Chapter 8

Future work and conclusions

8.1 Suggestions for future work

There are several issues with regard to FOX which should be further investigated:

1. An algorithm for automatic synthesis of the A, B, and C matrices needs to be created. Such an algorithm would take as input a system's impulse response and a desired eligibility order. It would synthesize an eligibility model which is as accurate as possible within the trace precision requirements. A possible method is least squares minimization using a gradient search technique.

2. The most time consuming part of FOX controller design is the selection of the error functions and learning rates. Currently the learning rates are selected by trial and error. A more systematic method is desirable. Such a method would need information about the likely distribution of the error signal and the desired (approximate) magnitude of the FOX output. Methods to avoid overtraining should also be considered. Currently the best approach is to use overshoot training, as fewer parameters need to be guessed.

3. Other types of error function need to be investigated. The currently used ones (output limiting, overshoot etc) are useful but surely there are others.

4. As far as practical applications go, there obviously needs to be a lot more testing of FOX in real systems. More design experience is needed, and more design rules and guidelines should be formulated.

5. A FOX approximation theorem needs to be formalized. That is, exactly how much eligibility profile approximation can be tolerated for training to succeed? The current understanding of this issue is rather poor.

   An extension of this would be to show to what extent nonlinear systems can be controlled. Given a FOX approximation theorem, the way to proceed is to show that the nonlinear system local linear approximations are within the desired bounds in all useful regions of the FOX input space.

6. Currently FOX training must encompass all potentially useful regions of the CMAC input space. This is a limitation of the CMAC's local generalization, and it means that careful selection of the CMAC inputs and input parameters is required. FOX could be adapted to some more globally generalizing CMAC variant, for example a CMAC with adaptive input encoding. Also the properties

of multiple CMAC modules which each use a subset of the available inputs should be investigated (this is after all the way it works in the real cerebellum).

7. A better understanding of how to use one or more FOXs to service multiple constraints needs to be obtained. This is particularly true when the FOX eligibility profiles are approximations—how should they be chosen to prevent the independent error signals from simply having an additive effect on the FOX outputs?

8. FOX should be generalized to non-binary CMACs, and also to other neural network paradigms if this is possible.

9. The trace precision problem is a constant annoyance. Is there an implementation that avoids it?

10. FOX should be re-formulated for use with time-delay systems (this is a common class of system where the control signals are not manifested in the sensor readings until some minimum time has passed). Possibly this can be done by holding weight/eligibility values in "stasis" for the time delay period until their effect on the system is measurable, at which time they will be modified by the error signal. This would also be useful for modeling eligibility profiles with a significant initial zero portion.

11. A CMAC with a wrap-around (toroidial) weight table should be created as an alternative to using a two dimensional closed loop (circular) input trajectory as a means to avoid the local generalization discontinuity problem.

It has been demonstrated that FOX is a useful component in legged robot controllers, but obviously *much* more work needs to be done to explore the possibilities here. A much more comprehensive robot controller should be constructed that allows all aspects of movement to be learned, not just the handful of parameters for the single stereotyped walking motion studied here. In general a much better "brain design" approach is required. The existing methods using the DND language (see Appendix J) have problems when the system gets too large: many unforeseen interactions occur between system components, and the system also becomes hard to understand and visualize. Possibly a graphically based design approach needs to be used.

Finally, a robust biped brain needs to have far more built-in reflexes to compensate for a wider range of situations.

8.2 Conclusions

This thesis has developed the FOX adaptive controller from a theoretical consideration of the optimal control problem. FOX is based on the CMAC neural network, which is a simple model of the cerebellum, a part of the brain which helps control movement. FOX improves the cerebellar model by associating eligibility values with each CMAC weight. FOX is used in a configuration similar to the feedback-error (FBE) controller. With FBE an explicit reference trajectory must be generated to suit different situations. In contrast FOX will automatically find an error-minimizing trajectory that approaches a single reference path in a well defined way. It has also been shown that FOX can control some systems that FBE can not.

FOX implements a form of reinforcement learning which assigns each CMAC weight an eligibility value which controls how that weight is updated. Unlike other systems, FOX eligibilities can be vector quantities. The optimal eligibility update dynamics were shown to be the same as the system that is being controlled. Previous studies [68] have only provided loose guidelines for updating eligibilities.

A design methodology for using FOX in an adaptive control system was developed. The designer's expert knowledge about how a system should be controlled is split between an "internal" feedback controller added to the system and the choice of error signal (and error function) given to FOX. The system's impulse response is used to synthesize an eligibility profile, which determines how the eligibility values are updated. It was shown that a large amount of inaccuracy is allowable, so eligibility profile approximations can be made easily, although numerical precision issues limit the choices that can be made.

A highly efficient implementation was developed whose speed is independent of the number of CMAC weights and the controlled system parameters. This implementation overcomes the need to update each weight's eligibility at each time step. This is more sophisticated, faster and more useful than previous approaches such as [48].

FOX was successfully used in three real-time learning control system experiments, including the control of the classically difficult inverted pendulum and other nonlinear systems. It was discovered that FOX is easy to design with, and highly effective as long as training encompasses a large enough area of the state space. The most time consuming part of controller design was found to be the selection of error functions and learning parameters. Both feed-forward and feedback modes of control were investigated—feed-forward needs less CMAC inputs but requires known starting states, and feedback has more CMAC inputs but is more flexible when unanticipated states are encountered.

Finally, FOX was shown to be a useful learning component in two autonomous legged robot controllers—a simulated hopping monoped and walking biped. It was used to learn a handful of parameters and reference adjustments for hard-wired controllers, to coordinate their existing behavior. A cunning use of FOX gates, training signals and error functions was necessary. It was found that with a number of FOX modules in a system, a badly trained one could be disguised by all the well trained ones, so careful checking was required.

Successful walking was implemented in the biped robot. Despite the seeming sophistication of this walking behavior, the biped controller is not robust enough for real world walking because of its simplicity. The problem of controlling a biped which must survive in an unstructured environment may be too hard for the relatively simple design techniques presented here.

Finally, a lot of further work needs to be done to make FOX a practical industrial adaptive controller. FOX can be part of the toolbox for autonomous robot design, but of course many other techniques will be necessary too. Most of the techniques that will be useful in future designs have not even been discovered yet!