DEALING WITH CHANGING CONTEXTS IN MYOELECTRIC CONTROL

Anna J. Koop[1], Alexandra Kearney[1,2], Michael Bowling[1], Patrick M. Pilarski[2]

[1] *Department of Computing Science, University of Alberta, Edmonton, AB, Canada*
[2] *Division of Physical Medicine and Rehabilitation, University of Alberta, Edmonton, AB, Canada*

ABSTRACT

Myoelectric prostheses approximate the motion and flexibility of biological limbs, especially when compared to their mechanical counter-parts. Machine learning enhances the functionality of these devices; however, in an ever-changing environment, the effectiveness of conventional approaches is impeded. We introduce Partition Tree Learning—a method for learning predictions in an ongoing fashion whilst being able to identify and adapt to new contexts automatically. We compare the performance of PTL to that of a stochastic gradient descent learner on a stream of data from a participant actuating a myoelectrically controlled robot arm. In a consistent context both learners' predictions are comparable. After a context switch, PTL is able to adapt to the change and outperform the gradient descent learner. These preliminary results indicate that PTL may effectively deal with change in real-world prosthetic use, lending its ability to learn over varying situations to the constantly changing environment of powered prosthetics.

INTRODUCTION

For ongoing, every-day use, it would be ideal for a prosthetic device to adapt to the unavoidable changes in the user and environment [1-4]. These changes take many forms: muscle fatigue degrading the control signal; shifts in the position of the residual limb while performing a single pattern of movement; the changing task profiles as a user moves from driving a car to putting away groceries. These changes, operating at different semantic and temporal scales, create challenges for control-related machine learning techniques that have been developed mainly for stationary environments—where the signals are coming from the same distribution and the learner aims for a universal best-fit solution [5]. In particular, changing circumstances are known to be problematic for pattern recognition in myoelectric control [1-3]. A solution that is learned a priori may be robust in general but unable to adapt to the particular circumstances of the moment. On the other hand, an ongoing learner can adapt as needed but may not provide the stability necessary for both user acceptance and effective control [2, 4, 6, 7].

Recent work has focused on using techniques to adapt across several of the many contexts that arise in myoelectric control. Sensinger et al. compared supervised and unsupervised adaptive approaches to improve pattern recognition for a single user across multiple sessions, where previous work had looked only at single-session performance [1]. Tommasi et al. were concerned with the transition from stable pre-trained models to customization for a particular user, and used adaptive combinations of the pre-trained models to reduce training time [4]. Prior work by our group has explored the use of real-time machine learning to adapt and improve both prediction and control policies during ongoing use by a single user [8-10]. Together these approaches to adaptive learning form a solid basis to approach problems inherent in building myoeletric control schemes for specific users and their varied patterns of use.

In this paper we are concerned with the changing contexts that arise from task switches during persistent use of an assistive device. Specifically, we present initial findings on one method by which a system learns automatically during ongoing multi-context use. Our meta-learning approach, termed Partition Tree Learning (PTL), is able to adapt to changing contexts without requiring pre-processing or explicit context identification. PTL therefore promises to complement existing learning methods and further expand the adaptability, robustness, and functionality of myoelectric human-machine interfaces.

PARTITION TREE LEARNING

Many machine-learning algorithms are developed primarily for use in stationary environments: expecting that either the task of interest does not change, or the state representation (that is, the features the learner uses to predict) is detailed enough that every context is uniquely identified. This is a necessary simplification that is sometimes sufficient, however, the many contexts a prosthetic user naturally encounters are diverse and too complex to represent in a single computationally efficient predictor. Thus, we are interested in studying online or continual learning systems where the learner is able to adapt to the specific context. Moreover, we aim to create a learning system that adapts to and identifies changing contexts automatically.

To that end, we are developing Partition-Tree Learning (PTL)—a meta-learning algorithm. Through PTL we can adapt existing learning algorithms to improve their performance in non-stationary environments, such as those encountered in myoelectric control. PTL increases the base learner's accuracy without drastically increasing their computational complexity. This is a direct extension of the Partition-Tree Weighting algorithm for probabilistic modeling by Veness et al. [11], which provides theoretical guarantees on the performance for minimal computational and memory costs.

A key aspect of PTL is that it coordinates learners across different time scales. These learners are designed to converge on the best prediction over the long term and are unable to adapt to local context. PTL allows these stationary learners to be used effectively in non-stationary environments by coordinating their predictions and limiting the data over which they learn.

PTL uses a binary partition tree (visualized in Fig. 1) to split the data into discrete binary segments. Each node of the tree represents a distinct learner over a specific segment. The root learner, sitting at the top of the tree, operates on all the data (up to 2^d time-steps): it behaves identically to the base learner. The leaf nodes operate over a single time-step, and therefore their predictions are mainly determined by the initial settings of the learning algorithm. On the levels in between, each node at depth i operates on 2^d time-steps.

Because PTL operates online it never has to store or compute the entire tree at once. Instead, it keeps a list of d learners, and up to d statistics summarizing the error for each completed subtree. At each time step PTL updates each of the d learners: from the learner at the leaf node monitoring only the current time-step, through all intermediate learners with their longer segments, ending with the root node. If this is the first time-step of a new partition for any depth (as is always the case at the leaf nodes), it will create a new learner at that node and update its own records for the newly completed subtree.

When making a prediction, PTL consults each of the d currently active learners and reports the weighted combination of their predictions. Each prediction is weighted according to a prior that considers long-term learners more likely than short-term, the statistics for the relevant completed subtrees, and the performance of the particular learner over its particular segment. This allows PTL to adjust automatically to changes in the environment: when the environment changes such that the shorter-term learners predict better than the long-term learner, the weight shifts to favour predictions from the short-term learner. When the environment is stable, the weight is mainly on the long-term learner and PTL makes predictions accordingly.

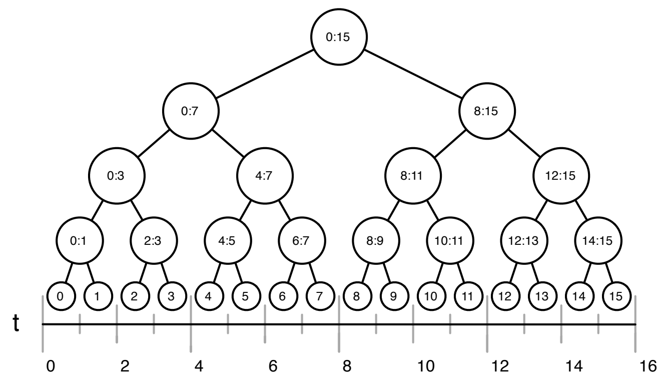

Figure 1: A full binary partition tree, showing all the binary intervals over the time 0...16.

METHODS

Interactive data was gathered from multiple able-bodied subjects—participants without amputations. We used a myoelectrically controlled robot arm which replicated the functionality of a commercial prosthetic device. Informed subject consent was acquired as per ethics approval by the University of Alberta Health Research Ethics Board. The experiment was composed of a simplified conventional prosthetic training task: a square movement. Each subject moved in the square pattern for 8 minutes.

For the duration of the trial the subjects moved the arm in a square pattern, where the learner predicted the joint activity of the robot's arm. If the elbow joint was in motion on any given time-step, then the value of the joint activity would be one, when stationary, the value would be zero.

For comparing the learners we used a 2048-timestep-length segment from the middle of the trial. To create a clear domain switch, we presented this time series twice to the learning agents: first where the signal to be predicted was the sum of the joint activation signal over a horizon of 50 time-steps, and second where the target prediction was the sum of the negation (so that if the joint was moving the one time-step signal would be -1, otherwise zero).

The base learner was a simple stochastic gradient-descent learner that used tile coding over the trace signals on the activation of the two joints together with the position data on each joint. These settings are similar to those used in Pilarski et al. [8–10], but rather than using a discounted sum of future signals we used a fixed-horizon sum for the gradient descent learner. The weights were initialized to 0 and the learning rate α was set to 0.05, which was the best on the square task.

PTL used the same learner and parameters, with a complete reset at the segment boundaries: each new learner was re-instantiated with its weights starting at 0. The performance of the base learner was by measuring the cumulative prediction error. We repeated trials across several different users and the results were consistent: one is singled out for discussion in detail.

RESULTS & DISCUSSION

The cumulative error is shown in Figure 3. Before the task change, the two algorithms made near-identical predictions for all parameter settings tested. After around 300 time steps the rate of error accumulation has levelled off. This can also be seen in the profile of the predictions, shown in Figure 2, where both lines overlap and closely track the target before the switch. After the switch point, they both predict as before but PTL has lower peaks, more quickly compensating for the change. The long-term benefit from the predictions immediately following the switch can be seen in the cumulative error, where there is clear separation between the cumulative errors.

At the switch point, there is a sharp penalty visible in both the cumulative error graph and the prediction graph. The error rate re-stabilizes but takes slightly longer for the gradient descent learner than the initial learning phase, approximately 500 time-steps compared to 300. After this stabilization period the predictions of both algorithms again overlap.

The weight visualization in Figure 2 provides insight into how PTL handles the context switch. Before the switch, PTL places the most weight on the long-term learners, shown. There is a deviation from this around the 1024 time-step mark. At that point, and more consistently after the switch, the weight is distributed across more learners. The medium-term learners are given weights equal to the long-term learner. These shorter learners, not being misled by previous experience in the pre-switch domain, are able to adapt faster to the new signal. As the task continues without another switch, the long-term learner eventually catches up, and the weight again shifts to favour the long-term learner. Introducing another or more frequent switches will shift the weight more towards the short-term learners.

In this preliminary work, we used artificially imposed switch points to understand how the PTL algorithm behaves during contextual shifts. As part of our ongoing work, we are investigating the performance of PTL in a variety of tasks where switching boundaries were naturally occurring in the data stream.

CONCLUSION

In this work we introduced Partition Tree Learning—a method for learning predictive information during ongoing myoelectric control. This approach helps to maintain consistency while still providing the flexibility to adapt to changes in the user and their situation. Our results suggest that PTL is a beneficial a way to learn and adapt during long term, contextually varying prosthesis use. PTL is capable of adapting to changing situations without requiring explicit contextual identification. As PTL is a meta-learning approach, it is also complementary to many existing control optimization techniques that function both prior to and during the control of a myoelectric device. PTL therefore promises a new approach to enhancing the versatility and utility of future myoelectric devices.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge support from the Alberta Innovates Centre for Machine Learning (AICML), Alberta Innovates – Technology Futures (AITF), and Compute Canada.

REFERENCES

[1] J. W. Sensinger, B. A. Lock, and T. A. Kuiken, “Adaptive pattern recognition of myoelectric signals: exploration of conceptual framework and practical algorithms,” *IEEE Trans. Neur. Syst. Rehab. Eng.,* vol. 17, pp. 270-278, 2009.

[2] C. Castellini, P. Artemiadis, M. Wininger, A. Ajoudani, M. Alimusaj, A. Bicchi, et al., “Proceedings of the first workshop on Peripheral Machine Interfaces: Going beyond traditional surface electromyography,” submitted *to Frontiers in Neurorobotics*, 2014.

[3] E. Scheme and K. B. Englehart, “Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use,” *The Journal of Rehabilitation Research and Development*, vol. 48, no. 6, pp. 643–660, 2011.

[4] T. Tommasi, F. Orabona, C. Castellini, and B. Caputo, “Improving control of dexterous hand prostheses using adaptive learning,” *IEEE Transactions on Robotics*, vol. 29, iss. 1, pp. 207–219, 2013.

[5] R. S. Sutton, A. Koop, and D. Silver, “On the role of tracking in stationary environments,” in *Proc. 24th International Conference on Machine Learning*, Corvallis, OR, USA, 2007.

[6] C. W. Antuvan, M. Ison, and P. Artemiadis, “Embedded human control of robots using myoelectric interfaces,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, in press.

[7] K. Ganguly and J. M. Carmena, “Emergence of a stable cortical map for neuroprosthetic control,” *PLoS Biology,* vol.7, iss. 7, pp. e1000153, 2009.

[8] P. M. Pilarski, M. R. Dawson, T. Degris, J. P. Carey, K. M. Chan, J. S. Hebert, and R. S. Sutton, "Adaptive artificial limbs: a real-time approach to prediction and anticipation," *IEEE Robotics & Automation Magazine*, vol. 20, pp. 53–64, 2013.

[9] P. M. Pilarski, M. R. Dawson, T. Degris, J. P. Carey, and R. S. Sutton, "Dynamic switching and real-time machine learning for improved human control of assistive biomedical robots," in *Proc. 4th IEEE RAS & EMBS Int. Conf. Biomedical Robotics and Biomechatronics (BioRob)*, Roma, Italy, pp. 296–302, 2012.

[10] P. M. Pilarski, T. B. Dick and R. S. Sutton, "Real-time prediction learning for the simultaneous actuation of multiple prosthetic joints," in *Proc. of the 2013 IEEE International Conference on Rehabilitation Robotics (ICORR)*, Seattle, USA, pp. 1–8, 2013.

[11] J. Veness, M. White, M. Bowling, A. György, “Partition tree weighting,” in *Proc. DCC*, pp. 321-330, 2013.

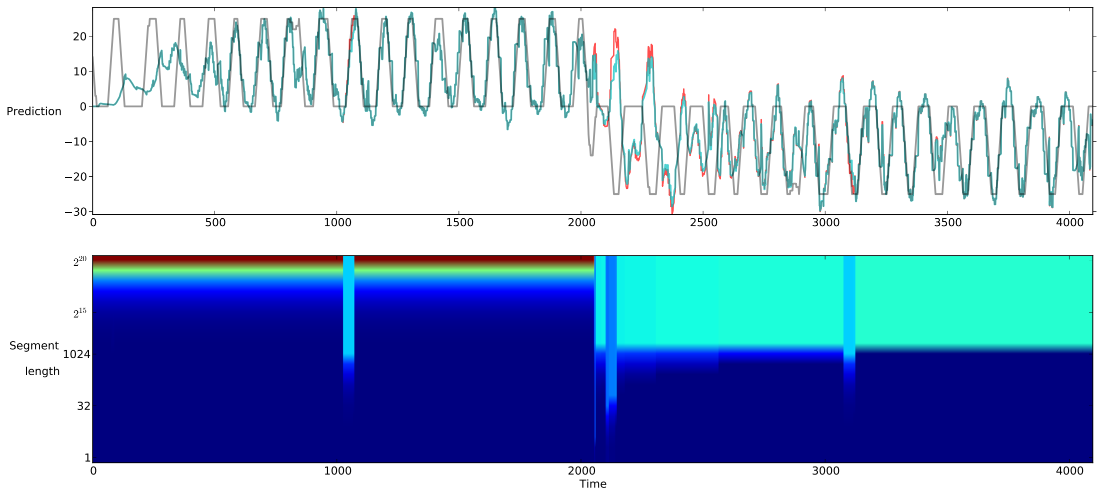

Figure 2: Predictions made by the gradient descent learner (red) and PTL (blue) compared to the true target (grey). The lower figure visualizes the weight on each of the segment lengths over time.

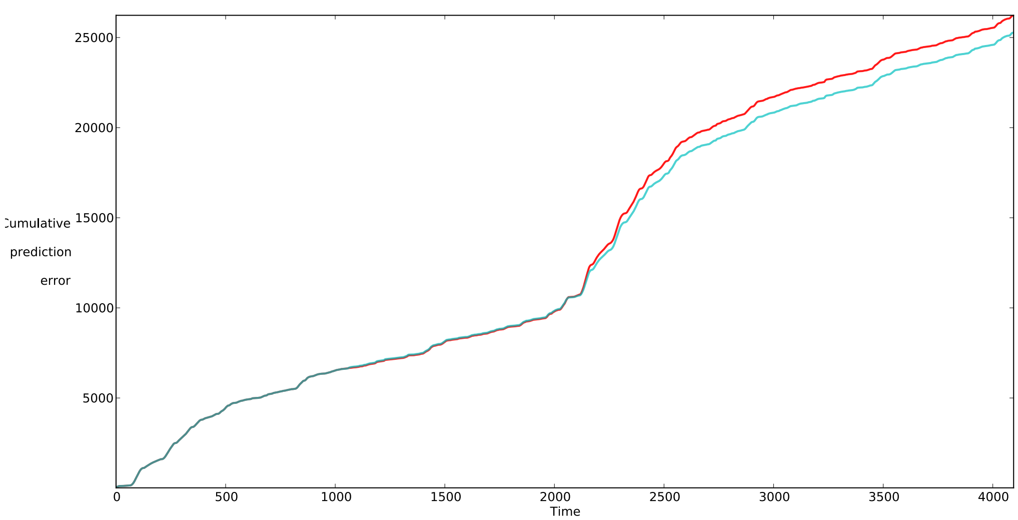

Figure 3: Cumulative error for the gradient descent learner (red) and PTL (blue) over time.