Towards improving incremental learning of manipulator kinematics with inter-robot knowledge transfer

Ndivhuwo Makondo
Mobile Intelligent Autonomous Systems
Council for Scientific and Industrial Research (CSIR)
Pretoria, South Africa
School of Computer Science and Applied Mathematics
University of the Witwatersrand
Johannesburg, South Africa
nmakondo@csir.co.za

Benjamin Rosman
Mobile Intelligent Autonomous Systems
Council for Scientific and Industrial Research (CSIR)
Pretoria, South Africa
School of Computer Science and Applied Mathematics
University of the Witwatersrand
Johannesburg, South Africa
brosman@csir.co.za

***Abstract*—This paper investigates the improvement of learning sensorimotor models for developmental robots, in particular robot arm kinematics models, with inter-robot knowledge transfer. Developmental robots progressively learn through embodied interaction with the physical environment. In the single-robot case, exploration in the world is performed in isolation and the robot explores its own capabilities. In a multi-robot case, with one or more experienced robots, we argue that it may be beneficial for the robots to be able to share the knowledge they have acquired through their individual exploration. We explore knowledge transfer in the context of learning arm kinematics models, where an experienced robot shares its kinematic data with a new robot that is autonomously exploring its environment. We show that the sensorimotor models of the new robot can be bootstrapped by the shared knowledge, converge faster and also achieve a better asymptotic performance compared to individual exploration from scratch. We perform an analysis of knowledge transfer in simulation, ranging from simple two-link planar robots to redundant systems.**

***Index Terms*—Model learning, Transfer learning, Robot kinematics, Robot learning, Developmental robots**

I. INTRODUCTION

In robot behavior modeling and control, kinematics modeling plays an important role. It enables, for example, a robot to make quick predictions about whether it can reach for objects in the surrounding environment using its end-effectors. Moreover, for motions decribed in the task space, or the sensory space, of the robot, a robot kinematics model is required to map them into the joint space, or the motor space, where control typically takes place.

Conventionally, robot kinematics models are designed analytically by an engineer, using parameters of the robot released by the manufacturer, such as link dimensions, number of degrees-of-freedom (DoFs), the configuration of the joints and the connections of the links. If these parameters are not available the engineer must estimate them through analysis of the robot's structure, which can cause modeling errors, requiring further calibration [1]. In such cases, machine learning techniques are employed as an alternative, where robot kinematics models are learned from data generated by the robot through its sensors and actuators. Unknown non-linearities can be taken into account as the model is estimated directly from measured data, while they are typically neglected by the standard analytical modeling techniques [2]. As a result, learning of robot models has attracted much interest and has been used successfully in recent years [2], [3].

In particular, developmental robotics aims to incrementally learn models progressively via embodied interaction with the physical world. Here, a robot is equipped with an exploration mechanism with which it can autonomously explore its surrounding environment in order to collect data from which to learn its kinematics models.

However, to successfully learn robot models, in this case kinematics models, a large amount of data must be collected from the robots, and this can be a time-intensive process for manipulators and humanoids with high DoFs and large state spaces. This is considerably worse in a multi-robot case, where each robot must learn its own models from scratch. In this paper we investigate the improvement of incremental learning of kinematics models for developmental robots in the multi-robot setting using knowledge transfer, where kinematic data generated by pre-existing robots is used to accelerate learning for new robots.

II. PROBLEM STATEMENT

We consider the relation between the motor commands $\boldsymbol{q} \in Q \subset \Re^d$ and their consequences in the sensory space $\boldsymbol{x} \in X \subset \Re^m$ (e.g., the position of the hand), where d is the number of DoFs (or the dimensionality of the joint space) and m is the dimension of the sensory space (e.g., $m = 3$ for the 3D spatial position of the hand). The forward kinematics function $L(\boldsymbol{q}) = \boldsymbol{x}$ describes the unique mapping from the motor space to the sensory space. It can be used to predict the consequences $\boldsymbol{x}^\star$ of setting the joints to some configuration $\boldsymbol{q}^\star$. For robot control, an inverse function $L^{-1}(\boldsymbol{x}) = \boldsymbol{q}$ is required to predict the control command $\boldsymbol{q}^\star$ required to place the robot end-effector at

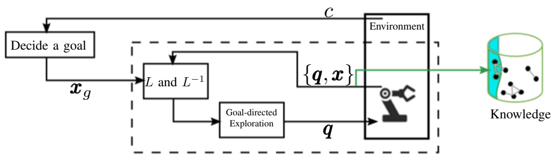


Fig. 1: An illustration of a goal babbling framework. Based on an environmental context c, a high-level learning architecture decides the next goal $\boldsymbol{x}_g$ that a learning agent must attempt to reach (randomly in our case). Without prior knowledge of the inverse model L^{-1}, the agent explores its environment by executing random motor commands $\boldsymbol{q}$ and observing the consequences $\boldsymbol{x}$, and uses the generated data $\{\boldsymbol{q}, \boldsymbol{x}\}$ to update its models, L and L^{-1}.

some desired position $\boldsymbol{x}^{\star}$, which is complex and not uniquely defined if the number of DoFs d exceeds the dimension of the sensory space m ($m < d$).

The two functions model the sensorimotor mappings of the agent, and in developmental robotics they are referred to as sensorimotor models. The robotic agent learns its sensorimotor models by collecting samples $(\boldsymbol{q}^{(i)}, \boldsymbol{x}^{(i)})$ through its interaction with the environment, i.e., by executing motor commands $\boldsymbol{q}$ and observing the sensory consequences $\boldsymbol{x}$ (see Figure 1). One of the challenges in learning L and L^{-1} from autonomous robot exploration is that since Q and X can be high-dimensional, exploration can be a long process. In the next section we briefly review work in accelerating kinematics learning for autonomous robots, after which we present our knowledge transfer approach.

III. Related Work

Kinematics learning has been approached from mainly two angles that compliment each other. In model learning the focus is on developing machine learning algorithms that can learn regression models from large amounts of data in high-dimensional spaces, used to represent the sensorimotor mappings L and L^{-1}. Online and incremental learning algorithms are highly desired as they allow the robot to adapt to changes in the environment and/or the robot itself. Developmental robotics on the other hand attempts to study robotics from the perspective of building capabilities progressively via embodied interaction with the physical world. The focus here has been on designing exploration mechanisms that equip robots with the capability to autonomously explore their surrounding environment in order to collect data from which to learn the sensorimotor mappings.

Combining the two approaches equips individual robots with the capabilities for autonomous exploration and learning of their sensorimotor mappings. Key to this is exposure to good quality data in abundance sufficiently early, as long explorations risk damaging the robot and from wear and tear. A robot can achieve this internally, through implementation of efficient exploration mechanisms, or through some external guidance, using techniques from learning from demonstration and transfer learning. We discuss the two approaches below.

A. Curiosity-driven Exploration

Work in developmental robotics for kinematics learning has led to several mechanisms for *internal* exploration, ranging from exploration in motor space Q, referred to as motor babbling [4] (e.g., exploring the joint space of a robot), to exploration in sensory space X, referred to as goal babbling [5] (e.g., exploring in end-effector space of a robot). Robots can explore these spaces randomly – random motor babbling and random goal babbling – or they can explore actively, making use of the so-called interest models to explore in such a way that maximizes some measure of progress – active motor babbling [6] and active goal babbling [7].

It has been shown that goal babbling can learn inverse models for redundant systems more efficiently, compared to motor babbling, since the sensory space is usually of much lower dimensionality than the motor space for manipulators [5], [7], [8]. Furthermore, efficiency in exploration has been shown to further improve when employing curiosity-driven techniques, where the robot explores in such a way as to maximize some measure of learning progress [7], [8]. However, in a multi-robot setting, these strategies do not take advantage of potential prior knowledge generated by other robots, and so learn from scratch for every agent.

B. Socially-guided Exploration

Alternatively, exploration and learning can be accelerated by an external agent, which can take a form of another robot or a human trainer. These techniques are typically studied under transfer learning, social robotics and learning from demonstration, in which some form of prior knowledge is transferred between agents. However, very little work attempts to transfer knowledge across robots for accelerating learning of kinematics models.

In [9], Procrustes Analysis (PA) was employed to transfer knowledge for learning forward kinematics, where data was generated using analytical models of the robots. In a later study, it was shown that PA is limited to robots with similar kinematics, as it assumes a linear relationship between robotic domains; and that in general non-linear mappings are required between robot kinematic domains [10]. As a result, Local Procrustes Analysis (LPA) was developed and shown to be superior to PA for learning forward kinematics in an offline setting [10].

Here, we provide experiments analyzing knowledge transfer for inverse kinematics in an online setting, in which we employ PA and LPA. These transfer learning models have been successfully applied in other learning settings, such as for learning dynamics models [11], in robot learning from demonstrations [12], [13] and human activity recognition [14].

In [15], an approach for transferring skills from human demonstrations for learning inverse kinematics of a soft-tendon driven manipulator was proposed. Their targeted application was minimally invasive surgical tasks. Work related to ours in this area is in social robotics, where robots interact and learn from humans [16] or socially interact with each other [17]. However, none of the work in social robotics has been applied to transfer knowledge for learning kinematics models. Thus in this paper we present, to the best of our knowledge, the first attempt to transfer knowledge across developmentally learning robots for learning kinematics models.

We aim to transfer knowledge across developmental learning robots. Developmental learning for robots has attracted much interest recently and has shown success in enabling robots to progressively learn their sensorimotor mappings through embodied interaction with the sorrounding environment. However, much of the work for learning kinematics models has focused on single-robot cases, and this typically involves long training times.

IV. Guided Exploration with Knowledge Transfer

In this work we investigate accelerating exploration and learning with knowledge transfer, where an experienced source robot Ω_s, with its sensorimotor models, L_s and L_s^{-1}, shares its kinematics data with a new, inexperienced target robot Ω_t, that is to learn its own sensorimotor models L_t and L_t^{-1}. Our aim is to utilize source agent data $\xi_s = \{\boldsymbol{q}_s^{(i)}, \boldsymbol{x}_s^{(i)}\}_{i=1}^{T}$, generated from source agent sensorimotor models to initialize the parameters of the target agent sensorimotor models, so as to accelerate learning of the target agent.

In general, the source and target agents have different embodiments, resulting in differing motor and/or sensory spaces, i.e., $d_s \neq d_t$ and $m_s \neq m_t$, and different data distributions. Thus, to effectively transfer the source data into the target agent domain, the source data must be configured such that it is useful to the target agent. This can be achieved by learning a domain mapping f from samples of correspondences X_s and X_t generated by the robots, which is then used to transfer source agent data ξ_s into the target agent domain to obtain estimated target agent data $\hat{\xi}_t$, which is subsequently used to initialize the parameters of the target agent sensorimotor models L_t and L_t^{-1}.

In this study we analyze transfer in the case of robots having different data distributions, but with the same dimensionality of their motor and sensory spaces, in order to analyze the possibility and benefit of knowledge transfer. Next, we discuss how to collect correspondences X_s and X_t in Section IV-A, followed by a treatment of how to learn the mapping f used to transfer source robot data ξ_s into target robot data $\hat{\xi}_t$ in Section IV-B.

A. Collecting Correspondences

To collect correspondences X_s and X_t, we propose an algorithm for guiding exploration of the target robot with motor commands generated by the source robot. Similar transfer approaches have been proposed in other learning domains, such as those based on ideas from adaptive control for accelerating learning of dynamic models [18], where they assume the source and target agents have similar motor and/or sensory spaces (i.e., $d_s = d_t$ and/or $m_s = m_t$).

We assume we know correspondences $\boldsymbol{q}_s^{(i)}, \boldsymbol{q}_t^{(i)} \in Q \subset \Re^d$ between the motor spaces of the robots. The correspondences

Algorithm 1 Guided exploration

IN: A sequence of corresponding Ω_t motor commands $\{\boldsymbol{q}_s^{(i)}\}_{i=1}^{T_{guided}}$
$X_s = \{\}$
for $i \in [1, T_{guided}]$ **do**
  Execute motor command $\boldsymbol{q}_s^{(i)}$ on Ω_s
  $X_s = X_s \cup \{\boldsymbol{q}_s^{(i)}, \boldsymbol{x}_s^{(i)}\}$
end for
OUT: Source correspondence samples X_s

are easy to determine manually for the robots used in our experiments. For exploration we employ a goal babbling strategy illustrated in Figure 1, however any strategy is applicable. In this strategy, the robot randomly explores its sensory space X, by choosing random goals $\boldsymbol{x}_g$, and employs the current estimate $\hat{L}^{-1}$ of the inverse model to predict the command $\hat{\boldsymbol{q}}_g$ required to reach $\boldsymbol{x}_g$ with its end-effector. The robot then executes the predicted command and observes the sensory consequences, generating the training pair $(\boldsymbol{q}^{(i)}, \boldsymbol{x}^{(i)})$. The reader is referred to [5] for more details about this strategy.

Typically, a learning algorithm is embedded in the system that incrementally learns $\hat{L}^{-1}$ as the robot explores and generates data, and also guides exploration by predicting the motor input for the next selected target to be explored. In this study, we use a simple weighted nearest neighbor regression model for learning L^{-1}. A forward model $\hat{L}$ can also be learned if needed from the generated data, but is not required by the goal babbling strategy.

In our proposed guided exploration algorithm, the target agent explores its environment for some period T_{guided}, using for example the goal babbling strategy illustrated in Figure 1, to collect $X_t = \{\boldsymbol{q}_t^{(i)}, \boldsymbol{x}_t^{(i)}\}_{i=1}^{T_{guided}}$. The source agent Ω_s then executes the sequence of corresponding commands $\{\boldsymbol{q}_s^{(i)}\}_{i=1}^{T_{guided}}$ to collect $X_s = \{\boldsymbol{q}_s^{(i)}, \boldsymbol{x}_s^{(i)}\}_{i=1}^{T_{guided}}$ corresponding to X_t, using Alg. 1. The domain mapping f can then be learned from X_s and X_t discussed in Section IV-B. The source agent experience, or synthesized, data ξ_s is then transferred to the target agent domain to obtain estimated target agent data $\hat{\xi}_t$, which is subsequently used to initialize target agent sensorimotor models L_t and L_t^{-1}. The target agent then continues exploring autonomously using standard goal babbling as illustrated in Figure 1.

B. Learning the Transfer Model

Given the samples of correspondences $\{\{\boldsymbol{q}_s^{(i)}, \boldsymbol{x}_s^{(i)}\}_{i=1}^{T_{guided}}, \{\boldsymbol{q}_t^{(i)}, \boldsymbol{x}_t^{(i)}\}_{i=1}^{T_{guided}}\}$ with the same dimensionality $d + m$, the domain mapping f thus learns the (non-linear) transformation of sensory signals $\boldsymbol{x}_s^{(i)}$ and $\boldsymbol{x}_t^{(i)}$ given correspondences in the motor space Q. One approach to learning f is using standard regression techniques. However, this would require to collect the same amount of data, if not more, than needed to learn the target robot sensorimotor models L_t and L_t^{-1} without knowledge transfer, which defeats the purpose of transferring knowledge.

Another approach is to use manifold alignment techniques, which allow for knowledge transfer between two seemingly disparate data sets, by aligning their underlying manifolds [19]. Furthermore, they represent the mapping f as an alignment function which they are able to learn from as few samples as possible. In this work we employ Procrustes Analysis and Local procrustes Analysis, as they have proved to be sample efficient when learning f [11].

Procrustes Analysis Learning the mapping f with PA is as follows. First the data X_s and X_t is preprocessed by subtracting the mean and whitening it, obtaining the preprocessed samples $\boldsymbol{s} \in M_s$ and $\boldsymbol{t} \in M_t$, using $\boldsymbol{s} = B_s(\boldsymbol{x}_s - \boldsymbol{\omega}_s)$ and $\boldsymbol{t} = B_t(\boldsymbol{x}_t - \boldsymbol{\omega}_t)$. The values $\boldsymbol{\omega}_s = \mathbb{E}\{X_s\}$ and $\boldsymbol{\omega}_t = \mathbb{E}\{X_t\}$ are the means of the data, where $\mathbb{E}\{\cdot\}$ denotes the expectation operator. Matrices B_s and B_t are obtained such that X_s and X_t are whitened, respectively.

The alignment function is then modeled as a linear mapping $f : M_s \mapsto M_t$, with $f(\boldsymbol{s}) = A\boldsymbol{s}$ where $A^{d\times d}$ is a transformation matrix. The expression for A was derived in [9], and is given as $A = \boldsymbol{\Sigma}_{ss}^{-1}\boldsymbol{\Sigma}_{ts}$, where $\boldsymbol{\Sigma}_{ss}$ is the covariance matrix of the source matrix M_s and $\boldsymbol{\Sigma}_{ts}$ is the covariance between the source and target matrices M_s and M_t. The reader is referred to [9] for a full derivation.

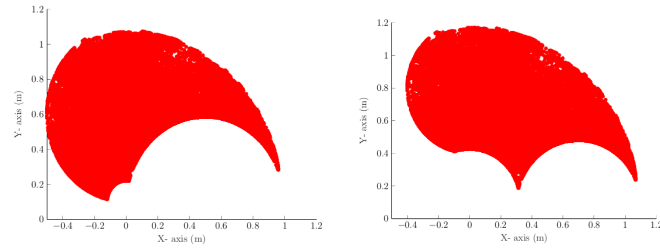
A new point $\boldsymbol{s}_\star = B_s(\boldsymbol{x}_s^\star - \boldsymbol{\omega}_s)$ from the source robot can then be mapped to the target robot using $\hat{\boldsymbol{x}}_t^\star = B_t{}^{\#} A\boldsymbol{s}_\star + \boldsymbol{\omega}_t$, where $\hat{\boldsymbol{x}}_t^\star$ is the transferred point and $B^{\#}$ is the Moore-Penrose inverse of B.

Local Procrustes Analysis LPA extends PA to handle non-linear mappings, by approximating a global non-linear manifold alignment with locally linear functions [10]. To achieve this, LPA first clusters the two data sets X_s and X_t into K local clusters. Then a linear mapping for each cluster is computed using PA. A new data point from the source robot can then be mapped to the target robot by a weighted sum of the linear mappings.

In LPA, clustering is typically performed in the input space of one of the robot domains (the source in our experiments) using Gaussian Mixture Modeling (GMM), and the clusters are transferred to the target domain using correspondence information. Clustering in input space ensures we obtain efficient clusters, because the input and output spaces of the data sets are expected to be correlated. In our case the motor space of the robot $\boldsymbol{q}$ is correlated with the sensory effect $\boldsymbol{x}$ through kinematics of the robot, so clustering is performed in the motor space. The reader is referred to [10] for more information about training and knowledge transfer with LPA.

V. Experiments

We present an analysis of knowledge transfer for accelerating learning of inverse kinematics using two sets of experiments. In Section V-A we present results for transfer between two-link planar robots, to illustrate our approach and to compare transfer with PA and LPA. In Section V-B we analyze knowledge transfer in a more complex scenario, where we transfer knowledge between three-link redundant robots.



(a) Source robot. (b) Target robot.

Fig. 2: 2D task spaces of the robots.

All robots were simulated using Peter Corke's Matlab Robotics Toolbox [20].

We assume knowledge of source agent sensorimotor models and that we can synthesize experience data from them. To evaluate knowledge transfer, we compare the time taken by the target robot to learn from scratch and the time taken to learn when provided with prior knowledge transferred from the source robot. We only conduct experiments for learning inverse kinematics, since learning forward kinematics is easier and the same procedure for transfer is applicable.

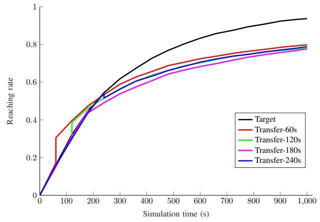
For all learning setups, we evaluate the learning progress by testing the learned target sensorimotor model at evenly spaced time intervals on some test data evenly distributed in the target robot's task space. We use one measure of progress: the *reaching rate*. We calculate the error of reaching all the test points and calculate the reaching rate as the ratio of the points reached within some error threshold – 0.01 m in all experiments. All the experimental results are averaged over 10 runs.

A. Simple Two-link Planar Robots

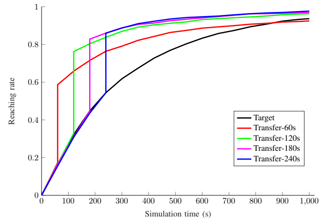
In this experiment we analyze knowledge transfer between two 2-link planar robots with differing link lengths, in order to illustrate our knowledge transfer method. The kinematic parameters of the two robots are as follows: Link 1 and 2 of the source are both 0.5 m long and those of the target are 0.7 and 0.4 respectively, and the joint limits of both robots are $[-\pi/2, 0]$ and $[0.4, 2.9]$ for Motor 1 and 2 respectively. The dimensionality of the motor and sensory spaces of both robots is 2, i.e., $\boldsymbol{q}_s, \boldsymbol{q}_t, \boldsymbol{x}_s, \boldsymbol{x}_t \in \Re^2$, and thus $X_s, X_t, \xi_s, \xi_t \in \Re^4$. The two robots share the same motor space, so correspondences are easily defined in their original motor spaces.

Figure 2 illustrates the differences in distributions of the sensory spaces of the two robots due to the differences in their link lengths. This makes direct transfer of ξ_s into the target domain infeasible. We employ PA and LPA to transfer ξ_s to the target to obtain $\hat{\xi}_t$ to initialize the target robot sensorimotor model. For learning the target robot sensorimotor model from scratch, we perform goal babbling for 1000 seconds.

To evaluate knowledge transfer, we perform guided exploration using Alg. 1 for 60, 120, 180 and 240 seconds and analyze the effect of transfer in the early stages as well as in later stages of learning. After performing guided exploration

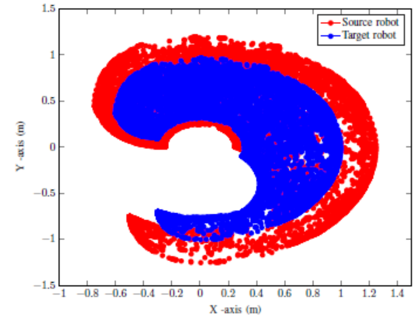


(a) Procrustes Analysis.

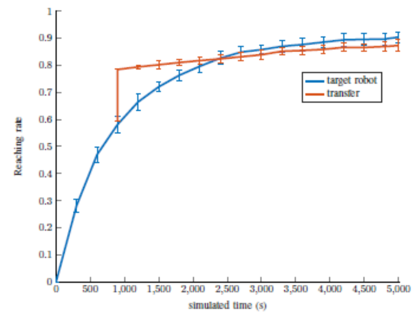


(b) Local Procrustes Analysis.

Fig. 3: Transfer results for two-link robots.



(a) Three link robot spaces.



(b) Three link transfer.

Fig. 4: Transfer with LPA for three link robots.

and transfer from the source robot, the target robot continues to explore for the remainder of the time. For all learning setups, we evaluate the learning progress by testing the sensorimotor model on test points evenly distributed in the task space, at 60 seconds intervals.

Figure 3a and 3b show the average results of knowledge transfer with PA and LPA, respectively, measured in terms the reaching rate. The black curve indicates the progress of the target robot learning from scratch, and the other curves indicate learning with knowledge transfer where the transfer model was learned and transfer applied at different intervals. We observe that the learning progress is boosted instantly when the knowledge is transferred using LPA, and the target robot achieves higher learning rates compared to learning from scratch. However, transfer is beneficial with respect to the final learning performance when applied at 180 seconds and beyond. This is because at 180 seconds there is enough correspondence data collected by the target robot and LPA achieves a better transfer accuracy when given more data [10]; otherwise negative transfer can occur.

Procrustes Analysis on the other hand fails to accelerate learning, with only a slight boost in progress when transfer is applied early. This is due to the linear mapping failing to capture the complex non-linear relationship between the robot kinematic spaces. These results here confirm those of [10]. Although the target robot continues to explore on its own after transfer, the negative transfer caused by the limited linear mappings of PA significantly degrades the learned nearest neighbor-based sensorimotor model.

B. Redundant Planar Robots

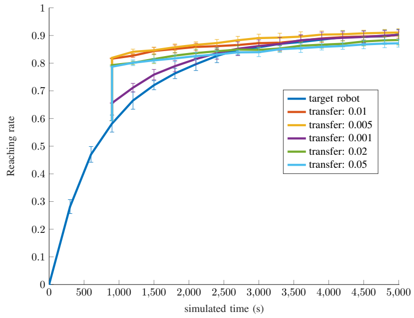
In this experiment we analyze knowledge transfer in a more complex scenario with two 3-link rendundant planar robots with kinematic parameters as follows: Link 1 of the source is 0.4 m long, and Link 2 and 3 are 0.3 m, whereas Link 1 and 3 of the target are 0.5 m long and Link 2 is 0.25 m. The joint limits of both robots are $[-\pi/2, \pi/2]$ for Motor 1 and 3 and $[0, \pi/2]$ for Motor 2. The redundancy in the systems results in more complex mappings between the joint and task spaces, and the mappings required between the robots is also more complex. This is particularly due to the many joint angle configurations mapping to the same points in the task space. Figure 4a shows the task spaces of the source and target robots.

Due to the high dimensionality of the motor space and a larger sensory space, learning sensorimotor models requires more exploration; and thus we perform goal babbling for 5000 seconds to learn from scratch, and apply transfer at 900 seconds. Figure 4b shows the results of transfer using LPA, in which there is an instant performance boost; however, this degrades after 1800 seconds, indicating the occurance of negative transfer. This is in contrast with the simple, non-redundant case in Section V-A, in which no negative transfer occured when transfer was applied at 180 seconds and beyond.

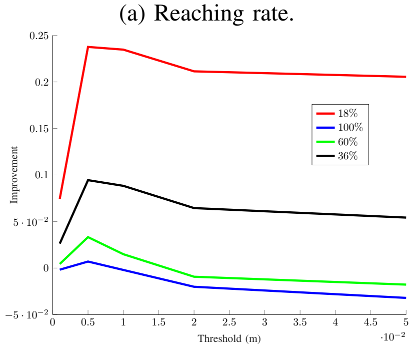
We noticed that LPA achieves a transfer accuracy of about 92.75% on a threshold of 0.01 m. Due to the use of an instance-based sensorimotor model (the nearest neighbor regression model), the 7.25% of the transferred data is inaccurate and thus negatively impacts the prediction of the sensorimotor model.

We illustrate this further in Figure 5, in which we apply transfer for different transfer thresholds based on the ground-truth target robot data. Here, we only use points in the target domain whose transfer errors, as compared to the ground-truth, is below some threshold. We consider thresholds between 0.001 m and 0.05 m.

Results in Figure 5a show that for the threshold of 0.001 there is only a small improvement when transfer is applied, because only a few points could be transferred with an accurancy within the threshold, resulting in very little knowledge being transferred – about 32.1% of source knowledge. For thresholds of 0.005 and 0.01 there is a bigger improvement – with 82.6% and 92.75% of source knowledge reused respectively, with the former leading to a better convergence rate. Finally,



(a) Reaching rate.



(b) Thresholding.

Fig. 5: Transfer analysis for three-link robots.

thresholds of 0.02 and 0.05 have a slightly worse performance boost than 0.005 and 0.01 and their asymptotic performance suffers from negative transfer, with 98.25% and 99.79% of source knowledge reused. This is because bigger thresholds allow more erroneous points to be reused by the target robot, therefore degrading the performance.

Figure 5b shows transfer for different thresholds evaluated at different time steps, indicated by the percentages of the total time it takes to learn from scratch. In all evaluations, transfer achieves the best performance at the threshold of 0.005, and the benefit of transfer is maximum in the early stages of learning, and over time the target robot eventually learns an accurate sensorimotor model. These results show that transfer with LPA in this case is beneficial in the early stages of learning, but negative transfer degrades the asymptotic performance of the target robot.

We suspect this is due to the use of instance based nearest neighbor regression model for learning the sensorimotor models, since it stores and uses all the data in memory, including points that were inaccurately transferred even though it obtains more useful data as it continues learning.

VI. CONCLUSIONS

This paper investigated the improvement of incremental learning of kinematics models for manipulators in a developmental robotics context. Our results demonstrated the possibility and benefit of knowledge transfer, and discovered that in more complex scenarios with redundant robots negative transfer occurs which degrades the learning performance of the target robot.

As future work, we suggest weighting transferred data less, especially since the target robot generates new data as it continues to explore. Alternatively, parameterized model representations can be used, such as neural networks, polynomial regression, etc., whose parameters are updated when new data is generated. This has the potential that the parameters initialized with knowledge transfer may be good enough in the early stages of learning, compared to learning from scratch, and will be updated by new data and quickly adapt, thereby forgetting the inaccurately transferred knowledge.

REFERENCES

[1] N. Makondo, J. Claassens, N. Tlale and M. Braae, ”Geometric technique for the kinematic modeling of a 5 DOF redundant manipulator,” 2012 5th Robotics and Mechatronics Conf. of South Africa, Gauteng, 2012, pp. 1-7.

[2] D. Nguyen-Tuong and J. Peters, Model learning for robot control: A survey,Cognitive Processing, vol. 12, no. 4, pp. 319-340, 2011.

[3] B. Bócsi, D. Nguyen-Tuong, L. Csató, B. Schölkopf, and J. Peters, Learning inverse kinematics with structured prediction, in IEEE Int. Conf. on Intelligent Robots and Systems, pp. 698-703, 2011.

[4] Y. Demiris and A. Dearden, From motor babbling to hierarchical learning by imitation: a robot developmental pathway, 2005.

[5] M. Rolf, J. J. Steil and M. Gienger, ‘Goal Babbling Permits Direct Learning of Inverse Kinematics,’ in IEEE Trans. on Autonomous Mental Development, vol. 2, no. 3, pp. 216-229, Sept. 2010.

[6] R. Saegusa, G. Metta, G. Sandini, and S. Sakka, Active motor babbling for sensorimotor learning, in 2008 IEEE Int. Conf. on Robotics and Biomimetics, pp. 794-799, Feb 2009.

[7] A. Baranes and Oudeyer, ‘Active learning of inverse models with intrinsically motivated goal exploration in robots,’ Robotics and Autonomous Systems, vol 61, no 1, pp.49-73, 2013.

[8] C. Moulin-Frier and P. Y. Oudeyer, ‘Exploration strategies in developmental robotics: A unified probabilistic framework,’ 2013 IEEE Third Joint Int. Conf. on Development and Learning and Epigenetic Robotics (ICDL), Osaka, 2013, pp. 1-6.

[9] B. Bócsi, L. Csató, and J. Peters, ‘Alignment-based transfer learning for robot models,’ Int. Joint Conf. on Neural Networks, 2013, pp. 1-7.

[10] N. Makondo, B. Rosman, and O. Hasegawa, ‘Knowledge transfer for learning robot models via Local Procrustes Analysis,’ IEEE-RAS Int. Conf. on Humanoid Robots, 2015, pp. 1075-1082.

[11] N. Makondo, B. Rosman and O. Hasegawa, ”Accelerating Model Learning with Inter-Robot Knowledge Transfer,” 2018 IEEE Int. Conf. on Robotics and Automation (ICRA), Brisbane, QLD, 2018, pp. 2417-2424.

[12] M. Hiratsuka, N. Makondo, B. Rosman, and O. Hasegawa, ”Trajectory learning from human demonstrations via manifold mapping,” IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2016, pp. 3935-3940.

[13] N. Makondo, M. Hiratsuka, B. Rosman, and O. Hasegawa, 2018. A non-linear manifold alignment approach to robot learning from demonstrations. J. of Robotics and Mechatronics, 30(2), pp.265-281.

[14] R. A. Hamad, E. Järpe, and J. Lundström. ”Stability analysis of the t-SNE algorithm for human activity pattern data,” 2018 IEEE Int. Conf. on Systems, Man, and Cybernetics, In Print.

[15] J. Chen, H. Y. K. Lau, W. Xu, and H. Ren, Towards transferring skills to flexible surgical robots with programming by demonstration and reinforcement learning, in 2016 8th Int. Conf. on Advanced Computational Intelligence (ICACI), pp. 378-384, Feb 2016.

[16] S. M. Nguyen and P.-Y. Oudeyer, Socially guided intrinsic motivation for robot learning of motor skills, Autonomous Robots, vol. 36, pp. 273-294, Mar 2014.

[17] S. Chernova and M. Veloso, Confidence-based multi-robot learning from demonstration, Int. J. of Social Robotics, vol. 2, pp. 195-215, Jun 2010.

[18] J. C. G. Higuera, D. Meger, and G. Dudek, Adapting learned robotics behaviours through policy adjustment, in 2017 IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 5837-5843, May 2017.

[19] C. Wang, and S. Mahadevan, ‘Manifold alignment without correspondence,’ Int. Joint Conf. on Artificial Intelligence, 2009, pp. 1273–1278.

[20] P.I. Corke, Robotics, Vision & Control, Springer 2011, ISBN 978-3-642-20143-1.