Chapter 3.2
Combining Software and Hardware LCS for Lightweight On-chip Learning

Andreas Bernauer, Johannes Zeppenfeld, Oliver Bringmann, Andreas Herkersdorf, and Wolfgang Rosenstiel

Abstract In this article we present a novel two-stage method to realise a lightweight but very capable hardware implementation of a Learning Classifier System for on-chip learning. Learning Classifier Systems (LCS) allow taking good run-time decisions, but current hardware implementations are either large or have limited learning capabilities.

In this work, we combine the capabilities of a software-based LCS, the XCS, with a lightweight hardware implementation, the LCT, retaining the benefits of both. We compare our method with other LCS implementations using the multiplexer problem and evaluate it with two chip-related problems, run-time task allocation and SoC component parameterisation. In all three problem sets, we find that the learning and self-adaptation capabilities are comparable to a full-fledged system, but with the added benefits of a lightweight hardware implementation, namely small area size and quick response time. Given our work, autonomous chips based on Learning Classifier Systems become feasible.

Keywords Autonomic System-on-Chip (ASoC) · System-on-Chip (SoC) · Learning classifier system (LCS) · XCS · Learning

1 Introduction

As the number of functions integrated in a single chip increases, the complexity of a chip grows significantly. Furthermore, increasing transistor variability [4, 6], process variation [1], and degradation effects [18] make it increasingly difficult to ensure the reliability of the chip [16]. The International Technology Roadmap for Semiconductors (ITRS) [13] estimates that until 2015, up to 70% of a chip’s design must be reused to keep up with the increasing complexity.

Autonomic System-on-Chip (ASoC) ([15], see also Chap. 4.7) add a logical, autonomic layer to contemporary SoCs that helps the designer to manage the com-
plexity and reliability issues: decisions that are hard to take at design time because many parameters are uncertain, can be taken at run time by the autonomic layer. Learning Classifier Systems (LCS) have been shown to be able to take the right run-time decisions ([2, 3], see also Chap. 5.3) and even adapt to events that due to the chip complexity have not been foreseen at design time. LCS use a genetic algorithm and reinforcement learning to evolve a set of rules, the interaction of which propose a preferably optimal action to any situation the chip may encounter. Although LCS allow very capable systems for autonomous run-time decisions and self-adaptation, current hardware implementations either require large portions of the chip [5], increasing total chip costs, or have limited learning capabilities [24].

In this article, we present a novel two-stage method to realise an on-chip Learning Classifier System (LCS) that is small, takes the good run-time decisions, and can adapt to unexpected events. In the first stage at design time, we learn a rule set in software using a particular LCS, the XCS [23]. In the second stage, we use the rule set to initialise the lightweight LCS hardware implementation Learning Classifier Table (LCT) [24]. The idea is that the XCS learns just enough rules so that the LCT can adapt to the actual manifestation and conditions of a particular chip and even to unexpected events, albeit in a limited way.

We first compare our method to other LCS implementations using the multiplexer problem, a traditional testbed for LCS [23], and then apply it to two chip-related problems, namely task-allocation and SoC component parameterisation. We show that the LCT can adequately learn and still react to unexpected events. To the best of our knowledge, this is the first study of a lightweight but still capable hardware implementation of an LCS. We think that our work makes using LCS to control chips conceivable.

This work is structured as follows. Section 2 gives an overview of related work. Section 3 introduces the XCS and the hardware implementation LCT. Section 4 describes our proposed method. Section 5 presents the three benchmarks multiplexer, task-allocation and SoC component parameterisation that we use to assess our method. Section 6 shows the results of our assessment and Sect. 7 concludes this article.

2 Related Work

Learning Classifier Systems were originally introduced in [12]. The XCS was first presented in [21] and later refined in [23]. The XCS has been used in a large range of learning and classification problems, including controlling a robotic mouse [10], a system-on-chip (SoC) [3], the lights of a traffic junction [17], and for finding suitable partitions in hardware-software co-design [11]. A first hardware implementation of an XCS has been presented in [5], named XCS\_i, which uses fixed-point arithmetic. The implementation shows good learning rates of the XCS\_i, but is quite large. In [24], the authors present an optimised hardware implementation of an LCS, called the Learning Classifier Table (LCT), which is small but has no mechanism to create new classifiers. Using a hand-crafted initial rule set, the authors show that the LCT can adjust the frequency of a SoC according to a given objective function.