Symbolic Finite Automata (SFA) have attracted a lot of attention recently. For an overview of SFA see [https://pages.cs.wisc.edu/~loris/symbolicautomata.html](https://pages.cs.wisc.edu/~loris/symbolicautomata.html). SFA are finite-state automata (FSA) where the alphabet is not a finite set of letters as in classical FSA but a set of formulas coming from a boolean algebra. For example, in the automaton in Figure 1 transitions are labeled by formulas defining intervals of natural numbers. SFA recognize words over a typically infinite domain like the natural numbers. The automaton of Figure 1 recognizes for example all words which are sequences of natural numbers between 0 and 99 followed by a number bigger or equal than 100.

\[ 0 \leq x < 100 \]

\[ x \geq 100 \]

Figure 1: A SFA

In [2] the problem of active learning of SFA was considered. Active learning here means that a learner tries to learn an unknown (for him) SFA by asking a teacher membership queries ("Is a given word accepted by the automaton?") and equivalence queries ("Is this hypothesis automaton correct?"). Membership queries are answered by yes/no whereas equivalence queries are answered by yes or a counterexample (a word in the symmetric difference between the hypothesis and the automaton to be learned). This problem has been well studied for FSA (first by Angluin in her seminal paper [1]).

For FSA, another natural problem is that of passive learning (for an overview see the book [3] where the learner only has a sample \( S = (S^+, S^-) \)
(where $S^+$ and $S^-$ are sets of words). $S^+$ are words which are accepted by an unknown automaton and $S^-$ words rejected by it. From this sample, the learners goal is to infer an automaton which correctly classifies all words in the sample while also generalizing and accepting more words. For example, from the sample $S^+ = \{ab, aab, aaab\}$ and $S^- = \{ba, bba\}$, the learner might infer an automaton corresponding to the language $a^*b$. Several algorithms for passive learning have been proposed, like RPNI [5] which is based on building an automaton accepting exactly the words of $S^+$ and then merging states as long as no word of $S^-$ is accepted.

In this internship the problem of extending RPNI to the framework of SFA will be studied. First results for passive learning SFA have been published recently [4] and will be adapted to the case of RPNI. There have also been results on passive learning of subclasses of timed automata [6]. Some of the ideas there should be adapted to the more general setting of SFA.

Time permitting, the passive learning algorithm will be implemented.

References


