Bold Hearts 2003: Skill Learning Via Information-Theoretical Decomposition of Behaviour Features

Santiago Franco and Daniel Polani

Abstract

The ideas underlying the soccer simulation team *Bold Hearts* are described. At the heart of the team concept, we describe SIVE, a method for *Selective Information Variant Extraction*. It is used to learn behaviours in a "holistic" fashion. SIVE concentrates on behaviours at the boundary between different possible outcomes. This boundary is analyzed in an information-theoretic fashion and decomposed into such parameters which are maximally predictive with respect to the behaviour outcome (the *variants*) and such parameters which are maximally insensitive (*invariants*) with respect to the behaviour outcomes. SIVE is used to train the passing and goal-shot skill, showing that it provides an effective framework to train skills.

1 Introduction

A challenge in the field of autonomous agents is the training of specific skills which can then be used in the agents' individual quests. The RoboCup scenario with all its complexity and the simulation league in particular, is no exception.

Many approaches to construct powerful RoboCup agents therefore concentrate on creating specific skills and capabilities. These are often constructed with knowledge about specific properties of the world physics as simulated by the soccer server. Use of explicit knowledge in agents, however, limits the flexibility and robustness of the approach should the simulated world change and it makes it difficult to generalize to other fields. In addition, the capability of learning is one of the key questions in Artificial Intelligence, and it would thus be desirable to address this question in the RoboCup context.

The central importance of learning has been identified early in the history of RoboCup [11] and approaches have been developed to tackle the intricate structure of the RoboCup learning scenario [10]. In these approaches, the principles of Reinforcement Learning algorithms are generalized to the specific needs of RoboCup, such as team play with limited world knowledge. Here, the learning of skills in continuous domains is of specific interest, and pioneering work has been done e.g. in [1].

Reinforcement Learning methods are attractive for learning approaches because the setting they work in is highly general, mathematically accessible and well understood. The generality of the task class that can be addressed by Reinforcement Learning,

however, comes at a price. The learning algorithms are slow, particularly in large search spaces. To alleviate that, dedicated decompositions of the representation of the state space have to be performed that deconstruct the task hierarchically into manageable parts (e.g. [3]). Still, a large number of learning steps has to be taken to learn a more complex task. In addition, convergence problems can arise in continuous domains (as RoboCup) [12].

One of the significant issues in typical Reinforcement Learning methods is the fact that to train a skill properly, basically the whole possible event space has to be sampled, leading to a huge number of required training runs. This, however, does not seem to be the way that e.g. humans attain a skill. To see this, it is instructive to consider a typical RoboCup scenario involving humans.

In the GermanOpen 2001 there was the opportunity, witnessed by the second author, to perform a simulated soccer game between an agent team controlled by humans and a team of autonomous agents. Open Zeng [7] provided an interface through which the humans could control an agent team which then could play in the simulator against e.g. an autonomous team. With Open Zeng, the humans obtained exactly the same blurred and imprecise information as their autonomous agent opponent team and had to base their actions on this information.

The autonomous agent team turned out to be much faster and accurate and was clearly playing a stronger game than the human-controlled team. This was expected, given the level of specific adaption and optimization that autonomous teams undergo during preparation for the tournament and that the humans had never before trained to use the OpenZeng agent controls which were of limited ergonomy. It was striking, however, that the human-controlled team exhibited an extremely steep learning curve. While during the early games the autonomous agent opponent scored a large number of goals, the human-controlled team improved very quickly and increasingly prevented the autonomous team from scoring, although the human accuracy in estimating ball positions and performing actions was nowhere as accurate as that of the autonomous team. This is a clear indication that the “exhaustive learning” character exhibited by typical learning algorithms is inadequate to obtain the directedness and generalization power that we find in human learning. It would be desirable to mimic some of the properties exhibited by human learning: extremely fast generalization and adaptation, “holistic” learning and the capability to combine skills. The present work attempts to do a step in that direction.

2 An Approach to Behaviour Learning

2.1 Philosophy

To approach aforementioned goal, it makes sense to picture a phenomenological scenario how one would like to see agents being trained, motivated by human learning. When training e.g. a human team, the training is not limited to just playing games over and over until a satisfactory level is achieved. Instead, one concentrates on training certain skills and then successively combining these skills. Thus, learning is undertaken in increasing complexity and more primitive skills are combined into larger ones. In

addition, as opposed to the fundamental dogma of Reinforcement Learning, skills in humans are often learned in an "holistic" fashion, i.e. not split up into time slices, but as complete patterns. Also, humans often train skills at the limits of their capability, ignoring more well-established capabilities.

In Bold Hearts 2003, we undertook an approach which mimics some of these phenomenological considerations. In this approach, called *SIVE* (Selective Information Variant Extraction[1], we use specific skill training scenarios in which behaviours are learned, not timestep-wise, but as complete patterns, concentrating on critical (risk) regions, i.e. of uncertain outcome. In addition, the approach is grounded information theory and we believe it to open the path towards generalization and the rapid acquisition of new skills. This is achieved by separating factors of influence and of invariance.

2.2 Earlier Work

The SIVE method is inspired by many different sources. Motivated by classical reinforcement learning [16, 12], it does not follow the standard paradigm of partitioning time and thus sequences of actions into equally distributed intervals; in fact, even reinforcement learning models with continuous time [15] still structure time. SIVE does not do that and instead captures the whole behaviour sequence as whole, classifying that behaviour according to its outcome, e.g. *success* or *failure*, respectively. This (external) classification can be considered a minimal reinforcement, or else, similar to a standard classification problem. While the support vector machine formalism [14, 4] would have provided a transparent approach to solve such a classification problem, it was decided to develop the approach closely guided by information-theoretical perspectives. It should be mentioned that the structural risk minimization propagated in [14] has information-theoretic ramifications, as the VC dimension is obtained from the size of the space of the possible classificator configurations. The SIVE framework was furthermore inspired by the explicit model to compute an optimal scoring strategy developed in [6].

3 The SIVE Method: Overview

The SIVE method incorporates several aspects. It consists of training individual skills in explicitly given scenarios which correspond to scenarios set up by coaches for human teams. Behaviours are trained by classifying the overall outcomes and the training concentrates on critical (risky) behaviours, where the outcome can not be predicted safely. Finally, a representation is sought for the behaviours that attempts to capture the properties that most and those that least affect the outcome of a behaviour. We illustrate SIVE in the concrete setting of the RoboCup scenario, although it is by no means limited to that.

[1] *Lat.* sive: be it that ... or that ...

3.1 Scenario Choice

To achieve high results in a human soccer team, one would not just have them play complete games indefinitely until their goal-scoring skills would globally improve. Instead, one would have a coach pick specific skills to be trained until the players are capable of handling them. A further step would then be to integrate these individual skills into more global skill clusters.

This is the framework that is adopted by the SIVE method. It concentrates on the mastery of individual skills; the choice of the scenarios themselves, however, is not part of the present SIVE framework. In the present team, the scenario(s) are chosen by the human "coach" the same way a coach would plan training sessions for human players to enhance specific skills.

In Bold Hearts 2003, we concentrated entirely on the *pass* skill. Our scenario consists of one player (the *kicker*) kicking the ball, the other player (the *catcher*) trying to capture this kicked ball inside a fixed time limit (typically 20-30). if the ball is not captured in this time window, the ball is considered lost.

3.2 Outcome Classes

As opposed to classical reinforcement learning, SIVE distinguishes the two cases of capture and loss, but does not a priori evaluate the cases, as classical MDP learning methods. Thus, this enables considering different types of situations using the same framework. E.g., Capturing the ball may be considered "success" if the catcher is our team mate or "failure" if the catcher is our opponent.

3.3 Instances of Interest

Another aspect of SIVE is that only "critical" ("interesting", "risky") behaviours are considered. E.g., in our scenario, SIVE ignores kicks where the ball is always captured or always lost by the catcher. Instead, SIVE concentrates on kicks where the outcome (capture or loss) cannot be safely predicted. By concentrating the actions on the critical region, SIVE aims at establishing the boundary between the different outcome classes. As opposed to classical classification methods, SIVE exploits the fact that the agent can actively probe the critical regions and does not have to restrict itself to a given sample of training data.

The concentration on critical, "risky" areas of behaviour has an attractive correspondence to the observation that risk-taking is part of animal and human behaviour, whether for exploratory benefits (e.g. to exploit the limits of their capabilities) and for the provocation of novelty stimuli [2, 8] or for reasons of attaining a profitable social status [17].

3.4 Variants and Invariants

On this set of critical behaviours, SIVE proceeds to find additional structure. To be able to generalize, it is useful to know which are the parameters that most predictive for the outcome of an action (*variants*) and which are most insensitive to that outcome

(*invariants*). The variants tell the agent which aspect of an action is most important for achieving a particular outcome. The invariants contain additional information about the structure of the problem which is not used in Bold Hearts 2003, but will be exploited in the future.

4 A Detailed Description of the SIVE Method

In the present section, the SIVE concept and its application to the RoboCup scenario is elaborated in more formal detail. As has been mentioned in Sec. 2.1, the SIVE method is has strong roots in information theory which we will thus use to formulate it.

4.1 Definitions

We consider random variables over discrete or continuous domains. Denote random variables by uppercase letters $X, Y, Z \ldots$, their respective instantiations by lowercase letters $x, y, z \ldots$, and their respective domains by calligraphic letters $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$. In the following we will assume that the domains are discrete. Most of the statements made are generalizable to continuous domains in a suitable way [4]. Since in our implementation we operate only with discrete finite domains and discretized continuous systems, this will be sufficient to understand the following algorithms.

Be $P(X = x)$ the probability that a random variable X with domain $\mathcal{X}$ assumes a value $x \in \mathcal{X}$. Instead of $P(X = x)$, we will often write more compactly as $p(x)$ by abuse of notation if no confusion is possible. In the same way, the conditional probability distribution of a random variable Y conditioned on a random variable X will be denoted as $P(Y = y|X = x)$ or, shorter, by $p(y|x)$.

In the following we define several information-theoretic quantities. Given a random variable X, the *entropy* of X is given by

$$H(X) := -\sum_{x \in \mathcal{X}} p(x) \log p(x) \ .$$

It measures the uncertainty of the outcome of a single sample of the variable X. If $\log$ is the binary logarithm, the entropy is measured in *bits* and denotes how many bits it would take to encode the given outcome.

The entropy of jointly distributed random variables (X, Y) is written $H(X, Y)$ instead of $H((X, Y))$ and is thus given as

$$H(X, Y) := -\sum_{(x,y) \in \mathcal{X} \times \mathcal{Y}} p(x, y) \log p(x, y)$$

Having jointly distributed variables X and Y, the marginal distributions are given as

$$p(x) = \sum_{y \in \mathcal{Y}} p(x, y)$$

and analogously for $p(y)$.

The *conditional entropy* of Y given X is defined by

$$H(Y|X) := \sum_{x \in \mathcal{X}} p(x) H(Y|X = x)$$

with

$$H(Y|X = x) := -\sum_{y \in \mathcal{Y}} p(y|x) \log p(y|x)$$

for $x \in \mathcal{X}$.

The *mutual information* between two (jointly distributed) random variables X and Y is given by

$$\begin{aligned} I(X;Y) &:= H(X) - H(X|Y) \\ &= H(Y) - H(Y|X) \\ &= H(X) + H(Y) - H(X,Y) \end{aligned}$$

and quantifies how much information X conveys about Y. It is a fundamental quantity of information theory and quantifies predictability and dependence of random variables.

SIVE makes heavy use of the information-theoretic notions introduced above to profit from their universality and interrelation with other fields.

4.2 Overview over the method

For the convenience of the reader we recapitulate the main phases of the SIVE method:

1. identify and sample the regions of "critical" behaviours for which the outcome cannot be safely predicted;
2. construct the variants and invariants for the attained sample;
3. use the variants to identify desired actions in a given situation;
4. and use the invariants to collapse or simplify the identification of features in other contexts.

A side note: All the phases of SIVE can be formulated in information-theoretic terms. However, in making the concrete construction in the present case, we occasionally still needs to resort to other methods, e.g. neural networks. Note also that item 4 has not yet been realized in the present version of SIVE.

4.3 Identifying the Critical Actions

We consider the following training scenario. Assume an agent in possession of the ball (the *kicker*) for which we wish to train the passing skill. The agent is to learn under which conditions it may be able to pass the ball to another player (the *catcher*) and under which the catcher will not be able to catch the ball.

For this, we devise a training scenario as follows: the kicker is placed at a defined position (e.g. the origin). Relative to it, the catcher is positioned. The catcher's relative position is assumed to be known to the kicker. The kicker controls kick strength and angle, it cannot control the position of the catcher. On kicking the ball, after a a priori given cut-off time, the outcome of the experiment is recorded which is either *catch* or *loss* (by the catcher).

For the experiments performed in our case, we fix the kick strength to the maximum possible value. The rationale for this is that, if one wishes to prevent an opponent player from catching the ball and excludes the own player currently in ball possession, as a first approximation one will kick the ball as strongly as possible[2]. On the other hand, if the catcher is a player from the kicker's own team, then, again, it is a good first approximation to require the catcher to perform the catch as early as possible. Fixing the kick strength to the maximum value, allows us to reduce the dimensionality of the input space and simplifies the construction of variants and invariants.

The remaining degrees of freedom are kick angle with respect to the kicker orientation and the opponent position relative to kicker orientation and position of which the kicker can only control kick angle. During training, kick angles and catcher positions are varied. For these trials it is recorded whether the kick is being caught or lost (inside the cut-off time limit). Using a sliding window, the critical samples are identified, i.e. those in the region where it cannot be safely predicted whether the outcome is *catch* or *loss*. The critical samples are displayed in Fig. 1 as a data cloud. The data cloud consists of two nearly 2-dimensional "sheets". The sheets form the boundary between caught and lost kicks. The region between the sheets belongs to the kicks which are always caught, the region outside of the sheets is the region of the kicks which are always lost by the catcher.

4.4 Variants and Invariants

The sheets from Fig. 1 contain the information about the variants and invariants of the given task. In the present scenario we will make use of the fact that both sheets can be regarded as an approximation to a two-dimensional manifold. A more general approach will be developed in the future.

The next step in the SIVE framework is to look for a nonlinear transformation of the original data such that a high amount of information about the original data is preserved, but at the same time variants and invariants are being separated. Let X be the random variable denoting the original data, and R the result (outcome) of the action (i.e. *catch* or *loss*). Then we are seeking an information-preserving transformation $T : \mathcal{X} \to \mathcal{Z}_v \times \mathcal{Z}_i$ such that, if we write $T(X) = Z = (Z_v, Z_i)$, Z_v becomes a variant

[2]This may be no longer true in more complex strategic scenarios, but we concentrate here on simple approximations first.

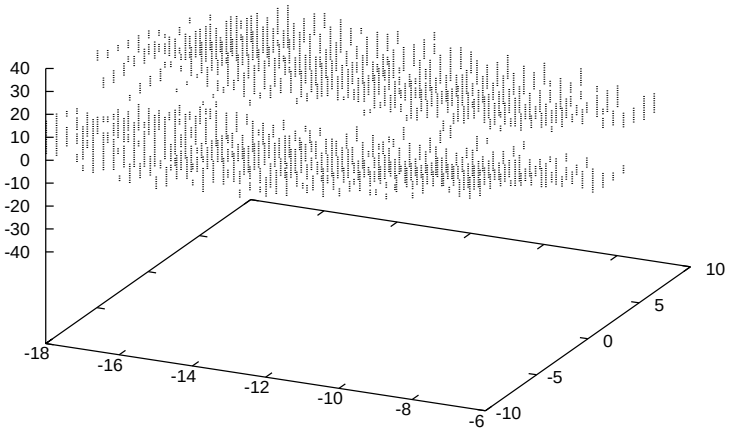

Figure 1: The critical samples for the pass scenario. The x and y axes denote the position of the catcher relative to the kicker (including relative orientation), and the z axis denotes the kick angle. The three-dimensional structure of the data cloud is not easy to represent in a static black-and-white figure. The cloud consists of two "sheets" which are extended in approx. 2 dimensions and are quite localized in the 3rd, modeled approximately by two 2-dimensional manifolds. For more details, see text.

and Z_i becomes an invariant. Ideally, we would require T such that

$$I(X;Z) = I(X;Z_v,Z_i) = H(X) \quad \text{(information preservation)}, \quad (1)$$

$$I(Z_v;R) = I(X;R) \quad \text{(variant property)} \quad (2)$$

$$I(Z_i;R) = 0 \quad \text{(invariant property)} \quad (3)$$

$$I(Z_v;Z_i) = 0 \quad \text{(independence)} \quad (4)$$

Information preservation requires that the transformation will not lose any information about the structure of the original data. The variant property requires that the transformation identifies a *variant* parameter which is as predictive about the outcome of the action as the original data. The invariant property separates an *invariant* parameter which is completely insensitive to the outcome. Finally, the variant and invariant are to be completely independent to remove redundancies.

It cannot be expected that in a real-world constellations all these properties be fulfilled or even a suitable transformation be practically identifiable. So, for a practical application the equations (1)-(4) are to be modified as to seek a transformation that maximizes the left-hand side of (1) and (2) and minimizes the left-hand side of (3) and (4). Since X is in general a continuous-valued variable, for the maximization of (1) to make sense, a suitable normalization of the transformation (e.g. fixed variance) has to be assumed.

Formulated as an optimization task still it is often not possible to fulfil all the properties at once. There are different possible approaches to solve that problem. One way is to formulate a Lagrangian optimization problem not unlike in the Information Bottleneck scenario of [13]. The other approach is to use a multiobjective optimization method, e.g. Evolutionary Algorithms.

In the present case, the scenario is however so simple that a straightforward heuristic approach was chosen.

4.5 Constructing the Variant and Invariant Parameters using Self-Organizing Maps

The boundary data are concentrated in two sheets which are approximations of two-dimensional manifolds. In addition, the two-dimensional extension is roughly invariant to the outcome, while the perpendicular direction points in the direction of the most significant change of outcome, namely from inside the region where the ball is always caught to the outside region where the ball is always lost.

So, the idea was to model the data sheets using Self-Organizing Maps (SOMs) [9]. We used one SOM per sheet. The result is shown in Fig. 2. To obtain a heuristic approximation for a suitable value for the variant Z_v, we use a geometric heuristic. We know that for any sample between the two sheets, the outcome is a clear *catch* and for a sample outside of the two sheets, the outcome of the experiment is a clear *loss*. Thus, we are interested in knowing whether an event is between or outside the sheets.

Let a sample x be given. Denote by $x^{(1)}$ that weight of the SOM in the first sheet which is closest to x, and similarly $x^{(2)}$ for the second sheet. Consider now the vectors $x - x^{(1)}$ and $x - x^{(2)}$. The cosine between these two vectors is negative for a sample x between the sheets and positive for x outside of the sheets. Thus, this cosine can be used as a measure for the variant component of the event. We use the scalar product of the two vectors to obtain the cosine.

Note that this approximation relies on the fact that the sheets are roughly parallel to each other. Also, it assumes that the sample for which one wishes to determine whether it is caught or lost has a corresponding point on the SOMs. In other words, for a well-defined classification, the event should lie either between the sheets or project approximately perpendicularly onto the closest points on each SOM. An outlier event that lies sideways outside the range of the sheets is factually not being covered by training knowledge and cannot be expected to be classified correctly by the given system. In the the Bold Hearts 2003 team the SIVE classifier is disabled in this case.

4.6 Use of the Variant

Using the SOMs and the approach from Sec. 4.5, we obtain a variant Z_v for the passing scenario. This means that we have an estimate for the quality of a pass, both with respect to how easy it is to catch by a team mate, and with respect how difficult it is to intercept by an opponent. The whole strategy of the Bold Hearts 2003 team is built around this variant calculation.

To calculate passes, the Bold Hearts strategy chooses a target position which can be either static and predefined or dynamic (position relative to a team mate). Using

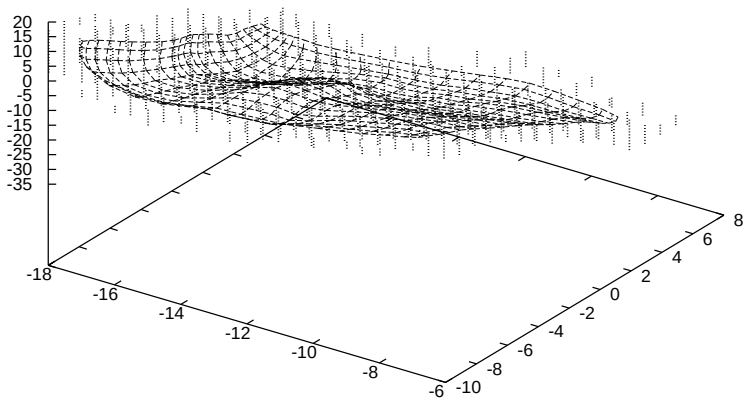

Figure 2: The lower sheet of the same data cloud as in Fig. 1, together with an embedded SOM, trained with these data. For each sheet, a SOM is trained individually.

Z_v, it calculates the possible instances of interceptions and successful passes, sorted by priority, i.e. distance. Negative Z_v indicate a capture by the respective player, positive a loss (in the given cut-off time). As a continuous variable, Z_v gives a measure how close the ball is to capture (or loss). A threshold is used to fine-tune the passing behaviour. A similar strategy was used against the goalie.

5 Discussion

The passing skill obtained by the SIVE method proved to be very effective. It gave a good percentage of strong and efficient passes, and allowed a very aggressive play using only very limited additional strategy elements. The team was occasionally able to break through the defenses of an extremely strong team like e.g. Tsinghuaeolus 2003 and occasionally try a goal shot (unsuccessfully, though, since the tactical component was not sufficiently developed in the present team). In the competition, it beat a team of average strength and, being used as fixed opponent, won all the games in the coach competition.

A few problems remained. Occasionally, possible interceptors have been overlooked or team-mate was not able to see the ball in time. As the training took place against the simple UvA 2002 agents, very strong teams that had a different reaction pattern (e.g. Tsinghuaeolus 2003) could outsmart the passing skill. A possible solution to that problem is to train the passing skill also using Tsinghuaeolus as opponent. We can expect slightly different boundary areas for this case which would work better against Tsinghuaeolus and teams based on Tsinghuaeolus.

6 Conclusion and Outlook

The SIVE framework combines different concepts using information-theoretical principles to obtain a "holistic" action learning for fixed scenarios. Properties of SIVE include the concentration on critical regions where the outcome of actions cannot be safely predicted, the separation of variants and invariants in based on an principled information-theoretic framework, identifying the features which most affect the outcome. The invariant features, not yet used in the present team open the path to generalization of skills in the future. In particular, SIVE is open to the possible training of the combination of skills. This will be the topic of future work.

7 Acknowledgements

We wish to thank the UvA team for providing the basis code (UvA 2002) [5] on which the present team is based and which allowed us to concentrate on the development of the SIVE method.

References

[1] Buck, S., and Riedmiller, M., [2000]. Learning situation dependent sucess rates of actions in a robocup scenario. In *Proceedings of PRICAI '00, Melbourne, Australia, 28.8.-3.9.2000*, 809.

[2] Commission on Behavioral and Social Sciences and Education, [1999]. Pathological Gambling. Open Book.
`http://books.nap.edu/books/0309065712/html/R1.html`, November 2003

[3] Dietterich, T. G., [1999]. Hierarchical reinforcement learning with the MAXQ value function decomposition. *Submitted to Machine Learning*.

[4] Haykin, S., [1999]. *Neural networks: a comprehensive foundation*. Prentice Hall.

[5] Kok, J., and de Boer, R., [2002]. UvA Trilearn. Software.
`http://carol.wins.uva.nl/ jellekok/robocup/`, October 2003

[6] Kok, J. R., de Boer, R., and Vlassis, N., [2002]. Towards an optimal scoring policy for simulated soccer agents. In Gini, M., Shen, W., Torras, C., and Yuasa, H., editors, *Proc. 7th Int. Conf. on Intelligent Autonomous Systems*, 195–198. Marina del Rey, California: IOS Press.

[7] Nishino, J., Morishita, T., and Kubo, T., [2001]. Open Zeng: An Open Style Distributed Semi-cooperative Team Development Project from Japan. In Stone, P., Balch, T., and Kraetzschmar, G., editors, *RoboCup-2000: Robot Soccer World Cup IV*, vol. 2019 of *LNCS*, 473. Springer.

[8] O'Donoghue, T., and Rabin, M., [2000]. Risky Behavior Among Youths: Some Issues from Behavioral Economics. Technical Report E00'285, UC Berkeley Department of Economics.

[9] Ritter, H., Martinetz, T., and Schulten, K., [1994]. *Neuronale Netze*. Addison-Wesley.

[10] Stone, P., [2000]. *Layered Learning in Multiagent Systems: A Winning Approach to Robotic Soccer.* MIT Press.

[11] Stone, P., and Veloso, M., [1998]. A layered approach to learning client behaviors in the RoboCup soccer server. *Applied Artificial Intelligence*, 12.

[12] Sutton, R. S., and Barto, A. G., [1998]. *Reinforcement Learning*. Cambridge, Mass.: MIT Press.

[13] Tishby, N., Pereira, F. C., and Bialek, W., [1999]. The Information Bottleneck Method. In *Proc. 37th Annual Allerton Conference on Communication, Control and Computing, Illinois*.

[14] Vapnik, V., [1995]. *The Nature of Statistical Learning Theory*. New York: Springer.

[15] Vollbrecht, H., [1998]. Three Principles of Hierarchical Task Composition in Reinforcement Learning. In Niklasson, L., Bodén, M., and Ziemke, T., editors, *Proc. of the 8th International Conference on Artificial Neural Networks, Skövde, Sweden, 2-4 September 1998*, vol. II, 1121–1126. Springer.

[16] Watkins, C. C. J. H., and Dayan, P., [1992]. Q-learning. *Machine Learning*, 8(3):279–292.

[17] Zahavi, A., [1975]. Mate selection - A selection for a handicap. *Journal of Theoretical Biology*, 53:205–213.