# Collaborative goal and policy learning from human operators of construction co-robots

**Harshal Maske**
Mechanical And Aerospace Engineering
Oklahoma State University
maske@okstate.edu

**Milecia Matthews**
Mechanical and Aerospace Engineering
Oklahoma State University
milecia.matthews@okstate.edu

**Allan Axelrod**
Mechanical and Aerospace Engineering
Oklahoma State University
allanma@okstate.edu

**Hossein Mohamadipanah**
Mechanical and Aerospace Engineering
Oklahoma State University
hossein.mohamadipanah@okstate.edu

**Girish Chowdhary**
Mechanical and Aerospace Engineering
Oklahoma State University
girish.chowdhary@okstate.edu

**Christopher Crick**
Computer Science
Oklahoma State University
chriscrick@cs.okstate.edu

**Prabhakar Pagilla**
Mechanical And Aerospace Engineering
Oklahoma State University
pagilla@okstate.edu

## Abstract

Human operators of real-world co-robots, such as excavator, require extensive experience to skillfully handle these complicated machines in uncertain safety-critical environments. We consider the problem of human-robot collaborative learning and task execution, where efficient human-robot interaction is critical to safely and efficiently accomplish complex tasks in uncertain environments. Our collaborative learning algorithm enables a construction co-robot to learn latent task subgoals from the demonstrations of skilled human operators which can then be used to guide novice human operators in completing complex tasks under uncertainty. The effectiveness our algorithm is demonstrated through experimentation on a scaled model of an excavator with guided and unguided human operators. Our results demonstrate that when the co-robot's inferred subgoals are communicated back to the novice human operator, task performance significantly improves.

## 1 Introduction

We consider the problem of **human-robot collaborative learning and task execution** where efficient human-robot interaction is essential to safely and efficiently perform construction tasks using co-robots. Skilled human operators are good at decomposing a loosely defined task (such as loading a truck) into a series of actionable goals. Our goal is to develop a robust solution framework that enables co-robots to (i) learn to decompose loosely defined task into semantics-based subgoals by learning from skilled human operators and (ii) guide novice operators in efficiently decomposing the task to speed up their learning.

We utilize the construction excavator as a co-robot platform to demonstrate the ideas, formulate the collaborative learning problem, and develop algorithm. To corroborate the results we show

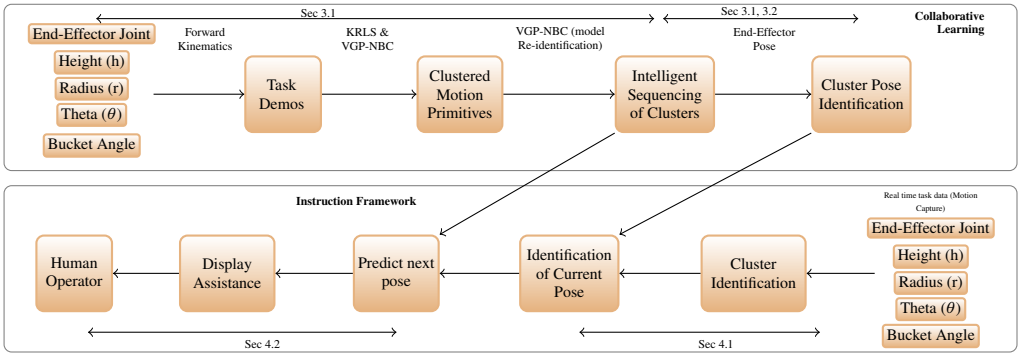



Figure 1: Overview of the LfD and Instruction Framework

Figure 2: An excavator co-robot and its human operator performing a truck loading task. (Image credit http://www.hulcher.com/)

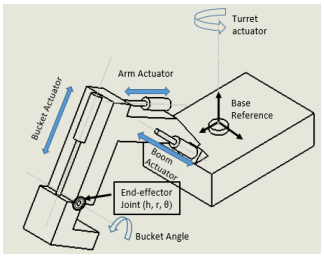



Figure 3: Excavator: 4-DOF arm, Input: End-effector joint $(h, r, \theta)$ w.r.t base frame and bucket angle, Observation: Position of actuators

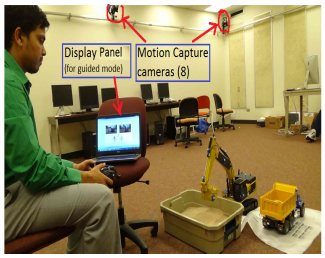



Figure 4: Experimental set-up: a) Motion Capture System, b) Scaled Excavator Model, c) Display panel for guided demos.

experimental implementation on a $1/14^{th}$ scaled-model excavator on a benchmark truck-loading operation performed by a number of guided and unguided human operators. Our approach is to use tools from reinforcement learning, Markov decision processes, and cognitive engineering constructs. Although these tools have been used in learning from demonstration in other areas, there is no known application of these tools to construction robots or to facilitate collaborative learning between co-robots and human operators. As such the contributions of this paper are: (i) A collaborative goal and policy learning algorithm from human demonstration that is scalable to real-world multi-input-output tasks, (ii) A computationally efficient Vector-valued Gaussian Processes Non-Bayesian Clustering (VGP-NBC) algorithm for real time clustering of vector-valued motion primitives, (iii) A semantically motivated instructional framework to train or assist novice users of construction co-robots that lays the foundation for collaborative human co-robot learning, in which a co-robot can transfer learned skills from experts to novice human operators. A block diagram of the presented architecture is in Figure 1. In section 3 the formulation of the presented collaborative learning architecture is laid out. The experimental results with a scaled excavator are discussed in section 4.

## 2 Related Work

Research in Learning from Demonstration (LfD) has focused on enabling robots to learn tasks through task demonstration. LfD has been previously successful in teaching robots tennis swings [14], walking gaits [10], and complex helicopter maneuvers [1]. Recent research has focused on performing automatic segmentation of demonstrations into simpler reusable primitives [7, 8], and then sequencing them in an intelligent manner by learning associated semantics [11].

Several approaches [8, 12] have proposed guidelines to sequence the segmented motion primitives. The recent research by [11] has improved upon these existing methods by utilizing associated coordinate frames of references or state visitation data to perform intelligent sequencing of segmented motion primitives by means of learning associated semantics. In contrast we propose to utilize an

intrinsic frame of reference, that is, a frame of reference which is natural and internal to many construction co-robots, such as the end-effector joint frame (Figure 3) to enable semantically-associated learning of motion primitives.

Currently, [9, 11] are regarded as the state of the art algorithms in policy segmentation. In [11], the Beta Process Autoregressive Hidden Markov Model (BP-AR-HMM) is used to perform offline auto-segmentation of time series data. However BP-AR-HMM is computationally intensive, as it requires solving numerous state equations whose dimensions increase with the length of time, hence affecting the scalability of the BP-AR-HMM. In [9], Dirichlet process mixture models are used to partition the demonstration data and obtain simpler subgoals. However, this process requires thousands of samples to infer the partitions by Gibbs sampling, which again leads to poor scalability. We argue that a computationally efficient method that can perform auto-segmentation in real time is required to enhance the applicability and utility of LfD approaches to the co-robotic domain.

## 3 Formulation

The goal of our work is to learn from an unsegmented demonstrations of a truck loading operation in real time and generate automated guidelines for a novice operator who can then learn to accomplish the same task. To that effect, the algorithm should be able to decompose the demonstrated task into human-understandable motion primitives and also identify their sequence of execution.

### 3.1 Proposed Vector-valued Gaussian Processes and Non-Bayesian Clustering (VGP-NBC)

**VGP :** Time series data with multiple observations $f \in R^D$ can be modeled as a Vector-valued Gaussian Process [3], $f \sim \mathcal{GP}(m, K)$ where $m \in R^D$ is a vector whose components are the mean functions $m_d(x)_{d=1}^D$ of each output and K is a positive matrix valued function with $ND \times ND$ entries. In our work the objective of VGP is to perform prediction over the sparse data set obtained using Kernel Recursive Least Squares (KRLS) algorithm [4]. For a set of inputs $X$, the prior distribution over the vector $f(X)$ is given by $f(X) \in \mathcal{N}(m(X), K(X, X))$, where $m(X)$ is a vector that concatenates the mean vectors associated to the outputs and the covariance matrix $K(X, X)$ is an $ND \times ND$ with entries $(K(x_i, x_j))_{d,d'}$, for $i, j = 1, ..., N$ and $d, d' = 1, ..., D$. For a Gaussian likelihood, the predictive distribution and the marginal likelihood can be derived analytically. The predictive distribution for a data set $X_*$ is [13]

$$p(f(X_*)|S, f, X_*, \phi) = \mathcal{N}(f_*(X_*), K_*(X_*, X_*)) \tag{1}$$

with

$$\begin{aligned} f_*(X_*) &= K_{X_*}^T (K(X, X) + \mathbf{\Sigma})^{-1} \bar{y}, \\ K_*(X_*, X_*) &= K(X_*, X_*) - K_{X_*} (K(X, X) + \mathbf{\Sigma})^{-1} K_{X_*}^T, \end{aligned} \tag{2}$$

where $\mathbf{\Sigma} = \Sigma \otimes I_N$, $K_{X_*} \in R^{DXND}$ has entries $(K(x_*, x_j))_{d,d'}$ for $i, j = 1, ..., N$ and $d, d' = 1, ..., D$ and $\phi$ denotes a set of hyper-parameters of the covariance function. We use multi-output separable kernels as in [3], the equivalent kernel matrix expression is $K(x, x') = k(x, x')B$, where $B \in \mathbb{R}^{D \times D}$ is a symmetric positive semi-definite matrix.

**VGP-NBC :** Next we define the extension of the non-Bayesian clustering method used in GP-NBC [2, 5, 6] to the case of clustering in VGP. Note that this extension is not straightforward, as it requires additional insight to make clustering decisions, which then has dramatic impacts on cluster formation and hence change point detection (CPD) and model re-identification.

Our algorithm maintains a set of points S which are considered unlikely to have arisen from the current model $M_c \sim \mathcal{VGP}(m(X), K(X, X))$. For a VGP, the log likelihood of a set of $D$-dimensional observation points $y \epsilon R^D$ can be evaluated as

$$logP(y|x, M) = -\frac{1}{2}(y - \mu(X_*))^T \Sigma_{X_* X_*} (y - \mu(X_*)) - log|\Sigma_{X_* X_*}|^{1/2} + C \tag{3}$$

where $\mu(X_*) = K(X, X_*)^T (K(X, X) + \omega_n^2 I)^{-1} Y$ is the matrix valued mean prediction using the model $M_c$ and has the dimension $N \times D$, N denotes the size of set $X_*$ ($\epsilon$ $S$) and

**Algorithm 1** VGP Clustering

**Input:** KRLS model $(X, Y)$, lps size $l$, model deviation $\eta$
Initialize VGP Model 1 for $(X, Y)$.
Initialize set of least probable points $S = \emptyset$.
**while** new data is available **do**
  Update the KRLS data dictionary
  Expand the current VGP model $M_c$ using updated KRLS data
  If data is unlikely with respect to $M_c$, include it in $S$ and build corresponding KRLS model $K_S$
  **if** $|S| == l$ **then**
    **for** each model $M_i$ **do**
      - Calculate $D \times D$ log-likelihood matrices of data set $S$ with respect to each of the generated models $M_i$ using (3)
      - Calculate the Frobenius norm for each of these matrices and find lowest likelihood model $M_h$, having lowest norm.
      - Make $M_h$ the current model $M_c$.
      - Create new VGP $M_S$ from $K_S$.
      - $KL \leftarrow \frac{1}{l}(\log(S|M_S) - \log(S|M_c))$
      **if** $\|KL\|_F > \eta$ **then**
        Add $M_S$ as a new model.
      **end if**
    **end for**
  **end if**
**end while**

$\Sigma_{X_*X_*} = K(X_*, X_*) - K(X, X_*)^T(K(X, X) + \omega_n^2 I)^{-1}K(X, X_*)$ is the conditional variance plus the measurement noise. The log-likelihood contains two terms which account for the deviation of points from the mean, $\frac{1}{2}(y - \mu(X_*))^T\Sigma_{X_*X_*}(y - \mu(X_*))$, as well as the relative certainty in the prediction of the mean at those points $\log|\Sigma_{X_*X_*}|^{1/2}$. The set S is used to create a new VGP $M_S$, which is tested against the current ($M_c$) as well as the existing models $M_i$ using a non-Bayesian hypothesis test to determine whether the new model $M_S$ merits instantiation as a new model. This test is defined as

$$\frac{P(y \mid M_i)}{P(y \mid M_j)} \underset{\hat{M}_j}{\overset{\hat{M}_i}{\gtrless}} \eta \tag{4}$$

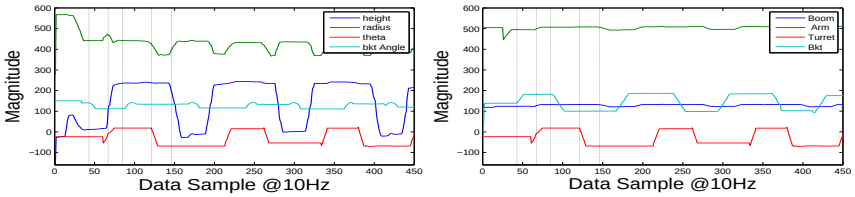
where $\eta = (1 - p)/p$, and $p = P(M_1)$. If the quantity on the left hand side is greater than $\eta$, then the hypothesis $M_i$ (i.e. that the data $y$ is better represented by $M_i$) is chosen, and vice versa. A major difference in the extension to VGP-NBC is due to the fact that the left hand term in (4) is a fraction of two $D \times D$ matrices, which makes it impossible to use (4) in our case. We calculate the Frobenius norm of each of these $D \times D$ matrices, justified by the fact that, being an entry-wise norm, it allows us to consider the cumulative effect of deviations resulting from each observation component interacting with itself as well as with each of the other $D$ observations. The overall algorithm is described in Algorithm 1.

### 3.2 METHODOLOGY

To demonstrate our approach of on-line learning for the benefit of an instruction framework, we propose a novel method to first learn from the time series data available from a given unsegmented demonstration and then assist a novice to perform the task. This approach uses a combination of the KRLS algorithm [4] (to learn) and VGP-NBC to predict the next motion primitive to assist.

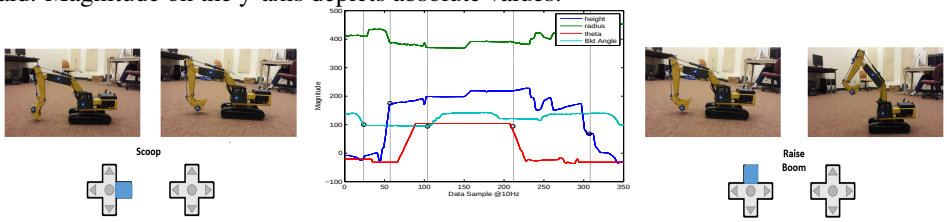
## 4 Simulation Results

Experiments were performed on a $1/14^{th}$ scaled 345D Wedico excavator model, controlled by an RDS 8000 Airtronics radio transmitter. The model robot lacked joint-angle encoders and internal proprioception, so all the experiments were performed inside a motion capture facility to provide real-time data to the algorithm. The experiment required human operators to execute two truck loading cycles in succession. An ideal truck loading cycle is comprised of six different transition




(a) Input data (b) Observation data

Figure 5: Demonstrated data for three cycles of the truck loading task with cluster segmentation overlaid. Magnitude on the y-axis depicts absolute values.




(a) Flash card for Pose 1 (b) Circles depicting cluster end poses (Pose 1 is the extreme left) (c) Flash card for Pose 2

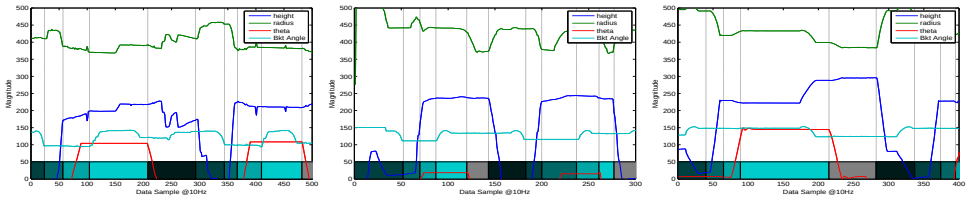
Figure 6: Pose changes associated with the changes in clusters. Flash cards in Fig (a) and Fig (b) were used for instructions in assisted mode

poses, two are shown in figure 6. The data from the expert set of demonstrations was given as an input to the algorithm 1. Next, we designed two sets of experiments to test and compare the performance of novice users with and without the guidance framework. Before each test, the participant learned the control scheme and was given a brief description about the truck loading task. In guided mode, instructions were provided during the task on a display panel facing the operator.

### 4.1 On-line learning from Demonstration

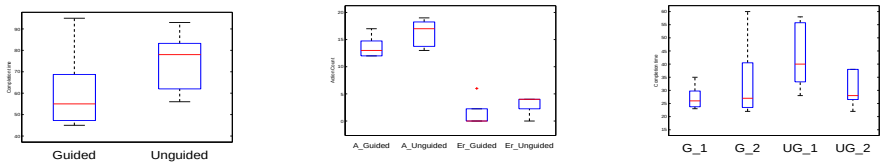
On-line demonstration data for the truck loading task, made available by the motion capture system, is processed by the algorithm (1), to cluster different motion primitives. Height, radius and theta of the end-effector joint $(h, r, \theta)$ along with the bucket angle (figure 3), constituted the input data (figure 5a) to the algorithm, whereas the corresponding actuator positions represented the observations (figure 5b). Data in figure 5 denotes one such sample task demonstration. The dotted vertical lines in figure 5 denote demarcation of different clusters produced by the algorithm (1) in real time.

The model re-identification feature of VGP-NBC ensures that demonstration data which depicts similar motion primitives is identified with previously clustered models, and hence is not reclustered to form a new model. Across multiple demonstrations, sequence of these clusters is identified, which was then associated with the end-effector poses as shown in figure 6. The algorithm was also tested on data possessing variable temporal and spatial characteristics, and the key poses of the end-effector identified by the algorithm were found to be similar, as shown in the figure 7.



(a) Two cycles in 50 sec (b) Two cycles in 30 sec (c) One cycle in 30 sec

Figure 7: VGP-NBC segmentation of data set with different temporal and spatial characteristics. Time (in tenths of a second) is shown on the x-axis




(a) Task completion time in seconds

(b) A: Actions, Er: Erroneous actions

(c) Performance compared over two cycles of truck loading

Figure 8: Comparison between guided (G) and unguided modes (UG)

### 4.2 Instruction Framework

The cluster output of our algorithm matches with the actual end-effector poses which a human trainer might utilize to train novice operators. Based on this result, we then demonstrate the instruction framework, which is founded upon the on-line re-identification of clusters, coupled with the learned sequence of clusters for a given task and the corresponding end-effector poses as illustrated in figure 1. We implemented this framework by providing instructions to the operator performing the task, in the form of flash cards on a display panel, which represent the end-effector poses corresponding to the identified clusters. These flash cards depicted the current pose as well as the target pose, along with the required control actuation as seen in figure 6.

Box plots depict the mean completion time, total number of actions, and erroneous actions for the two cases as shown in figure 8a & 8b; these figures clearly show that (except for a single outlier), the task was performed more efficiently in the "guided" mode as compared to the "unguided" mode. Figure 8c compares performance over the two cycles of truck loading operation. Unguided operator seems to improve their performance over the next cycle, but only manage to match the first cycle performance of guided operators. Further insights into the proposed instructional framework will be generated as a result of psychometric analysis that corroborates subjective feedback from the participants of guided demonstrations.

## 5 Conclusion

Our architecture enables the co-robot to learn latent task subgoals through demonstration from skilled human operators and then use that information to guide novice human operators in completing complex tasks. Our experiment with a scaled model excavator demonstrated that when the co-robot inferred subgoals are communicated back to the novice human operator, task performance significantly improves. Psychometric analysis should yield key insights into ideal human-robot interaction strategies for collaborative learning and task execution.

## References

[1] Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 1. ACM, 2004.

[2] Rakshit Allamaraju, Hassan Kingravi, Allan Axelrod, Girish Chowdhary, Robert Grande, Jonathan P How, Christopher Crick, and Weihua Sheng. Human aware uas path planning in urban environments using nonstationary mdps.

[3] Mauricio A Alvarez, Lorenzo Rosasco, and Neil D Lawrence. Kernels for vector-valued functions: A review. *arXiv preprint arXiv:1106.6251*, 2011.

[4] Yaakov Engel, Shie Mannor, and Ron Meir. The kernel recursive least-squares algorithm. *Signal Processing, IEEE Transactions on*, 52(8):2275–2285, 2004.

[5] Robert Grande, Thomas Walsh, Sarah Fergusson, Girish Chowdhary, and Jonathan How. Online regression for non-stationary data using gaussian processes and reusable models. *Transactions of Neural Networks and Learning Systems*, 2013. submitted.

[6] Robert C. Grande. Computationally efficient Gaussian process changepoint detection and regression. Master's thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, Cambridge MA, June 2014.

[7] Daniel H Grollman and Odest Chadwicke Jenkins. Incremental learning of subtasks from unsegmented demonstration. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 261–266. IEEE, 2010.

[8] George Konidaris, Scott Kuindersma, Roderic Grupen, and Andrew Barto. Robot learning from demonstration by constructing skill trees. *The International Journal of Robotics Research*, page 0278364911428653, 2011.

[9] Bernard Michini, Mark Cutler, and Jonathan P How. Scalable reward learning from demonstration. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 303–308. IEEE, 2013.

[10] Jun Nakanishi, Jun Morimoto, Gen Endo, Gordon Cheng, Stefan Schaal, and Mitsuo Kawato. Learning from demonstration and adaptation of biped locomotion. *Robotics and Autonomous Systems*, 47(2):79–91, 2004.

[11] Scott Niekum, Sachin Chitta, Andrew G Barto, Bhaskara Marthi, and Sarah Osentoski. Incremental semantically grounded learning from demonstration. In *Robotics: Science and Systems*, volume 9, 2013.

[12] Stefanos Nikolaidis and Julie Shah. Human-robot cross-training: computational formulation, modeling and evaluation of a human team training strategy. In *Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction*, pages 33–40. IEEE Press, 2013.

[13] Carl Edward Rasmussen. Gaussian processes for machine learning. 2006.

[14] Stefan Schaal. Dynamic movement primitives-a framework for motor control in humans and humanoid robotics. In *Adaptive Motion of Animals and Machines*, pages 261–280. Springer, 2006.