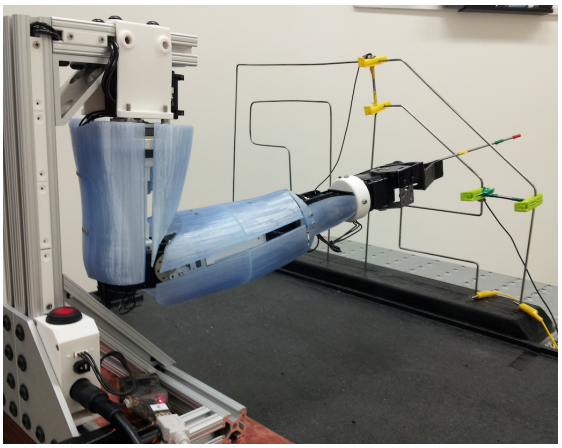


Fig. 2. The Bento Arm [12], shown configured for the Angle Maze experiment with a conductive rod attached to the gripper.

However, it is still not clear what form of partial autonomy would be preferred by amputees, and in fact it may be different for each person and change depending on the situation.

In this work we explore how machine learning of predictions may help manage the integration of user and automatic control. Research suggests that prediction is a key component in movement planning and that anticipatory action plays a role in producing coordinated movements, or synergies [7]. With this in mind, recent work by our group explored several methods for producing synergies using anticipatory movements of unattended joints in a prosthetic arm based on predictions made about a target angle [8]. In Pilarski et al. (2013), a user controlled the elbow and gripper of a three-DOC arm, while automation controlled wrist rotation with the goal of moving the wrist joint to an anticipated target angle based on the current state. Put differently, a prediction about the target angle, some time in the future, was used to direct the movement of the wrist joint. This target angle was programmatically provided to the system in some way. To make predictions about the target wrist angle, Pilarski et al. used a generalization of the reward-based value functions used in reinforcement learning (RL), known as general value functions (GVFs), which are capable of learning multi-step predictions about any measurable signal [9]. GVFs incrementally learn predictions in an online setting, with linear computation. One particular control policy explored by Pilarski et al. was called *Direct Predictive Control* (DPC), which directly maps predictions about target angles into control commands [8]. It was shown to be effective in quickly learning a good control policy given a provided target angle.

Our first contribution in the present work is the description and evaluation of an extension of this method, which we term *Direct Predictive Collaborative Control* (DPCC). In order to compare the performance of our method against TPC we adopt the commonly held perspective that it is better to complete a manipulation task faster, and that manual interactions with the system (toggles) places a burden on the user. We show that DPCC achieves movement synergies and improves user performance by reducing task time and the number of toggles. As a second contribution, we demonstrate, that unlike previous work, it is possible to learn target angles without the need to programmatically provide these target angles to the system. Rather, they can be learned directly by observing the user's behavior.

One of the goals of the present work was to develop technology that could be readily translatable to clinical application. As such, our work respects constraints inherent in conventional myoelectric control under what is arguably the most limited case—i.e., where the user can only produce a scalar signal and a toggle signal, as might be the case with a transhumeral amputee. Additionally, we sought to work within the input and output constraints of conventional prosthetic hardware. As such, we were interested in ways to improve control without adding extra user interface channels or additional sensor modalities like those described in [16].

II. Direct Predictive Collaborative Control

In what follows, we use the term collaborative control to mean two or more agents working together towards a common goal, with only one agent acting on any one DOC at a time. Our proposed approach, Direct Predictive Collaborative Control (DPCC), is a method where a user can choose to attend to any controllable function, but is limited to attending to only a subset at a given time. Unattended DOC are then controlled by automation (Fig. 1). As noted, DPCC extends the DPC method of Pilarski et al. [8]. We employ DPCC in prosthetic control with the user controlling one joint at a time while automation controls the other joints. Predictions about future joint angles are used to generate velocity commands for a given joint according to Eq. 1.

$$V_{t+1} = (P_{t+1}^{(\tau)} - \theta_{t+1}) \cdot r \cdot k \tag{1}$$

Here the difference between the current position, θ_{t+1} and the predicted position, P_{t+1} (looking τ timesteps in expectation into the future), is used with the update rate, r, given in Hz, to calculate the velocity needed to achieve that position in one timestep. This value is then scaled by $0 < k < 1$. Unattended joints only move when the user is moving an attended joint and after each joint toggle the system reverts to manual conttrol for 0.5 s during which unattended joints are held still. For clarity, note that a single update is equivalent to one timestep, i.e., for an update rate of 30 Hz there are 30 timesteps per second.

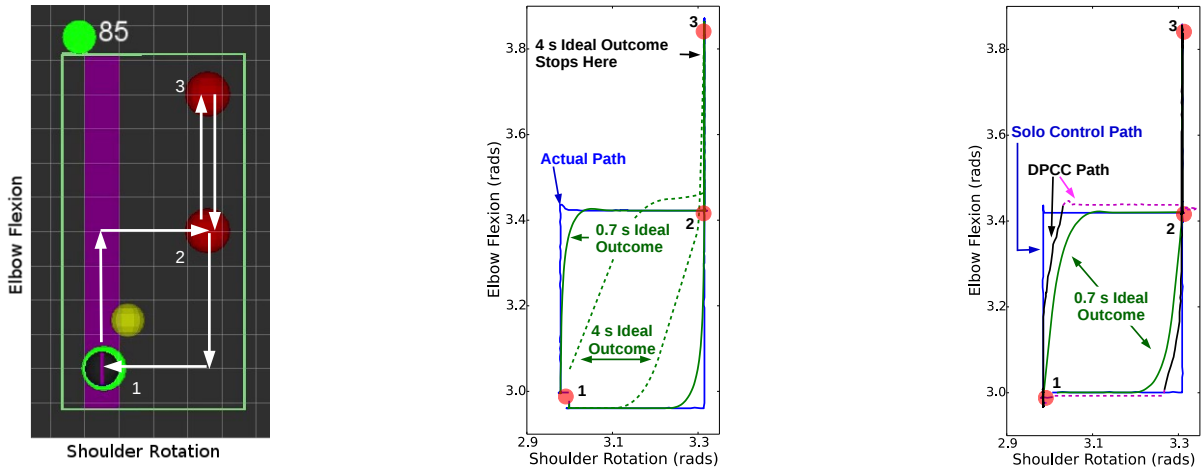
Temporally extended predictions of joint angles (θ) are made using GVFs that are learned using the True Online TD(λ) algorithm [10], as defined by three update equations below (shown for a single joint θ).

$$\delta_t = R_{t+1} + \gamma \mathbf{w}_t^\mathsf{T} \boldsymbol{\phi}_{t+1} - \mathbf{w}_{t-1}^\mathsf{T} \boldsymbol{\phi}_t \tag{2}$$

$$\mathbf{e}_t = \gamma\lambda \mathbf{e}_{t-1} + \alpha_t \boldsymbol{\phi}_t - \alpha_t \gamma\lambda [\mathbf{e}_{t-1}^\mathsf{T} \boldsymbol{\phi}_t] \boldsymbol{\phi}_t \tag{3}$$

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \delta_t \mathbf{e}_t + \alpha_t [\mathbf{w}_{t-1}^\mathsf{T} \boldsymbol{\phi}_t - \mathbf{w}_t^\mathsf{T} \boldsymbol{\phi}_t] \boldsymbol{\phi}_t \tag{4}$$

This algorithm learns a weight vector, $\mathbf{w}$, that is used to make the prediction P_{t+1} about the future of the pseudo-reward signal R, where the ideal prediction is the scaled



(a) Waypoint visualization. (b) Ideal outcomes at various timescales. (c) DPCC performance.

Fig. 3. a) Visualization of the waypoint navigation task. Joint bounds are shown by the green box. The current joint angles are indicated by the black circle, while the yellow circle indicates the current prediction. Red circles indicate waypoints and when the black circle is within a waypoint it turns bright green indicating the joints are within the specified tolerance. The purple strip indicates the direction the user can move. Circuit number is indicated at the top along with a state indicator. The path taken in this task (white arrows) starts at 1, proceeds up and right to 2, then up to 3, down to 2 and finally down and left to 1. b) The path of a single training circuit (blue), and predicted ideal outcomes for 20 timesteps, or 0.7 s (solid green) and 120 timesteps or 4 s (dashed green). c) Direct Predictive Collaborative Control. Compare a circuit made in training (blue) and its ideal outcome (green) against the path taken during a DPCC circuit (dashed magenta and solid black). Dashed magenta and black indicate user control of the shoulder and elbow, respectively.

sum of future rewards $\sum_{i=1}^{\infty} \gamma^{i-1} R_{t+i}$. The psuedo-reward signal is defined by $R_{t+1} = (1-\gamma)\theta_{t+1}$, where the $(1-\gamma)$ factor is used to select the instantaneous angle at the pseudo-termination of the prediction. The γ term specifies how much weighting is given to future rewards and, in expectation, is related to the number of timesteps used for prediction by $\gamma = 1 - \frac{1}{\text{timesteps}}$; $\gamma = 1$ looks to infinity and $\gamma = 0$ looks one timestep. At each timestep a feature vector, $\boldsymbol{\phi}$, is used to calculate the temporal-difference error in Eq. (2). The trace vector in Eq. (3) assigns credit to various features, with a decay factor λ specifying how far into the past to assign credit ($\lambda = 1$ is fully Monte Carlo, $\lambda = 0$ is full bootstrapping), and α specifying a per-timestep learning rate. Finally, Eq. (4) updates our weight vector, $\mathbf{w}$. Predictions, P_{t+1}, are made for a given joint, θ, in expectation, looking τ timesteps in the future, using an inner product $P_{t+1}^{(\tau)} = \mathbf{w_{t+1}}^{\top} \boldsymbol{\phi_{t+1}}$. GVFs were initialized to predict the minimum angle values for each joint ($\mathbf{w}[:] = \text{min\_angle}/\text{num\_active\_features}$), and a fixed learning rate was used, $\alpha = 0.3/\text{num\_active\_features}$, and $\lambda = 0.95$. Each predictor used the same fixed-length binary feature representation, $\boldsymbol{\phi}$, as input, consisting of a single bias unit and a 4-dimensional tile-coding [11] of shoulder angle (θ_S), elbow angle (θ_E), and decaying traces of the same, with a decay rate of 0.99. Angles were normalized over the effective joint range. To balance generalization and accuracy, 100 coarse tilings with width 1 were hashed to a memory size of 2048, for a total feature vector size of 2049 with 101 active features per step.

Each experiment consisted of a TPC training phase (i.e., no automated joint control) followed by DPCC, with learning kept on during all phases to allow for continued adaptation and user correction. Experiments were conducted by a non-amputee subject using the Bento Arm (Fig. 2), a non-compliant robot arm developed in our lab [12]. For these experiments, only shoulder rotation and elbow flexion were used, with all other joints held rigid via motor commands. The words "shoulder" and "elbow" were audibly played to inform the user about the outcome of their toggling action. Joint angles were constrained to limit joint action to a range covering the effective workspace of each experiment by a small margin. The Bento Arm was desk mounted and fixed in place. Control software for the Bento Arm and all experiments ran on the Robot Operating System. Sensory updates, motor commands and real-time learning of the GVFs were all performed at 30 Hz. It is important to note that while the current experiments were limited to only 2 DOC, the methods employed are theoretically applicable to any number of DOC.

III. Experiment 1: Navigating Waypoints

The purpose of our first experiment was to investigate the behavior of DPCC during ongoing human-robot interaction. Here the user operated a joystick to move the robot arm through a series of waypoints, while also observing a visualization, as shown in Fig. 3a, which translated the joint space of the shoulder and elbow into horizontal and vertical components. Waypoints were indicated by red markers. When joints were within 0.0175 rads (1 degree) of a waypoint center, the marker would turn green, a sound was played and the user attempted to hold the joints within the marker until a second sound played three seconds later. The circuit started at the lower-left waypoint (1), moved up and then right to the midpoint (2), up to the top-right waypoint (3), down to the midpoint (2), and finally down and then left to the start point (1). These waypoints are symbolic for places in joint space where the amputee would complete another task before moving on, such as grasping or releasing an object. In typical operation of a prosthesis, an amputee may not have any

conscious recognition of these waypoints and they may change over time. Joystick control mimicked the signals produced by an EMG-based TPC system, i.e., a single joystick provided a scalar signal [-1,1], and a button press allowed the user to toggle between joints.

Fig. 3b shows a sample circuit made during training alongside ideal outcomes. Ideal outcomes are the ideal path we would expect to follow when using GVF predictions as control actions when looking ahead at a specific timescale. In this case, the ideal outcomes are defined to be the ideal predictions made by the GVFs (i.e., the computed temporally extended predictions for the observed data). Outcomes for 0.667 s or 20 timesteps ($\gamma = 1 - \frac{1}{20}$) are shown (solid green), as well as outcomes for 4 s or 120 timesteps ($\gamma = 1 - \frac{1}{120}$, dashed green). We see that these predictions produce a rounding of the upper-left and lower-right corners of the circuit. It is this rounding that we exploit in order to preemptively activate joints and achieve desired joint angles. As would be expected, the rounding of the corners for 4 s predictions is much more pronounced, and it looks past the 3 s pause at the waypoints.

Training lasted 29 circuits (≈11 min), followed by 50 circuits of DPCC. Predictions of 20 timesteps were used, with a scaling factor k of $0.5/\mathsf{num\_timesteps} = 0.025$. Fig. 3c compares the path taken in the final DPCC circuit against a typical circuit taken from the training set. We can see that the DPCC followed a similar path to the ideal outcome for the training circuit, albeit not as pronounced. Figure 4 shows the temporal behavior of the final DPCC circuit. In the shaded regions of this example, the shoulder was moved by the system while the user controlled the elbow, as highlighted by the change in the dark blue line in each of the two black circles. In the first circle we clearly see that the shoulder joint angle is increasing as the elbow joint (light blue) increases. In the second circle we see that shoulder joint angle is decreasing as the elbow joint decreases.

These results demonstrate that our approach was able to learn target angles for each joint without explicitly providing them to the system. Simple, but potentially beneficial, joint synergies were learned in real time purely by observing ongoing user behavior, and effected by a direct mapping of predictions to control commands for the unattended joint.

IV. Experiment 2: Navigating an Angle Maze

As one example of the need for simultaneous multi-joint prosthetic motion, amputees operating one joint at a time must use compensatory body motions to create diagonal (off-joint-axis) movements. We expect diagonal movements and related motions should be straightforward to perform using DPCC. We conducted a limited test of this hypothesis by navigating an angled portion of a wire maze with the robot arm. The user controlled the arm using proportional EMG signals (TPC) with a metal rod attached to the gripper as shown in Fig. 2. A green barrier marked the start of a circuit and a yellow barrier marked the turn-around point. Contact with each barrier was detected by electrical connection between the rod and a 2-cm-long exposed metal region at the center of each barrier. A circuit consisted of contacting the green barrier, moving to the yellow barrier, then returning to the green barrier. The system

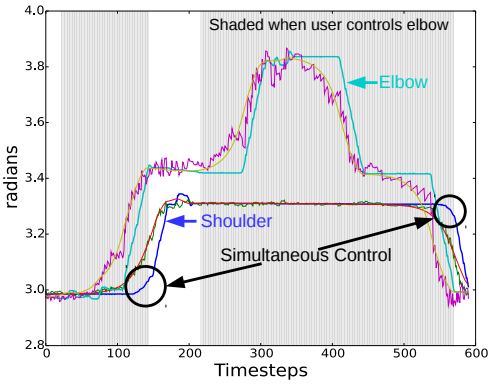


Fig. 4. Simultaneous control of multiple joints though a single control channel is made possible by our proposed automation. Angles of the joints are shown with respect to time. For each joint the angle at each step is indicated along with the ideal outcome for a 0.667 s prediction and the prediction given by our system. Shoulder: angle (blue), ideal outcome (red), actual prediction (green). Elbow: angle (cyan), ideal outcome (yellow), actual prediction (purple). Shading indicates the user was actively controlling the elbow. Areas of simultaneous joint activation are shown inside the black circles.

played a sound when either barrier was contacted. The user would then hold position on the exposed portion until a second sound was played 5 s later. The user was to avoid contacting the walls, but was not penalized for doing so. The proportional signal was generated using EMG recorded from two sites on the user's forearms (each thresholded and normalized) at 200 Hz, calculating the mean absolute value of those signals over a 10-sample sliding window and then taking the difference. The toggle signal was produced using a third EMG signal on the user's other forearm. User controlled joint speed was limited to 0.2 rad/s for ease of control, while no limitation was placed on those controlled by automation. This task was designed to reflect real-life precision movement tasks in a constrained environment, and was inspired by a similar challenge in an upcoming competition for parathletes (Cybathlon 2016).

TPC training lasted for 30 circuits (≈16 min), followed by 53 circuits of DPCC. As expected, when the user controlled the arm alone, the path was noticeably stepped as shown in Fig. 5 (blue). However, with DPCC based on predictions made for 20 timesteps, or 0.667 s ($\gamma = 1 - \frac{1}{20}$) in the future, the achieved trajectory was considerably smoothed due to the learned, simultaneous joint actuation (dashed magenta indicates user was controlling the shoulder, and solid black indicates user was controlling the elbow). A value of 0.1 was used for k. This figure overlays the 46th circuit of DPCC, a particularly good circuit, where we see the path of the rod move along at an angle without the stepped behavior characteristic of TPC. One point of contact with the walls occurs on the return portion of the circuit near the lower bend in the maze near the yellow barrier. The user corrected by switching to elbow, moving up and then switching back to shoulder and moving the rest of the way to the right. While angled movement was evident in most DPCC circuits, not all had so few collisions with the bounds. To accommodate the resulting slight shifts to the maze during operation, the lines

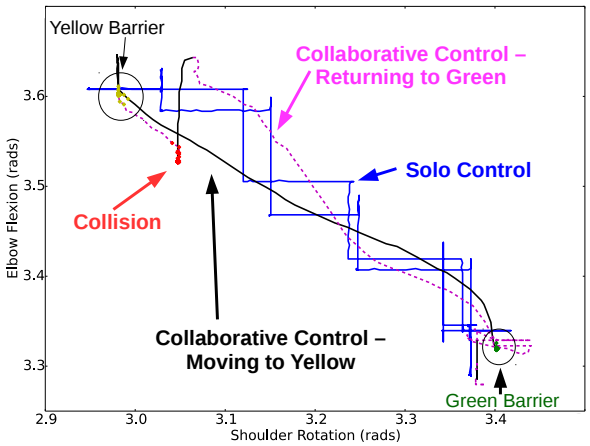


Fig. 5. Key result: Comparison of solo and collaborative control through the angle maze. While not perfect, we see clearly that DPCC enables the user to achieve an angled trajectory not possible for a user on their own.

in Fig. 5 denoting the DPCC circuit were registered to the initial circuit for accurate visual comparison.

Table I compares performance between TPC and DPCC phases. Ideally a circuit could be completed with no joint toggles in 12.8 s. The fastest time achieved by DPCC was 14.9 s with a single toggle. DPCC performance showed significant improvement over performance during the training phase, with average circuit time falling by 19% from 32.3 s to 26.3 s, and average toggle counts falling nearly 50% from 15.1 to 7.83. Improvement was not due to human learning; the user had prior experience with the system and task.

TABLE I. RESULTS FOR THE ANGLE MAZE EXPERIMENT

Control	Toggle Count			Task Time (s)		
	AVG	STDEV	Min	AVG	STDEV	Fastest
Training (TPC)	15.1	3.85	10	32.3	8.13	21.9
DPCC	7.83	4.07	1	26.3	6.14	14.9

V. Discussion and Future Work

Several studies have shown users are willing to accept a degree of automation. As examples, one study examined the use of intelligent wheelchairs [5] and another examined able-bodied use of prosthetic hands [6] in a shared control system. However, it is not clear to what degree actual amputees will accept automatic control of prosthetic movements. Amputees have an intimate relationship with their artificial limbs; arm motions not felt to be self-initiated by the user may be difficult to accept, or may undermine the illusion of ownership created by a user with respect to their prosthesis. It is important that these interactions between automation and ownership be investigated and that potential methods be compared in terms of effectiveness and amputee acceptance.

A. Reinforcement and Learning

Assuming that some form of blended autonomy is beneficial, there are many ways that one might approach multi-joint coordination during the use of a robotic arm. The DPCC approach is a fairly straightforward one—given predictions about where a joint angle will be in the future we simply move toward that angle. One potential benefit of DPCC, as compared to others we might imagine, is that the user operates the arm in the same way regardless of whether or not they receive automation assistance. Further, by keeping learning on during all phases of operation the system is able to continually update predictions based on user behavior and adapt to user correction. It effectively demonstrates that *use can be its own form of reinforcement*. However, it should be noted that the present approach has the potential to reinforce both good and bad behavior, as has been observed in biological learning [13].

In terms of limitations, DPCC makes the assumption that it is beneficial to move towards where the system predicts the user will be, with predictions being made at a single level of temporal abstraction (i.e., one time scale). We can easily identify situations where a single time scale is not the best basis for control. A chosen time scale may be a strong choice for some aspect of a task, as in the diagonal portion of the angle maze experiment. However, looking a fixed distance into the future can also lead to collisions with objects from cutting a corner, or bypassing important locations in space, as was demonstrated in the 4 s predictions in our first experiment (see Fig.3b). One potential solution is to choose the predictive distance based on some aspect of the system's state, allowing us to look anywhere from immediate to remote predictions. This state-dependent behavior seems desirable in general. However, it is non-trivial to select or learn the correct temporal distance to use in each state.

While typical RL methods involve the more complex process of learning a policy based on a reward signal, we have first explored what can be achieved with the simpler direct method described here. In our scheme, system and user behavior reinforce the predictions and behavior without the use of a reward signal. A reward signal would open up additional options for control improvement, one of which might be a mechanism for learning how far in the future to look at any given instant. Reinforcement might be used to select from a list of predictors at different timescales, to blend predictions from multiple timescales, or to select a state-dependent γ value for a single GVF. One reward signal already present in the system is the toggle event. A reinforcement learning algorithm could seek to minimize the number of toggle signals required from the user to complete a task. This is an interesting area for future work.

B. Confidence

A measure of confidence in the system's abilities may improve behavior in previously unseen situations. Confidence might consist of several measures including: accuracy of recent predictions [14], how often the system has seen the current state before, accuracy of past predictions when in the current state, predictive convergence [15], risk level of the current situation, and a measure of the user's confidence in the system.

First, confidence could remove the need for the user to explicitly turn DPCC on and off. If the system gave a high enough confidence value then it could gradually engage in DPCC; if confidence drops the system could return to TPC. Second, being able to automatically move in and out of

DPCC based on confidence measures could further improve the system when it encounters previously unseen states. While the trajectories shown by the ideal prediction lines in Fig. 3b appear straightforward for us to follow, in reality, once the system begins to follow these trajectories it moves the arm out of state spaces it has seen before and its predictions suffer as a result. Some of this can be accounted for by choosing a representation that generalizes well, which is the reason why many very coarse tilings were used in these experiments. However, this is not always sufficient. When using confidence measures, the system could disengage from DPCC when it encounters a new state and simply watch the user in order to build up its predictive certainty until its confidence is restored.

C. Time and State-Space

The results achieved here should be reproducible on any commercial prostheses that provides joint angle feedback. This is possible because, unlike many alternative approaches, GVFs allow us to effectively use time as a signal. The temporal nature of GVF predictions enables automation to recognize and leverage patterns of usage, adapt to changing conditions, and capture how joints are temporally related to one another. It seems appropriate to design control systems that do not waste the user's time. While our results do not depend on modifying existing hardware, improvements should be possible by increasing the amount of information available to the machine learner from the user, from the system, and from the environment. Our methods are ideally suited to efficient implementation in the face of increasing state-space size, being linear in computation and memory (something of great importance for learning on devices intended for wearable operation).

VI. Conclusion

We have described and tested a new collaborative control approach, denoted Direct Predictive Collaborative Control, whereby a user, limited to controlling a single joint at a time, can effect multi-joint synergies in conjunction with an intelligent prosthetic control system. The control system used here *learns predictions directly from user action*, without the need for predefined target angles to be specified, and maps these predictions directly into command signals. Unlike many other approaches, the system's *predictions can be learned in real time* during ongoing, uninterrupted use of the device. Our preliminary results from a single able-bodied participant on a simple angle-maze experiment demonstrate that our methods enable coordinated multi-joint movements and are able to improve task performance by reducing the number of switches and the time needed to complete the task. While our approach was demonstrated for the control of a prosthetic arm, the same approach should be applicable to many domains involving assistive devices where the numbers of functions exceed a user's ability to attend to them. To our knowledge, the present study is also the first demonstration of the combined use of the True Online TD(λ) with general value functions for online control. While larger user studies are needed to confirm the improvements seen here, these results support the continued exploration of our approach.

Acknowledgment

The authors would like to thank Harm van Seijen and Richard Sutton for a number of helpful discussions and critical feedback, and are extremely grateful for support from the Alberta Innovates Centre for Machine Learning (AICML), Alberta Innovates – Technology Futures (AITF), and the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] B. Peerdeman, D. Boere, H. Witteveen, R. Huis in 't Veld, H. Hermens, S. Stramigioli, H. Rietman, P. Veltink, and S. Misra, "Myoelectric forearm prostheses: State of the art from a user-centered perspective," *JRRD*, vol. 48, no. 6, p. 719–738, 2011.

[2] E. Scheme and K. Englehart, "Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use," *JRRD*, vol. 48, no. 6, p. 643–659, 2011.

[3] L. Resnik, M. R. Meucci, S. Lieberman-Klinger, C. Fantini, D. L. Kelty, R. Disla, and N. Sasson, "Advanced upper limb prosthetic devices: Implications for upper limb prosthetic rehabilitation," *Arch. Phys. Med. Rehab.*, vol. 93, no. 4, pp. 710–717, 2012.

[4] P. Parker, K. Englehart, and B. Hudgins, "Myoelectric signal processing for control of powered limb prostheses," *Journal of Electromyography and Kinesiology*, vol. 16, no. 6, pp. 541–548, 2006.

[5] P. Viswanathan, J. L. Bell, R. H. Wang, B. Adhikari, A. K. Mackworth, A. Mihailidis, W. C. Miller, and I. M. Mitchell, "A wizard-of-oz intelligent wheelchair study with cognitively-impaired older adults: Attitudes toward user control," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems Workshop on Assistive Robotics for Individuals with Disabilities: HRI Issues and Beyond*, 2014.

[6] C. Cipriani, F. Zaccone, S. Micera, and M. C. Carrozza, "On the shared control of an EMG-controlled prosthetic hand: Analysis of user-prosthesis interaction," *IEEE Trans. Robotics*, vol. 24, no. 1, pp. 170–184, 2008.

[7] D. M. Wolpert, Z. Ghahramani, and J. R. Flanagan, "Perspectives and problems in motor learning," *Trends in Cognitive Sciences*, vol. 5, no. 11, pp. 487–494, 2001.

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

[9] R. S. Sutton, J. Modayil, M. Delp, T. Degris, P. M. Pilarski, A. White, and D. Precup, "Horde: A scalable real-time architecture for learning knowledge from unsupervised sensorimotor interaction," *Proc. 10th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*, Taipei, Taiwan, 2011, pp. 761–768.

[10] H. van Seijen and R. S. Sutton, "True Online TD (lambda)," in *Proc. 31st Int. Conf. Machine Learning (ICML)*, Beijing, China, 2014, pp. 692–700.

[11] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction.* Cambridge, MA: MIT Press, 1998.

[12] M. R. Dawson, C. Sherstan, J. P. Carey, J. S. Hebert, and P. M. Pilarski, "Development of the Bento Arm: An improved robotic arm for myoelectric training and research," in *Proc. of MEC'14: Myoelectric Controls Symposium*, Fredericton, NB, 2014, pp. 60–64.

[13] S. H. Jang, "Motor function-related maladaptive plasticity in stroke: A review," *NeuroRehabilitation*, vol. 32, no. 2, pp. 311–316, 2013.

[14] A. White, J. Modayil, and R. S. Sutton, "Surprise and curiosity for big data robotics," in *AAAI Workshop on Sequential Decision Making with Big Data*, Quebec City, QC, 2014, pp. 19–23.

[15] M. White and A. White, "Interval estimation for reinforcement-learning algorithms in continuous-state domains," in *Advances in Neural Information Processing Systems*, 2010, pp. 2433–2441.

[16] D. Novak and R. Riener, "A survey of sensor fusion methods in wearable robotics," *Robotics and Autonomous Systems*, in press, 2014, DOI: 10.1016/j.robot.2014.08.012