Principled Domain Extension for Analytic IK

Thomas Cohn*, Seiji Shaw*, Nicholas Roy*, and Russ Tedrake*
*Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, United States.
Email: [tcohn,seijis,nickroy,russt]@mit.edu

I. Introduction

Motion planning algorithms often inherently model configuration space (C-space) as a Euclidean space. But when a robot must satisfy end-effector constraints during its motion (e.g. carrying an object with two hands), the set of valid configurations collapses to a measure-zero submanifold in the ambient configuration space. Recent work has examined the use of *analytic inverse kinematics* (analytic IK), a closed-form mapping from end-effector space to joint angles, to construct charts with a large domain [2, 18]. By planning in that parameterized space, the constraint is eliminated, and the path adheres to the manifold by construction. The mapping is made well-defined by including *self-motion parameters* as input to resolve the many-to-one problem of IK.

Recent works have applied analytic IK to trajectory optimization under end-effector constraints [2]. Decision variables describe a trajectory in parameterized space, costs and constraints are imposed in C-space, and gradient-based optimization is enabled by differentiating through the IK function. A key challenge is the limited domain of an IK function: nonlinear trajectory optimizers do not guarantee workspace reachability at each iteration of the optimization [16, 9]. Bespoke IK solutions made for particular manipulators can be made to return approximate solutions outside of the reachable workspace [8, 4], but popular automatic IK solvers return "no solution" for non-reachable targets [6, 14]. **This is catastrophic for general nonlinear optimizers.**

We extend the domain of an analytic IK function by leveraging least-squares solutions, yielding a closed-form Jacobian for end-effector poses outside the reachable workspace. This preserves the gradient flow outside of the reachable workspace, allowing the optimizer to restore feasibility. Inspired by the numerical IK literature, we present efficient approximations to this Jacobian. Empirically, we demonstrate that the overall framework begins to close the gap between bespoke and automatically generated analytic IK solutions.

II. Background

Some of the following background text is reproduced from [3]. We use monogram notation [15, §3.1] to describe poses: ${}^AX^B \in \mathrm{SE}(3)$ is the pose of frame B relative to (and expressed in) frame A. When this pose is a function of a variable q, we write ${}^AX^B(q)$. The forward kinematics are $\mathrm{FK} : \mathbb{R}^n \to \mathrm{SE}(3)$. An analytic IK function is a mapping

$$\mathrm{IK} : \mathcal{U} \times \Psi \times \mathscr{K} \to \mathbb{R}^n, \tag{1}$$

where $\mathcal{U} \subseteq \mathrm{SE}(3)$ is a subset of the reachable end-effector positions, Ψ parameterizes the set of continuous self-motions and $\mathscr{K}$ enumerates the discrete self-motions (e.g. "up-elbow" vs "down-elbow"). We restrict ourselves to a specific $\kappa \in \mathscr{K}$, and drop it from the notation. Taking derivatives through the parameterization enables gradient-based optimization with variables in minimal coordinates but costs and constraints in C-space. Recent work used the inverse function theorem (IFT) [13, Thm. 2.11] to differentiate through automatically-generated IK solutions without modification [3].

Because robot manipulators often have bounded workspaces, optimization problems in the parameterized coordinates must impose a *reachability constraint* [2]. Even if the IK function returns approximate solutions outside the reachable workspace, there are two issues with the IFT strategy: the contribution of the approximation procedure to the gradient is not accounted for, and the boundary of the reachable workspace is composed of singular points, so the conditions to compute IFT-based gradients are unsatisfied.

III. Methodology

For a given analytic IK solution (1), the *augmented* forward kinematics [7], $\mathrm{FK}_A : \mathbb{R}^n \to \mathrm{SE}(3) \times \Psi$, is constructed to be precisely the inverse of IK, such that $\mathrm{IK}(\mathrm{FK}_A(q)) = q$. Next, to extend the domain of IK, we define

$$\widehat{\mathrm{IK}}({}^WX^G, \psi) = \mathrm{argmin}_q \, \| \, \mathrm{FK}_A(q) - ({}^WX^G, \psi) \|_2^2. \tag{2}$$

If ${}^WX^G$ is reachable, then $\widehat{\mathrm{IK}}({}^WX^G, \psi) = \mathrm{IK}({}^WX^G, \psi)$; for non-reachable targets, it returns a solution that provides the nearest end effector position in a least-squares sense.

We can compute the derivative of $\widehat{\mathrm{IK}}$ via sensitivity analysis. Let $q^* = \widehat{\mathrm{IK}}({}^WX^G, \psi)$, $J_A = \nabla \mathrm{FK}_A(q^*)$, $r_i = \mathrm{FK}_A(q^*) - ({}^WX^G, \psi)$, and $H_i = \nabla_q [J_A]_i$ is the Hessian slice corresponding to the ith row of J_A. Then

$$\nabla\widehat{\mathrm{IK}}({}^WX^G, \psi) = \Big(J_A^T J_A + \sum_{i=1}^{n} r_i H_i \Big)^{-1} J_A^T. \tag{3}$$

This derivative has several important properties. First, observe that if ${}^WX^G$ is reachable, $r_i = 0$, so

$$\nabla\widehat{\mathrm{IK}}({}^WX^G, \psi) = \left(J_A^T J_A \right)^{-1} J_A^T = J_A^\dagger. \tag{4}$$

This project was supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. 2141064, and the MIT Siegel Family Quest for Intelligence. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or the other sponsors acknowledged in this work.

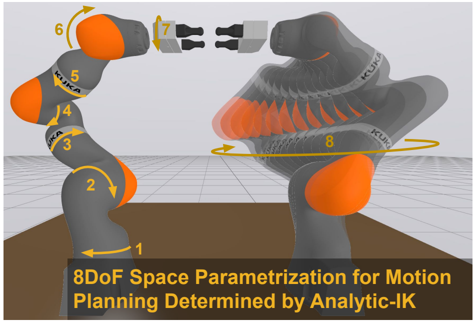

Fig. 1: A minimal coordinates parameterization for a bimanual carrying task. Figure reproduced from [5].

Thus, if J_A is full rank, we recover the IFT formulation [3].

As for the non-reachable case, a nonzero residual means the Hessian term is not eliminated, potentially requiring expensive computations. However, our experimental results will demonstrate that an approximation of (3) is actually more effective. These approximations are inspired by the rich numerical inverse kinematics literature [1], which considers formulations very similar to (3). We now describe the approximations we compare, in order of increasing complexity.

Zero Gradients:

$$\nabla\widehat{\mathrm{IK}}({}^W X^G, \psi) \approx 0. \tag{5}$$

This effectively matches how Cohn et al. [2] handled derivatives for non-reachable end-effector targets.

Pseudoinverse:

$$\nabla\widehat{\mathrm{IK}}({}^W X^G, \psi) \approx J_A^\dagger. \tag{6}$$

If J_A is singular, this should still preserve gradient flow in the non-singular directions while avoiding a numerical blowup.

Levenberg-Marquardt:

$$\nabla\widehat{\mathrm{IK}}({}^W X^G, \psi) \approx \left(J_A^T J_A + \lambda I\right)^{-1} J_A^T. \tag{7}$$

The classic damped least-squares [17]. For λ, we consider both a **Constant** and **Singular Value Thresholding** (SVT), where $\lambda = \lambda_{\max}(1 - (\frac{\sigma_{\min}(J_A)}{\epsilon})^2)$ if $\sigma_{\min}(J_A) \leq \epsilon$, and $\lambda = 0$ otherwise, to only apply damping if J_A is ill-conditioned.

Residual Damping:

$$\nabla\widehat{\mathrm{IK}}({}^W X^G, \psi) \approx \left(J_A^T J_A + \lambda \|r\| I\right)^{-1} J_A^T. \tag{8}$$

For a larger residual, the Jacobian at the projected configuration is less informative.

Anisotropic Damping: Let $J_A = U\Sigma V^T$ be the SVD, $\Lambda_j = \lambda(\|r\|^2 + 3(u_j^T r)^2)$, $\Lambda = \mathrm{diag}(\Lambda_j)$. Then

$$\nabla\widehat{\mathrm{IK}}({}^W X^G, \psi) \approx \left(J_A^T J_A + V\Lambda V^T\right)^{-1} J_A^T. \tag{9}$$

A damping factor is introduced along each singular direction that scales with the magnitude of the residual in that direction.

Full Newton:

$$\nabla\widehat{\mathrm{IK}}({}^W X^G, \psi) \approx \left(J_A^T J_A + \sum_{i=1}^{6} r_i H_i + \lambda I\right)^{-1} J_A^T. \tag{10}$$

TABLE I: Downstream runtimes (seconds) by gradient strategy. IrisNp2 runtimes are averaged over 17 seed points. The more general IFT approach only suffers a minor runtime increase over bespoke autodiff gradient implementations.

Gradient Method	**Trajopt**	**IrisNp2**
Baselines (AD methods)		
Autodiff, Direct Reachability [2]	1.15	1.47
Autodiff, Probing Reachability [4]	0.99	1.40
Inverse Function Theorem with Direct Reachability		
Zero gradients	1.69	5.28
Pseudoinverse	1.58	5.08
Levenberg-Marquardt Constant	0.88	3.70
Singular Value Thresholding	1.39	3.39
Full Newton	1.16	3.33
Residual Damping	1.05	2.21
Anisotropic Damping	1.64	2.20

Newton's method is well-understood for least-squares problems [11, §10]. We only use the first 6 components of the Hessian, corresponding to the end-effector pose, as these can be obtained in closed-form. (The self-motion components would require second-order automatic differentiation.)

IV. Experiments

We evaluate the efficacy of our approach on two downstream optimization tasks for a constrained bimanual carrying setup (Figure 1). First, we test trajectory optimization [15, §6.2] through the parameterization, using bidirectional RRT [10] with shortcutting [12] for the initial guess. Second, we test IrisNp2, an algorithm for constructing collision-free convex polytopes in configuration space [18]. A key step of IrisNp2 is solving a *counterexample search program* to find configurations for which a constraint (e.g. reachability) is violated. Thus, applying IrisNp2 with an IK parameterization requires high-quality gradients outside the reachable workspace.

We compare against a bespoke autodiff-compatible implementation. Timing results are listed in Table I. Our experimental results suggest that the residual damping strategies are the most performant, with smallest difference in runtime performance against the bespoke baseline. The full Newton method's higher runtime is likely due to the higher expense of computing the kinematic Hessian. But it may also reflect the diminishing returns of more accurately approximating something that is itself only an approximation – recall that in practice, IK extensions do not actually return a least-squares solution. We aim to further close the performance gap between the bespoke autodiff-compatible solutions and domain extended-generated solutions in future work.

V. Conclusion

We present a method to extend the domain of analytic IK functions and derive the gradient of the extension. These gradients are essential when solving optimization problems that leverage analytic IK, as the solvers may not guarantee end-effector reachability at every iteration. Importantly, we present approximations of this Jacobian that are verified to be performant for downstream optimization problems that are known to have infeasible iterates.

References

[1] Samuel R Buss. Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods. *IEEE Journal of Robotics and Automation*, 17(1-19):16, 2004.

[2] Thomas Cohn, Seiji Shaw, Max Simchowitz, and Russ Tedrake. Constrained bimanual planning with analytic inverse kinematics. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6935–6942. IEEE, 2024.

[3] Thomas Cohn, Seiji Shaw, Nicholas Roy, and Russ Tedrake. Planning along differentiable charts of constraint manifolds with the inverse function theorem. In *ICRA 2026 Workshop on Geometry in the Age of Data-Driven Robotics*, 2026.

[4] Thomas Cohn, Lihan Tang, Alexandre Amice, and Russ Tedrake. A framework for combining optimization-based and analytic inverse kinematics. *arXiv preprint arXiv:2602.05092*, 2026.

[5] Thomas B Cohn. Motion planning along manifolds with geodesic convexity and analytic inverse kinematics. Master's thesis, Massachusetts Institute of Technology, 2024.

[6] Rosen Diankov. *Automated construction of robotic manipulation programs*. PhD thesis, Carnegie Mellon University, USA, 2010.

[7] Alexander J Elias and John T Wen. Redundancy parameterization and inverse kinematics of 7-dof revolute manipulators. *Mechanism and Machine Theory*, 204: 105824, 2024.

[8] Alexander J Elias and John T Wen. IK-Geo: Unified robot inverse kinematics using subproblem decomposition. *Mechanism and Machine Theory*, 209:105971, 2025.

[9] Philip E Gill, Walter Murray, and Michael A Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM review*, 47(1):99–131, 2005.

[10] James J Kuffner and Steven M LaValle. RRT-connect: An efficient approach to single-query path planning. In *Proceedings 2000 ICRA. Millennium conference. IEEE international conference on robotics and automation. Symposia proceedings (Cat. No. 00CH37065)*, volume 2, pages 995–1001. IEEE, 2000.

[11] Jorge Nocedal and Stephen J Wright. *Numerical optimization*. Springer, 2006.

[12] Sepanta Sekhavat, Petr Svestka, Jean-Paul Laumond, and Mark H Overmars. Multilevel path planning for nonholonomic robots using semiholonomic subsystems. *The international journal of robotics research*, 17(8): 840–857, 1998.

[13] Michael Spivak. *Calculus on manifolds: a modern approach to classical theorems of advanced calculus*. CRC press, 2018.

[14] Siddhartha Srinivasa. ssik: Analytical inverse kinematics for 6R and 7R revolute arms, June 2026. URL https://doi.org/10.5281/zenodo.20561101.

[15] Russ Tedrake. *Robotic Manipulation*. 2024. URL http://manipulation.mit.edu.

[16] Andreas Wächter and Lorenz T Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1):25–57, 2006.

[17] Charles W Wampler. Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods. *IEEE Transactions on Systems, Man, and Cybernetics*, 16(1):93–101, 1986.

[18] Peter Werner, Thomas Cohn, Rebecca H Jiang, Tim Seyde, Max Simchowitz, Russ Tedrake, and Daniela Rus. Faster algorithms for growing collision-free convex polytopes in robot configuration space. *The International Journal of Robotics Research*, 2026.