On-Board Evolution of a Model-Free Adaptive Controller for a Robotic Fish

Anthony Clark, Philip K. McKinley, and Xiaobo Tan

Michigan State University, East Lansing, MI, USA
ajc@msu.edu

Introduction Many physical systems experience fluctuating dynamics throughout their lifetime. Variations can be attributed in part to material degradation and decay of mechanical hardware. Designing control strategies that mitigate the negative effects of such variations can be difficult. One approach is to utilize model-free adaptive control (MFAC), which learns how to control a system by continually updating link weights of an artificial neural network (ANN) (Cheng, 2004). However, determining the optimal values of various control parameters, as well as the structure of the ANN, is challenging. In this study, we investigate how to enhance the on-board adaptability of MFAC-based systems through computational evolution.

Methodology We apply the proposed method to a robotic fish propelled by a flexible caudal fin. The differential evolution (DE) algorithm (Storn and Price, 1997) is utilized to optimize MFAC parameters; the MFAC controls a robotic fish, which can have changing fin characteristics (i.e. fin length and fin flexibility). It is the job of the MFAC to adapt to these variations, which are meant to mimic changes in the caudal fin material that occur after deployment, in order to realize effective locomotion. DE, a global optimization algorithm for real-valued problems, was chosen because studies have shown that it will converge faster than real-valued genetic algorithms for problems similar to ours.

Figure 1 shows a block diagram of the MFAC and the controlled robotic fish. The input to the entire system is a reference signal r, which can be any physical signal relating to the robotic fish. The signal r would most likely originate from a higher-level controller. For this study, r refers to a *desired* speed, and the output of the robotic fish y is the *actual* (measured) speed. The objective of an MFAC is to produce a control signal u such that y closely tracks r. That is, an effective controller will force the robotic fish to closely match the desired speed, and have little error e between y and r.

The motivating problem for this study is how to specify the MFAC parameters in such a way that they effectively allow the controller to adapt to differences in caudal fin be-

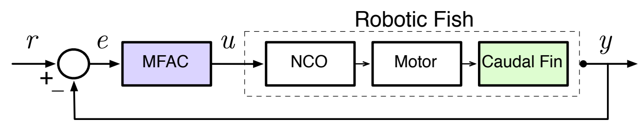

Figure 1: Block diagram of the MFAC and the robotic fish.

havior. Values for these parameters are typically set based on expert knowledge; however, our experiments show that more effective parameters can be found with the DE algorithm.

Discussion Preliminary results demonstrate that a robot controlled by an MFAC is able to adapt to subtle changes in material properties, as well as a range of different commands from a high-level controller. Changes beyond a certain extent, however, will cause an MFAC to lose its adaptability. For example, if the robot fish caudal fin is cut in half, the MFAC parameters may need to be re-evolved.

In the future, our research will investigate how our MFAC-based technique will benefit from both offline evolution and more complex on-board methods such as *self-modeling* (Bongard et al., 2006). Starting evolution offline will likely reduce the number of on-board evaluations needed to find an adequate set of MFAC parameters. Self-modeling will allow the entire system to handle larger changes, including component failures.

References

Bongard, J., Zykov, V., and Lipson, H. (2006). Resilient machines through continuous self-modeling. *Science*, 314(5802):1118–1121.

Cheng, G. S. (2004). Model-free adaptive (MFA) control. *Computing and Control Engineering*, 15(3):28–33.

Storn, R. and Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359.