Reconstructing Null-space Policies Subject to Dynamic Task Constraints in Redundant Manipulators

Matthew Howard Sethu Vijayakumar*

September 13, 2007

Abstract

We consider the problem of direct policy learning in situations where the policies are only observable through their projections into the null-space of a set of dynamic, non-linear task constraints. We tackle the issue of deriving consistent data for the learning of such policies and make two contributions towards its solution. Firstly, we derive the conditions required to exactly reconstruct null-space policies and suggest a learning strategy based on this derivation. Secondly, we consider the case that the null-space policy is conservative and show that such a policy can be learnt more easily and robustly by learning the underlying potential function and using this as our representation of the policy.

1 Introduction

Redundant manipulators are characterised as having degrees of freedom in excess of those needed to perform some task. In the control of such systems a popular paradigm is to utilise redundancy through secondary movement policies that complement the primary task goals in some way. Such policies prefer actions that, for example, avoid joint limits [1], kinematic singularities [11] or self-collisions [9]. Traditionally these policies were implemented as optimising some carefully selected *instantaneous cost function* or potential. The approach was first proposed by Liégeois [3] in the context of Resolved Motion Rate Control (RMRC) [10], but has since been extended to other control regimes (notably force based control) and a wider variety of secondary policies. For example Nakamura [5] studied the problem extensively with particular emphasis on optimal RMRC and Resolved Acceleration Control (RAC) of arms with pre-defined tasks and time-integral cost functions.

However, the secondary policy need not be the result of optimisation and the formalism extends to a variety of constraint-based control scenarios. For example, in humanoid robots, a secondary goal might be to maintain some posture when performing some task [2], to perform multiple prioritised tasks at once [8] or perform control subject to contact constraints [6]. Furthermore, many tasks are best described as constrained with respect to certain variables.

*M. Howard and S. Vijayakumar are with the School of Informatics, University of Edinburgh, Edinburgh EH9 3JZ, United Kingdom. E-mail: matthew.howard@ed.ac.uk.

Consider running on a treadmill: the centre of mass and tilt of the torso are constrained and the policy controlling the gait is projected into the null-space of these constraints.

The focus in this paper is on modelling secondary control policies from observations of constrained motion using statistical learning methods. This is a form of direct policy learning, with the difference that the policy is only partially observable. For the learning, most supervised learning techniques require consistent, convex training data. In this paper, we look at the problem of deriving this data and offer two contributions for its solution.

The first concerns the general case where the policy can be any non-conservative vector function of the state. We take a geometric approach to reconstructing the policy based on Euclid's theorem and outline the necessary conditions for exact reconstruction at any given point. Furthermore, we show that this approach suggests an iterative training algorithm for our learner.

The second contribution is to show that, by restricting the class of permissible policies to those that optimise some potential, the policy can be inferred in a simpler and more robust way using a form of inverse optimal control. In this approach, identifying the unconstrained policy from constrained observations, is equivalent to seeking the potential being optimised. We show that by modelling the policy through its potential we can side-step several of the restrictions of the geometric approach.

Finally, we present experimental results for a simulated robot arm in which the reconstructed null-space policy is used to replace that of the original and the resultant behaviour is compared across a variety of consistent task goals.

2 Problem Formulation

We consider control policies of the form

$$\mathbf{u} = \mathbf{u}_{task}(\mathbf{x}, t) + \mathbf{u}_{null}(\mathbf{x}, t) = \mathbf{u}_{task}(\mathbf{x}, t) + \mathbf{N}(\mathbf{x}, t)\mathbf{a}(\mathbf{x}) \tag{1}$$

where $\mathbf{u}$ is the control signal, $\mathbf{u}_{task}$ is the component of $\mathbf{u}$ that satisfies a set of non-linear, time-varying task constraints, $\mathbf{a}(\mathbf{x})$ is a policy pursuing secondary movement goals, and $\mathbf{N}(\mathbf{x}, t)$ is a projection matrix. $\mathbf{N}(\mathbf{x}, t)$ prevents violation of the task constraints by projecting the policy into the null-space. Our goal is to model $\mathbf{a}(\mathbf{x})$ from observations of $\mathbf{u}_{null}$.

In general, $\mathbf{a}(\mathbf{x})$ can be any arbitrary vector field. According to the Helmholtz decomposition, any vector field may be comprised of rotational and divergent components

$$\mathbf{a}(\mathbf{x}) = \nabla_{\mathbf{x}} \times \mathbf{\Phi}(\mathbf{x}) + \nabla_{\mathbf{x}}\phi(\mathbf{x}) \tag{2}$$

where $\mathbf{\Phi}$ and ϕ are vector and scalar potentials. Assuming that $\mathbf{a}(\mathbf{x})$ is conservative[1] an equivalent goal is to model $\phi(\mathbf{x})$. Policies of the form (1) occur in both velocity ($\mathbf{u} \equiv \dot{\mathbf{q}}$) and force ($\mathbf{u} \equiv \boldsymbol{\tau}$) based control [4].

Example 2.1. *Velocity-based Control*
A standard velocity-based control scheme is RMRC [10, 3]

$$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q}, t)^{\dagger}\dot{\mathbf{r}} + \mathbf{N}(\mathbf{q}, t)\mathbf{a} \tag{3}$$

[1] A necessary and sufficient condition for this is that $\nabla_{\mathbf{x}} \times \mathbf{a}(\mathbf{x}) = 0, \forall \mathbf{x}$.

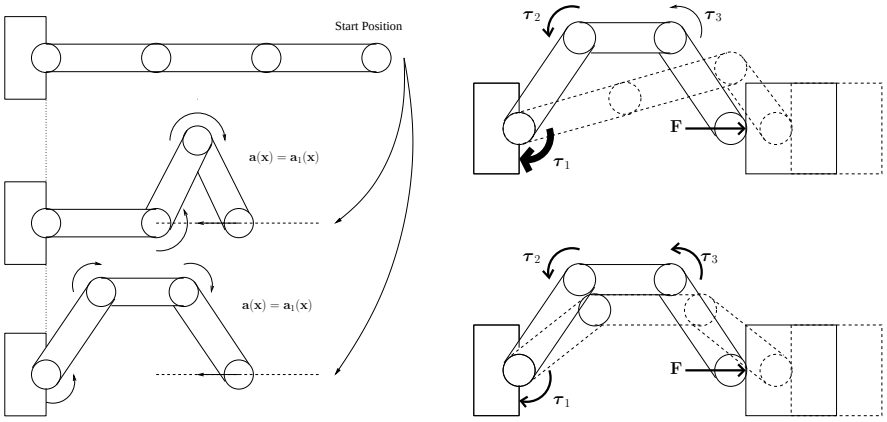

Figure 1: Effect of different null-space policies $\mathbf{a}(\mathbf{x})$ on behaviour when tracking a linear task-space trajectory in RMRC (left) and applying a force $\mathbf{F}$ to a mass in force control (right).

where $\mathbf{r}, \dot{\mathbf{r}} \in \mathbb{R}^k$ *and* $\mathbf{q}, \dot{\mathbf{q}} \in \mathbb{R}^n$*, denote the task- and joint-space positions and velocities and* $\mathbf{J}(\mathbf{q}, t)$ *is the Jacobian with* $\mathbf{W}$*-weighted pseudoinverse* $\mathbf{J}^\dagger = \mathbf{W}^{-1}\mathbf{J}^T(\mathbf{J}\mathbf{W}^{-1}\mathbf{J}^T)^{-1}$*.* $\mathbf{N}(\mathbf{q}, t) = (\mathbf{I} - \mathbf{J}^\dagger(\mathbf{q}, t)\mathbf{J}(\mathbf{q}, t))$ *is the null-space projection matrix (where* $\mathbf{I}$ *is the identity matrix). Note that, in general, the Jacobian and the projection matrix are time-dependent reflecting the fact that the task-space may change in time* [1]*.*

Example 2.2. *Force-based Control*
A general formulation for force-based control is [7]

$$\boldsymbol{\tau} = \mathbf{W}^{-1/2}(\mathbf{A}\mathbf{M}^{-1}\mathbf{W}^{-1/2})^\dagger(\mathbf{b} - \mathbf{A}\mathbf{M}^{-1}\mathbf{F}) + \mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}, t)\mathbf{a} \tag{4}$$

where $\boldsymbol{\tau} \in \mathbb{R}^n$ *is the applied torque/force,* $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \mathbb{R}^n$ *are joint-space positions, velocities and accelerations,* $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$ *is an inertia/mass matrix and* $\mathbf{F} \in \mathbb{R}^n$ *describes perturbing forces such as centrifugal, Coriolis and gravity forces. The weighting matrix* $\mathbf{W} \in \mathbb{R}^{n \times n}$ *determines the control paradigm used, such as RAC (*$\mathbf{W} = \mathbf{M}^{-2}$*) or the Operational Space Formulation (*$\mathbf{W} = \mathbf{M}^{-1}$*)* [7]*. The task is described through constraints of the form* $\mathbf{A}(\mathbf{q}, \dot{\mathbf{q}}, t)\ddot{\mathbf{q}} = \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}, t) \in \mathbb{R}^k$ *and the null-space projection matrix is given by* $\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{W}^{-1/2}(\mathbf{I} - (\mathbf{A}\mathbf{M}^{-1}\mathbf{W}^{-1/2})^\dagger \mathbf{A}\mathbf{M}^{-1}\mathbf{W}^{-1/2})\mathbf{W}^{1/2}$*.*

The correspondence between (1), (3) and (4) can easily be shown by appropriate substitution of variables. In both cases, the second term arises when there is redundancy, i.e. the task dimensionality is lower than that of the action space ($k < n$), allowing secondary goals to be pursued. Fig. 1 shows examples of how different null-space policies affect behaviour.

3 Reconstructing Nullspace Policies

Theorem 3.1. *Reconstruction of Projected Policies*
Given observations $\mathbf{a}^{(i)} = \mathbf{N}^{(i)}(\mathbf{x})\mathbf{a}(\mathbf{x})$*,* $i = 1, \ldots, n$ *of a policy* $\mathbf{a}(\mathbf{x})$ *projected into the null-space of a set of* n *task constraints which that span the action space, the unconstrained policy is given by*

$$\mathbf{a}(\mathbf{x}) = \mathbf{x}^\times - \mathbf{x} \tag{5}$$

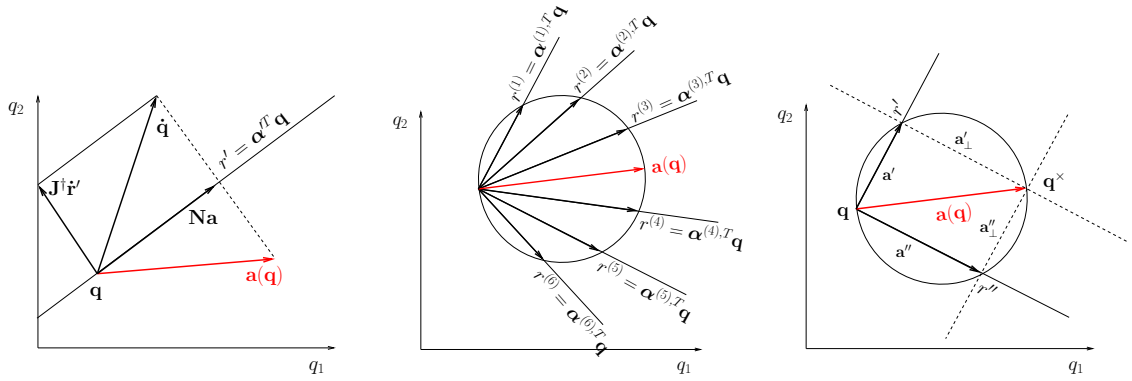

Figure 2: Under the task constraints (7), the null-space policy is projected onto a manifold $r = \boldsymbol{\alpha}^{(t)T}\mathbf{q}$ (left), orthogonal to the task space motion. Under multiple constraints the projected policy vectors lie inscribed in a hypersphere in state-space (centre). Euclid's Theorem can be used to reconstruct $\mathbf{a}(\mathbf{q})$ given observations under different constraints (right).

where $\mathbf{x}^\times$ *is the solution to the linear system*

$$\mathbf{A}\mathbf{x}^\times = \mathbf{d} \tag{6}$$

where $\mathbf{A} \equiv (\mathbf{a}', \ldots, \mathbf{a}^{(m)})^T$ *and the elements of* $\mathbf{d}$ *are given by* $d_i = \mathbf{a}^{(i)T}(\mathbf{x} + \mathbf{a}^{(i)})$.

Proof Consider the RMRC control of a manipulator with two-dimensional joint space, $\mathbf{q} \equiv (q_1, q_2)^T$, and one-dimensional task space $r^{(i)}$, $i = 1, \ldots, n$. The Jacobian of this system

$$\mathbf{J}^{(i)}(\mathbf{q}) = (\alpha_1, \alpha_2)^{(i)} = \boldsymbol{\alpha}^{(i)} \tag{7}$$

is locally linear in the region of $\mathbf{q}$. Under task constraint i the null-space policy is constrained to a line in joint-space with intersection $r^{(i)}$ (Fig. 2, left). When the active constraint changes the rotation of this line changes so that the observed projections lie inscribed within a circle (hypersphere in n-d space) of diameter $||\mathbf{a}(\mathbf{q})||$ (Fig. 2, centre). Euclid's theorem states that any triangle inscribed in a semi-circle is a right-angle triangle. Hence $\mathbf{a}(\mathbf{q})$ is given by the intersection of the lines orthogonal to any two projections $\mathbf{a}', \mathbf{a}''$ (Fig. 2, right). By the same argument, in n-dimensional space, if observations are such that they form a basis set of the space, we can construct planes normal to the projections and solve for the intersection point $\mathbf{q}^\times$. This yields the linear system (6) with the unprojected vector given by (5). □

Theorem 3.1 also suggests the following lemma.

Lemma 3.1. *Given observations* $\mathbf{a}^{(i)} = \mathbf{N}^{(i)}(\mathbf{x})\mathbf{a}(\mathbf{x})$, $i = 1, \ldots, n$ *of a constrained policy* $\mathbf{a}(\mathbf{x})$, *the observation with the largest norm* $||\mathbf{a}^{(i)}||$ *lies closest to the unconstrained policy.*

Proof By inspection of Fig. 2, or by considering that $\mathbf{N}(\mathbf{x}, t)$ is a projection matrix, with k eigenvalues of value 0 and $n - k$ eigenvalues of value 1. Fewer constraints (smaller k) results in larger norms. □

Lemma 3.1 suggests an iterative approach to training whereby if multiple observations are made around the same point, those with the largest norm should be used for learning. This is particularly true of highly redundant systems ($k << n$) where there the policy is much less

constrained. Furthermore, in the limit that observations are made under a single, constant constraint, a consistent policy $\mathbf{u}_{null}(\mathbf{x}) = \mathbf{N}(\mathbf{x})\mathbf{a}(\mathbf{x})$ will be learnt.

The condition in Theorem 3.1 that a spanning set of projections are required to exactly reconstruct the policy is somewhat restrictive, and in real data sets unlikely. However if the policy is conservative (i.e. the first term of (2) is zero) we can side-step these restrictions with the following proposition.

Proposition 3.1. *Reconstruction of Conservative Policies*
Under the same conditions as Theorem 3.1, a conservative policy $\mathbf{a}(\mathbf{x})$ *can be represented by its underlying potential function, which can be learnt without the need for multiple observations or iterative training.*

Consider again the case of RMRC of a redundant manipulator. The potential underlying a conservative $\mathbf{a}(\mathbf{q})$ can be reconstructed through inverse optimal control [4]. The simplest method requires trajectories sampled at some rate ρ resulting in a set of via-points $(\mathbf{q}_1 \dots \mathbf{q}_{\rho\tau})^T$ where

$$\mathbf{q}_{t+1} = \mathbf{q}_t + \mathbf{N}(\mathbf{q}_t)\nabla_{\mathbf{q}}\phi(\mathbf{q}_t) \tag{8}$$

for a trajectory of duration τ and $\mathbf{u}_{task} = 0$. Training samples of $\phi(\mathbf{x})$ can be generated by integrating along trajectories using, for example, the Euler method

$$\phi(\mathbf{q}_{t+1}) = \phi(\mathbf{q}_t) + (\mathbf{q}_{t+1} - \mathbf{q}_t)^T\mathbf{N}(\mathbf{q}_t)\nabla_{\mathbf{q}}\phi(\mathbf{q}_t). \tag{9}$$

The key observation is that the integration in (9) occurs in the direction locally orthogonal to the constraints. We refer the reader to results reported in [4] for empirical evidence supporting Proposition 3.1.

In Fig. 3 the left-hand plot shows the true (blue) and reconstructed (cyan) potential along trajectories under a variety of constraints. Contours show the true (quadratic) potential function over two of the joints of the arm. The trajectories are reconstructed up to a translation in the ϕ-dimension (trajectories have been translated in Fig. 3 for comparison). In the middle and right-hand plots a modified Euler method was used to learn two policies; that derived from a quadratic potential (top row) and a sinusoidal one (bottom row); The middle plots show the true and reconstructed policy subject to constraints on the hand. The right-hand plots show a time-lapse of the arm tracking a linear trajectory using the true and learnt policy in the null-space.

4 Conclusion

We have presented the mathematical basis for direct policy learning of policies subject to dynamic, non-linear constraints. We have shown that in the general case of non-conservative policies exact reconstruction of the policy requires solution of a system of equations constructed from observations under task constraints that span the state-space. We have noted that this suggests an iterative training scheme based on the norm of observed projections. Finally, we have suggested a more robust approach to learning conservative policies through numerical integration techniques and simulation results have been presented for the learning of such policies for a kinematically-controlled three link arm.

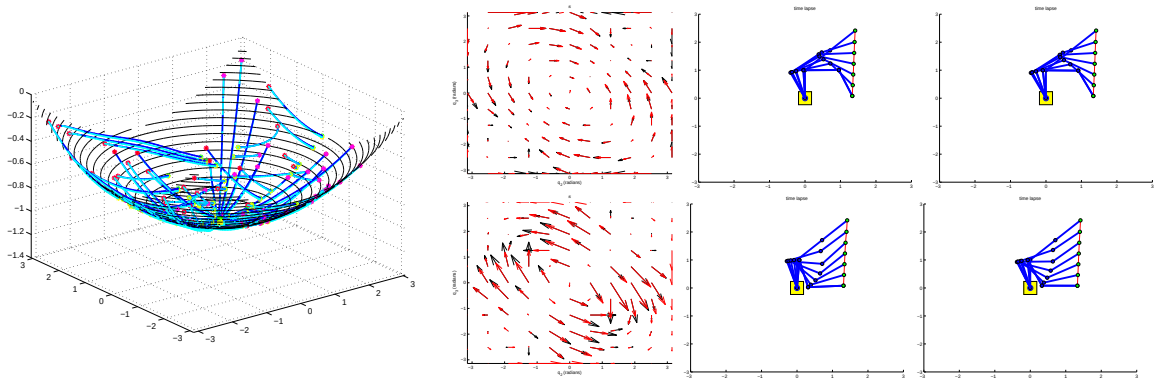

Figure 3: True (blue) and reconstructed (cyan) values of the quadratic potential (contours) along trajectories subject to different constraints (constraints on the hand, wrist, elbow, and unconstrained trajectories shown). True (black) and learnt (red) null-space policies subject to hand constraints for the quadratic (top) and sinusoidal (bottom) potentials. Time-lapse of the arm tracking a linear trajectory using the true (left) and learnt (right) null-space policies.

References

[1] M. Gienger, H. Janssen, and C. Goerick. Task-oriented whole body motion for humanoid robots. In *IEEE-RAS Int. Conf. on Humanoid Robots*, 2005.

[2] O. Khatib, J. Warren, V. De Sapio, and L. Sentis. Human-like motion from physiologically-based potential energies. In J. Lenarcic and C. Galletti, editors, *On Advances in Robot Kinematics*. Kluwer Academic Publishers, 2004.

[3] A. Liégeois. Automatic supervisory control of the configuration and behavior of multibody mechanisms. In *IEEE Trans. Syst., Man, Cybern.*, volume 7, 1977.

[4] Howard M., M. Gienger, C. Goerick, and S. Vijayakumar. Learning utility surfaces for movement selection. In *IEEE Int. Conf. on Robotics and Biomimetics (ROBIO)*, 2006.

[5] Y. Nakamura. *Advanced Robotics: Redundancy and Optimization*. Addison Wesley, Reading, MA, 1991.

[6] J. Park and O. Khatib. Contact consistent control framework for humanoid robots. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2006.

[7] J. Peters, M. Mistry, F. Udwadia, R. Cory, J. Nakanishi, and S. Schaal. A unifying methodology for the control of robotic systems. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2005.

[8] L. Sentis and O. Khatib. A whole-body control framework for humanoids operating in human environments. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2006.

[9] H. Sugiura, M. Gienger, H. Janssen, and C. Goerick. Real-time self collision avoidance for humanoids by means of nullspace criteria and task intervals. In *IEEE-RAS Int. Conf. on Humanoid Robots*, 2006.

[10] D. E. Whitney. Resolved motion rate control of manipulators and human prostheses. 10(22), 1969.

[11] T. Yoshikawa. Manipulability of robotic mechanisms. *Int. J. Robotics Research*, 4(2), 1985.