How should a robot discriminate between objects?

de Jong, E.; Vogt, P.A.

Published in:
From animals to animats 5

Publication date:
1998

How Should a Robot Discriminate Between Objects? A comparison between two methods

Edwin de Jong, Paul Vogt
Vrije Universiteit Brussel
Pleinlaan 2, B-1050 Brussels, Belgium
{edwin, paul}@arti.vub.ac.be

Abstract

In order to discriminate among the different objects in its environment, an agent may develop a primitive notion of concepts based on the sensor data it receives. In this paper, this phenomenon is investigated by having software agents play *discrimination games* with the sensor data of autonomous robots. We have compared the Simple Prototype method and the Adaptive Subspace method. Both methods achieve high discrimination-success rates. The Adaptive Subspace method accomplishes this with a converging and relatively low number of categories. The purpose of these discrimination games is to serve as a basis for lexicon formation experiments. From the experiments in this paper, we conclude that the Adaptive Subspace method is more attractive for discrimination games.

1. Introduction

When a robotic agent investigates its environment it may, depending on its task, need to discriminate between the different objects it encounters. Deciding whether or not two observations concern the same object implies a categorization of the world. Provided a robot has appropriate and sufficiently precise sensors at its disposal, it can classify observations of the objects surrounding it into a number of different categories. We investigate this by letting agents play *discrimination games*. The discrimination games the robots play and the theory behind these one-player games were introduced in (Steels, 1996b); see also (Vogt, 1997). These games are investigated in the context of experiments on lexicon formation, for an overview see (Steels, 1997).

In other experiments at our lab called *naming games* (Steels and Vogt, 1997)(Steels, 1996a), the goal is to let robots communicate about their environment. A problem for these robots is that repeated encounters with the same object generally do not yield identical observations. The differences are caused by changes of the viewing angle, different lighting conditions, reflections, or sensor inaccuracies. In spite of these differences, an agent should categorize its observations such that an object consistently falls into the same category on separate encounters. If this is not the case, the development of meaningful communication about the objects will be severely hindered. Thus, in these experiments, a meaningful conceptualization of the environment will yield a limited number of categories; the smaller the number of categories, the larger the possibility that two agents will each consistently label an object as belonging to a certain category. This in turn facilitates the development of names for such categories which can be used in communication to identify objects. The number of possible categories to be considered grows rapidly with the number of sensors.

We are interested in the question of how a robot should discriminate between objects. In these experiments, what constitutes an object is directly based on the robot's sensory information. This usually, but not always, corresponds to a light source. The agent can autonomously determine whether its categorization allows it to discriminate between the objects in a context. Thus, discrimination success can be calculated by the agent itself, and is not externally given as is the case in reinforcement learning problems. Since no categorization is favored over other categorizations (as is the case with classification), the problem here is unsupervised learning. Many methods for unsupervised learning exist; in this paper, we investigate two possible methods. Both methods adapt the features which determine the agent's categories. Arguments from a cognitive perspective in favor of feature adaptation as opposed to fixed features are discussed in (Schyns et al., 1998).

The first method is based on *Simple Prototypes*. A prototype is a type of feature that can be represented by a sensor-value pair. New prototypes are created whenever discrimination fails. The second method is an adaptive subspace method that resulted from an attempt to combine the Exploration Buckets algorithm (De Jong, 1997) with generalization. This method generalizes over the sensor space by distinguishing only between features that enhance discrimination.

2. The robots and their environment

The experiments are carried out with the two mobile robots shown in figure 1. These robots are controlled by the sensory-motor board SMBII that was developed at the VUB AI-Lab (Vereertbrugghen, 1996). The robots are equipped with several sensors and actuators, as well as a radio-link. The sensors include a white light sensor, a modulated light sensor and three infrared (IR) sensors. One sensor of each type is mounted on the front of the robot and is used for perception. A virtual sensor is introduced to denote the robot itself, since one of the robots may be the subject of a naming game as well. Furthermore, the robots have IR sensors on both sides that are used for IR-taxis and -orientation (Steels and Vogt, 1997) and four bumpers for obstacle avoidance. The actuators include four IR emitters to make each robot visible to the other and two motors to produce the movements of the robots.

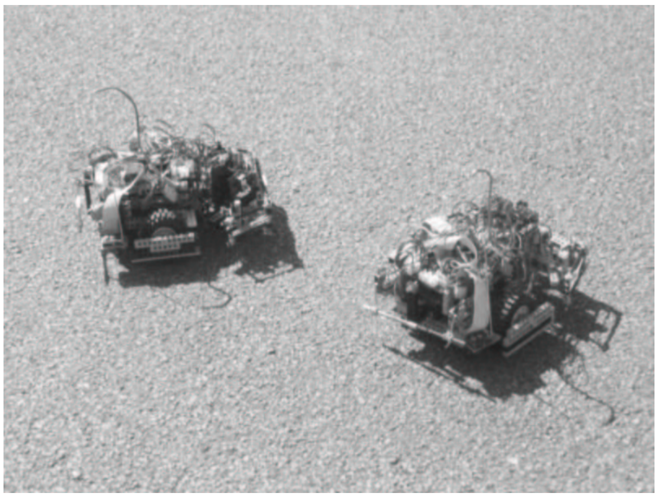

Figure 1 The robots used in the experiments.

In the experiments, the robots engage in a series of so-called naming games. As part of the naming game, the robots perform a perception task. This involves rotating 360 degrees and detecting objects using sensors as explained below, thus obtaining a context for the naming game. The objects the robots can detect are made visible by different light sources; the objects include the robots themselves. The light sources correspond to the sensors: there is a white light source, a source emitting light at a modulated frequency and an infrared source. During the perception task, the robots scan their sensors, which are pre-filtered by a threshold to filter out background noise.

The first object in the context is a virtual detection of the robot itself. When a robot encounters an object as it turns, one or more of the sensors will increase until a maximum is reached, after which the sensor value(s) will drop again. Rotating 360 degrees results in a graph like that shown in figure 2. A bounding box is drawn around every hill in the graph in the interval for which the sensor was activated. Since the robot turns (approximately) 360 degrees, the graph is a panoramic view. Therefore, if a sensor is active at both the beginning and the end of the graph, they should be interpreted as a single object.

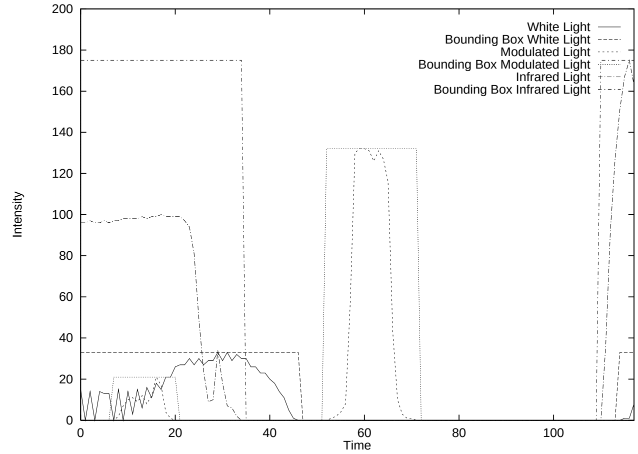

Figure 2 An object as perceived by the robot.

The two outer bounding boxes for infrared in figure 2 are thus part of the same bounding box. A bounding box is interpreted as an object located in the middle of that box. If the positions of two boxes detected are within a distance of 5 time steps, these boxes are considered to represent one object.

For every object, two values for every sensor are obtained as sensory channels. The first one is the filling ratio of the detected hill inside the bounding box and the second value is the height of the hill. The representation for the robot itself is added at the beginning of the context. For reasons of consistency, this virtual sensor also has two sensory channels, thus making the total number of sensory channels 8. This method for object detection differs from previous work. The resulting context contains the detected objects with values for the 8 sensory channels. This data is transmitted to a radio station linked to a PC for further off-line processing.

3. Discrimination Games

In the perception task, a robot has gathered sensor data of the objects it has observed in its environment. Every object can be related to a set of features that the agent has constructed. These objects together constitute the *context*. The agent then chooses one object from the context, called the *topic*. The aim of the discrimination game is to distinguish the topic from every other object in the context. If this discrimination is possible, the game ends in success. If not, it ends in failure, and the agent will try to improve its ability to discriminate by generating or modifying its features. The two methods that have been investigated, take different approaches to relating objects to features and for adapting features. These methods are described in the following two sections.

With the two methods discussed, the agent has a set of features that it adapts based on its experiences in order to increase its chances of success in the discrimination games. The features are quite primitive, in the sense that they directly correspond to a subset of the values of a certain sensory channel. If an object's value v for that sensory channel is a member of this set, the feature is said to *cover* v. Features can be combined into *feature sets*. A feature set covers an object if all features in the set cover the value of the feature's sensory channel for that object. For both methods, two features in a feature set never correspond to the same sensory channel. Thus, the number of features in a feature set is never greater than the number of sensory channels. A feature set is *discriminative* iff it covers the topic but none of the other objects in the context. The discrimination game results in a set of distinctive feature sets. This set may be empty, in which case the discrimination game fails and the system is adapted, as described in the following sections.

4. The Simple Prototype Method

The Simple Prototype method is based on the generation of features which can be viewed in some sense as prototypes. These prototypes should not be confused with the prototypes that are normally used in AI, which are frames for whole objects. The prototypical features discussed here act at the subsymbolic level. Sensor values observed for objects are mapped onto an appropriate prototypical feature. These features are compared in the discrimination games in order to distinguish the topic from other objects in the context. If the discrimination game is unsuccessful, a new feature may be generated for which a sensory channel of the topic is an example. The successful feature sets remain in the memory of the agents. In this way the features are selected for their discriminative powers. So, the method constitutes a selectionist approach; see also (Popescu-Belis, 1997), where Edelman's theory of neuronal group selection (Edelman, 1987) is used. Note that the method is different from the one used in previous work (Vogt, 1997)(Steels, 1996b).

Now, suppose an agent that has a set of prototypical features $\{f_i =< sc_i, v_i, s_i, u_i >\}$, where the sc_i is the attribute, which is the sensory channel on which the feature works, v_i is a value, s_i is the success score and u_i is the use score for f_i. Suppose furthermore that the agent has a context C of objects o_i, where every object has a sensorvalue $o_{i,j}$ pairs for every sensory channel j. The context is obtained from the perception of the robot. The agent has chosen one of the objects from the context to be the topic t. The method can now be described as follows:

- Every object $o_k \in C$ is covered by a set of features

  $$\Lambda^{o_k} = \{f_i \mid \forall f_j : (((f_j \neq f_i) \wedge (sc_j = sc_i = l)) \rightarrow (|v_i - o_{k,l}| \leq |v_j - o_{k,l}|)) \wedge (v_i > 0)\}$$

  where $o_{k,l}$ is the detected value for sensory channel l. This method for covering is shown schematically in figure 3.

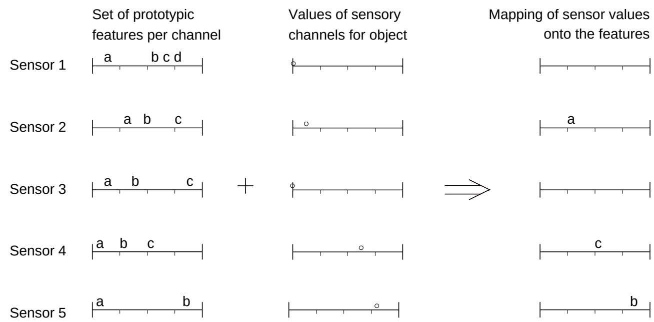

Figure 3 Determining which features cover an object using the Simple Prototype method

- The agent tries to distinguish topic t from the other objects in the context. The result of the discrimination is the set of distinctive feature sets:

  $$D^t = \{D_n^t \mid D_n^t \subseteq \Lambda^t \wedge \forall o \in C \backslash \{t\} : (\exists f_j \in D_n^t : \neg(f_j \in \Lambda^o))\}$$

  Every D_n^t is associated with a pair of $< s_n, u_n >$, where s_n is a success score and u_n is a usage score. The usage u_n for all $D_n^t \in D^t$ is incremented.
- If $|D^t| \geq 1$, we prefer the set D_n^t for which:
  (1) $\forall D_m^t \in D^t : |D_n^t| \leq |D_m^t|$, or
  (2) if this holds for more than one n, we prefer the set for which holds: $s_n/u_n \geq s_m/u_m$ for all $m \neq n$. The success score s_n is incremented for the chosen D_n^t.
- If $D^t = \emptyset$, the features need to be adapted. A new feature $f =< sc, v >$ is generated, so that $sc = sc_i$ and $v = o_{t,i} > 0$ for an arbitrary sensory channel i.

Initially, every agent has exactly one feature for every sensory channel, with a value that is in the middle of the range for that sensory channel.

Let us look at the example given in table 1. Agent r1 has observed 5 objects **o0**,. . .,**o4**. Object **o0**, for example, has value 1 for sensory channels 0 and 1 and value 0 for the remaining channels, and **o1** has values 200 and 1 for sensory channel 4 and 5 respectively.

Feature sets r1:
o0: {sc0-1,sc1-1}
o1: {sc4-127,sc5-127}
o2: {sc2-127,sc3-127}
o3: {sc4-66,sc5-127}
o4: {sc2-127,sc3-127,sc6-127,sc7-127}

In this example, *sc0-1* is a prototypical feature, where *sc0* is the sensory channel and the feature has the value 1; use and success scores are not given. For each object, the features are determined. These together constitute the object's feature set. Object **o0** is covered by features *sc0-1* and *sc1-1*, and object **o1** is covered by *sc4-127* and *sc5-127*. Now, suppose that the agent has chosen object

o1 to be the topic. Since the feature set that covers this object does not cover any other object, this feature set is discriminative. Furthermore, feature *sc4-127* does not cover any other object and is therefore also discriminative. On the other hand, we see feature *sc5-127* in **o3**, and this is not a distinctive feature set. The game is successful and since {*sc4-127*} is the smallest set, this set is preferred.

Now suppose object **o2** was the topic. In this case both features that cover **o2** can also be found in the feature set that covers **o4**. The distinctive feature set would be empty, and a new feature would be made. The feature that would be made is either *sc2-245* or *sc3-57*, since these are the only non-zero sensor values for **o2**.

Since a discrimination game may yield several distinctive feature sets, the number of feature sets may be larger than necessary. This increases the search space considerably, and therefore it is attractive to forget unsuccessful feature sets. Starting from discrimination game 420, every 30 games the agent starts to forget those feature sets that sofar have not been successful ($s = 0$), and that have been acquired more than 300 games before. This way, the unsuccessful feature sets may have the opportunity to become successful.

sc	0	1	2	3	4	5	6	7
o0	1	1	0	0	0	0	0	0
o1	0	0	0	0	200	1	0	0
o2	0	0	245	57	0	0	0	0
o3	0	0	0	0	57	3	0	0
o4	0	0	245	57	0	0	113	130

Table 1 An example of a context with 5 objects. The table lists the values of 8 sensory channels for each object.

5. The Adaptive Subspace Method

The Adaptive Subspace method is based on the principle that different sets of sensory inputs should only be treated as different if this distinction is meaningful. Which distinctions are meaningful is determined by the application. In the case of discrimination games, a distinction should be made if and only if this increases the ability of the agent to discriminate the topic from the other objects, in one or more of the discrimination games it has played. This method results from research on generalization, an important issue in machine learning. An overview is beyond the scope of this paper, but for interesting contributions on generalization, including other subspace methods, see (Landelius, 1994; Murao and Kitamura, 1997; McCallum, 1996). In this paper, orthogonal splits were used; an interesting variation would be to allow non-orthogonal splits as well. Oja discusses subspace methods where, unlike here, subspaces have a lower dimensionality than the original data (Oja, 1983).

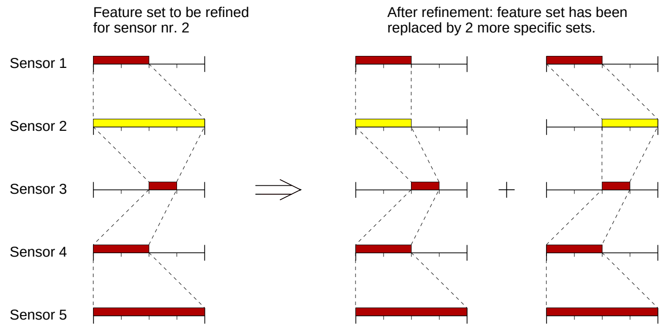

Figure 4 Refining a feature set using the Adaptive Subspace method.

When a discrimination game is played, the agent determines for each object it has observed, by which feature set it is covered. If the feature set of the topic covers no other object, the agent can discriminate the topic from the other objects, and the game is a success. When a game fails, the feature set of the topic is investigated to see whether it should be refined. Refining a feature set amounts to splitting one of the ranges in half and replacing the existing feature set with two new feature sets, one for each half. This is illustrated in figure 4. We will now describe the refinement procedure in greater detail, and then give an example.

- For sensory channel j, feature F_j is a tuple of two integers $< F_{j,l}, F_{j,u} >$, which represent the lower and upper values of an interval. An object o_i is represented by its values for the n sensory channels $o_{i,j}$, where $j \in [0 \ldots n-1]$. F_j covers $o_{i,j}$ if $F_{j,l} \leq o_{i,j} \leq F_{j,u}$. This is written as $covers(F_j, o_{i,j})$. Every feature set contains a feature for each sensory channel. A feature set $F = <F_0, F_1, \ldots F_{n-1}>$ covers an object o_i if $\forall j \in [0 \ldots n-1] : covers(F_j, o_{i,j})$.
- Every point in the statespace is covered by exactly one feature set. This property is not lost during a refinement. The method thus adapts a division of the statespace into regions, hence the name. The contexts C here contains all the objects in a discrimination game except the topic t. The success ratio s is the fraction of the objects in C that are not covered by the feature set F that covers t, and can thus be distinguished from the topic:

$$s = \frac{|\{o | o \in C \wedge \neg covers(F, o)\}|}{|C|}$$

- For each dimension, we compute the success increase that a split in that dimension would yield. After a split, F is replaced by 2 new feature sets FA and FB. The success increase Δs_d for a split in dimension d is calculated as follows:

$$m_j = \frac{F_{j,l} + F_{j,u}}{2}$$

$$FA_j = \begin{cases} <F_{j,l}, F_{j,u}> & if \quad j \neq d \\ <F_{j,l}, m_j> & if \quad j = d \end{cases}$$

$$FB_j = \begin{cases} <F_{j,l}, F_{j,u}> & if \quad j \neq d \\ <m_j, F_{j,u}> & if \quad j = d \end{cases}$$

$$\Delta s_d = \begin{cases} \frac{|\{o|o \in C \wedge covers(FB,o)\}|}{|C|} & if \quad covers(FA,t) \\ \frac{|\{o|o \in C \wedge covers(FA,o)\}|}{|C|} & if \quad covers(FB,t) \end{cases}$$

This is the fraction of the objects in the context (not including the topic) that can be discriminated from the topic after, but not before a split in dimension d.

– The highest success increases of the past h games are stored in a history window named sh. If the inequality

$$\max_d \ \Delta s_d \geq \max_h \ sh_h$$

holds, the split in the corresponding dimension is actually performed. Sometimes,

$$\max_d \ \Delta s_d = 0 \wedge s < 1.0$$

This means no split would have yielded an increase of the success for this discrimination game, but discrimination is nevertheless imperfect. This is the case when all of the objects in the topic's feature set have all of their sensor values in the same halve of the corresponding feature's interval as the topic. In that situation, a *distance split* is performed. During a distance split, the feature set covering the topic is split in the dimension dim with the highest average distance between the sensor values of the objects covered by the feature set and the corresponding sensor value of topic t:

$$dim = arg \max_{j \in [0 \ldots n-1]} \sum_{i \in [1 \ldots |C|]} \frac{||o_{i,j} - t_j||}{i}$$

It is crucial that the refinement in the range of the sensory channel is made only for the current feature set (= subspace). This is what allows the method to handle the higher-dimensional statespaces that result when several sensory channels are used (8 sensory channels or dimensions in these experiments).

We now give an example of discrimination games using the Adaptive Subspace method. Let us assume agent r1 has the following feature sets:

$$FS1 = [0,255]\ [0,255]\ [0,255]\ [0,255]\ [0,255]\ [0,255]\ [0,127]\ [0,255]$$
$$FS2 = [0,255]\ [0,255]\ [0,255]\ [0,255]\ [0,255]\ [0,255]\ [128,255]\ [0,255]$$

Furthermore, suppose the agent chooses **o4** (see table 1) to be the topic. Since all objects have a value below 128 for sensory channel 6, all objects are covered by FS1. Thus, none of the 4 objects in the context can be discriminated from the topic, yielding a success score of $\frac{0}{4} = 0$. Since discrimination fails, the agent considers splitting the subspace of the topic, FS1, for each dimension. Dimensions 0 through 5 yield no improvement, but if FS1 is split in dimension 6, it would be replaced by the following two feature sets:

$$FS1A = [0,255]\ [0,255]\ [0,255]\ [0,255]\ [0,255]\ [0,255]\ [0,63]\ [0,255]$$
$$FS1B = [0,255]\ [0,255]\ [0,255]\ [0,255]\ [0,255]\ [0,255]\ [64,127]\ [0,255]$$

After this split, the topic would be covered by FS1B, but all 4 other objects would be covered by FS1A. Thus, the new success is $\frac{4}{4} = 1$. If the success increase (1 - 0 = 1.0) is greater than or equal to the maximal success increase in the history window, the split is actually performed.

6. Results

In the experiments described here, the Simple Prototype method and the Adaptive Subspace method have been applied to discrimination games with the same robot data. The data set contains 321 contexts. For each context the data contains the sensory channel values of the objects. During an experiment, the sequence of contexts encountered by a robot is repeatedly presented to an agent in the original order. In this section we describe the results of these experiments, and compare the two methods. The robot data has been collected in runs of about 45 minutes (a period limited by the robot's battery). To collect the data, the robots have both been active for 9 hours in total.

Figure 5 shows the success of the Simple Prototype method. The initial success is already quite high. It increases and then continuous to vary, mostly between 0.87 and 1. Figure 6 shows the success of the Adaptive Subspace method. The success steadily increases over time, and a very high performance is attained; after 4,000 games the average success varies mostly between 0.96 and 1. Figure 7 shows the total number of feature sets used as a function of the number of games played for both methods. For the Simple Prototype method, the number of feature sets is rather capricious, and does not seem to converge during the 10,000 games that have been observed. With the Adaptive Subspace method the number of feature sets used does converge, and is substantially smaller.

7. Conclusions

Two methods for discrimination between objects based on sensor data have been compared. Both methods are based on the principle that an agent adapts its categorization of the world to increase its ability to distinguish different objects from each other. The development

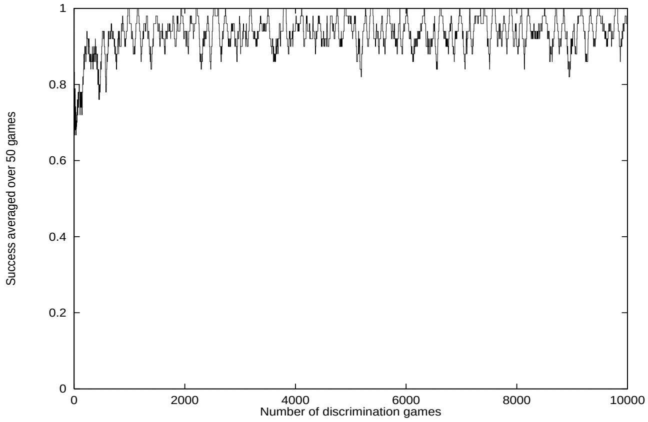

Figure 5 Success using the Simple Prototype method.

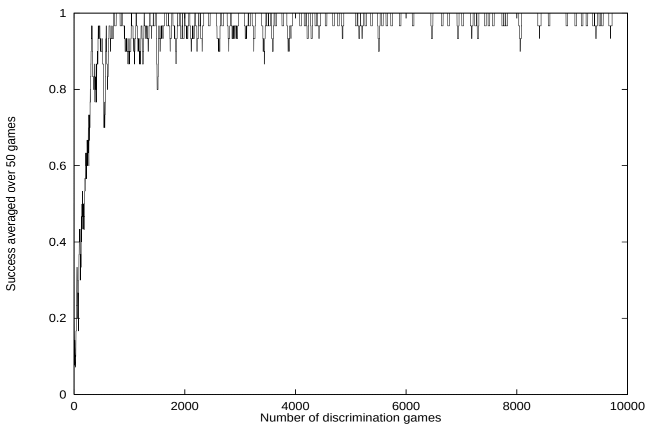

Figure 6 Success using the Adaptive Subspace method.

of discrimination success for both methods makes clear that this objective is attained. Since it is essential that the number of feature sets is as low as possible, the development of this number over time has also been measured. The Simple Prototype method yielded a high success rate, but the number of feature sets does not appear to converge within 10,000 games. The Adaptive Subspace method refines a feature set only after some consideration. This yields an even higher success rate, and has the advantage that the number of feature sets converges to a relatively low number. The question we addressed in this paper is how a robot should discriminate between objects. From the results, we have seen that the Adaptive Subspace method is more attractive with respect to both aims, i.e. a high success rate and a low number of feature sets.

Acknowledgments

The authors want to thank Ruth Aylett, Dave Barnes, Bart de Boer, Michiel de Jong, and Will Lowe for various comments and suggestions.

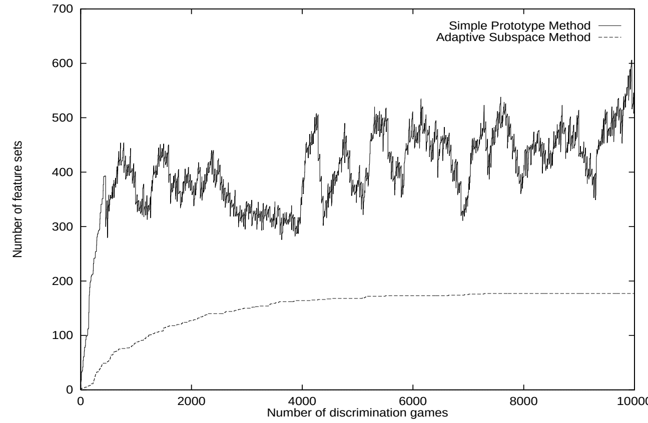

Figure 7 Number of feature sets.

References

De Jong, E. D. (1997). An accumulative exploration method for reinforcement learning. In *Proceedings of the AAAI'97 Workshop on Multiagent Learning, available as AAAI Technical Report WS-97-03.*

Edelman, G. (1987). *Neural Darwinism.* Basic Books Inc., New York.

Landelius, T. (1994). Behaviour representation by growing a learning tree. Master's thesis, Linkoping University.

McCallum, A. K. (1996). Reinforcement learning with selective perception and hidden state. *Ph.D. Thesis.*

Murao, H. and Kitamura, S. (1997). Q-learning with adaptive state space construction. In Birk, A. and Demiris, J., editors, *6th European Workshop on Learning Robots.*

Oja, E. (1983). *Subspace Methods of Pattern Recognition.* Research Studies Press Ltd.

Popescu-Belis, A. (1997). Design of an adaptive multi-agent system based on the "neural darwinism" theory. In *First International Conference on Autonomous Agents*, pages 484–485, Marina del Rey, California.

Schyns, P. G., Goldstone, R. L., and Thibaut, J.-P. (1998). The development of features in object concepts. *Behavioral and Brain Sciences*, In Press.

Steels, L. (1996a). Emergent adaptive lexicons. In Maes, P., editor, *From Animals to Animats 4: Proceedings of the Fourth International Conference On Simulating Adaptive Behavior*, Cambridge MA. The MIT Press.

Steels, L. (1996b). Perceptually grounded meaning creation. In Tokoro, M., editor, *Proceedings of the International Conference on Multi-Agent Systems*, Menlo Park CA. AAAI Press.

Steels, L. (1997). The synthetic modeling of language origins. *Evolution of Communication*, 1(1):1–34.

Steels, L. and Vogt, P. (1997). Grounding adaptive language games in robotic agents. In Husbands, C. and Harvey, I., editors, *Proceedings of the Fourth European Conference on Artificial Life*, Cambridge MA and London. MIT Press.

Vereertbrugghen, D. (1996). Design and implementation of a second generation sensor-motor control unit for mobile robots. Master's thesis, VUB.

Vogt, P. (1997). Perceptual grounding in robots. In Birk, A. and Demiris, J., editors, *6th European Workshop on Learning Robots.*