A. Conclusions

In this paper we have shown how the stereographic projection may be used in combination with the pseudoinverse redundancy resolution method to achieve global control of a partial end effector orientation. In order to demonstrate the approach, a robot equipped with a stereo vision camera is controlled to follow and point its camera toward a leader robot controlled freely by a human operator. Experimental results validated the proposed control methodology. The control law is applicable to similar tasks where the direction of the end effector is important, such as spray painting curved surfaces.

The stereographic projection does not introduce any new singularities to the system. It was shown that if the robot is in a singular configuration, then there exists at most one reference such that the pseudoinverse is undefined. This reference, if it exists, is given in closed form as a function of the robots joint angles.

The coordination control problem where a follower robot is given the task to monitor a leader robot with a camera was efficiently solved using pseudoinverse redundancy resolution. Joint angle increments were generated for the follower in real-time using only joint position measurements. The proposed control strategy is verified through experiments on two 6-DOF industrial manipulators.

In order to develop smaller offshore oil and gas fields, more cost effective solutions are required. Robots may constitute part of such a solution and the results presented in this paper form steps toward such technology progress.

### B. Future work

The control law proposed in this paper and the experimental system used is further elaborated on in [17]. Pseudoinverse weighting and nullspace control is used to achieve obstacle avoidance while respecting the inherent joint range limitations

With the current approach, the follower's knowledge of the leader's state is limited to position measurements. Velocity and acceleration estimates of the leader robot may improve the tracking error, especially for high speed camera tracking.

## X. ACKNOWLEDGMENTS

This work is part of the research project "Next Generation Robotics for Norwegian Industry". The project partners are The Research Council of Norway, SINTEF, NTNU, Statoil, Hydro, Tronrud Engineering, SbSeating, Glen Dimplex Nordic and RobotNorge.

### APPENDIX I

#### DERIVATION OF THE TASK JACOBIAN

The derivation of the task Jacobian is notationally shortest using the differentiation of vectors in rotating coordinate systems, see [18] p.242. Define  $(\mathbf{p} - \mathbf{x})/\|\mathbf{p} - \mathbf{x}\| = \mathbf{R}\mathbf{y}$

as  $\mathbf{y}^i = \mathbf{R}\mathbf{y}^b$  where the superscripts denote the inertial and body fixed coordinate systems. The vector of interest is the time derivative of  $\mathbf{y}^b$ , and in coordinate fixed form we have

$$\dot{\mathbf{y}}^i = \mathbf{R} \left[ \dot{\mathbf{y}}^b + \boldsymbol{\omega}^b \times \mathbf{y}^b \right]. \quad (33)$$

Isolating  $\dot{\mathbf{y}}^b$  and using  $\boldsymbol{\omega} = \mathbf{R}\boldsymbol{\omega}^b$  and  $\mathbf{y}^b = \mathbf{y}$  yields

$$\dot{\mathbf{y}}^b = \mathbf{R}^T \frac{d}{dt} \frac{\mathbf{p} - \mathbf{x}}{\|\mathbf{p} - \mathbf{x}\|} + \mathbf{y} \times \mathbf{R}^T \boldsymbol{\omega}. \quad (34)$$

Defining  $\mathbf{z} = \mathbf{p} - \mathbf{x}$  shortens the first term in (34), and we calculate it using the chain rule

$$\frac{d}{dt} \frac{\mathbf{z}}{\|\mathbf{z}\|} = \frac{\dot{\mathbf{z}}}{\|\mathbf{z}\|} - \frac{\mathbf{z}\mathbf{z}^T}{\|\mathbf{z}\|^3} \dot{\mathbf{z}}. \quad (35)$$

One can verify elementwise that

$$\mathbf{I}_{3 \times 3} - \frac{\mathbf{z}}{\|\mathbf{z}\|} \frac{\mathbf{z}^T}{\|\mathbf{z}\|} = -\mathbf{S}^2 \left( \frac{\mathbf{z}}{\|\mathbf{z}\|} \right), \quad (36)$$

such that, using  $\mathbf{R}^T \mathbf{S}(\mathbf{z}) = \mathbf{S}(\mathbf{R}^T \mathbf{z}) \mathbf{R}^T$ , and substituting (36) into (34) gives the result

$$\frac{d\mathbf{y}}{dt} = -\frac{1}{\|\mathbf{z}\|} \mathbf{S}(\mathbf{y})^2 \mathbf{R}^T \dot{\mathbf{z}} + \mathbf{S}(\mathbf{y}) \mathbf{R}^T \boldsymbol{\omega} \quad (37)$$

The final result stated in III-D follows by using  $\dot{\mathbf{z}} = \dot{\mathbf{p}} - \dot{\mathbf{x}}$ ,  $\dot{\mathbf{x}} = \mathbf{J}_v \dot{\mathbf{q}}$ ,  $\dot{\boldsymbol{\omega}} = \mathbf{J}_v \dot{\mathbf{q}}$  and factoring out  $\mathbf{J}^T = [\mathbf{J}_v^T, \mathbf{J}_w^T]^T$ .

### APPENDIX II

#### PROOF OF PROPOSITION 1

If we assume that there exists a singular configuration  $\mathbf{q}_s$  and a view point  $\mathbf{p}_s$  such that the task Jacobian is singular, then we have

$$\text{rank}\{\mathbf{J}_t(\mathbf{q}_s, \mathbf{p}_s)\} = \text{rank}\{\mathbf{E}\mathbf{N}\mathbf{J}(\mathbf{q}_s)\} = 1. \quad (38)$$

Where  $\mathbf{E}$  is given by (20) and  $\mathbf{N} = \text{blockdiag}\{\mathbf{R}^T, \mathbf{R}^T\}$ . It is convenient for the analysis to factor out  $2(y_3 + 1)^{-1}$  and to separate the view distance as

$$\mathbf{E}_k = \begin{bmatrix} k\mathbf{v}_1^T & \mathbf{v}_2^T \\ k\mathbf{v}_2^T & -\mathbf{v}_1^T \end{bmatrix}, \quad k = \frac{1}{\|\mathbf{p}_s - \mathbf{x}\|}, \quad (39)$$

and let

$$\mathbf{J}_R = \begin{bmatrix} \mathbf{R}^T(\mathbf{q}_s) & \mathbf{I}_{3 \times 3} \\ \mathbf{I}_{3 \times 3} & \mathbf{R}^T(\mathbf{q}_s) \end{bmatrix} \mathbf{J} \quad (40)$$

such that  $\text{rank}\{\mathbf{J}_t\} = \text{rank}\{\mathbf{E}_s \mathbf{J}_R\}$ . The only loss of generality is in the assumption that  $(y_3 + 1) \neq 0$  since  $\text{rank}\{c\mathbf{A}\} = \text{rank}\{\mathbf{A}\}$  for any nonzero constant  $c$ . We note that the two vectors  $\mathbf{v}_i \in \mathbb{R}^3$ ,

$$\mathbf{v}_1 = \begin{bmatrix} -\frac{y_3+1-y_1^2}{y_3+1} \\ \frac{y_1 y_2}{y_3+1} \\ y_1 \end{bmatrix}, \quad \mathbf{v}_2 = \begin{bmatrix} \frac{y_1 y_2}{y_3+1} \\ -\frac{y_3+1-y_2^2}{y_3+1} \\ y_2 \end{bmatrix} \quad (41)$$

together with  $\mathbf{y}$  form an orthonormal basis of  $\mathbb{R}^3$  for all  $y_3 \neq -1$  such that

$$\mathbf{v}_1(\mathbf{y}) \perp \mathbf{v}_2(\mathbf{y}), \quad \|\mathbf{v}_1(\mathbf{y})\| = \|\mathbf{v}_2(\mathbf{y})\| = 1, \quad \mathbf{v}_1 \times \mathbf{v}_2 = \mathbf{y}. \quad (42)$$

The two rows of  $\mathbf{J}_t$  are linearly dependent since  $\mathbf{J}_t$  has rank 1, such that there exist constants  $a, b$  that are not both zero which satisfy

$$a\mathbf{E}_{k_1}^T \mathbf{J}_R + b\mathbf{E}_{k_2}^T \mathbf{J}_R = \mathbf{0} \quad (43)$$

where we denote  $\mathbf{E}_k^T = [\mathbf{E}_{k_1} \mathbf{E}_{k_2}]$ ,  $\mathbf{E}_{k_i} \in \mathbb{R}^6$ . Let  $\boldsymbol{\eta}$  span the left nullspace of  $\mathbf{J}_R$  such that  $\boldsymbol{\eta}^T \mathbf{J}_R = \mathbf{0}$ , then Eq. (43) is true if and only if

$$a\mathbf{E}_{k_2} + b\mathbf{E}_{k_1} = \boldsymbol{\eta}, \quad (44)$$

for some possibly new constants  $a, b$ . Equation (44) written in terms of  $\mathbf{v}_i, k$  is given by

$$a \begin{bmatrix} k\mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix} + b \begin{bmatrix} k\mathbf{v}_2 \\ -\mathbf{v}_1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{\eta}_1 \\ \boldsymbol{\eta}_2 \end{bmatrix} \quad (45)$$

We may now state the condition on the vector  $\boldsymbol{\eta}$  under which (44) is true. First we note that if  $\boldsymbol{\eta}^T = [\boldsymbol{\eta}_1^T, \boldsymbol{\eta}_2^T]$ , then  $\boldsymbol{\eta}_1 \perp \boldsymbol{\eta}_2$  which may be seen by taking the inner product

$$\boldsymbol{\eta}_1^T \boldsymbol{\eta}_2 = (ak\mathbf{v}_1^T + bk\mathbf{v}_2^T)(a\mathbf{v}_2 - b\mathbf{v}_1), \quad (46)$$

which is zero since  $\mathbf{v}_1$  and  $\mathbf{v}_2$  are orthonormal

$$\boldsymbol{\eta}_1^T \boldsymbol{\eta}_2 = -kab\|\mathbf{v}_1\|^2 + kab\|\mathbf{v}_2\|^2 = 0. \quad (47)$$

We may also observe that  $\boldsymbol{\eta}_1$  and  $\boldsymbol{\eta}_2$  are never zero, since they are expressed as a sum of orthonormal vectors. It is also apparent, by taking the inner product  $\mathbf{y}^T \boldsymbol{\eta}_i$  that  $\boldsymbol{\eta}_1 \perp \mathbf{y}$  and  $\boldsymbol{\eta}_2 \perp \mathbf{y}$ . The last four linearly independent equations which need to be satisfied are given by the following inner products

$$ak = \mathbf{v}_1^T \boldsymbol{\eta}_1, \quad bk = \mathbf{v}_2^T \boldsymbol{\eta}_1, \quad -b = \mathbf{v}_1^T \boldsymbol{\eta}_2, \quad a = \mathbf{v}_2^T \boldsymbol{\eta}_2. \quad (48)$$

Since either  $a$  or  $b$  are nonzero, we may find that the view distance  $k^{-1}$  required is

$$k^{-1} = \|\mathbf{p}_s - \mathbf{x}(\mathbf{q}_s)\| = \left\{ \begin{array}{l} \mathbf{v}_2^T \boldsymbol{\eta}_2 \\ \mathbf{v}_1^T \boldsymbol{\eta}_1 \end{array}, -\frac{\mathbf{v}_1^T \boldsymbol{\eta}_2}{\mathbf{v}_2^T \boldsymbol{\eta}_1} \right\}. \quad (49)$$

One can verify that these two possible solutions are identical, but the sign depends upon whether or not  $\{\boldsymbol{\eta}_1, \boldsymbol{\eta}_2\}$  forms a right handed coordinate system in the tangent plane to  $\mathbf{y}$ . If  $\{\boldsymbol{\eta}_1, \boldsymbol{\eta}_2\}$  forms a right handed coordinate system, we have for some angle  $\theta$

$$\mathbf{v}_1^T \boldsymbol{\eta}_1 = \|\boldsymbol{\eta}_1\| \cos(\theta), \quad \mathbf{v}_2^T \boldsymbol{\eta}_2 = \|\boldsymbol{\eta}_2\| \cos(\theta), \quad (50)$$

and for a left handed coordinate system we have

$$\mathbf{v}_1^T \boldsymbol{\eta}_1 = \|\boldsymbol{\eta}_1\| \cos(\theta), \quad \mathbf{v}_2^T \boldsymbol{\eta}_2 = -\|\boldsymbol{\eta}_2\| \cos(\theta). \quad (51)$$

For the left handed coordinate system we get  $k = -\|\boldsymbol{\eta}_1\|/\|\boldsymbol{\eta}_2\|$ , but since  $k > 0$  this cannot be a solution. The valid solution for  $k$  is given by  $k = \|\boldsymbol{\eta}_1\|/\|\boldsymbol{\eta}_2\|$ , and since  $\boldsymbol{\eta}_i \perp \mathbf{y}$ , we have

$$\mathbf{y}^T \boldsymbol{\eta}_i = \left( \mathbf{R}^T \frac{\mathbf{p}_s - \mathbf{x}}{\|\mathbf{p}_s - \mathbf{x}\|} \right)^T \boldsymbol{\eta}_i \Rightarrow (\mathbf{p}_s - \mathbf{x})^T \boldsymbol{\nu}_i = 0. \quad (52)$$

The vector  $\boldsymbol{\nu}^T = [\boldsymbol{\nu}_1^T, \boldsymbol{\nu}_2^T]$  with  $\boldsymbol{\nu}_i = \mathbf{R}\boldsymbol{\eta}_i$  is the basis vector of the left nullspace of the manipulator Jacobian. Since  $(\mathbf{p}_s - \mathbf{x})^T \boldsymbol{\nu}_i = 0$  and we need a right handed coordinate system, we have the following unique solution to the equations (48).

$$(\mathbf{p}_s - \mathbf{x}(\mathbf{q}_s)) = \frac{\boldsymbol{\nu}_1 \times \boldsymbol{\nu}_2}{\|\boldsymbol{\nu}_1\| \|\boldsymbol{\nu}_2\|} k. \quad (53)$$

The final result which completes the proof is given by substituting for  $k$  in (53) and isolating  $\mathbf{p}_s$ .

$$\mathbf{p}_s = \mathbf{x}(\mathbf{q}_s) + \frac{\boldsymbol{\nu}_1 \times \boldsymbol{\nu}_2}{\|\boldsymbol{\nu}_1\|^2}. \quad (54)$$

## REFERENCES

- [1] E. Conkur and R. Buckingham, "Clarifying the definition of redundancy as used in robotics," *Robotica*, vol. 15, no. 5, pp. 583–586, 1997.
- [2] A. Liegeois, "Automatic Supervisory Control of the Configuration and Behavior of Multibody Mechanisms, IEEE trans," *Syst., Man, Cybernetics*, vol. 7, pp. 868–871, 1977.
- [3] J. Baillieul, J. Hollerbach, and R. Brockett, "Programming and control of kinematically redundant manipulators," in *Proc. IEEE Int. Conf. on Decision and Control*, vol. 23. IEEE, 1984, pp. 768–774.
- [4] A. Maciejewski and C. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *The international journal of robotics research*, vol. 4, no. 3, p. 109, 1985.
- [5] N. Hogan, "Impedance Control: An Approach to Manipulation: Part 1–3," *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 107, pp. 1–24, 1985.
- [6] D. Whitney, "The mathematics of coordinated control of prosthetic arms and manipulators," *Journal of Dynamic Systems, Measurement, and Control*, vol. 94, p. 303, 1972.
- [7] J. Hollerbach and K. Suh, "Redundancy resolution of manipulators through torque optimization," *IEEE J. of Robotics and Automation*, vol. 3, no. 4, pp. 308–316, 2002.
- [8] S. Chiaverini, "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators," *IEEE Trans. on Robotics and Automation*, vol. 13, no. 3, pp. 398–410, 2002.
- [9] B. Siciliano, "Kinematic control of redundant robot manipulators: A tutorial," *Journal of Intelligent and Robotic Systems*, vol. 3, no. 3, pp. 201–212, 1990.
- [10] P. Corke, "Visual control of robot manipulators—a review," in *Visual servoing*. Citeseer, 1994.
- [11] T. Yoshikawa, "Dynamic hybrid position/force control of robot manipulators. Description of hand constraints and calculation of joint driving force," *IEEE journal of robotics and automation*, vol. 3, no. 5, pp. 386–392, 1987.
- [12] M. von Wattenwyl, M. Clerici, and H. Brauchli, "Independent Hybrid Force/Motion Control of Constrained Six-Degrees-of-Freedom Manipulators," *Multibody System Dynamics*, vol. 6, no. 4, pp. 327–342, 2001.
- [13] M. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*. Wiley New Jersey, 2006.
- [14] F. Caccavale, C. Natale, B. Siciliano, and L. Villani, "Resolved-acceleration control of robot manipulators: A critical review with experiments," *Robotica*, vol. 16, no. 05, pp. 565–573, 1998.
- [15] J. Latombe, *Robot motion planning*. Springer Verlag, 1990.
- [16] T. Chan and R. Dubey, "A weighted least-norm solution based scheme for avoiding joint limits for redundant joint manipulators," *Robotics and Automation, IEEE Transactions on*, vol. 11, no. 2, pp. 286–292, 2002.
- [17] M. Bjerkeng, A. A. Transeth, K. Pettersen, E. Kyrkjebø, and S. A. Fjerdingen, "Active camera control with obstacle avoidance for remote operations with industrial manipulators: Implementation and experimental results," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, San Francisco, USA, Sept 2011, submitted.
- [18] O. Egeland and J. Gravidahl, *Modeling and simulation for automatic control*. Marine Cybernetics, 2002, vol. 76.