Approved for Public Release; Distribution Unlimited
Case # 05-0674

Adaptive value function approximations in classifier systems

Lashon B. Booker
The MITRE Corporation
7515 Colshire Drive
McLean, VA 22102-7508, USA
booker@mitre.org

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning—*Induction, Parameter Learning*

General Terms

Algorithms, Design, Experimentation, Performance, Theory

Keywords

Learning Classifier Systems, Reinforcement Learning, Function Approximation, Tile Coding, Hyperplane Coding

Extended Abstract

Considerable attention has been paid to the issue of value function approximation in the reinforcement learning literature [3]. One of the fundamental assumptions underlying algorithms for solving reinforcement learning problems is that states and state-action pairs have well-defined values that can be approximated and used to help determine an optimal policy. The quality of those approximations is a critical factor in determining the success of many algorithms in solving reinforcement learning problems.

In most classifier systems, the information about the value function is stored and computed by individual rules. Each rule maintains an independent estimate of the value of taking its designated action in the states that match its condition. From this standpoint, each rule is treated as a separate function approximator. The quality of the approximations that can be achieved by simple estimates like this is not very good. Even when those estimates are pooled together to compute a more reliable collective estimate, it is still questionable how good the overall approximation will be. It is also not clear what the best way is to improve the quality of those approximations.

One approach to improving approximation quality is to increase the computational abilities of individual rules so that they become more capable function approximators [4]. Another idea is to look back to the original concepts underlying the classifier system framework and seek to take advantage of the properties of distributed representations in classifier systems [2]. This paper follows in the spirit of the latter approach, looking for ways to tap the distributed representational power present in a collection of rules to improve the quality of value function approximations.

Previous work [1] introduced a new approach to value function approximation in classifier systems called *hyperplane coding.* Hyperplane coding is a closely related variation of tile coding [3] in which classifier rule conditions fill the role of tiles, and there are few restrictions on the way those "tiles" are organised. The basic idea is to treat rules as features that collectively specify a linear gradient-descent function approximator. The hypothesis behind this idea is that classifier rules can be more effective as function approximators if they work together to implement a distributed, coarse-coded representation of the value function.

Experiments with hyperplane coding have shown that by carefully using the resources available in a random population of classifiers, continuous value functions can be approximated with a high degree of accuracy. This approach computes much better approximations than more conventional classifier system methods in which individual rules compute approximations independently. The results to date also demonstrate that hyperplane coding can achieve levels of performance comparable to those achieved by more well-known approaches to function approximation such as tile coding. High quality value function approximations that provide both data recovery and generalisation are a critically important component of most approaches to solving reinforcement learning problems. Because hyperplane coding substantially improves the quality of the approximations that can be computed by a classifier system using relatively small populations of classifiers, it may provide the foundation for significant improvements in classifier system performance.

One open question remaining about hyperplane coding is how the quality of the approximation is affected by the set of classifiers in the population. A random population of classifiers is sufficient to obtain good results. Would a more carefully chosen population do even better? The obvious next step in this research is to use the approximation resources available in a random population as a starting point for a more refined approach to approximation that reallocates resources adaptively to gain greater precision in those regions of the input space where it is needed. This paper shows how to compute such an adaptive function approximation. The goal is learn a population of classifiers that

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
GECCO'05, June 25–29, 2005, Washington, DC, USA.
Copyright 2005 ACM 1-59593-097-3/05/0006 ...$5.00.

reflects the structure of the input space (Dean & Wellman, 1991). This means more rules (ie. more tiles) should be used to approximate regions which are sampled often and in which the function values vary a great deal. Fewer rules should be used in regions which are rarely sampled and in which the function is nearly constant. We discuss how to adaptively manage the space in the population, as well as how to structure the search for tiles that reduce the approximation error.

† Acknowledgements

This work is based on research originally funded by the MITRE Sponsored Research program. That support is gratefully acknowledged. The author's affiliation with The MITRE Corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions or viewpoints expressed by the author.

1. REFERENCES

[1] Lashon B. Booker. Approximating value functions in classifier systems. In Larry Bull and Tim Kovacs, editors, *Foundations of Learning Classifier Systems.* Springer, 2005.

[2] Lashon B. Booker, David E. Goldberg, and John H. Holland. Classifier Systems and Genetic Algorithms. *Artificial Intelligence*, 40:235–282, 1989.

[3] Richard S. Sutton and Andrew G. Barto. *Introduction to Reinforcement Learning.* MIT Press, Cambridge,MA, 1998.

[4] Stewart W. Wilson. Classifiers that approximate functions. *Natural Computing*, 1(2-3):211–234, 2002.