A Step Towards Autonomy in Robotics via Reservoir Computing

E. Antonelo, X. Dutoit, B. Schrauwen, D. Stroobandt, H. Van Brussel, M. Nuttin

Autonomous mobile robots form an important research area due to their applicability in the real world as domestic service robots. The autonomy of a robot strongly relies on its ability to extract information from the environment. A robot must also be aware of the current situation for an improved interaction with humans or other robots. However, it is very difficult to achieve robotic autonomy in a simple way. Robot tasks such as event detection, robot localization, plan execution, intent recognition are examples of complex tasks that can not be easily solved by standard robotic approaches. These tasks have to be performed using a limited number of sensors with low accuracy, as well as with a restricted amount of computational power.

This work uses the recently emerged paradigm of Reservoir Computing (RC) [1] for grasping information in several contexts of autonomous robotic tasks. RC is a new concept for efficient handling of recurrent networks. With RC, the states of a random and high-dimensional dynamical system made of a reservoir of recurrent nodes are mapped onto the desired output via a simple linear readout. Only the readout output layer is trained using standard linear regression techniques which are simple and fast to implement (the recurrent part or reservoir is left fixed).

This work presents recent and ongoing research carried jointly by two research groups on several robotic tasks. By using RC, we can solve the tasks without any model of the environment neither of the task itself. We consider distinct levels of complexity for the robotic tasks: event detection, localization, plan recognition and intent detection [2]. Event detection consists of detecting simple occurrences local in time and space. It is not a trivial task: for instance, the events of 'passing through door 1' and 'passing through door 2' can seem identical for a robot. It is very important that we are able to distinguish between very similar events. The next step is towards robot localization, in which we rather want to predict the current location of the robot based on the same kind of sensory information. We consider three different granularities for the problem of robot localization: coordinate detection in the cartesian map; location detection in a grid of small discrete areas; location detection in a more realistic environment composed of rooms of distinct sizes. The following robotic task corresponds to plan recognition. Particularly, we are interested in recognizing robot actions during a navigation task such as: the robot is navigating to goal number 1 (which means a possible robot action in a number of distinct goals). Finally, with intent detection we aim at predicting the current intent of a human. It is different from action in that we consider here the case of disabled user who can not always act according to his/her intents. For all four complexity levels, we use RC as a black-box model for learning the task, where the inputs are only distance sensors. In addition, in the context of robot localization, we further extend the experiments to map and path generation. This is achieved by using the RC network in a generative setting, which makes possible to easily extract the maps and the trajectories stored internally by the reservoir.

This work aims at achieving increasing degrees of robotic autonomy by using a simple and efficient technique, namely, RC. We show that we can efficiently solve all aforementioned robotic tasks with RC. Finally, we believe that RC can be applied to a wide range of robotic tasks, enhancing the robot's autonomy, its interaction with (disabled) humans or other cooperating robots.

References

[1] Verstraeten, D., Schrauwen, B., D'Haene, M., Stroobandt, D.: A unifying comparison of reservoir computing methods. Neural Networks **20** (2007) 391–403

[2] Dutoit, X., Antonelo, E., Schrauwen, B., Stroobandt, D., Van Brussel, H., Nuttin, M.: Towards Robotic Awareness using Reservoir Computing. (submitted).

Plan Recognition and Execution with Reservoir Computing

X. Dutoit, H. Van Brussel, M. Nuttin

Autonomous service robots are becoming a major research area in the field of robotics. In this area, Human-Robot interaction (HRI) is of prime importance. The quality of the HRI relies on the ability of the robot to implicitly understand the user's needs and its ability to assist her/him to perform these needs.

To achieve a better HRI, we consider the relatively recent technique of Reservoir Computing (RC), and more specifically of Echo State Networks (ESNs) [2], to perform plan recognition and provide assistance to a human user. The key idea of RC is very similar to a kernel methods, as it consists of projecting the input data into a high-dimensional space, the *reservoir*, acting as a collection of dynamical functions. The desired output can then be extracted from the reservoir by a simple linear readout.

We restrict ourselves in the present work to the context of 2-D navigation. The task is two-folds: first, an ESN is trained to perform plan recognition; then, it is trained to perform plan execution.

For the recognition part, the goal is to estimate the desired position of a user in a maze. We show that an ESN can correctly predict towards which corner a mobile platform driven by the user is currently going with a high probability [1].

For the execution part, the ESN is used to assist the driving and steer the platform towards the desired position. Here again, the ESN can reach the desired goal with a relatively good performance when considering the complexity of the task.

Both results show that RC is able to perform complex tasks based only on distance sensors data and with a simple example-based training method. An intelligent system based on RC could thus be used to estimate the desired action and then assist its execution. This would make possible to have a user-friendly and efficient robotic assistant for everyday tasks.

References

[1] X. Dutoit, E. Antonelo, B. Schrauwen, D. Stroobandt, H. Van Brussel, and M. Nuttin. Towards Robotic Awareness Using Reservoir Computing. Submitted.

[2] H. Jaeger. The "echo state" approach to analysing and training recurrent neural networks. Technical report, 2001.

Reinforcement Learning with Multiple Demonstrations

Adam Coates, Pieter Abbeel, Andrew Y. Ng
Department of Computer Science
Stanford University

Many tasks in robotics can be described as a trajectory that the robot should follow. Unfortunately, specifying the desired trajectory is often a non-trivial task. For example, when asked to describe the trajectory that a helicopter should follow to perform an aerobatic flip, one would have to not only (a) specify a complete trajectory in state space that intuitively corresponds to the aerobatic flip task, but also (b) ensure that the state space trajectory is consistent with the helicopter's dynamics. This is a non-trivial task for systems with complicated dynamics.

In the apprenticeship learning setting, where an expert is available, one can instead have the expert demonstrate the desired trajectory. Unfortunately, this means that we must have an essentially optimal expert available—since any learned controller, at best, will only be able to repeat the demonstrated trajectory. Such a perfect demonstration may be hard, if not impossible, to acquire. However, even suboptimal expert demonstrations often embody many of the desired qualities. Even stronger, repeated expert demonstrations are often suboptimal in different ways, suggesting that a large number of suboptimal expert demonstrations could implicitly encode the optimal demonstration. In this piece of work we propose an algorithm that approximately extracts this implicitly encoded optimal demonstration from multiple suboptimal expert demonstrations. In doing so, the algorithm learns a target trajectory that not only mimics the behavior of the expert, but can even be significantly better.

The problem of extracting the underlying ideal trajectory from a set of suboptimal trajectories is not a matter of merely averaging the states observed at each time-step. A simple arithmetic average of the states would result in a trajectory that does not obey the dynamic constraints of the model. Also, in practice, each of the demonstrations will occur at different rates so that attempting to combine states from the same time-step in each trajectory will not work properly.

Our algorithm uses a generative model that describes the expert demonstrations as noisy observations of the hidden, optimal target trajectory, with each demonstration possibly occurring at a different rate. An EM algorithm is developed to infer the hidden target trajectory and the necessary model parameters using a Kalman smoother and an efficient dynamic programming algorithm to perform the E-step.

We also show how prior knowledge can be easily incorporated to further improve the quality of the resulting "averaged" trajectory. For example, often only an approximate dynamics model is known, and our algorithm can estimate an improvement to the general dynamics model specific to the trajectory being performed by incorporating data from multiple demonstrations. Our formulation also allows us to take out known expert flaws. For example, for a helicopter performing in-place flips, it is known that the helicopter can be centered around the same position over the entire sequence of flips. Our model incorporates this prior knowledge, and factors out the position drift in the expert demonstrations.

Our experimental results show that the resulting trajectories are not only good, feasible trajectories that can be used in reality, but also that the resulting performance meets or exceeds that of the expert (as evaluated by our expert helicopter pilot). The presented algorithm significantly extends the state of the art in aerobatic helicopter flight ([1], [4]). Specifically, the learned trajectories resulted in significantly better in-place flips and rolls than previously possible. The presented algorithm also resulted in the first autonomous tic-tocs, a maneuver considered even more challenging than flips and

rolls. Movies of the flight results can be found at the Stanford Autonomous Helicopter homepage:
http://www.cs.stanford.edu/groups/helicopter

Related work. In recent work on apprenticeship learning and inverse reinforcement learning ([2], [5], [7], [6]), the reward function is assumed be a linear combination of a known set of features (rather than being defined by a trajectory), and the weighting of the features is then estimated from expert demonstrations. Most similar to our work, Atkeson and Schaal [3] also estimate the desired trajectory (for a pendulum swing-up task) from a demonstration. However, they learn from a single demonstration only, which can significantly limit the performance obtained (or, equivalently, increase the requirements on the expert) as discussed in previous paragraphs.

References

[1] P. Abbeel, A. Coates, M. Quigley, and A. Y. Ng. An application of reinforcement learning to aerobatic helicopter flight. In *NIPS 19*, 2007.

[2] P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *Proc. ICML*, 2004.

[3] Christopher G. Atkeson and Stefan Schaal. Robot learning from demonstration. In *Proc. ICML*. Morgan Kaufmann, 1997.

[4] V. Gavrilets, I. Martinos, B. Mettler, and E. Feron. Control logic for automated aerobatic flight of miniature helicopter. In *AIAA Guidance, Navigation and Control Conference*, 2002.

[5] Gergely Neu and Csaba Szepesvari. Apprenticeship learning using inverse reinforcement learning and gradient methods. In *Proc. UAI*, 2007.

[6] N. Ratliff, J. Bagnell, and M. Zinkevich. Maximum margin planning. In *Proc. ICML*, 2006.

[7] N. Ratliff, D. Bradley, J. Bagnell, and J. Chestnutt. Boosting structured prediction for imitation learning. In *Neural Information Processing Systems 19*, 2007.

CRF-Based Semantic and Metric Maps

Bertrand Douillard Dieter Fox Fabio Ramos

Map building is a fundamental task in robotics. This poster presents a methodology to create metric maps augmented with object information [1]. The algorithm combines 2D laser scans with vision data to produce maps in which each laser point is annotated by the type of object it represents (see Fig.1).

Metric information is obtained by projecting the laser data into the global reference frame. Symbolic information is obtained through probabilistic classification. The underlying model is based on Conditional Random Fields (CRFs) where each scan is represented by a CRF chain model, and each return is a node with categorical values. In this work, each label belongs to one of the following seven classes: "car", "trunk", "foliage", "people", "wall", "grass", "other". The links in the chain network encode spatial correlations between neighbor returns.

CRF chains representing consecutive scans are linked across time in order to represent temporal correlations. Temporal links are instantiated based on the connections found by the Iterative Closest Point algorithm and allow smoothing of classification results over time. The resulting network from the connections of multiple CRF chains is the data structure defining the map. The model allows both real time filtering and smoothing via loopy belief propagation.

The real-time analysis of the network also permits to identify moving objects. Moving objects are characterized by a trace of associated points in the map which change over time (e.g. the blue traces generated by the car in Fig. 1). This trace is identified and replaced by the estimated object trajectory. Non-moving nodes connected by temporal links are merged together to limit the growth in the number of nodes of the map.

The CRF model is trained with the Virtual Evidence Boosting (VEB) algorithm, which enables the incorporation of high-dimensional, continuous features into CRF training [2]. We also show how VEB can be used with partially labeled data, thereby significantly reducing the burden of manual data annotation.

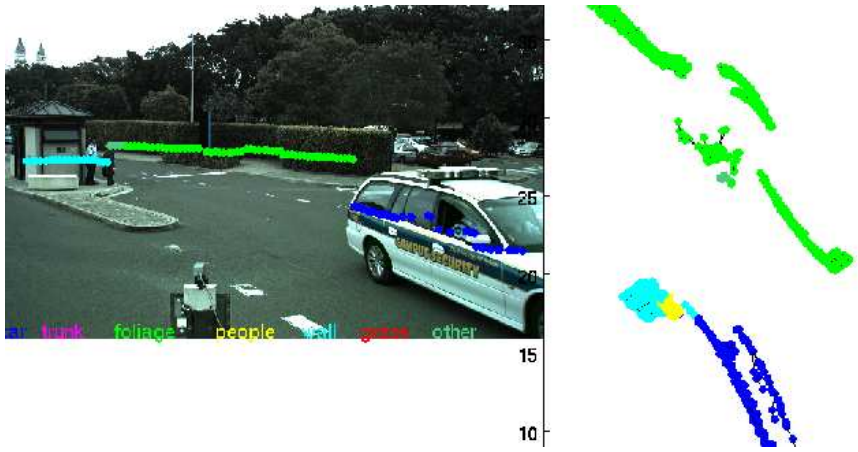


Figure 1: Left: One data frame consisting of a laser scan projected into a camera image. The colors of the laser points indicate the inferred object label. The correspondence between colors and labels is indicated at the bottom of the image. Right: Map generated from multiple scans. Each dot is a node of the underlying CRF network. The dark blue dots correspond to the car moving through the scene.

References

[1] B. Douillard, D. Fox, and F. Ramos. A spatio-temporal probabilistic model for multi-sensor multi-class object recognition. In *Proc. of the International Symposium of Robotics Research (ISRR)*, 2007.

[2] L. Liao, T. Choudhury, D. Fox, and H. Kautz. Training conditional random fields using virtual evidence boosting. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2007.

Learning to Associate with CRF-Matching

Fabio Ramos Dieter Fox

Data association remains a difficult and fundamental problem for many robotics tasks. It is a crucial component in problems such as tracking, image registration, reconstruction, and simultaneous localisation and mapping (SLAM).

Most existing data association algorithms consider only a limited set of features and require substantial manual tuning to work in practice. For instance, in robotics SLAM, data associations are typically determined solely based on the locations of landmarks, thereby ignoring important appearance information. However, incorporating more complex information makes manual tuning extremely cumbersome. Another limitation of existing data association algorithms is the fact that they only provide a deterministic result on the association. This makes them less robust and difficult to incorporate into probabilistic filtering approaches since no uncertainty on the association is returned.

This poster introduces CRF-Matching; a general multi-sensor data association framework that can be learnt from data [5]. CRF-Matching is a supervised probabilistic model able to jointly reason about the association of points. This is obtained by overcoming the independence assumption through the use of Conditional Random Fields (CRFs) [2]. CRFs are an extremely flexible technique for integrating different features in the same probabilistic framework. Features can be defined over different sensor modalities or designed to capture neighbourhood information. The power of CRFs is enhanced through the possibility to use statistical measurements (such as the likelihood of the data given the model) to learn a parametrisation of the model given some training data. This process estimates weights for the features, thus quantifying the importance of each feature for the particular task. Inference can be performed efficiently using Loopy Belief Propagation once the model has been fully specified.

We demonstrate the capabilities of CRF-Matching in two data association tasks: laser scan matching and image feature matching. In the first case data association between two laser scans is accomplished by converting the individual measurements of one laser scan into hidden nodes of a CRF. The states of each node range over all measurements in the other scan. The CRF models arbitrary information about local appearance and shape of the scans. Consistency of the association is achieved by connections between nodes in the CRF. CRF-Matching learns model parameters discriminatively from sets of aligned laser scans. When applied to a new pair of scans, maximum *a posteriori* estimation is used to determine the data associations, which in turn specify the spatial transformation between the scans. Extensive experiments show that CRF-Matching significantly outperforms ICP when matching laser range-scans with large spatial offset. Furthermore, they show that our approach is able to reliably match scans without a priori information about their spatial transformation, and to incorporate visual information so as to further improve matching performance.

An extension of the previous approach can be employed for association of image features. This is obtained through the use of the Delaunay triangulation [4] as the graph structure for CRF-matching. The graph defines neighbour points by respecting interesting geometric constraints such as the *empty circle property*. This creates a graph that is not over-connected while still encoding most of the geometric relationship between neighbour points. We demonstrate how pairwise potential functions can be defined over edges to jointly reason about the associations. In addition to pairwise potential functions, local potential functions can also be defined to directly incorporate sensor observations into the model. In our implementation, SIFT features and descriptors [3] were used, although any image feature descriptor or detector could also be employed. As opposed to the SIFT match procedure described in [3] where Euclidean distance is used to measure the compatibility of matches, we show how a boosting classifier can be learnt and integrated in CRF-Matching to capture non-linear relationships between image descriptors to best match the features. We perform extensive experiments in challenging indoor and outdoor datasets where images were obtained while the robot was in motion. Some of complexities in the datasets include occlusion, different illumination conditions, blurring, translation and rotation transformations. We show that CRF-Matching outperforms the commonly applied RANSAC procedure [1] when there is a small set of detected features.

References

[1] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Comms. Assoc. Comp. Mach.*, 24(6):381–395, 1981.

[2] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of the International Conference on Machine Learning (ICML)*, 2001.

[3] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[4] F. R. Preparata and M. I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, New York, USA, 1985.

[5] F. Ramos, D. Fox, and H. Durrant-Whyte. Crf-matching: Conditional random fields for feature-based scan matching. In *Proc. of Robotics: Science and Systems*, 2007.

Self-Supervised Learning from High-Dimensional Data for Autonomous Offroad Driving

Ayse Naz Erkan[1], Raia Hadsell[1], Pierre Sermanet[1,2], Koray Kavukcuoglu[1], Marc'Aurelio Ranzato[1], Urs Muller[2], Yann LeCun[1]
(1) Courant Institute of Mathematical Science, New York University
(2) Net-Scale Technologies, Morganville, NJ 07751
naz|raia|ranzato|koray|yann@cs.nyu.edu, psermanet|urs@net-scale.com

Designing a vision-based autonomous robot that can navigate through complicated outdoor environments is an extremely challenging problem that is far from solved. However, through use of a self-supervised learning strategy and compact feature representations, we have developed a long-range vision system that succeeds in accurately classifying obstacles and traversable areas that are 60 meters or more distant, bringing us closer to the goal of human-level autonomous driving. The learning system is trained in realtime in rapidly changing terrain, and it must balance the need for stability against the need for plasticity and adaptability. This dilemma is addressed by employing a dynamic ensemble of experts.

Motivation: Shortcomings of Stereo Vision

The existing paradigm for vision-based mobile robots relies on hand-tuned heuristics: a stereo algorithm produces a (x, y, z) point cloud and heuristics assign traversability costs to points based on their proximity to a ground plane. However, stereo algorithms that run in realtime frequently produce traversability costs that are short-range, sparse, and noisy. Our learning strategy uses these stereo labels for online supervision to train a realtime classifier. The classifier then predicts the traversability of all visible areas, from close-range to the horizon. For accurate recognition of ground and obstacle categories, it is best to train on large, discriminative windows from the image, since larger windows give contextual information that is lacking in color and texture features.

Feature Representation

The visual windows are high dimensional (16x11x3 pixels), necessitating a concise and informative representation. This is crucial in order to reduce processing time as well as removing statistical redundancies to enable a consistent learning algorithm over these features. We use an unsupervised two layer autoencoder network trained offline with 150 log files from typical outdoor environments [4]. The network takes inputs from a scale invariant image pyramid [1] [2] and returns 100 dimensional features vectors corresponding to the windows in pyramid bands. These features are provided to the online learning module.

Dynamic Ensemble Learning

Concept drifts in visual data is common in off-road robot navigation. The statistical properties of the visual features shift over time, sometimes dramatically, as when the robot moves from forest to clearing. In cases where the robot is exposed to previously unseen scenarios, online learning is required to provide adaptivity. However, with the real-time processing requirements, it is often impractical to maintain a single high capacity classifier. On the other hand, with a a low capacity learner, it is not possible to capture the characteristics of very high-dimensional and diverse features, and therefore a stability-plasticity tradeoff becomes unavoidable. To overcome this problem, we are using a dynamic online ensemble of localized classifiers combined with a mixture of experts [3] controller. We experiment on various policies to localize experts, to merge their decisions and to detect outliers.

Evaluation

The long-range vision system has been implemented and tested using the LAGR (Learning Applied to Ground Robots) platform. Enabling the classifier allows the planner to avoid dead ends and navigate towards distant paths. The long-range vision runs at 2-3 Hz, so it must be processed in a separate thread from the main control loop, which runs at 10-15 Hz. This multiple-thread architecture allows the vehicle to nimbly avoid close obstacles while using the long-range vision for strategic navigation.

References

[1] A. Erkan, R. Hadsell, P. Sermanet, J. Ben, U. Muller, and Y. LeCun. Adaptive long-range vision in unstructured terrain. In *Proc. of Int'l Conf on Intelligent Robots and Systems (IROS)*, 2007.

[2] R. Hadsell, P. Sermanet, J. Ben, A. Erkan, J. Han, B. Flepp, U. Muller, and Y. LeCun. Online learning for offroad robots: Using spatial label propagation to learn long-range traversability. In *Proc. of Robotics: Science and Systems (RSS)*, 2007.

[3] R. Jacobs, M. Jordan, S.J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. In *Neural Computation*, 1991.

[4] M. Ranzato, Y. Boureau, and Y. LeCun. Sparse feature learning for deep belief networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2007.

(Machine) Learning Robot Control Policies

Daniel H Grollman, Odest Chadwicke Jenkins
{dang,cjenkins}@cs.brown.edu

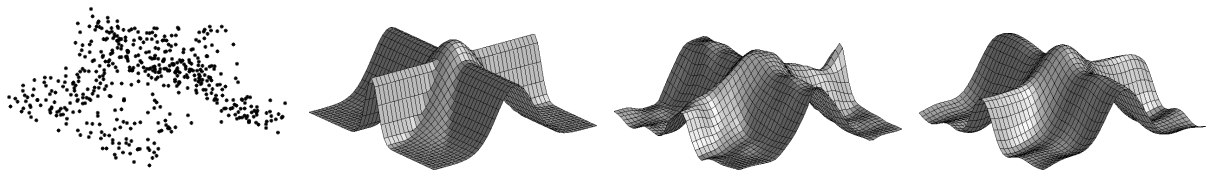
It currently requires years of education and practice before a skilled user can successfully program a sophisticated robot platform to perform a given task. We are exploring ways in which statistical machine learning techniques can enable **Learning from Demonstration**, an approach where users 'reprogram' a robot without writing code. In this scenario, a user demonstrates the desired task and the robot learns to perform the task by observing its performance. We treat this learning as a form of **Policy Transfer**, where the decision making policy latent in the demonstrator is transitioned onto the robot.

Taking perception and motion processes as fixed, we represent each policy as a functional mapping from perceived states to desired actions ($\pi(\hat{s}) \rightarrow a^*$). Using teleoperation, a demonstrator guides the robot through an instance of the desired behavior, creating a set of matched inputs and outputs. Function approximation techniques can then be applied to find an approximation of the control policy ($\hat{\pi}$).

We have left the tasks undefined, as we are interested in how robots can be made to learn **Unknown Tasks**, tasks not predefined during construction and original programming. Robots that exhibit **Lifelong Learning**, learning over extended periods (years) and in multiple domains, will likely need to deal with this issue. We have thus been exploring nonparametric function approximators. In addition, by using an algorithm capable of fast ($\sim$ 30Hz) inference and prediction on our system, we can enable interactive tutelage, where a demonstrator can observe and correct a learned behavior in realtime using **Mixed-Initiative Control**.

In our work so far [2], we have explored two such algorithms: Locally Weighted Projection Regression (LWPR) [4] and Sparse Online Gaussian Processes (SOGP) [1]. Figure 1 compares these two algorithms on a synthetic data set. Our initial robot-based experiments have focused on soccer-related tasks with robot dogs, and have shown successful learning from both hand-coded controllers and human demonstration.

Currently, we assume that the desired mapping (π) is functional, that each input has only one correct output. This is not the case in all contexts, as a robot may be able to perform two or more task-appropriate actions. We are interested in techniques that can learn such non-deterministic mappings directly from input-output pairs, such as mixtures of experts [3]. In addition, by incorporating aspects of reinforcement learning we hope to further our ability to perform task performance refinement and task structure learning.



(a) Noisy Data (b) Ground Truth (c) LWPR, MSE = 0.0202 (d) SOGP, MSE = 0.0244

Figure 1: SOGP and LWPR with default parameters compared on the cross function. We limit SOGP's capacity to the number of receptive fields used by LWPR (22).

References

[1] Lehel Csató and Manfred Opper. Sparse online gaussian processes. *Neural Computation*, 14(3):641–669, 2002.

[2] Daniel H Grollman and Odest Chadwicke Jenkins. Learning robot soccer skills from demonstration. In *6th IEEE Intl. Conference on Development and Learning*, 2007.

[3] Edward Meeds and Simon Osindero. An alternative infinite mixture of gaussian process experts. In *Neural Information Processing Systems (NIPS)*, pages 883–890, 2006.

[4] Sethu Vijayakumar, Aaron D'Souza, and Stefan Schaal. Incremental online learning in high dimensions. *Neural Computation*, 17(12):2602–2634, 2005.

Efficient Sample Reuse by Covariate Shift Adaptation in Value Function Approximation

Department of Computer Science, Tokyo Institute of Technology,
2-12-1, O-okayama, Meguro-ku, Tokyo, 152-8552, Japan

Hirotaka Hachiya hachiya@sg.cs.titech.ac.jp
Takayuki Akiyama akiyama@sg.cs.titech.ac.jp
Masashi Sugiyama sugi@cs.titech.ac.jp

Policy iteration is a general framework to obtain the optimal policy by iteratively performing value function approximation and policy improvement [6]. A traditional practice of policy iteration is, when policies are updated, new data samples are gathered following the new policy and are used for value function approximation. However, this approach is inefficient particularly when the sampling cost is high since previously gathered data samples are simply discarded; it would be more efficient if we could reuse the data collected in the past. A situation where the behavior policy (a policy used for gathering data samples) and the current target policy are different is called *off-policy* reinforcement learning [6].

In the off-policy situation, simply employing a standard policy iteration method (such as *least-squares* policy iteration [2]) does not lead to the optimal policy due to the bias caused by the difference between behavior and target policies. This policy mismatch problem could be eased by the use of *importance sampling* techniques [1]—the bias caused by the policy mismatch is asymptotically canceled. However, the approximation error is not necessarily small when the bias is reduced by importance sampling; the variance of estimators should also be taken into account since the approximation error is the sum of squared bias and variance. Due to large variance, existing importance sampling techniques tend to be unstable [6], [3].

To overcome the instability problem, we propose using an *adaptive importance sampling* technique used in statistics [4]. The proposed adaptive method, which smoothly bridges the ordinary estimator and importance-weighted estimator, allows us to control the trade-off between bias and variance. Thus, given that the adaptive parameter is chosen carefully, the optimal performance can be acheived in terms of both bias and variance. However, the optimal value of the adaptive parameter is heavily dependent on problems, and therefore using a prefixed adaptive parameter may not be always effective in practice.

For optimally choosing the value of the trade-off parameter, we reformulate the value function approximation problem as a supervised regression problem and propose using an automatic model selection method based on a statistical machine learning theory [5]. The method called *importance-weighted cross-validation* enables us to estimate the approximation error of value functions in an unbiased manner even under off-policy situations. Thus we can actively determine the adaptive parameter based on data samples at hand. We demonstrate the usefulness of the proposed approach in standard chain-walk and mountain-car benchmark problems.

REFERENCES

[1] G. S. Fishman. *Monte Carlo: Concepts, Algorithms, and Applications*. Springer-Verlag, Berlin, 1996.
[2] M. G. Lagoudakis and R. Parr. Least-squares policy iteration. *Journal of Machine Learning Research*, (4):1107–1149, 2003.
[3] D. Precup, R. S. Sutton, and S. Singh. Eligibility traces for off-policy policy evaluation. Morgan Kaufmann, 2000.
[4] H. Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.
[5] M. Sugiyama, M. Krauledat, and K.-R. Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8:985–1005, May 2007.
[6] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, 1998.

Improving Gradient Estimation by Incorporating Sensor Data

Gregory Lawrence
Computer Science Division
U.C. Berkeley
gregl@cs.berkeley.edu

Policy search algorithms have been effective in learning good policies in the reinforcement learning setting. Successful applications include learning controllers for helicopter flight [3] and robot locomotion [1]. A key step in many policy search algorithms is estimating the gradient of the objective function with respect to a given policy's parameters. The gradient is typically estimated from a set of policy trials. Each of these trials can be rather expensive, and so we prefer to minimize the total number of trials required to achieve a desired level of performance. During each policy trial, an agent may receive a considerable amount of sensory data from its environment. While the agent's controller may use this information in deciding which actions to take, the sensory data is usually ignored in the gradient estimation task. In this poster we show that by viewing the task of estimating the gradient as a structured probabilistic inference problem, we can improve the learning performance. We argue that in many instances, reasoning about sensory data obtained during policy execution is beneficial. We demonstrate the effectiveness of this approach by showing a reduction in the variance of the gradient estimates for a simulated dart throwing problem and quadruped locomotion task. Our prior work shows one method of exploiting the sensor data in the case of perfect sensing of the control noise [2]. This work removes this assumption by allowing the agent access to a sensor model that only measures the positions of observable joints.

The performance measure f for our dart throwing task is defined as the negative squared distance to the target, and in the quadruped problem it is the distance travelled during a single policy trial. Our policies specify the desired trajectories of each controllable joint, and a PD-controller applies torques in an attempt to follow these paths. Actuator noise is added in the simulation, and the agent observes the actual joint angles for each policy execution. By using an appropriate encoding of the sensor data obtained during each policy trial, unexpected sensor values can be used to explain away the deviations in the observed performance. For example, in the dart throwing problem suppose that the agent executes a policy and notices that the dart missed the target. Normally, an agent would want to change it's desired trajectory so that the next throw moves closer to the target. However, suppose that during the previous throw the agent noticed that it let go of the dart too soon. This helps to explain the miss and allows the agent to better infer which direction it needs to move in policy space to improve its performance.

We consider parameterized policies $\pi \in \mathbb{R}^d$ that encode how an agent chooses its actions given its past observations, and the reinforcement learning goal is to find an optimal policy π^* that maximizes the performance measure f. The gradient is estimated from a collection of policy trials by learning two components. The first component is a linear model between the policy parameters π and a transformation of the sensor data $\phi(s)$ that is almost uncorrelated with the policy parameters. At each time step we predict what the next observation should be as a function of the current observation and controls. This prediction is formed by using estimates, which are obtained in a pre-processing routine, of the current mass matrix, gravity compensation, and inertial terms used in the equations of motion. We project the difference between the observed states and the predicted states down to a low-dimensional subspace using a set of basis functions. The second component is a linear model between the policy parameters π augmented with the sensor data $\phi(s)$ and the observed performance f. If the sensor data correlates with the noise in the performance measure, then this relationship will be easier to learn when compared to a model that ignores the sensor data.

References

[1] N. Kohl and P. Stone. Machine learning for fast quadrupedal locomotion. In *Proceedings of the Nineteenth National Conference on Artificial Intelligence*, 2004.

[2] Gregory Lawrence, Noah Cowan, and Stuart Russell. Efficient gradient estimation for motor control learning. In *Proceedings of the Nineteenth International Conference on Uncertainty in Artificial Intelligence*, 2003.

[3] Andrew Y. Ng, H. Jin Kim, Michael Jordan, and Shankar Sastry. Autonomous helicopter flight via reinforcement learning. In *Advances in Neural Information Processing Systems*, 2003.

Relocatable Action Models for Autonomous Navigation

Bethany R. Leffler **Michael L. Littman**
Rutgers University
Piscataway, NJ USA

A reinforcement-learning agent, in general, uses information from the environment to determine the value of its actions. Once the agent begins acting in the world, there is no further modification of its behavior by humans. This lack of human control makes the use of reinforcement learning a natural solution to the autonomous navigation and exploration problem. However, implementing algorithms from this field has not always been possible in the robotic domain.

Early work in reinforcement learning focused on approximating values of different locations by mapping them to a lower dimension function that generalized across the environment (Sutton, 1988, Tesauro, 1995). More recent work has sought to illuminate foundational issues by proving bounds on the resources needed to learn near optimal behavior (Brafman and Tennenholtz, 2002). Unfortunately, these later papers treat all locations in the world as being completely independent. As a result, learning times tend to scale badly with the size of the state space—experience gathered in one state reveals nothing about other states. The generality of these results makes them too weak for use in real-life problems in robotics and other problem domains.

The work reported in this poster builds on advances that retain the formal guarantees of recent algorithms while moving toward algorithms that generalize across states. These results rely critically on assumptions. The main assumption adopted in our current work (Leffler et al., 2007) is that states belong to a relatively small set of *types* that determine their transition behavior. In a navigation task, states can be grouped by terrain type—in areas of similar terrain, a robot's action model (how its actions affect its state) will be similar. Once correlations are discovered, not all actions have to be explored in every state to fully determine a near optimal policy. Through the use of *relocatable action models*, transition functions learned in one state can be used to calculate the results of actions in similar states. By sharing information between homogeneous states (Leffler et al., 2005), learning time is greatly reduced by limiting the number of actions that need to be performed before the world is well modeled.

Learning efficiency can be improved further by leveraging visual input. If type classification is performed visually, the robot does not have to visit a location to determine its terrain type. Visual segmentation on texture allows us to perform such classifications on terrain images.

With our proposed algorithm, the agent acquires an image of the world and classifies the states into types using texture segregation. The agent then explores all of its actions in a representative sample of states in the environment. Exploiting the assumption of terrain types mentioned above, a generalized action model for the learned states is applied to other states of similar terrain, greatly improving learning time when there are many more states than terrain types.

This work was implemented on a Lego Mindstorm wheeled robot in a 4 foot by 4 foot domain with two ground surfaces. Using our algorithm, the agent was able to obtain action models for each terrain and reach the goal on the first run. After five runs, the agent performed the learned policy continuously. This approach was compared against RMAX, which after 50 runs was not able to reliably reach the goal.

References

Brafman, R. I. and Tennenholtz, M. (2002). R-MAX—a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 3:213–231.

Leffler, B. R., Littman, M. L., and Edmunds, T. (2007). Efficient reinforcement learning with relocatable action models. In *Proceedings of the Twenty-Second National Conference on Artificial Intelligence*.

Leffler, B. R., Littman, M. L., Strehl, A. L., and Walsh, T. J. (2005). Efficient exploration with latent structure. In *Proceedings of Robotics: Science and Systems*.

Sutton, R. S. (1988). Learning to predict by the method of temporal differences. *Machine Learning*, 3(1):9–44.

Tesauro, G. (1995). Temporal difference learning and TD-Gammon. *Communications of the ACM*, pages 58–67.

Machine learning for developmental robotics

Manuel Lopes, Luis Montesano, Francisco Melo
Institute for Systems and Robotics - IST
Lisboa, Portugal
[macl,lmontesano,fmelo]@isr.ist.utl.pt

Extended Abstract

Developmental robotics is the process whereby a robot incrementally acquires more and more complex cognitive skills. This approach draws inspiration from biology to tackle the ultimate goal of robotics, i.e. intelligent robust machines operating in open ended environments. The main idea is to equip the robot with a set of predefined (pre-programmed) skills and then follow learning processes to acquire new ones on top of current knowledge.

Acquiring new skills requires combining many learning process such as unsupervised self-exploration, learning by observation or reinforcement learning. The success of the approach also depends heavily on the models used by the robot to represent and use this knowledge. In this work we discuss which machine learning methods and models are and can be used to create a robot that develops autonomously. We discuss a possible developmental pathway whereby a robot acquires the capability to learn by imitation. It is composed by three levels: 1) sensory-motor coordination; 2) world interaction; and 3) imitation. At each stage the system learns more about its own body and about the world. The newly acquired knowledge enables and facilitates the learning at the next level.

We focus our work on two main problems related to the world interaction and imitation phases respectively: learning the properties and dynamics of objects (affordances) and inferring task descriptions from observations (imitation). Affordances represent the behaviour of objects in terms of the robot's motor and perceptual skills. This type of knowledge plays a crucial role in developmental robotics, since it is at the core of many higher level skills such as imitation. In our work, we propose a general affordance model based on Bayesian networks. This model describes inter-relations between actions, object features and observable effects. The robot learns the structure and parameters of the network by interacting with different objects.

Knowledge of the world in turn enables social interaction. The nature of this third phase is very different from the previous ones. It requires the robot to interact with a teacher which in turn provides supervision or reinforcement. We develop an imitation learning methodology for a humanoid robot that uses the general world model acquired previously to infer the task to be learnt from the teacher's demonstrations. The core of our algorithm is the recently proposed Bayesian inverse reinforcement learning algorithm. The challenges are to reuse a general task independent model and to estimate the appropriate rewards/policies.

The proposed framework gives rise to several important issues and future directions for research. For instance, generalizing robot-object interaction knowledge requires to take into account groups of objects, sequences of actions and delayed effects. Active learning strategies should be implemented to deal with huge search spaces. Another important point is the interaction between the different learning processes, e.g the evolution of actions from pure joint positions or velocities to (possibly parameterized) motion primitives. Finally, it is important to perceive the impact of inaccurate learnt models in subsequent steps. For example, in the imitation stage, errors in the recognition of the demonstration may affect the learning of the demonstrated task.

Acknowledgements

This work was partially supported by Programa Operacional Sociedade do Conhecimento (POS_C) that includes FEDER funds and by by the EU-Project RobotCub.

References

[1] M. Lopes and J. Santos-Victor. A developmental roadmap for learning by imitation in robots. *IEEE Trans. Systems, Man and Cybernetics - B*, 37(2):308–321, 2007.

[2] M. Lopes, F. Melo, and L. Montesano. Affordance-based imitation learning in robots. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2007 (to appear).

[3] L. Montesano, M. Lopes, A. Bernardino, and J. Santos-Victor. Modelling object affordances using bayesian networks. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2007 (to appear; extended version accepted in the IEEE Transactions on Robotics).

Policy gradient approach to multi-robot learning

Francisco Melo
Institute for Systems and Robotics
Instituto Superior Técnico
Lisboa, Portugal
fmelo@isr.ist.utl.pt

Extended Abstract

In theory, the formalism and methods of reinforcement learning (RL) can be applied to address any optimal control task, yielding optimal solutions while requiring very little a priori information on the system itself. However, in practice, RL methods suffer from the "curse of dimensionality" and exhibit limited applicability in complex control problems. Unfortunately, many actual control problems are inherently infinite, described in terms of continuous state variables. This is the case, for example, of optimal control of autonomous vehicles or complex robotic systems. However, the combination of value-based methods (such as Q-learning) and function approximation is far from trivial and the usefulness of the obtained solutions is still not clear. This has, perhaps, motivated the impressive advances in policy-gradient-based methods in recent years [3].

The motivation to extend these methods to multi-robot scenarios is evident. Many tasks found in practice are inherently too complex or even impossible for a single robot to execute. Furthermore, it is often the case that the use of several cheap robots is preferrable to the use of a single complex and expensive robot. On the other hand, the "traditional RL approach" makes use of game theoretic models such as Markov games. These approaches are generally unsuited to address problems envolving real robots, because they rely on several joint-observability assumptions inherent to these models that seldom hold in practice. Finally, more realistic models such as Dec-POMDPs are inherently too complex to be solved exactly.

It is in face of this inherent complexity in addressing complex multi-robot problems that policy gradient methods may prove of use. In this work, we conduct a preliminary study of policy-gradient methods in multi-robot problems. In particular, we analyze how successful policy-based approaches such as WoLF-PHC [1] can be adapted to accomodate the recent developments in policy gradient methods. The setting considered in this work is distinct from other approaches in the literature [2] in that we assume no joint-state or joint-action observability, which renders our approach more adequate to address multi-robot problems (where such assumptions seldom hold). We study how the existence of several independent learners in a common environment effects the overall learning performance of the different agents in several simple multi-robot scenarios and discuss how this approach can be extended to more complex problems.

Acknowledgements

This work was partially supported by Programa Operacional Sociedade do Conhecimento (POS_C) that includes FEDER funds. The author acknowledges the PhD grant SFRH/BD/3074/2000.

References

[1] M. Bowling and M. Veloso. Rational and convergent learning in stochastic games. In *Proc. 17th Int. Joint Conf. Artificial Intelligence*, pages 1021–1026, 2001.

[2] V. Könönen. Policy gradient method for team Markov games. In *Intelligent Data Engineering and Automated Learning*, *LNCS 3177*, pages 733–739, 2004.

[3] J. Peters, S. Vijayakumar, and S. Schaal. Policy gradient methods for robot control. Technical Report CS-03-787, University of Southern California, 2003.

Learned system dynamics for adaptive optimal feedback control

Djordje Mitrovic, Stefan Klanke, and Sethu Vijayakumar
{d.mitrovic, s.klanke, sethu.vijayakumar}@ed.ac.uk
School of Informatics, University of Edinburgh, United Kingdom

Optimal feedback control (OFC) has been proposed as an attractive movement generation strategy in goal reaching tasks for anthropomorphic manipulator systems. In contrast to classic open loop optimizers that produce "just" a minimal-cost trajectory with implicit resolution of kinematic and dynamic redundancies, the OFC framework additionally yields a feedback control law which corrects errors *only* if they adversely affect the task performance (minimum intervention principle).

Locally, the optimal feedback control law for systems with non-linear dynamics and non-quadratic costs can be found by iterative methods, such as the recently introduced iterative Linear Quadratic Gaussian (iLQG) algorithm [1]. So far this framework relied on an analytic form of the system dynamics, which may often be unknown, difficult to estimate for more realistic control systems or may be subject to frequent systematic changes.

We present a novel combination of learning a forward dynamics model within the iLQG framework, for which we employ Locally Weighted Projection Regression (LWPR) [2]. Utilising such adaptive internal models can compensate for complex dynamic perturbations of the controlled system in an online fashion. Moreover, through the availability of analytic derivatives of the learned model, the adaptive iLQG–LD framework we introduce lends itself to a computationally more efficient implementation of the iLQG optimization without sacrificing control accuracy, allowing the method to scale to large DoF systems.

Up to now, we studied iLQG–LD on two different joint torque controlled manipulators as simulated by the Matlab Robotics Toolbox: i) a planar 2 DoF manipulator, which is ideal for performing extensive (quantitative) comparison studies and to test the manipulator under controlled perturbations and force fields during planar motion, and ii) a 6 DoF manipulator with realistic physical parameters. We successfully tested our iLQG–LD framework with both stationary and non-stationary dynamics, simulating constant or velocity-dependent force fields.

Our current work concentrates on implementing the iLQG–LD framework on the 7-DOF robot arm hardware of the German Aerospace Centre (DLR), which raises challenges such as a very high-dimensional input space (7 commands + 14 motor states + 14 joint states) and a high sampling rate of 1kHz. In the future, we will aim for the biomorphic *variable stiffness* based highly redundant actuation system that is currently developed at DLR – this will not only explore an alternative control paradigm, but will also provide the only viable and principled control strategy for such a system.

References

[1] E. Todorov and W. Li. A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems. In *Proc. of the American Control Conference*, 2005.

[2] S. Vijayakumar, A. D'Souza, and S. Schaal. Incremental online learning in high dimensions. *Neural Computation*, 17:2602–2634, 2005.

Learning Robot Low Level Control Primitives: A Case Study

Diego Pardo, Cecilio Angulo, Ricardo Tellez

Technical University of Catalunya

Abstract

Controlling articulated mobile robots is associated with the manipulation of very complex dynamics, leading to a straightforward lack in the kind of motions these robots can complete, i.e. confined to those computed off-line from the very approximated models of its bodies' physics. Trajectory generation based on safety mathematical conditions, e.g. avoiding singularities or non-equilibrium states, limites its performance. It has been demonstrated that, by means of optimization process inside the low level control laws, robots may outperform its physical capabilities [1]. Inverse dynamics of very approximated and complex models are not helpful to design natural motions.

The system presented in this paper uses a modular control architecture, where joint's actuators share information between each other. The policies that gather and distribute the signals to the actuators are learned based on the task performance.

Here we defend the idea of learning low level control primitives to achieve coordination [2], allowing the system to generate trajectories autonomously by using Policy Gradient Reinforcement learning techniques (PGRL), i.e an optimal control framework is established.

In this poster a time-varying dynamics task is used as a test bed : The simulated version of the AIBO® robot completing a basketball-like task by means of PGRL (see process video); control actions and output signals are presented while comparing changes during learning; additionally, different kind of PGRL algorithms are proved and compared from a control systems' perspective.

References

[1] D. Pardo and C. Angulo, *Understanding sensori-motor coordination during a humanoid robot dynamic task*, in Proc. IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2007), London, UK. 2007.

[2] D. Pardo and C. Angulo, *Emerging Behaviors by Learning Joint Coordination in Articulated Mobile Robots* Lecture Notes in Computer Science, vol 4507, pp. 806-813 (IWANN 2007), Donostia, Spain. Springer, 2007.

TORO: Tracking and Observing Robot

Deepak Ramachandran and Rakesh Gupta
Honda Research Institute USA, Inc.
800 California Street, Suite 300,
Mountain View, CA 94041
dramcha@uiuc.edu | rgupta@hra.com

Abstract

A humanoid robot like ASIMO is required to accomplish tasks requiring interaction with people such as object delivery tasks. These tasks require keeping track of past observations of people and objects in the environment. Previous work has used ad-hoc methods to manually encode regions to explore to accomplish these tasks. In this paper, we describe how Region based Particle Filters can be used to maintain and update belief about objects and people location over extended time periods.

We have implemented and demonstrated this work on a Pioneer mobile robot. We first detect people and objects in the scene using depth and vision sensors. We create a map of the environment and use off the shelf techniques for localization and path planning using SICK laser. We describe use of Dynamic Bayesian Network to maintain and update belief about each people and object location. We run separate particle filters for each person and object and for robots' own location. We update these particle filters with information from observations using sampling and re-sampling to incorporate observations over time during the day.

1 Introduction

In this work, we introduce a region-based belief representation of peoples' locations. The idea is to model the person's location by a hierarchical process that first selects a *region* and then conditioned on that a position (X-Y coordinate) for the person. The regions are chosen to be resting places where people people typically stop and stay for long periods of time. For example office desks, the area in front of a TV, water cooler, printer etc. Typically the region is a discrete variable, and position is a linear gaussian conditional on region. This representation allows us to separate transitions that occur at different time scales into the different layers of the model (e.g. deliberate movement from conference room to office desk vs. fidgeting in an office chair). We propose using a Dynamic Bayesian Network (DBN) to track the state of our model over time. Even if the person goes out of view of the robot, it knows that individuals tend to move at a certain maximum speed, that the individuals (for example, someone sitting in their office) is more likely to stay there for a while before moving to a different location.

We demonstrate the use of our belief tracker to carry out simple delivery tasks: Pick up objects from a fixed location and deliver them to the corresponding individual. As a result of using multiple sensors, we have a more complicated observation model which updates measurement weights based on the current robot view and walls in the environment that limit the observation regions. Integrating these various sensors together(depth sensor, vision camera, laser scans, plus robot odometry) presented a significant engineering challenge.

2 Related Work

Past work concentrates on tracking movement of people in the immediate neighborhood of the mobile robot over short time periods. Montemerlo et. al. [2] used a probabilistic algorithm for simultaneously estimating the pose of the mobile robot and positions of people in a previously mapped environment. They used a laser sensor and tracked two people in the vicinity. Schulz et. al. [3] also used a laser sensor to keep track of people in vicinity and handle cases where people are obstructed over short time periods. Bennewitz et. al. [1] track people over long time periods but need to first learn the transitions in the environment.

3 Results

(a) Pioneer Robot in the room

(b) Particle filter with belief for robot and person position

Figure 1: (i) prior belief at starting (ii) belief after person is detected (iii) belief after person is no longer seen (iv) belief before turning around (v) belief after turning around (vi) belief after person is seen again

References

[1] M. Bennewitz, W. Burgard, G. Cielniak, and S. Thrun. Learning motion patterns of people for compliant robot motion. *The International Journal of Robotics Research*, 24(1):31–48, 2005.

[2] M. Montemerlo, S. Thrun, and W. Whittaker. Conditional particle filters for simultaneous mobile robot localization and people-tracking. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, volume 1, pages 695– 701, Washington D.C., May 2002.

[3] D. Schulz, W. Burgard, D. Fox, and A. B. Cremers. People tracking with mobile robots using sample-based joint probabilistic data association filters. *The International Journal of Robotics Research*, 22(2):99–226, 2003.

Active Learning for Robot Control

Philipp Robbel, MIT Media Lab, USA (robbel@mit.edu)
Sethu Vijayakumar, University of Edinburgh, UK (sethu.vijayakumar@ed.ac.uk)
Marc Toussaint, TU Berlin, Germany (mtoussai@cs.tu-berlin.de)

Abstract

The suggested poster summarizes and extends the approach presented in [1]. There, our work focused on learning the inverse dynamics of a robot arm using an efficient exploration strategy. We developed an active learning scheme for the LWPR algorithm [2] to guide data selection to regions of high model uncertainty until a task-specific trajectory can be realized by the manipulator. We position our work as an alternative to manual data collection (such as human guidance to specific points in the task space) and to simpler exploration schemes such as random flailing of the robot arm.

As in Vijayakumar et al's work, we use a compliant composite controller consisting of a learned feedforward model of the inverse dynamics and a low-gain corrective PID element. We address two difficulties with data selection for robot control: first, points cannot be chosen freely from the input distribution (order-sensitive scenario) and second, we would like to learn the inverse dynamics of the system online. Our active learning algorithm trades off between exploitation of the current LWPR model and exploration based on the confidence in the current LWPR predictions. In [1] we derive confidence bounds around the LWPR model and use these during learning as follows:

1. At every time step we determine model prediction and prediction confidence for the current query point $\mathbf{x}_q$. We postulate that the model generalization error is approximated by the size of the confidence intervals.
2. If the confidence is above a threshold, we apply the model prediction as a control signal to all joints and continue with step 1. Otherwise, we define $\mathbf{x}_{q-1}$, i.e., the last point that we trusted our model predictions, as a setpoint.
3. We execute a number of directed exploratory actions around $\mathbf{x}_{q-1}$ to reduce the confidence interval size. In our current implementation those are random control signals followed by "resetting" the arm to the setpoint via high-gain PID control. We then continue with step 1.

By focusing exploration to the subspace where a particular task requires control, data collection of our algorithm is trajectory-specific. In [1] we presented results for a simulated 2-DOF robot arm along an arbitrary trajectory with a bell-shaped velocity profile. The work presented in this poster extends the evaluation to a simulated 6-DOF Puma arm. For both we demonstrate that our algorithm significantly reduces the number of data points (and time) required over random flailing to learn an inverse model that successfully drives back the trajectory. Evaluations are presented as reduction in normalized mean-square error (nMSE) along the trajectory with respect to the number of data points used during learning.

References

[1] Philipp Robbel. Active learning in motor control. Master's thesis, University of Edinburgh, UK, August 2005. Available at http://www.inf.ed.ac.uk/publications/thesis/online/IM050311.pdf.

[2] Sethu Vijayakumar, Aaron D'Souza, and Stefan Schaal. Incremental online learning in high dimensions. *Neural Computation*, 17(12):2602–2634, December 2005.

Learning 3-D Object Orientation from Images

Ashutosh Saxena, Justin Driemeyer and Andrew Y. Ng
Stanford University

We consider the problem of estimating the 3-D orientation of objects from a single image, even in the presence of symmetries of the object. We apply our algorithm both to estimating orientation of new objects from a known object class, and to robotic manipulation, where the task is to grasp (pick up) an object from a previously unknown object class.

Estimating orientation is a fundamental problem in computer vision, but is difficult because (i) The space of orientations is intrinsically non-Euclidean and non-linear, (ii) The presence of symmetries means that orientation is ambiguous, in that there are multiple "correct" orientations.[1] These two properties make orientations difficult to learn using standard learning algorithms.

In this paper, we present a novel representation for orientation that is invariant to symmetries in the object. Our representation applies even to objects that exhibit arbitrary (rotational, reflective, axial, etc.) symmetries in 2-D or 3-D. Using this representation, we further develop novel learning and inference algorithms for estimating orientations of symmetric and asymmetric objects.

We first describe previous methods to learn 3-D orientation, and describe their deficiencies. Specifically, they fail when the range of angles considered does not lie within a small range.[2] In contrast, our method does not assume any restriction on the range of angles considered, and is accurate even in the fully general case.

We extensively evaluate previous methods, and our algorithm, on two tasks: (i) Estimating the 3-D orientation of a new object (drawn from a known object class), and (ii) Choosing the orientation of a robot arm/hand so as to enable it to grasp a new object (where here the previously-unseen object is drawn from a previously-unknown object class). We show that in all cases, our algorithm results in significantly lower error. We have also successfully applied these ideas to enabling our robot to grasp a variety of objects.

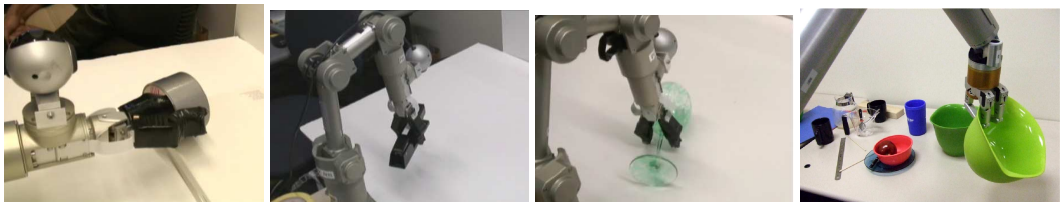


Figure 1: Figure shows the robots grasping a roll of duct tape, a stapler, a wine-glass and a cereal bowl.

References

[1] N. Fisher. *Statistical Analysis of Circular Data*. Cambridge University Press, 1993. 1

[2] P. Mittrapiyanuruk, G. N. DeSouza, and A. C. Kak. Calculating the 3d-pose of rigid objects using active appearance models. In *ICRA*, 2004. 1

Towards Active Learning for Socially Assistive Robots

Adriana Tapus **Maja Matarić**
Department of Computer Science
University of Southern California
Interaction Lab
Los Angeles, CA 90089
adriana.tapus@ieee.org ; mataric@usc.edu

Introduction: The recent trend toward developing a new generation of robots capable of operating in human-centered environments, interacting with people, and participating and helping us in our daily lives, has introduced the need for robotic systems capable of learning to use their embodiment to communicate and to react to their users in a social and engaging way. Social robots that interact with humans have thus become an important focus of robotics research. Nevertheless, Human-Robot Interaction (HRI) for assistive applications is still in its infancy. In this work, the target user population is post-stroke patients. Stroke is the leading cause of serious, long-term disability among adults, with over 750,000 people suffering a new stroke each year in the US alone. Therefore, in this work we investigated the role of robot's active learning in the assistive therapy process. We tried to address the following research question: How should the behavior and encouragement of the therapist robot adapt as a function of the user's personality, profile, preferences, and task performance?

Methodology: Creating robotic systems capable of adapting their behavior to user personality, user preferences, and user profile in order to provide an engaging and motivating customized protocol, is a very difficult task, especially when working with vulnerable user populations. Various learning approaches for human-robot interaction have been proposed in the literature, but none include the user's profile, preferences, and/or personality. To the best of our knowledge, no work has yet tackled the issue of robot personality and behavior adaptation as a function of user personality in the assistive human-robot interaction context. In the work described here, we address those issues and propose a reinforcement-learning-based approach to robot behavior adaptation. In the learning approach, the robot incrementally adapts its behavior and thus its expressed personality as a function of the user's extroversion-introversion level and the amount of performed exercises, attempting to maximize that amount. We formulated the problem as *policy gradient reinforcement learning (PGRL)* and developed a learning algorithm that consists of the following steps: (a) parametrization of the behavior; (b) approximation of the gradient of the reward function in the parameter space; and (c) movement towards a local optimum. The main goal of our robot behavior adaptation system is to enable us to optimize on the fly the three main parameters (interaction distance/proxemics, speed, and verbal and paraverbal cues) that define the behavior (and thus personality) of the therapist robot, so as to adapt it to the user's personality and thus improve the user's task performance. Task performance is measured as the number of exercises performed in a given period of time; the learning system changes the robot's personality, expressed through the robot's behavior, in an attempt to maximize the task performance metric.

Experimental Results: Two experiments were designed to test the adaptability of the robot's behavior to the participant's personality and preferences. The

experimental task was a common object transfer task used in post-stroke rehabilitation and consisted of moving pencils from one bin on the left side of the participant to another bin on his/her right side. The bin on the right was on a scale in order to measure the user's task performance. The system monitored the number of exercises performed by the user. The robot used PGRL algorithm to adapt its behavior to match each participant's preferences in terms of therapy style, interaction distance, and movement speed. The learning algorithm was initialized with parameter values that were in the vicinity of what was thought to be acceptable for both extroverted and introverted individuals, based on our previous experiments [2] and psychology literature. The PGRL algorithm used in our experiments evaluated the performance of each policy over a period of 60 seconds. The reward function, which counted the number of exercises performed by the user in the last 15 seconds, was computed every second and the results over the 60 seconds "steady" period were averaged to provide the final evaluation for each policy. The result is a novel stroke rehabilitation tool that provides individualized and appropriately challenging/nurturing therapy style that measurably improves user task performance.

Discussion: Due to the large number of combinations of parameter values that have to be investigated during the adaptation phase the optimal policy might be obtained only after a period of time that exceeds our session of exercise (i.e., 15 minutes). However, we feel that this does not reduce the efficiency of our approach or the relevance of our results, as our research targets interaction with patients for an extended period of time and where many therapy sessions are required for complete rehabilitation. Thus, if the optimal policy is not reached during one therapy session the adaptation process can be extended over several sessions, with most of the interaction occurring with the optimal policy in place. In fact, this is very similar to real-life situations where therapists get to know patients over several therapy sessions and respond to their clues to provide a more efficient recovery environment.

Conclusions: We presented a non-contact therapist robot intended for monitoring, assisting, encouraging, and socially interacting with post-stroke users during rehabilitation exercises. The experimental results provide first evidence for the effectiveness of robot behavior adaptation to user personality and performance.

References

[1] Tapus, A., Tapus, C., and Matarić, M., J. (2008) User-Robot Personality Matching and Assistive Robot Behavior Adaptation for Post-Stroke Rehabilitation Therapy. *Journal on Intelligence Service Robotics* (to appear).

[2] Tapus, A., and Matarić, M., J. (2006) User personality matching with hands-off robot for post-stroke rehabilitation therapy. In *Proc. International Symposium on Experimental Robotics* (ISER'06), Rio de Janeiro, Brazil, July.

The conditioning effect of stochastic dynamics in continuous reinforcement learning

Yuval Tassa,
Hebrew University, Jerusalem
tassa@alice.nc.huji.ac.il

Teaching robots to perform complex behaviors using reinforcement learning (RL) algorithms is a long-term goal of the machine learning community. This goal is underwritten by our familiarity with nervous systems which evidently implement some form of RL, while simultaneously setting the highest benchmark for motor control. When attempting to learn optimal controllers, *stochasticity* is often considered a hurdle to be overcome. Methods which deal with uncertainty, whether process noise, measurement noise or modeling uncertainty are seen as extensions of the observable deterministic case, as exemplified by the stochastic and robust extensions to linear-quadratic control.

Perhaps surprisingly, the assumption of stochastic dynamics can also *simplify* learning algorithms. Intuitively, the smoothing of our beliefs about the world into a distribution function entails a limiting of the spatial bandwidth of our predictions and therefore an effective reduction of dimensionality, which can consequently be exploited.

In order to benefit from this low-pass effect, our model and algorithm must be spatially continuous. An important class of algorithms which take advantage of continuity and smoothness are those which construct second-order approximations of their parameters, as epitomized by the classic *Newton's method*. When implementing second-order algorithms, stochasticity of the dynamics is shown to reduce the condition number of Hessian matrices, a chief measure of convergence quality. We present two recent results where such phenomena occur in two very different settings.

Restricting ourselves to the dynamic programming (DP) framework, we consider algorithms which learn the *value function*, or solve the Hamilton Jacobi Bellman (HJB) equation. Using a general-purpose feedforward neural network to approximate the value function, we apply a Levenberg-Marquardt algorithm to minimize the squared HJB residual [1]. When a Brownian noise term is introduced into the dynamics, an additional second-derivative term is added to the HJB equation. We show how the addition of this term to the residual, *without injecting any actual noise*, smoothes the value function and reduces the condition number of the Hessian by orders of magnitude.

In the second example we use Differential Dynamic Programming (DDP), a method which iteratively computes an explicit second-order model of the value function along a trajectory [2]. At points where small changes in the policy have large effects on the value, the local Hessian matrices can become ill-conditioned and lead to divergence or slow convergence. We show how the introduction of worst-case minimax noise in the H^{∞} control framework also has a conditioning effect on these local matrices.

[1] Tassa & Erez (2007). IEEE Transactions on Neural Networks, 18(4):1031-1041
[2] Tassa, Erez & Smart (2007). NIPS 2007 (accepted)

Reinforcement Learning and Weak Derivatives for Motor Learning in Robotics

E. A. Theodorou, J. Peters, S. Schaal
University of Southern California, MPI & ATR

Every day motor behavior consist of a plethora of challenging motor skill from discrete movements such as reaching and throwing to rhythmic movements such as walking, drumming and running. How this plethora of skills can be learned remains an open question. In particular, is there any unifying computational framework that could model the learning process of this variety of motor behaviors and at the same time be biologically plausible? In this work we aim to give an answer to these questions by providing a computational framework that unifies the learning mechanism of both rhythmic and discrete movements under optimization, ie, in a non-supervised trial and error fashion way.

Our suggested framework is based on Reinforcement Learning, which is mostly considered as too costly to be a plausible mechanism for learning complex limb movement. However, recent work on reinforcement learning with policy gradient methods and weak derivatives combined with the parametarized movement primitives allows novel and more efficient algorithms. By using the representational power of motor primitives we show how rhythmic motor behaviors such as walking, squashing and drumming as well as discrete behaviors like reaching and grasping can be learned. Furthermore we are proposing a new method of estimating the policy gradient by using the weak derivatives framework and also test a evaluate this new method into 3D arm simulator.

References

[1] B. Heidergott. A weak derivative approach to optimization of threshold parameters in a multi- component maintenance system. *Journal of Applied Probability*, 38:386–406, 2001.

[2] S. Schaal, J. Peters, J. Nakanishi, and A. Ijspeert. Learning movement primitives. In *International Symposium on Robotics Research (ISRR2003)*, Springer Tracts in Advanced Robotics, Ciena, Italy, 2004. Springer.

Bayesian Nonparametric Regression with Local Models

Jo-Anne Ting[1], Stefan Schaal[1,2]

[1]Computer Science, University of Southern California, Los Angeles, CA 90089
[2]ATR Computational Neuroscience Laboratories, Kyoto 619-0288, Japan

We propose a Bayesian nonparametric regression algorithm with locally linear models for high-dimensional, data-rich scenarios where real-time, incremental learning is necessary. Nonlinear function approximation with high-dimensional input data is a nontrivial problem. For example, real-time learning of internal models for compliant control may be needed in a high-dimensional movement system like a humanoid robot. Fortunately, many real-world data sets tend to have locally low dimensional distributions, despite having high dimensional embedding [1, 2]. A successful algorithm, thus, must avoid numerical problems arising potentially from redundancy in the input data, eliminate irrelevant input dimensions, and be computationally efficient to allow for incremental, online learning.

Several methods have been proposed for nonlinear function approximation, such as Gaussian process regression [3], support vector regression [4] and variational Bayesian mixture models [5]. However, these global methods tend to be unsuitable for fast, incremental function approximation. Atkeson et al. [6] have shown that in such scenarios, learning with spatially localized models is more appropriate, particularly in the framework of locally weighted learning. In recent years, Vijayakumar et al. [7] have introduced a learning algorithm designed to fulfill the fast, incremental requirements of locally weighted learning, targeting high-dimensional input domains through the use of local projections. This algorithm, called Locally Weighted Projection Regression (LWPR), performs competitively in its generalization performance with state-of-the-art batch regression methods and has been applied successfully to sensorimotor learning on a humanoid robot.

The major issue with LWPR is that it requires gradient descent (with leave-one-out cross-validation) to optimize the local distance metrics in each local regression model. Since gradient descent search is sensitive to the initial values, we propose a novel Bayesian treatment of locally weighted regression with locally linear models [8] that eliminates the need for any manual tuning of meta parameters, cross-validation approaches or sampling. Combined with variational approximation methods to allow for fast, tractable inference, our algorithm learns the optimal distance metric value for each local regression model. It is able to automatically determine the size of the neighborhood data (i.e., the "bandwidth") that should contribute to each local model. A Bayesian approach offers error bounds on the distance metrics and incorporates this uncertainty in the predictive distributions. By being able to automatically detect relevant input dimensions, our algorithm is able to handle high-dimensional data sets with a large number of redundant and/or irrelevant input dimensions and a large number of data samples. We demonstrate competitive performance of our Bayesian locally weighted regression algorithm with Gaussian Process regression and LWPR on standard benchmark sets.

References

[1] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323, 2000.

[2] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, 2000.

[3] Christopher K. I. Williams and Carl Edward Rasmussen. Gaussian processes for regression. In David S. Touretzky, Michael C. Mozer, and Michael E. Hasselmo, editors, *In Advances in Neural Information Processing Systems 8*, volume 8. MIT Press, 1995.

[4] A. Smola and B. Schölkopf. From regularization operators to support vector kernels. In M. I. Jordan, M. J. Kearns, and S. A. Solla, editors, *Advances in Neural Information Processing Systems 10*, pages 343–349, Cambridge, MA, 1998. MIT Press.

[5] Z. Ghahramani and M.J. Beal. Graphical models and variational methods. In D. Saad and M. Opper, editors, *Advanced Mean Field Methods - Theory and Practice*. MIT Press, 2000.

[6] C. Atkeson, A. Moore, and S. Schaal. Locally weighted learning. *AI Review*, 11:11–73, April 1997.

[7] S. Vijayakumar, A. D'Souza, and S. Schaal. Incremental online learning in high dimensions. *Neural Computation*, pages 1–336, 2005.

[8] J. Ting, A. D'Souza, S. Vijayakumar, and S. Schaal. A Bayesian approach to empirical local linearization for robotics. Submitted for publication, 2007.

Tekkotsu as a Framework for Robot Learning Research

David S. Touretzky
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15213
dst@cs.cmu.edu

Ethan J. Tira-Thompson
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
ejt@andrew.cmu.edu

Tekkotsu (the name means “framework”, literally “iron bones” in Japanese) is an open source application development framework for mobile robots originally created for the Sony AIBO. It has since been extended to support a variety of other platforms, and is available for free at Tekkotsu.org. Tekkotsu promotes a high level approach to robot programming we call “cognitive robotics” by providing primitives that make it easy to implement new applications. These include a “dual coding” vision system with map builder, particle filter-based localization, forward and inverse kinematics solvers, and an extensive collection of GUI tools for teleoperation and monitoring. Tekkotsu is implemented in C++, with the GUI tools in Java for portability.

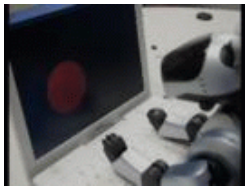
Although primarily intended for undergraduate robotics education, Tekkotsu has also proven useful as a research platform because of its powerful and well-integrated primitives. We have in the past used Tekkotsu to demonstrate classic machine learning problems such as the two-armed bandit, and negative patterning (XOR learning). The images below are screenshots from demo videos available at Tekkotsu.org. Current projects include addition of a SIFT-style object recognition facility, and development of visually-guided motor primitives for grasping and manipulation of small objects using a Lynx Motion arm. (The arm is part of a new prototype robot called Regis.) We encourage machine learning researchers to consider Tekkotsu if they wish to test their algorithms on real robots.

Touretzky, D.S., Daw, N.D., and Tira-Thompson, E.J. (2002) Combining configural and TD learning on a robot. *Proceedings of the Second International Conference on Development and Learning*, IEEE Computer Society, pp. 47-52.

Touretzky, D.S., Halelamien, N.S., Tira-Thompson, E.J., Wales, J.J., and Usui, K. (2007) Dual-coding representations for robot vision in Tekkotsu. *Autonomous Robots,* 22(4):425–435.

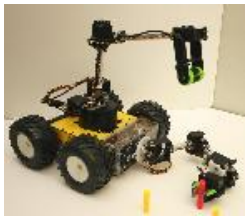
Supported by National Science Foundation awards CNS-0540521, DUE-0717705, and CNS-0742106.

Two-armed bandit learning demonstration. On each trial, the AIBO robot must choose whether to press the left or right side of the keyboard. Reward or non-reward is indicated by a sound and a color change of a bouncing ball on the display. The head moves to track the ball.



Combining configural and TD learning to demonstrate negative patterning (XOR). A red or blue ball stimulus is followed a few seconds later by a reward signal (thumbs up gesture). But both balls together are not followed by reward. The robot learns to make an appropriately-timed reward response (tail wag) only to single balls.

Regis prototype created specifically for research on visually guided manipulation in Tekkotsu. Features include a 600 MHz/128 MB gumstix processor, “goose neck” webcam (4-dof arm with camera at the tip), and a front-mounted 6-dof “crab arm” manipulator lying in the plane of the workspace for unobstructed visual monitoring.



Robot Perception Challenges for Machine Learning

Chieh-Chih Wang
Department of Computer Science and Information Engineering
Graduate Institute of Networking and Multimedia
National Taiwan University
Taipei, Taiwan
bobwang@ntu.edu.tw

Abstract

Establishing the spatial and temporal relationships among a robot, stationary entities and moving entities in a scene serves a basis for robot perception. *Localization* is the process of establishing the spatial relationships between the robot and stationary objects, *mapping* is the process of establishing the spatial relationships among stationary objects, and *moving object tracking* is the process of establishing the spatial and temporal relationships between moving objects and the robot or between moving and stationary objects. Localization, mapping and moving object tracking are difficult because of *uncertainty* and *unobservable states* in the real world.

Over the last two decades, the simultaneous localization and mapping (SLAM) problem has attracted immense attention in the robotics literature. Theoretical and practical issues such as representation, computational complexity and data association have been addressed. However, it is assumed that the scene is stationary in SLAM and moving objects are filtered out. In [1, 2], we pointed out that SLAM and moving object tracking are mutually beneficial. A mathematical framework was established to integrate SLAM and moving object tracking in which a solid basis is provided for understanding and solving the whole problem, simultaneous localization, mapping and moving object tracking, or SLAMMOT. In [3], we further relaxed the assumption that the robot and moving objects move independently of each other, and proposed a scene interaction model and a neighboring object interaction model to accomplish interacting object tracking in crowded urban areas. With the use of the interaction models, unusual activity recognition is accomplished straightforwardly.

It is believed by many that a solution to the SLAM problem will open up a vast range of potential applications for autonomous robots. We believe that a solution to robot perception will expand the potential for robotic applications still further, especially in applications which are in close proximity to human beings. In this poster, we summarize the approaches to learn the map, to classify moving and stationary entities, and to recognize short-term and long-term interactions between the robot and the dynamic scene. In addition, robot perception challenges and opportunities for machine learning are addressed.

References

[1] C.-C. Wang and C. Thorpe, "Simultaneous localization and mapping with detection and tracking of moving objects," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Washington, DC, May 2002.

[2] C.-C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte, "Simultaneous localization, mapping and moving object tracking," *The International Journal of Robotics Research*, vol. 26, no. 9, pp. 889–916, September 2007.

[3] C.-C. Wang, T.-C. Lo, and S.-W. Yang, "Interacting object tracking in crowded urban areas," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Roma, Italy, April 2007.

Maximum Entropy Inverse Reinforcement Learning

Brian D. Ziebart
Machine Learning Department
Carnegie Mellon University
Pittsburgh, PA 15213
bziebart@cs.cmu.edu

J. Andrew Bagnell
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
dbagnell@ri.cmu.edu

Anind K. Dey
Human-Computer Interaction Institute
Carnegie Mellon University
Pittsburgh, PA 15213
anind@cs.cmu.edu

1 Extended Abstract

In many domains, demonstrating good behavior is easier than tuning parameters of an agent so that it behaves in a desirable way. A powerful recent idea to approach problems of imitation learning is to structure the space of learned policies to be solutions to search, planning, or, more generally, Markov Decision Problems. The imitation learning problem then is reduced to recovering a reward function that induces the demonstrated behavior.

Ratliff et al. [2] cast this problem as one of structured maximum margin prediction (MMP). These authors consider a class of loss functions that directly measure disagreement between an expert and a learned policy, and then efficiently learn a reward function using a convex relaxation of the loss function using the structured margin method using only oracle access to an MDP solver. However, this method suffers from some significant drawbacks when a single policy is not significantly better than all other policies, which can occur frequently in the presence of noise.

Abbeel and Ng [1] provide an alternate approach based on Inverse Reinforcement Learning (IRL). They propose a strategy of matching feature expectations between an expert's policy and a learner's behavior. Unfortunately, both the IRL concept and the matching feature counts are ambiguous. Each policy can be optimal for many reward functions (e.g., all zeros) and many policies or distributions over state/action pairs can lead to the the same feature counts. No method is proposed to resolve the ambiguity.

In this work, we treat uncertainty about expert behavior in a thoroughly probabilistic way. As in [1] and [2], we require our policies to match feature expectations. However, we attempt to estimate the probability of an expert taking trajectories as $p(\xi)$ using the principle of maximum entropy, which suggests that the natural distribution on trajectories is the unique distribution that maximizes the entropy of the distribution of trajectories subject to meeting the expectation constraints. In the absence of additional knowledge, the problem then is clear: $\max H(p(\xi))$ subject to $E_p[f_i(\xi)] = c_i$ where c_i are feature counts experienced by the expert we wish to imitate.

Solving this optimization problem yields a distribution on trajectories of the form $p(\xi) \propto e^{-w^T f}$, for feature counts f. The most likely trajectory then is, naturally, the one which minimizes $w^T f$ (where f is the feature count over the trajectory). We show that MaxEntIRL is more robust to noise than MMP and removes the ambiguity about reward functions that occurs with methods based on IRL, while providing the same key guarantee. MaxEntIRL produces policies that achieve (nearly) the same reward as the expert demonstrating the policy on the expert's unknown reward function – even when that exact reward function is unrecoverable (without ambiguity) from the available data.

For problems where actions have deterministic outcomes, the gradient of this convex optimization is then the difference between our learned policy's feature expectations and those of the demonstrated policy (Equation 1), for which we provide efficient algorithms for fixed time thresholds.

$$\frac{\delta}{\delta w_k} P(\tilde{\xi}|w) = c_i - E_p[f_i(\xi)|w] \tag{1}$$

We apply this method to learn context-sensitive driving route preferences. The features of each road segment (e.g., road type, number of lanes) are mapped to a negative reward by parameters we optimize using over 100,000 miles of GPS trace data collected from Yellow Cab Pittsburgh taxi drivers. The resulting model is employed for recommending routes that use some of the traffic-avoding tricks that cab drivers employ.

References

[1] P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *Proc. ICML*, 2004.

[2] N. Ratliff, J. A. Bagnell, and M. Zinkevich. Maximum margin planning. In *Proc. ICML*, 2006.