On Solving Inverse Kinematics of Redundant Robots Using Invertible Neural Networks with Ex-Post Density Estimation

Mridul Mahajan, Gora Chand Nandi
Center of Intelligent Robotics
Indian Institute of Information Technology
Allahabad, India
mridulmahajan44@gmail.com, gcnandi@iiita.ac.in

Abstract

Learning inverse kinematics of humanoid and collaborative robots, which have inherent kinematic redundancy, is a challenging problem due to its multivalued nature. Since these robots hardly obey Pieper's recommendation (Pieper and Roth 1969), solutions to the inverse kinematics problem cannot always be obtained analytically. Recently, Invertible Neural Networks (INNs) have found success in solving such ill-posed inverse problems. In this work, we empirically show that density constraints on the latent variables while training INNs could be replaced by an ex-post density estimation step. The advantage is twofold; the latent variables could have an arbitrarily complex distribution, and posterior mismatch is no longer an issue. Through experiments on learning the inverse kinematics of planar redundant serial robotic manipulators, we validate the efficacy of our approach.

1 Introduction

Robots, especially humanoid and collaborative robots, with kinematic redundancy may not have a guaranteed closed-form inverse kinematics solution if they do not follow Pieper's recommendation (Pieper and Roth 1969). Alternatively, data-driven approaches could be used for such robots. However, inverse kinematics is an ill-posed problem, i.e., different joint space configurations might result in the same pose of the end-effector.

Recently, Invertible Neural Networks (INNs) have found success in solving inverse problems (including inverse kinematics) (Ardizzone et al. 2019). The key idea here is to use a bijective function to model the forward kinematics process and then use the inverse of the function to model the inverse kinematics process. INNs are invertible by construction and thus, can approximate such functions. The information loss due to the forward process is prevented by using additional latent output variables ($\boldsymbol{z}$). The combination of a supervised learning loss (to accurately model the forward kinematics process) and a maximum mean discrepancy loss (to enforce the additional latent output variables to follow a known prior $p(\boldsymbol{z})$) is used to train these networks.

However, this approach has two significant drawbacks. First, for the sake of simplicity, $p(\boldsymbol{z})$ is assumed to be the standard normal distribution. As a consequence, latent variables are not correlated, and thus, we compromise with flexibility. Second, in practical scenarios, $\boldsymbol{z}'s$ distribution never completely matches the assumed prior, $p(\boldsymbol{z})$. This situation is called posterior mismatch. As a consequence, we may sample invalid latent vectors that could produce incorrect solutions to the inverse kinematics problem. The work in (Ghosh et al. 2020) alleviates similar issues in Variational Autoencoders (VAEs) using ex-post density estimation. Essentially, they make the autoencoder completely deterministic and estimate the density of the latent variables post-training. We hypothesize that a similar modification could improve an invertible neural network's performance on inverse problems.

Rest of the paper is organized as follows. First, we give an overview of related work (**Sec. 2**). Next, we formally define the inverse kinematics problem and cast it as a learning problem using INN with ex-post density estimation (**Sec. 3**). We then describe the experimental setup (**Sec. 4**). Thereafter, we present and analyse the results for planar serial robotic manipulators (**Sec. 5**). Finally, we conclude the paper and highlight future research directions (**Sec. 6**).

2 Related Work

Analytical and Numerical Methods For non-redundant robots (e.g., most of the industrial robots), the inverse kinematics problem has closed-form solutions in terms of the $atan2(.)$ function. These solutions can be obtained analytically. However, for robots with kinematic redundancy, numerical methods (e.g., the Newton-Raphson method) are used in practice, but these methods can have very slow convergence.

Conventional Data-Driven Approaches (Jordan and Rumelhart 1992) train two neural networks such that the composition of the two is an identity function. The first network tries to approximate forward kinematics, while the second neural network tries to approximate inverse kinematics. However, jointly training these networks together is very unstable. In (Bócsi et al. 2011), the authors pose inverse kinematics as a structured output prediction problem. They use One Class SVM (OC-SVM) to build an inverse kinematics solver by estimating the joint density of the training data. However, their approach has very

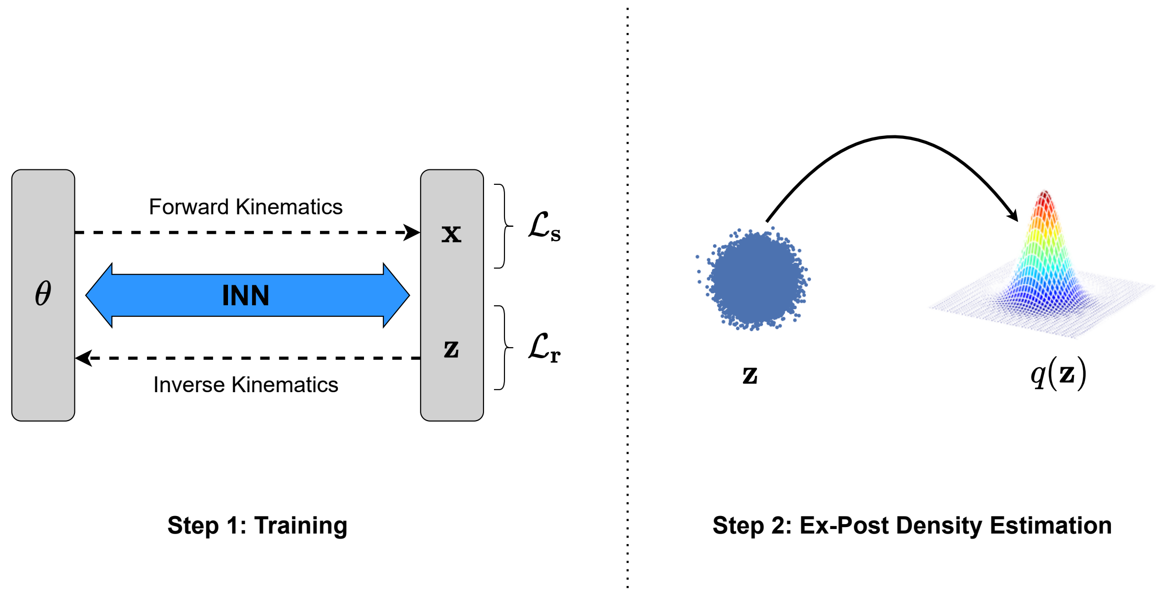

Figure 1: Overview of the proposed method.

high sample complexity and is local in nature. For a more comprehensive review, the reader is directed to (Aristidou et al. 2018).

INN-Based Solver for Inverse Problems The work in (Ardizzone et al. 2019) analyses several inverse problems through the lens of INNs. They use a bidirectional training procedure to train these networks. The forward process is augmented with additional latent output variables $\boldsymbol{z}$ to make it bijective. $\boldsymbol{z}$ is enforced to follow the standard normal distribution by minimizing maximum mean discrepancy (MMD). At test time, a latent vector is sampled from the known prior and fed to the INN for solving the inverse problem.

Ex-Post Density Estimation In (Ghosh et al. 2020), the authors address posterior mismatch in variational autoencoders (VAEs) by replacing the probabilistic component with an ex-post density estimation step. Their approach generates samples with improved quality.

3 Methodology

The position of the links of a manipulator can be specified by an n-dimensional joint vector ($\boldsymbol{\theta}$). The space of all such joint vectors is known as the joint space. The position and orientation ($\boldsymbol{x}$) of the end-effector is measured in the cartesian space. Forward kinematics is modelled by a function f such that $\boldsymbol{x} = f(\boldsymbol{\theta})$. Thus, $f^{-1}(.)$ models inverse kinematics. For redundant robots, f^{-1} is not a unique relationship between the cartesian space and the joint space. Thus, we call it an ill-posed problem.

To model the kinematics using an INN ($g_{\boldsymbol{\phi}}(.)$), the forward kinematics function f is augmented with additional latent output variables ($\boldsymbol{z}$), i.e., $[\boldsymbol{x}, \boldsymbol{z}] = f(\boldsymbol{\theta})$.

We construct the INN using a composition of RealNVP (Dinh, Sohl-Dickstein, and Bengio 2017) blocks. Given an input $\boldsymbol{u}$, a RealNVP block splits it into two halves, $\boldsymbol{u_1}$ and $\boldsymbol{u_2}$, and modifies these halves as follows:

$$\boldsymbol{v_1} = \boldsymbol{u_1} \odot exp(s_2(\boldsymbol{u_2})) + t_2(\boldsymbol{u_2})$$
$$\boldsymbol{v_2} = \boldsymbol{u_2} \odot exp(s_1(\boldsymbol{v_1})) + t_1(\boldsymbol{v_1})$$

Each such block is invertible by construction, which makes the INN invertible as well, since a composition of invertible functions is itself an invertible function.

Given a training set $\{(\boldsymbol{\theta}^{(i)}, \boldsymbol{x}^{(i)})\}_{i=1}^{N}$ of N examples generated using the known forward kinematics process of the system, we learn the kinematics using the INN in two steps.

(Step 1: Training) Unlike (Ardizzone et al. 2019), we train the INN by optimizing a combination of a supervised learning loss (forward kinematics should be accurately modelled), $\mathcal{L}_s$, and a regularization loss on $\boldsymbol{z}$ (the values should not explode), $\mathcal{L}_r$. In particular, we choose $\mathcal{L}_s = \left\| f(\boldsymbol{\theta}) - g_{\boldsymbol{\phi}}^{\boldsymbol{x}}(\boldsymbol{\theta}) \right\|_2^2$ and $\mathcal{L}_r = \left\| g_{\boldsymbol{\phi}}^{\boldsymbol{z}}(\boldsymbol{\theta}) \right\|_2^2$. An important thing to note here is that we do not enforce any constraint on $\boldsymbol{z}$'s distribution. As a result, we need not make a trade-off between the expressivity of the INN (due to the choice of $p(\boldsymbol{z})$) and the computational cost of training it.

(Step 2: Ex-Post Density Estimation) Next, we fit a density estimator $q(\boldsymbol{z})$ to $\{\boldsymbol{z} = g_{\boldsymbol{\phi}}^{\boldsymbol{z}}(\boldsymbol{\theta}^{(i)}) \mid i = 1 \ldots N\}$. In this work, we fit a simple density estimator: a full covariance Gaussian distribution.

4 Experimental Setup

Datasets For a comprehensive evaluation of our approach, we evaluate it on *three* planar serial robotic manipulators (4, 5, and 6 degrees-of-freedom, respectively), all having kinematic redundancy. The forward kinematics process for such a manipulator with n degrees-of-freedom (DOF) is given by the following equations:

$$y = \sum_{i=1}^{n} l_i sin(\sum_{j=1}^{i} \theta_j)$$

$$x = \sum_{i=1}^{n} l_i cos(\sum_{j=1}^{i} \theta_j)$$

Here, l_i denotes the length of the i^{th} link. The end-effector's position is (x, y). To inject kinematic redundancy, we do not consider the end-effector's orientation. For each manipulator, the dataset is constructed using Gaussian priors $x_i \sim \mathcal{N}(0, 0.5)$. Each link has length 0.5 unit, except the last link which is 1 unit long. 2^{20} examples are generated for each manipulator. Out of these, 10^4 examples are kept aside for validation and testing each, and the remaining examples are used for training.

Model Architecture The INN is constructed by stacking 6 RealNVP blocks. Between each consecutive pair of blocks, the transformed vector is permuted in a deterministic manner. Each parameter for affine transformations in a block is determined using a neural network with 2 hidden layers (512 neurons in each layer) and ReLU activations.

Training Setup We perform all experiments on a Tesla K80 GPU. To check the impact of using ex-post density estimation, we train *two* INNs for each task. INN 1 is trained using the bidirectional training procedure described in (Ardizzone et al. 2019). INN 2 is trained by optimizing $\mathcal{L}_{total} = \mathcal{L}_s + \mathcal{L}_r$, and uses a full covariance Gaussian distribution for ex-post density estimation.

Evaluation Metrics For each example in the test set, we sample 10 latent vectors from $p(\boldsymbol{z})$ and report the root-mean-square error (RMSE) on the inverse kinematics task. $p(\boldsymbol{z})$ is $\mathcal{N}(\mathbf{0}, \mathbb{1})$ for INN 1, and $q(\boldsymbol{z})$ for INN 2.

5 Results and Analysis

DOF	RMSE (INN 1)	RMSE (INN 2)
4	0.14	**0.08**
5	0.13	**0.09**
6	0.16	**0.08**

Table 1: RMSE scores for INN 1, and INN 2.

It is evident from Table 1 that ex-post density estimation performs better than bidirectional training. These results are in line with (Ghosh et al. 2020), in which VAEs trained with ex-post density estimation generate samples with improved quality.

6 Conclusion and Future Work

We cast the inverse kinematics problem for redundant robots as a learning problem using invertible neural networks with ex-post density estimation. We have shown that our approach performs better than solvers built using a bidirectional training procedure in solving the inverse kinematics task. As of now, we have implemented our approach for planar serial robotic manipulators having kinematic redundancy. In future, we aim to extend our approach by considering the complete spatial kinematics of real robots.

Acknowledgments

The presented research is jointly sponsored by the Indian Institute of Information Technology (IIIT), Allahabad, and the I-Hub Foundation for Cobotics (Technology Innovation Hub of IIT-Delhi, set up by the Department of Science and Technology, Govt. of India).

References

Ardizzone, L.; Kruse, J.; Rother, C.; and Köthe, U. 2019. Analyzing Inverse Problems with Invertible Neural Networks. In *International Conference on Learning Representations*.

Aristidou, A.; Lasenby, J.; Chrysanthou, Y.; and Shamir, A. 2018. Inverse Kinematics Techniques in Computer Graphics: A Survey. *Computer Graphics Forum*, 37.

Bócsi, B.; Nguyen-Tuong, D.; Csató, L.; Schölkopf, B.; and Peters, J. 2011. Learning inverse kinematics with structured prediction. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 698–703.

Dinh, L.; Sohl-Dickstein, J.; and Bengio, S. 2017. Density estimation using Real NVP. In *International Conference on Learning Representations*.

Ghosh, P.; Sajjadi, M. S. M.; Vergari, A.; Black, M.; and Scholkopf, B. 2020. From Variational to Deterministic Autoencoders. In *International Conference on Learning Representations*.

Jordan, M. I.; and Rumelhart, D. E. 1992. Forward models: Supervised learning with a distal teacher. *Cognitive Science*, 16(3): 307–354.

Pieper, D.; and Roth, B. 1969. The Kinematics of Manipulators Under Computer Control. In *Proceedings of the Second International Congress on Theory of Machines and Mechanisms*, volume 2, 159–169. Zakopane, Poland.