Scalability of Confidence-Based Autonomy Multi-Robot Demonstration Learning

Sonia Chernova Manuela Veloso

***Abstract*— In this paper, we present the first application of demonstration learning to more than two robots and perform an analysis of the scalability of the Confidence-Based Autonomy (CBA) multi-robot demonstration learning algorithm. Through experimental evaluation using up to seven Sony AIBO robots, we examine how the number of robots being taught by a human teacher at the same time affects the number of demonstrations required to learn the task, the time and attention demands on the teacher, and the delay each robot experiences in obtaining a demonstration. Additionally, we contribute an analysis of a special case of CBA learning in which all robots learn a common task policy.**

I. Introduction

Learning from demonstration, also known as teaching by demonstration, is a learning technique based on human-robot interaction that provides an intuitive interface for robot programming. In this approach, a teacher, typically a human, performs demonstrations of the desired behavior to the robot. The robot records the demonstrations as sequences of state-action pairs, which it then uses to learn a policy that reproduces the observed behavior.

Demonstration-based learning has been gaining widespread attention for providing a fast and intuitive method for transferring knowledge from humans to robots. Recent work has led to the development of a wide variety of *single-robot demonstration learning* algorithms, in which a single person teaches a single robot and a policy is learned based on underlying reinforcement learning [1], classification [8], [9] or regression [2], [7] learning methods.

However, solutions to complex tasks often require the coordination and cooperation of multiple robots. In our previous work, we introduced the *Confidence-Based Autonomy (CBA)* demonstration learning algorithm that enables a single person to teach small groups of autonomous robots to perform collaborative tasks [4], [6]. We believe this to be the first algorithm that enables multiple distributed robots to be taught at the same time using demonstration. The ability to teach multiple robots at the same time is particularly important for addressing collaborative domains as it enables each robot to learn to respond appropriately to the actions of others. In previous work, we demonstrated the feasibility of the CBA learning approach using two humanoid robots performing a joint ball sorting task [6].

In this paper, we extend our previous work and present an analysis of the scalability of the CBA algorithm. Through experimental evaluation, we examine how the number of robots being taught by the teacher at the same time affects the number of demonstrations required to learn the task, the time and attention demands on the teacher, and the delay each robot experiences in obtaining a demonstration. Based on our evaluation using up to seven Sony AIBO robots, we conclude that most of the demands on the teacher and robots grow at a roughly linear rate with respect to the number of learners. Most importantly, our analysis indicates that no strict upper bound exists on the number of robots due to the algorithm's limitations. Instead, the number of robots is likely to be limited by real-world factors particular to each learning domain, such as the amount of time the teacher is able to invest in training.

To allow for a direct comparison between different numbers of robots, each robot in our experiments was taught to perform the same task. As a final contribution of this paper, we present an analysis of a special case of CBA learning in which all robots consolidate their knowledge and share demonstration examples. We show that for this subset of multi-robot learning problems, the training time and number of demonstrations can be significantly reduced by learning a single common policy for all robots.

In the following section, we present the single-robot Confidence-Based Autonomy algorithm that is used to learn an individual policy for each robot. We then present the complete multi-robot learning framework in Section III, followed by a description of the evaluation domain in Section IV. Results of the scalability analysis are presented in Section V. We conclude by examining a special case of CBA learning in Section VI.

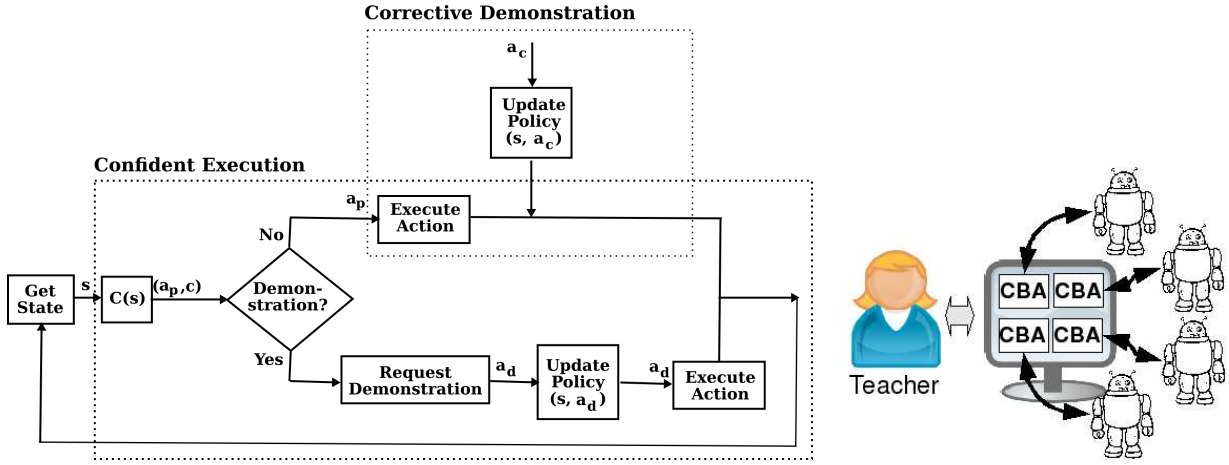
II. Confidence-Based Autonomy Algorithm

In this section, we present a summary of the single-robot Confidence-Based Autonomy algorithm that lies at the heart of our multi-robot demonstration learning framework. For full details and evaluation of CBA, please see [4], [5].

Confidence-Based Autonomy is a single-robot demonstration learning algorithm that enables a robot to learn a policy through interaction with a human teacher. In this learning approach, the robot begins with no initial knowledge and learns a policy incrementally through demonstrations acquired as it practices the task.

Each demonstration is represented by a state-action pair, (s, a), symbolizing the correct action to perform in a particular state. The robot's state s is represented using an n-dimensional feature vector that can be composed of continuous or discrete values. The robot's actions are bound to a finite set $a \in \mathcal{A}$ of action primitives, which are the

S. Chernova and M. Veloso are with the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, U.S.A. {soniac,veloso}@cs.cmu.edu



(a) (b)

Fig. 1. (a) Diagram of Confidence-Based Autonomy, showing the interplay between the Confident Execution and Corrective Demonstration components. (b) Visualization of multi-robot demonstration learning.

basic actions that can be combined together to perform the overall task. The goal is for the robot to learn to imitate the demonstrated behavior by learning a policy mapping states s_i to actions in $\mathcal{A}$. The policy is learned using supervised learning and is represented by classifier $\mathcal{C} : s \rightarrow (a, c)$, trained using state vectors s_i as inputs, and actions a_i as labels. For each classification query, the model returns the highest confidence action $a \in \mathcal{A}$ and action-selection confidence c. CBA can be combined with any supervised learning algorithm that provides a measure of confidence in its classification. In this work, the policy is represented by a one-against-one multiclass Support Vector Machine (SVM) with a radial basis function kernel [3].

The most important element of the CBA algorithm is the method for obtaining demonstration examples, which consists of two components, *Confident Execution (CE)*, an algorithm that enables the robot to learn a policy based on demonstration examples selected by regulating its autonomy, and *Corrective Demonstration (CD)*, an algorithm that enables the teacher to improve the learned policy by correcting mistakes made by the robot through additional demonstrations. Combined, these techniques provide a mechanism for obtaining teacher demonstrations, regulating the robot's autonomy during the learning process, and learning an individual policy for each robot. Figure 1(a) presents an overview of the combined learning process.

A. Confident Execution

The Confident Execution algorithm *enables the robot to select demonstrations* in real time as it interacts with the environment, targeting states that are unfamiliar or in which the current policy action is uncertain. At each timestep, the algorithm obtains the current robot state and determines whether a demonstration of the correct action in this state will provide useful information and improve the robot's policy. If demonstration is required, the robot stops and actively requests help from the teacher (through sound, speech or other means) and waits for the person to select an action using a graphical interface. Upon obtaining the demonstration, the algorithm updates the robot's policy using the acquired action label and performs the demonstrated behavior. If a demonstration is not required, the robot autonomously executes the action specified by its policy without consulting the teacher. Demonstrations are selected based on the action selection confidence of classifier $\mathcal{C}$.

B. Corrective Demonstration

The above Confident Execution algorithm enables the robot to identify unfamiliar and ambiguous states and prevents autonomous execution in these situations. However, states in which an incorrect action is selected with high confidence can still occur, typically due to over-generalization of the classifier. When allowing the robot to select demonstration and regulate its own autonomy, it is important to provide a mechanism for correcting unwanted behavior. The Corrective Demonstration algorithm *enables the teacher to correct the robot's mistakes by performing additional demonstrations*. If an incorrect action is selected for autonomous execution by the Confident Execution algorithm above, Corrective Demonstration allows the teacher to retroactively demonstrate what action should have been selected in its place. In addition to indicating that the wrong action was selected, this method also provides the algorithm with an additional training point, leading the robot to learn quickly from its mistakes.

III. Multi-Robot Learning

Multi-robot learning is achieved by replicating instances of the single-robot CBA architecture, as shown in Figure 1(b). This approach takes advantage of the adjustable autonomy provided by the Confident Execution component of CBA to enable a single teacher to work with multiple robots at the same time. Controlled by its independent instance of CBA, each robot will act autonomously only when highly confident in its actions, and pause to wait for a demonstration in low confidence states.

Using this approach, each robot acquires its own set of demonstrations and learns its individual task policy. Specifically, given a group of robots R, our goal is for each robot $r_i \in R$ to learn policy $\Pi_i : S_i \rightarrow A_i$ mapping from the robot's states to its actions. Note that each robot may have a unique state and action set, allowing distinct policies to be learned by possibly heterogeneous robots. In Section VI we explore a special case of CBA learning in which the desired policy Π_i is the same for all robots. We show that for this case, the number of demonstrations and overall training time can be reduced by learning a single *common* policy.

Algorithm 1 outlines the general procedure followed by the teacher in performing multi-robot demonstrations. Using this approach, the teacher alternates between responding to demonstration requests when they are present, and correcting any mistakes in the autonomous behavior of the robots. Note that the teacher interacts with only a single robot at any one time, while other robots are monitored in the background. The function $f(D)$, which regulates the selection demonstration requests, can be used to implement a variety of selection policies, such as a first-in-first-out or round-robin ordering. In the experiments presented in this paper, a demonstration request is selected arbitrarily from the set by the teacher.

Communication is an important part of many multi-robot tasks. To differentiate between data sources, we represent each robot's state as the union of subsets, $s = \{F_o \cup F_s \cup F_c\}$, such that:

- $F_o =$ set of private, locally observed state features
- $F_s =$ set of locally observed state features that are automatically communicated to teammates each time their value changes
- $F_c =$ set of state features containing data either directly contained in, or calculated based on, information communicated from teammates

This representation seamlessly combines local and communicated data, allowing each robot to make decisions based on all available information.

The CBA demonstration learning algorithm and the presented multi-robot framework have been applied to a wide variety of domains, ranging from a simulated driving task [4] to a ball-sorting task involving two Sony humanoid QRIO robots [6]. In the following section we introduce a new multi-robot domain that is used in the scalability analysis.

IV. Multi-Robot Beacon Homing Domain

Evaluation of the scalability of the Confidence-Based Autonomy algorithm was performed in a beacon homing domain using Sony AIBO robots. Figure 2 shows three examples of these distributed autonomous robots operating in the domain, which consists of an open area with three uniquely-colored beacons ($B = \{B1, B2, B3\}$) located around the perimeter. Each robot is able to identify the relative position of a beacon using its onboard camera, and to communicate information via the wireless network. The set of action available to each robot is limited to basic movement commands, $A = \{Forward, Left, Right, Search, Stop\}$, used by each robot to navigate in the environment.

Algorithm 1 Multi-robot demonstration procedure.

```
Let D be set of current demonstration requests
loop
  if D ≠ ∅ then
    - Select robot demonstration request r according to
    some function f(D)
    - Perform demonstration for robot r
  else
    - Observe autonomous execution of the robots
    if correction is required for robot r then
      - Perform correction for robot r
    end if
  end if
end loop
```

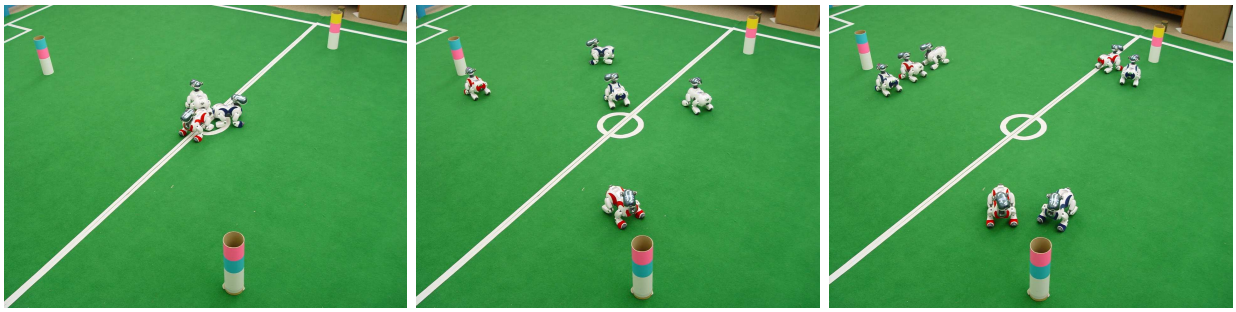
For this task, we represent the robot's state s by the following set of features:

- $F_o = \{B1_d, B1_a, B2_d, B2_a, B3_d, B3_a\}$
- $F_s = \{myBeaconID\}$
- $F_c = \{B1_{nr}, B2_{nr}, B3_{nr}\}$

The set of observed features, F_o, contains information about the robot's relative distance (b_d) and angle (b_a) to each beacon $b \in B$. For any beacon not currently in view, the distance and angle are set to the default values 4000 mm and 1.8 rad, respectively, to indicate that this beacon is far away. The set of shared state features, F_s, contains a single value, $myBeaconID$, which is set to a beacon's ID number if the robot is within a set distance r of a beacon, and -1 if the robot is not located near a beacon. Each robot communicates the value of this feature to its teammates. In turn, all robots use this shared information to determine the values of the calculated features F_c, which maintain the count of the current number of robots occupying each of the beacons.

In summary, using the above representation, each robot knows its position relative to beacons that it observes, and the number of other robots already located at each of the beacons. Using this information, we would like the teacher to teach each robot to navigate from a random initial location in the center of the open region to one of the colored beacons. Specifically, the selection of a beacon is governed by the following rules: *Given a maximum limit* `m` *for the number of robots that can occupy a marker, search for a beacon until one is found for which the number of robots,* b_{nr}*, is less than* `m`*. Navigate to that beacon and occupy it by stopping within a set distance* `r`*. If at any point the number of robots at the selected beacon exceeds* `m`*, search for another beacon.*

These explicit rules of the task are known only to the teacher. During the learning process, each robot in the experiment learns an independent policy representing this behavior from demonstrations. All robots were taught the same task to ensure a fair comparison between robots for the scalability evaluation. The maximum number of robots allowed per beacon for each experiment was set to $m = ceil(\frac{\#Robots}{\#Beacons})$, such that at least one beacon must contain the maximum



(a) (b) (c)

Fig. 2. Beacon homing domain: (a) Example starting configuration, 3 robots. (b) Example intermediate stage, 5 robots. (c) Example final configuration, 7 robots.

number of robots. Each experiment began with all robots located in the center of the open region (Figure 2(a)) and ended once all robots had reached a beacon (Figure 2(c)). Training was performed until all robots executed the desired behavior correctly without requesting demonstrations.

V. Evaluation

The scalability of the CBA algorithm was evaluated in the beacon homing domain using 1, 3, 5, and 7 robots. In this section, we discuss how the number of robots taught by the teacher at the same time affects the number of demonstrations required to learn the task, the demands for time and attention placed on the teacher, and the delay that each robot experiences in obtaining a demonstration.

All evaluation results presented in this paper were performed with a single teacher. As with all human user trials, we must account for the fact that the human teacher also learns and adapts over the course of the evaluation. To counter this effect, the teacher performed a practice run of each experiment, which was then discarded from the evaluation. An alternate evaluation method would be to eliminate the human factor by using a standard controller to respond to all demonstration requests in a consistent manner. This approach, however, would prevent us from evaluating the effect multiple robots have on teacher performance.

A. Robot Autonomy

Figure 3 shows how the level of autonomy, measured as the percentage of autonomous actions versus demonstrations, changes for an individual robot over the course of training. Data in the figure presents the average autonomy over time of robots in the 5-robot beacon homing experiment. The shape of the curve seen in this figure is typical of CBA learning, in which robots begin with no initial knowledge about the task and request many demonstrations early in the training process. The domain knowledge acquired from these initial demonstrations provides the robot with the experience for handling most commonly encountered domain states. As a result, following the initial burst of demonstration requests, the robot quickly achieves 80–95% autonomous execution. The remainder of the training process then focuses

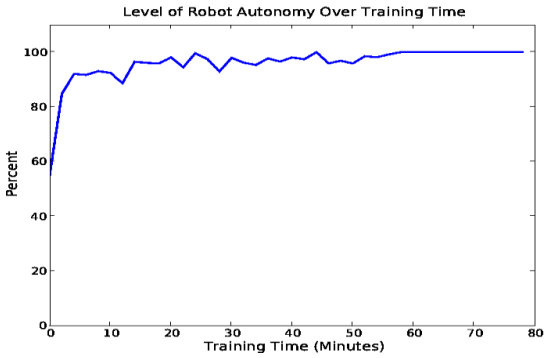


Fig. 3. Average level of autonomy of a single robot over the course of training (5-robot learning example).

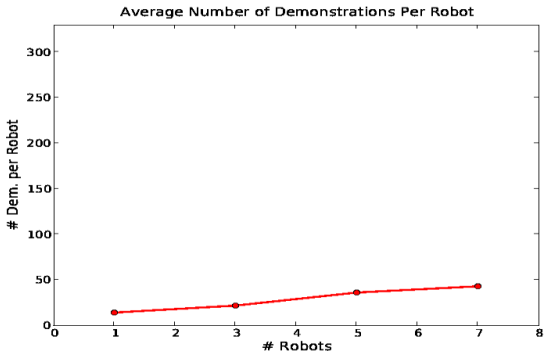


Fig. 4. Average number of demonstrations performed by the teacher for each robot.

on refining the policy and addressing previously unencountered states. The duration of this learning time is dependent upon the frequency with which novel and unusual states are encountered. Learning is complete once the correct action is selected for all states with high confidence.

B. Number of Demonstrations

In this section, we examine how the number of demonstrations performed by the teacher on average for each robot, and in total for each experiment, changes with respect to the number of robots. Figures 4 shows that as the number of robots grows, we observe a slight increase in the number of

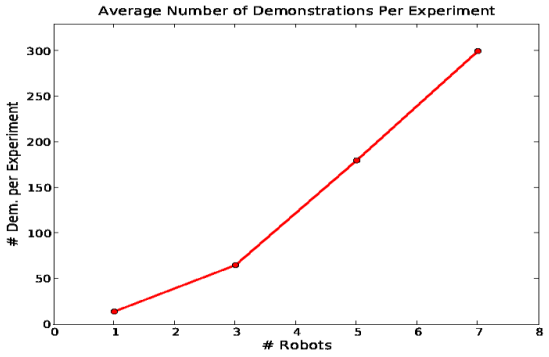


Fig. 5. Total number of demonstrations performed in each experiment.

Fig. 6. Total training time with respect to number of robots.

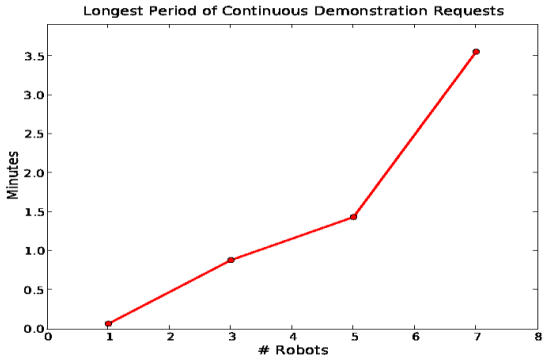


Fig. 7. (a) Attention demand on the teacher.

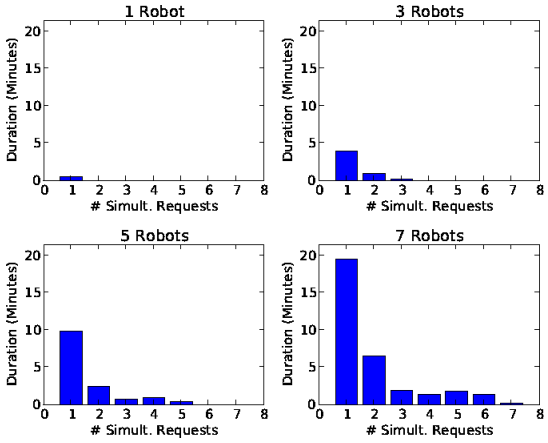


Fig. 8. Graphs showing the distribution of the number of simultaneous requests for each experiment.

demonstrations required per robot. This possibly surprising increase is due to the fact that, although the number of state features in the representation of our domain does not change, the *range* of possible feature values does. Specifically, in an N-robot experiment, the value of features representing the number of robots located at a beacon, b_{nr}, have the range [0,N]. As a result, extra demonstrations are required in the presence of a greater number of robots to provide guidance in the additional states. While similar effects are present in many domain representations, state features can often be designed or modified in such a way that their range is independent of factors such as the number of robots. For example, in the beacon homing domain this could be achieved by converting b_{nr} to a boolean feature that indicates whether the beacon's capacity has been reached or not.

Figure 5 shows how the total number of demonstrations required for each experiment changes with respect to the number of robots. The rate of growth is nearly linear, with seven robots requiring nearly 300 total demonstrations to learn the task. The overall number of demonstrations that must be performed has a significant effect on the overall training time, as discussed in the next section.

C. Training Time

Figure 6 presents the change in the overall experiment training time with respect to the number of robots. The data shows a strongly linear trend, with seven robots requiring just over 1.5 hours to train. This result is significant as it suggests that this approach will continue to scale to even larger tasks.

D. Attention Demand on the Teacher

In addition to the overall training time and number of demonstrations, it is important to understand the demands that multiple robots place on the teacher. The teacher experiences the greatest number of demonstration requests during the earliest stages of learning, possibly from multiple robots at the same time. To evaluate the demand on the teacher's attention during this most laborious training segment, we calculate the longest continuous period of time during which the teacher has at least one demonstration request pending. This value provides insight into the degree of mental effort that is required from the teacher.

Figure 7 plots the duration of the longest continuous period of demonstration requests for each experiment. The data shows that the length of this time period grows quickly, possibly exponentially, with the number of robots. In experiments with only a single robot, demonstration requests last only a few seconds at a time; as soon as the teacher responds to the request, the robot switches to performing the demonstrated action. As the number of robots increases, however, so does the number of simultaneous requests from multiple robots. In the 7-robot experiment, this results in a

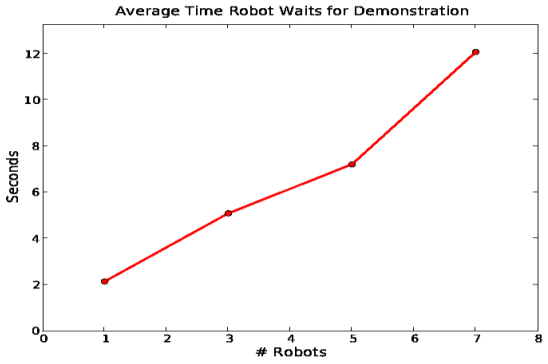


Fig. 9. Average amount of time a robot spends waiting for a demonstration response from the teacher.

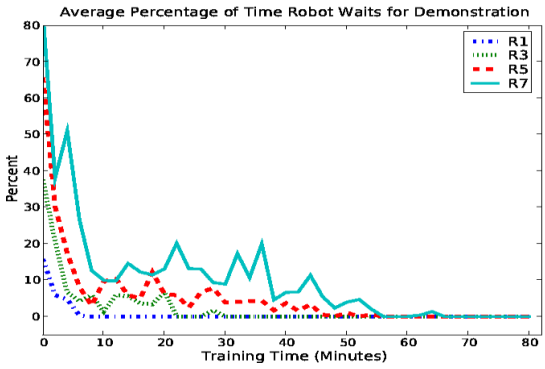


Fig. 10. Average percentage of time a robot spends waiting for a demonstration over the course of training.

3.5 minute uninterrupted segment of demonstration requests for the teacher.

Additionally, we examine the total time per experiment that multiple demonstration requests are pending. Figure 8 presents a set of graphs showing the distribution of the number of simultaneous requests for each experiment. This data indicates that for all experiments, the greatest percentage of time is spent with only a single demonstration request. However, the teacher spends over 3 minutes in the 5-robot experiment, and over 13 minutes in the 7-robot experiment, faced with multiple queries. This growing number of simultaneous queries has a significant impact on demonstration delay, the amount of time that passes between the robot's initial request and the teacher's response.

E. Demonstration Delay

As discussed in the previous section, simultaneous demonstration requests from multiple robots become common as the number of robots increases. As a result, robots are often required to wait while the teacher responds to other robots. Figure 9 shows that the average time a robot spends waiting for a demonstration grows with respect to the number of learners from only 2 seconds for a single robot to 12 seconds for seven robots. Figure 10 plots the percentage of time a robot spends waiting on average for a demonstration over the course of training. Not surprisingly, we observe that the demonstration delay is greatest early in the training process when the teacher is most busy with initial demonstration requests. A promising direction for future work is to examine the possibility of staggering the times at which novice robots are introduced to the task in order to reduce the demand of the initial training phase on the teacher.

F. Evaluation Summary

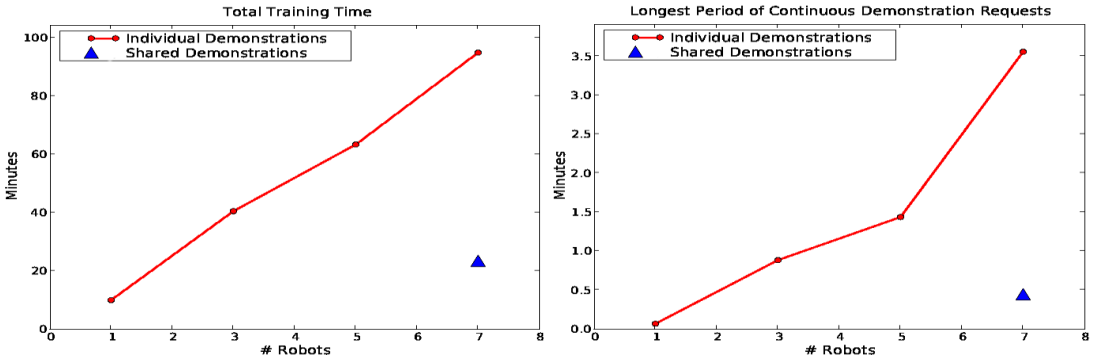
In summary, our findings show promising trends for the scalability of the presented multi-robot demonstration learning approach. Particularly significant is that the total training time grows linearly with the number of robots, allowing learning to scale easily to larger tasks. Somewhat unsurprisingly, we also found that increasing the number of robots also significantly increases the workload of the teacher, as measured by the number of pending demonstration requests. In our evaluation, we show that this in turn impacts demonstration delay and robots must spend more time waiting for the teacher's response. While this has no negative impact on learning in the presented domain, delay may impact performance in other tasks.

Further studies are required before making broad conclusions about the scalability of the presented approach. In particular, more research is needed to determine what impact state representation, action duration, and degree of collaboration between robots have on learning performance and scalability.

However, based on the presented case study we find that no absolute upper bound exists on the number of robots that can be taught at the same time. The maximum number of robots used in the experiments, seven, represents our own limitation in terms of time and the number of available robots, not a limitation of the algorithm. Furthermore, insights gained in this evaluation can be used as a guide for the development of future applications for CBA learning. For example, our knowledge of the trend in overall training time requirements can be used to limit the number of robots in other applications for which the availability of an expert teacher is limited to some fixed time. Similarly, the number of robots in other domains may be affected by the amount of time a robot may remain idle while waiting for a demonstration.

VI. Common Policy Learning

In the standard formulation of the CBA learning algorithm, analyzed above, the teacher provides each robot with an individual set of demonstrations from which a unique policy is derived. This generalized approach is highly suitable for domains in which robots perform different roles and functions. However, in some domains, as in our experiments, the same behavior may be desirable for each robot. In such cases, teaching the same policy to multiple robots results in a large number of redundant demonstrations. To address this case, we propose that all robots learning the same task learn a single, *common*, policy by consolidating all demonstration data. The sharing of information can occur by collecting all data within a single dataset, or by freely



(a) (b)

Fig. 11. Comparison of the single common policy and multiple individual policy CBA learning approaches. (a) Total training time. (b) Attention demand on the teacher.

exchanging each demonstration among all robots so as to maintain a distributed set of identical policies. In this section, we evaluate the performance of the common policy approach using the 7-robot beacon homing domain.

Our study found that using the common policy technique, the teacher was required to perform a total of only 44 demonstrations, compared to nearly 300 total demonstrations previously required for this task. Figure 11(a) shows the overall training time of the experiment, which was similarly reduced to only 23 minutes, compared to the 95 minutes for the standard approach. In fact, while seven robots learning a common policy require more of the teacher's time than a single learner, they require less time than three robots learning individual policies.

Figure 11(b) shows that a common policy similarly reduces the attention demand that seven robots require of the teacher. This effect can be attributed to the fact that frequently a demonstration performed for one robot addresses the queries of other currently waiting robots. An additional effect of this occurrence is that the average waiting time of the common policy approach is reduced from 12 seconds to 1.2 seconds. This evaluation clearly shows the benefits of the common policy approach over distributed policy learning in cases where a common policy is desired.

Note that performing common policy learning in multi-robot domains with independent robots, results in the same policy as when training a single robot alone. Using multiple robots in this case may speed up learning, however, since uncommon states are more likely to be encountered with many learners. In the case of multi-robot domains with non-independent robots, as in the case of the beacon homing domain, common policy learning differs from training and replicating a single robot policy as it additionally allows the robots to learn the collaborative aspects of the task.

VII. Conclusion

In this paper, we presented the first known application of demonstration learning to more than two robots, enabling a single person to train up to seven robots at the same time. We contributed an evaluation of the Confidence-Based Autonomy multi-robot demonstration learning algorithm and evaluated the scalability of this approach with regard to the number of demonstrations required to learn the task, the demands for time and attention placed on the teacher, and the delay that each robot experiences in obtaining a demonstration. The results of our case study indicate that no strict upper bound exists on the number of robots due to limitations of the algorithm. Instead, knowledge gained from this evaluation can be used to guide the design of real-world applications for CBA learning in which real-world constraints on teacher time and robot performance must be taken into account. Additionally, we contributed analysis of a special case of CBA learning in which a common policy is learned by all robots by sharing demonstrations. The presented work serves as a stepping stone for further research, opening the door to many promising directions for multi-robot demonstration learning reseach.

References

[1] P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *International Conference on Machine learning*, New York, NY, USA, 2004. ACM Press.

[2] D. C. Bentivegna. *Learning from Observation Using Primitives*. PhD thesis, College of Computing, Georgia Institute of Technology, Atlanta, GA, July 2004.

[3] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2007. Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm.

[4] S. Chernova and M. Veloso. Interactive policy learning through confidence-based autonomy. *Journal of Artificial Intelligence Research (to appear)*.

[5] S. Chernova and M. Veloso. Multi-thresholded approach to demonstration selection for interactive robot learning. In *Proceedings of 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI'08)*, March 2008.

[6] S. Chernova and M. Veloso. Teaching multi-robot coordination using demonstration of communication and state sharing (short paper). In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AMMAS '08)*, May 2008.

[7] D. Grollman and O. Jenkins. Dogged learning for robots. In *IEEE International Conference on Robotics and Automation*, pages 2483–2488, 2007.

[8] T. Inamura, M. Inaba, and H. Inoue. Acquisition of probabilistic behavior decision model based on the interactive teaching method. In *Ninth International Conference on Advanced Robotics (ICAR)*, pages 523–528, 1999.

[9] A. Lockerd and C. Breazeal. Tutelage and socially guided robot learning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004.