An Anticipatory Approach to Improve XCSF

Amin Nikanjam
Iran University of Science and Technology
Computer Engineering Department, Narmak
Tehran 16846_13114, Iran
21 22312494, +98
nikanjam@comp.iust.ac.ir

Adel Rahmani
Iran University of Science and Technology
Computer Engineering Department, Narmak
Tehran 16846_13114, Iran
21 73913324, +98
rahmani@iust.ac.ir

ABSTRACT
XCSF is a novel version of learning classifier systems (LCS) which extends the typical concept of LCS by introducing computable classifier prediction. In XCSF Classifier prediction is computed as a linear combination of classifier inputs and a weight vector associated to each classifier. Learning process takes place using a weight update mechanism. Initial results show that XCSF can be used to evolve accurate approximations of some functions. In this paper, we try to add an anticipatory component to XCSF improving its performance.

Categories and Subject Descriptors
I.2.6 [**Learning**].

General Terms
Algorithms, Experimentation.

Keywords
Learning classifier system, Anticipatory classifier system, XCSF, Function approximation.

1. INTRODUCTION
XCSF [7] extends the typical concept of learning classifier systems (as XCS [6]) through the introduction of a computable classifier prediction. In XCSF, classifier prediction is not memorized into a parameter but computed as a linear combination of the current input and a weight vector associated to each classifier. Wilson [7] applied XCSF to simple function approximation problems showing that computable prediction can be used to evolve accurate piecewise linear approximations of target function. In [3] authors had extended the XCSF beyond linear approximation showing promising results. At the other hand Anticipatory classifier system (ACS) [5] brought new ideas about anticipation in the area of learning classifier systems. In this paper we try to explain a method for adding the idea of anticipation into XCSF in order to improve its performance.

2. DISCRIPTION OF XCSF
XCSF [4] is a model of learning classifier system that extends the typical concept of classifiers through the introduction of a computed classifier prediction. To develop XCSF, XCS has to be modified in three respects: (i) classifier conditions are extended for numerical inputs; (ii) classifiers are extended with a vector of weights $\vec{w}$ that is used to compute the classifier prediction; finally, (iii) the original update of the classifier prediction must be modified so that the weights are updated instead of the classifier prediction.

These three modifications result in a version of XCS, XCSF that maps numerical inputs into actions with an associated calculated prediction.

3. ACS IN BRIEF
In [5] the basic structure of the ACS with its anticipatory learning process (ALP) was introduced. In [2] an enhancement of the application of the ALP and a genetic algorithm (GA) was introduced to the ACS.

3.1 The Basic Structure
An ACS [1] always interacts with an environment. At each time step t it perceives a state $\sigma(t) \in \{s_1, s_2, ..., s_m\}^L$, executes an action $\alpha(t) \in \{r_1, r_2, ..., r_n\}$ and receives payoff $\rho(t) \in \Re$. The ACS presents its knowledge in rules, which are called classifiers. A classifier consists out of a condition part C (state of environment), an action part A, an effect part E (anticipation of next state) and a mark M.

Learning in the ACS starts always with a completely general knowledge. A behavioral act at first makes a match set [M] from the current population considering the current state $\sigma(t+1)$. Next, the ACS decides with a probability of p_x whether to choose an action randomly or whether to choose a classifier by roulette-wheel selection with the bid $q*r$ in the match set and choose its action. Considering the action, an action set is formed. After the execution of the action, first the ALP (with respect to the resulting state $\sigma(t+1)$) and then the GA modify the action set as the main part of learning process. Finally, the reward measure r is updated considering the perceived payoff $\rho(t)$ and the next match set $[M](t+1)$.

4. XCSF WITH ANTICIPATION REWARD
In this section, we describe the idea and architecture of the proposed extension to XCSF. As we mentioned in previous section, in ACS when a classifier (or some classifiers) anticipates the result of an action correctly, ALP applies some modifications in the action set like increasing the Anticipation Quality of classifier(s). We shall use this idea in the XCSF to improve its performance. In fact we want to apply ALP in the XCSF.

GECCO'06, July 8–12, 2006, Seattle, Washington, USA.
ACM 1-59593-186-4/06/0007.

As it was described in section 2, each classifier in XCSF keeps a vector of weight $\vec{w}$ to compute the prediction of reward for any input. If we regard this vector and the calculation of reward as an anticipation of environmental reward, we can apply the ALP in XCSF. For this reason we add an ALP component to XCSF, the resulting system is called XCSA. In this component if a classifier in action set can anticipate the (environmental) reward correctly, the fitness of classifier (that estimates the accuracy of the classifier prediction) will be increased as the quality of anticipation by equation below:

$$cl.F = cl.F + \beta(1 - cl.F)$$

Where *cl.F* is the fitness of classifier *cl* and β is a learning rate. If such classifier can not anticipate the (environmental) reward correctly, the fitness of classifier will be decreased as in ACS:

$$cl.F = cl.F - \beta \times cl.F$$

Correct anticipation can be determined by absolute difference of reward (value of function) and calculated prediction (anticipation)

$$\textit{Anticipation is} \begin{cases} \textit{correct} & \textit{if } \left| r - cl.p(s_{t-1}) \right| < \theta_{ant} \\ \textit{incorrect} & \textit{if } \left| r - cl.p(s_{t-1}) \right| \geq \theta_{ant} \end{cases}$$

Where $cl.p(s_t)$ is the prediction of *cl* (classifier) computed in the state s_t, r is a reward from the environment and θ_{ant} is the *anticipation threshold*.

5. EXPERIMENTAL RESULTS

We now compare XCSF to XCSA. For this purpose, we have considered problems taken from the literature [4] and adapted them to integers, following the approach of [7] for the sine function. The functions that we use are reported below:

$$f_{ds}(x) = 100\sin(\frac{2\pi x}{100}),$$

$$f_{s3}(x) = 100\left(\sin(\frac{2\pi x}{100}) + \sin(\frac{4\pi x}{100}) + \sin(\frac{6\pi x}{100}) \right),$$

$$f_{s4}(x) = 100\left(\sin(\frac{2\pi x}{100}) + \sin(\frac{4\pi x}{100}) + \sin(\frac{6\pi x}{100}) + \sin(\frac{8\pi x}{100}) \right),$$

$$f_{abs}(x) = 100\left| \sin(\frac{2\pi x}{100}) + \left| \cos(\frac{2\pi x}{100}) \right| \right|.$$

Note that in all experiments $x \in [0,100]$.

Table 1 reports for each experiment (i) the average mean absolute error with the standard derivation (column $\overline{MAE} \pm \sigma$), (ii) the average mean square error with standard derivation (column $\overline{MSE} \pm \sigma$) for XCSF and XCSA.

6. CONCLUSION

As it was mentioned before we use MAE and MSE for performance measurement. In fact, these criteria indicate the average approximation error as difference between target function and approximated function. All values reported in table 1 are averaged over 50 run.

It was clear that all average errors reported in table 2 are below *error threshold* (e_0 = 5). It means that XCSF and XCSA both can approximate functions accurately. But as can be noted, XCSA (XCSF with Anticipation reward) evolves more accurate solution, in fact, the average error (MAE and MSE) are usually smaller than XCSF. This happened because the anticipation reward makes XCSA to be conducted to a set of accurate, most general classifiers sooner than XCSF.

Although work on the ALP and research in anticipatory systems, in general, is just in its beginning, we hope that our research about using anticipation in the area of learning classifier systems and results are helpful to this line of research.

Table 1. XCSF vs. XCSA

f(x)	XCSF		XCSA	
	MAE±σ	MSE±σ	MAE±σ	MSE±σ
f_{ds}	0.24±0.11	0.18±0.18	0.24±0.11	0.18±0.14
f_{s3}	2.36±0.73	16.04±23.91	2.06±0.52	11.08±8.43
f_{s4}	4.02±1.3	75.15±156.78	3.95±1.17	39.97±29.96
f_{abs}	1.84±0.46	12.04±16.50	1.68±0.36	9.03±3.77

7. ACKNOWLEDGMENTS

This research has been supported by a research grant from the Iran University of Science and Technology (IUST).

8. REFERENCES

[1] Butz, M., Goldberg, D. E., Stolzmann, W., 2000a. *Probability-Enhanced predictions in the Anticipatory Classifier System*, Technical Report No. 2000016, Illinois Genetic Algorithms Laboratory.

[2] Butz, M., Goldberg, D. E., Stolzmann, W., 2000b. *The Anticipatory Classifier System and Genetic Generalization*, Technical Report No. 2000032, Illinois Genetic Algorithms Laboratory.

[3] Lanzi, P. L., D. Loiacono, S. W. Wilson, and D. E. Goldberg 2005a. *Extending XCSF Beyond Linear Approximation*, Technical Report No. 2005006, Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign.

[4] Lanzi, P. L., D. Loiacono, S. W. Wilson, and D. E. Goldberg, 2005d. *Generalization in the XCSF Classifier System: Analysis, Improvement, and Extension*. Technical Report No. 2005012, Illinois Genetic Algorithms Laboratory.

[5] Stolzmann, W. 1998., *Anticipatory classifier systems*, in *Genetic* Programming *'98* (pp. 658-664), University of Wisconsin, Madison, Wisconsin: Morgan Kaufmann.

[6] Wilson, S. W., 1995. *Classifier Fitness Based on Accuracy*. Evolutionary Computation 3.

[7] Wilson, S. W., 2002. *Classifiers that approximate functions*. Journal of Natural Computations 1 (2-3), 211-234.