Bachelor thesis
Computing Science

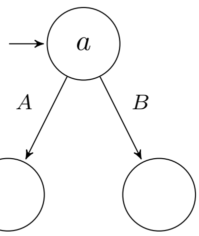
Radboud Universiteit

Exploring Adaptive Experiments

Author:
Cas Visser
s1045060

Supervisor:
Prof. Dr. Frits Vaandrager
F.Vaandrager@cs.ru.nl

Second assessor:
Bharat Garhewal, MSc
B.Garhewal@cs.ru.nl



Abstract

Adaptive Distinguishing Sequences (ADSs) have been used for state identification of Finite State Machines (FSMs). However, for some FSMs, no ADS exists. In this work, we formalize *adaptive experiments*, which can be constructed for any FSM. Additionally, we propose a way to measure the effectiveness of adaptive experiments, as well as a method to derive an adaptive experiment from an observation tree as described by [Vaandrager et al., 2021]. The lower and upper bound for our effectiveness measure are derived and it is demonstrated that adaptive experiments become more effective as the observation tree they are applied to is extended.

January 14, 2022

Contents

1 Introduction **3**

2 Preliminaries **5**
2.1 Mealy machines 5
2.2 Active Automata Learning 6
2.3 The $L^{\#}$ Algorithm 7

3 Research **9**
3.1 Extending Observation Trees 9
3.2 Adaptive Experiments 10
3.3 Observation Tree Experiment 13
3.4 Identification power 15
3.5 Identification Power of Extended Observation Trees 17

4 Conclusions **19**
4.1 Main contributions 19
4.2 Suggestions for future work 19

5 Acknowledgements **21**

Chapter 1

Introduction

It is a well known fact that, once compiled, computer software is not very transparent. Many programs and devices are so called “black boxes”, meaning that they can be interacted with, but their inner workings are not visible. Yet it is often desirable to be able to confirm that a system is working as intended. Testing whether an implementation conforms to a certain specification is called *conformance testing*.

This writing is concerned in particular with the conformance testing of Mealy machines, a specific type of *finite state machine* (FSM). Many programs, processes, devices, and systems can be modeled as Mealy machines. Characteristic of Mealy machines is that each transition between states is associated with exactly one input and one output symbol. The output of a Mealy machine is thus determined by the current state and the input.

A common problem in conformance testing is state identification: given an unknown FSM and its specification, determine the initial state of the FSM. In this context, “initial state” refers to the state the FSM is in at that moment and not for instance the initial state according to the specification. To determine the initial state, we run experiments on the FSM, which means we provide certain input symbols and try to derive information from the resulting output symbols.

One type of these experiments is separating sequences. These are input sequences that produce different outputs depending on which state they are run from. They can be used to determine what the starting state of the FSM was. There also exist adaptive separating sequences. These sequences bear some similarity to decision trees. They provide one input symbol at a time, and the next input symbol depends on the previous output symbols observed. This allows them to use more information to choose input symbols and identify the starting state in fewer steps than a non-adaptive separating sequence would. Adaptive separating sequences might also be able to identify some states that cannot be identified using non-adaptive separating sequences.

There exist more types of adaptive experiments besides adaptive separating sequences. Adaptive experiments are not a new occurrence in the world of conformance testing: the concept was already described by [Moore, 1956]. So far, research has mostly been focused on what [Lee and Yannakakis, 1994] call *adaptive distinguishing sequences* (ADSs). These experiments can uniquely identify any state of a specific Mealy machine, but they do not exist for all FSMs. However, even if no ADS exists for an FSM, there almost always exists an adaptive experiment which provides at least some useful information for state identification. For instance, [van den Bos and Vaandrager, 2021] find that their method still finds effective adaptive experiments for FSMs from the real world, even when an ADS usually does not exist.

These adaptive experiments which are not ADSs are potentially useful in *active automata learning* as well. When trying to figure out how something works, often the best strategy is to try things out and see what happens. Think for instance of trying to learn how to use a new kitchen appliance with too many buttons and vague symbols. This is in essence what active automata learning is. There is a system, which can be modeled as a Mealy machine, of which it is unknown how it functions internally. We are trying different inputs to find out how it works and effectively recreate the Mealy machine, preferably using as few inputs as possible. Conformance testing is one possible application of active automata learning. By comparing the learned Mealy machine to a specification, it can be confirmed that the implementation is functioning as intended.

Ever since it was introduced by [Angluin, 1987], the L^* algorithm has been the most common strategy for active automata learning. Recently however, [Vaandrager et al., 2021] published a new approach: the $L^{\#}$ algorithm. This algorithm makes use of a specific type of partial Mealy machine: an observation tree. Additionally, they use adaptive experiments to optimize their approach, providing a new relevance for adaptive experiments that are not ADSs. Although they describe how they create and use these adaptive experiments, not all the concepts in their report are defined formally.

This paper formalizes adaptive experiments and also explores some related concepts. We formalize the method described by [Vaandrager et al., 2021] for deriving an adaptive experiment from an observation tree and show that it provides useful experiments. We propose a method to measure the effectiveness (identification power) of an adaptive experiment and derive its lower and upper bound. Lastly, we find that adaptive experiments gain more identification power as the observation tree they are applied to is extended.

Ch. 2 gives some context and background on Mealy machines, active automata learning, and the $L^{\#}$ algorithm. Ch. 3 describes our findings in detail. Finally, Ch. 4 summarizes our main contributions and provides some suggestions for future research.

Chapter 2

Preliminaries

In this section, we introduce *mealy machines*, which are the type of models we learn using the L^* and $L^\#$ algorithms. We then provide an overview of *active automata learning*; and finally we briefly discuss the $L^\#$ algorithm, including the concept of *observation trees*, the primary data structure of the $L^\#$ algorithm.

2.1 Mealy machines

A nice intuitive definition of a Mealy machine by [Wikipedia contributors, 2021] is that "a Mealy machine is a finite-state machine whose output values are determined both by its current state and the current inputs". To achieve this, a transition between 2 states in a Mealy machine has both an input and an output symbol related to it.

Mealy machines were first described by [Mealy, 1955]. Mealy machines can be used to describe all sorts of systems, for example traffic lights, communication protocols, and control software.

The remainder of this section will define Mealy machines formally. Throughout this report, we fix a set of input symbols I (the input alphabet) and a set of output symbols O (the output alphabet). We write $f\colon X \rightharpoonup Y$ to denote that f is a partial function from X to Y. We write $f(x)\downarrow$ to indicate that f is defined on x, i.e. $\exists y \in Y\colon f(x) = y$. We write $f(x)\uparrow$ to indicate that f is undefined on x. A Mealy machine is then defined as follows:

Definition 1. *A* Mealy machine $\mathcal{M}$ *is a tuple* $\mathcal{M} = (Q, q_0, \delta, \lambda)$ *where*

- Q *is a set of states containing one initial state* q_0*,*
- $\delta : Q \times I \rightharpoonup Q$*, is a* transition function*, and*
- $\lambda : Q \times I \rightharpoonup O$*, is an* output function*.*

A transition from state q *to state* q' *with input* i *and output* o*, such that* $\delta(q, i) = q'$ *and* $\lambda(q, i) = o$*, is denoted by* $q \xrightarrow{i/o} q'$*. We generalize* δ *and* λ *to input sequences of length* n *by composing* δ *and* λ *with themselves* n *times. Whenever it is clear from context, we also use* δ *and* λ *for words. The functions* δ *and* λ *are defined on exactly the same inputs and states, i.e.* $\forall q \in Q, i \in I\colon \delta(q, i)\downarrow \iff \lambda(q, i)\downarrow$*. A Mealy machine* $\mathcal{M}$ *is* complete *if* δ *is complete, i.e.* $\forall q \in Q, i \in I\colon \delta(q, i)\downarrow$*.*

The transition function defines the next state given a current state and an input. The output function defines the output given a current state and an input. Two Mealy machines $\mathcal{M}$ and $\mathcal{N}$ are equivalent if and only if starting from their initial states, they produce the same output sequence for all input sequences.

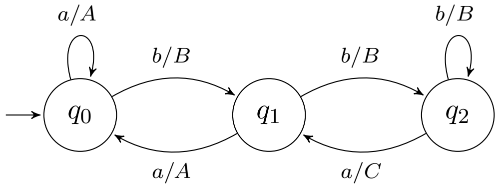


Figure 2.1: A simple Mealy machine.

When there are multiple different Mealy machines being discussed, say $\mathcal{M}$ and $\mathcal{N}$, to avoid confusion, the set of states belonging to the Mealy machine $\mathcal{M}$ is denoted by $Q^{\mathcal{M}}$ and the set of states belonging to the Mealy machine $\mathcal{N}$ is denoted by $Q^{\mathcal{N}}$. The same notation is used for the other parts of a Mealy machine, e.g. $\delta^{\mathcal{M}}$ and $\lambda^{\mathcal{M}}$.

As an example, Fig. 2.1 shows a visualization of a Mealy machine $\mathcal{M}$ with $Q^{\mathcal{M}} = \{q_0, q_1, q_2\}$, $q_0^{\mathcal{M}} = q_0$. Furthermore, $\delta^{\mathcal{M}}$ and $\lambda^{\mathcal{M}}$ are defined as shown in Tab. 2.1. As an illustration, for the input $\sigma = abbba$, the ending state is $\delta^{\mathcal{M}}(q_0, \sigma) = q_1$ and the output is $\lambda^{\mathcal{M}}(q_0, \sigma) = ABBBC$.

Table 2.1: Definitions for the δ and λ functions of the Mealy machine in Fig. 2.1.

δ:

Q \ I	a	b
q_0	q_0	q_1
q_1	q_0	q_2
q_2	q_1	q_2

λ:

Q \ I	a	b
q_0	A	B
q_1	A	B
q_2	C	B

Two states $q, q' \in Q$ are *apart* — denoted $q \# q'$ — if there exists an input sequence σ such that $\lambda(q, \sigma) \neq \lambda(q', \sigma)$. In this case, σ is a *witness* for $q \# q'$, denoted by $\sigma \vdash q \# q'$. Witnesses are also called *separating sequences*.

Formally, apartness is defined as a relation $\# \subset Q \times Q$, where $\# = \{(q, q') \in Q \times Q \mid q \# q'\}$. It is irreflexive, as for any input sequence, a state always gives the same outputs as itself, and symmetric, as a witness that separates q and q' also separates q' and q.

2.2 Active Automata Learning

Active automata learning can be described as a two player game between a learner, which poses the queries, and a teacher, who answers the queries. The teacher has a complete Mealy machine $\mathcal{M}$ which the learner is trying to "learn". Initially, the learner only knows the input alphabet I and the output alphabet O. To learn the Mealy machine, the learner may pose queries of the following 2 types:

1. OutputQuery(σ): The learner provides an input sequence $\sigma \in I^*$ and the teacher responds with the corresponding output sequence $\lambda^{\mathcal{M}}(q_0^{\mathcal{M}}, \sigma) \in O^*$. Remember that $\lambda^{\mathcal{M}}$ and $q_0^{\mathcal{M}}$ are the output function and initial state belonging to the Mealy machine $\mathcal{M}$, which in this case is the target Mealy machine that the teacher has.

2. EquivalenceQuery($\mathcal{H}$): The learner provides a hypothesis $\mathcal{H}$ where $\mathcal{H}$ is a complete Mealy machine, and the teacher replies whether $\mathcal{H}$ is equivalent to its model $\mathcal{M}$. If it is equivalent, the teacher replies `yes`. In this case, the learner has successfully learned $\mathcal{M}$. If it is not equivalent, the teacher replies `no` and provides a counterexample $\sigma \in I^*$ for which $\mathcal{H}$ and $\mathcal{M}$ give different output sequences.

In the context of this writing, it is assumed that it is also possible to pose an OutputQuery one symbol at a time and observe the output symbol at each step. This allows for OutputQuerys that let their later input symbols depend on the output for their earlier input symbols, so for instance something like "Start with input symbol a. If the output is A, input another a, otherwise input a b", etc.

Lastly, it is also assumed that all Mealy machines can be reset to their initial state at the cost of 1 input.

2.3 The $L^{\#}$ Algorithm

Recently [Vaandrager et al., 2021] presented the $L^{\#}$ (pronounced "el sharp") algorithm. The key discerning feature of this algorithm compared to other active model learning algorithms is how it stores observations, namely in a structure called an *observation tree*. Using this structure allows for storing more information about the responses it gets from the teacher than some other model learning algorithms can.

An observation tree $\mathcal{T}$ is a partial Mealy machine with a unique path to each state, giving it its tree structure. This paper uses the same definition for observation trees as [Vaandrager et al., 2021]. The following definitions were taken from their report.

In our learning setting, an *undefined* value in the partial transition map represents lack of knowledge. We consider maps between Mealy machines that preserve existing transitions, but possibly extend the knowledge of transitions:

Definition 2 (Functional Simulation)**.** *For Mealy machines $\mathcal{M}$ and $\mathcal{N}$, a* functional simulation *$f\colon \mathcal{M} \to \mathcal{N}$ is a map $f\colon Q^{\mathcal{M}} \to Q^{\mathcal{N}}$ with*

$$f(q_0^{\mathcal{M}}) = q_0^{\mathcal{N}} \qquad \text{and} \qquad q \xrightarrow{i/o} q' \text{ implies } f(q) \xrightarrow{i/o} f(q').$$

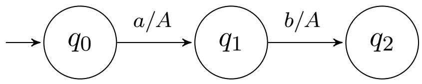
Intuitively, a functional simulation preserves transitions.

For a given machine $\mathcal{M}$, an observation tree is simply a Mealy machine itself which represents the inputs and outputs we have observed so far during learning. Using functional simulations, we define it formally as follows.

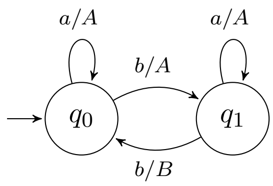
Definition 3 ((Observation) Tree)**.** *A Mealy machine $\mathcal{T}$ is a* tree *if for each $q \in Q^{\mathcal{T}}$ there is a unique sequence $\sigma \in I^*$ s.t. $\delta^{\mathcal{T}}(q_0^{\mathcal{T}}, \sigma) = q$. We write* $\mathsf{access}(q)$ *for the sequence of inputs leading to q. A tree $\mathcal{T}$ is an* observation tree *for a Mealy machine $\mathcal{M}$ if there is a functional simulation $f\colon \mathcal{T} \to \mathcal{M}$.*

Initially, the observation tree only contains one state which is also the initial state and no transitions. During the execution of the algorithm, all information obtained from OutputQuerys and EquivalenceQuerys is stored in the observation tree. If for instance an observation tree only has an initial state q_0, and OutputQuery(ab) gives output AA, the observation tree is expanded with 2 new states q_1 and q_2 and transitions $q_0 \xrightarrow{a/A} q_1$ and $q_1 \xrightarrow{b/A} q_2$. Fig. 2.2a shows this observation tree.

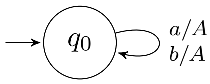
Note that it is possible that the states $q_2^{\mathcal{T}}$ and $q_0^{\mathcal{T}}$ actually correspond to the same state in the target Mealy machine $\mathcal{M}$, like in Fig. 2.2b. It is even possible that $q_0^{\mathcal{T}}$, $q_1^{\mathcal{T}}$, and $q_2^{\mathcal{T}}$ are all the same state, like in Fig. 2.2c. Figuring out whether 2 states in the observation tree actually relate to the same state in the target Mealy machine is one of the challenges the $L^{\#}$ algorithm faces.



(a) An observation tree.



(b) A possible Mealy machine.



(c) Another possible Mealy machine.

Figure 2.2: Possible Mealy machines for the same observation tree.

The states in the observation tree are divided into three disjoint sets: the basis $S \subset Q^{\mathcal{T}}$, the frontier $F \subset Q^{T}$, and the remaining states. The basis is constructed such that its states are pairwise apart. This means that every state in the basis is apart from every other state in the basis. If each state in the basis has a transition for every input $i \in I$, then the basis is *complete*. The frontier simply consists of the direct successors of the basis, or more formally $F = \{q' \in Q^{\mathcal{T}} \setminus S \mid \exists q \in S, i \in I : \delta^{\mathcal{T}}(q, i) = q'\}$.

Chapter 3

Research

In this section, we define the *extending* of observation trees, *adaptive experiments* and their application to Mealy machines, and *adaptive distinguishing sequences*. Additionally we define a method for deriving an adaptive experiment from an observation tree and a method to measure the effectiveness (*identification power*) of an adaptive experiment. Lastly, we derive the lower and upper bound for this method and show that the effectiveness of an adaptive experiment according to this measure goes up as the observation tree it is applied to is extended.

3.1 Extending Observation Trees

As mentioned in Sec. 2.3, the $L^{\#}$ algorithm uses observation trees and extends them. We formally define extension as follows.

Definition 4 (Observation Tree Extension). *Let $\mathcal{T}$ and $\mathcal{T}'$ be observation trees. Then $\mathcal{T}'$ is an* extension *of $\mathcal{T}$ if*

- $q_0^{\mathcal{T}} = q_0^{\mathcal{T}'}$,
- $Q^{\mathcal{T}} \subseteq Q^{\mathcal{T}'}$,
- $\delta^{\mathcal{T}'}(q, i) = \delta^{\mathcal{T}}(q, i)$ *for all* $q \in Q^{\mathcal{T}}$ *and all* $i \in I$*, and*
- $\lambda^{\mathcal{T}'}(q, i) = \lambda^{\mathcal{T}}(q, i)$ *for all* $q \in Q^{\mathcal{T}}$ *and all* $i \in I$.

Put more simply, extending an observation tree leaves all the original parts in place and adds states and transitions to those states. See Fig. 3.1 for an example.

Figure 3.1: An observation tree (left) and a possible extension (right).

3.2 Adaptive Experiments

Sec. 2.1 discusses separating sequences, which are sequences that show that 2 states are not equivalent. To show that a Mealy machine is minimal, in the worst case one separating sequence is needed for every pair of states, which would amount to $(|Q|-1)^2$ sequences. In the worst case, each of these sequences has a length of $|Q|-1$, making a total of $(|Q|-1)^3$ inputs.

If the goal is to be able to show that no state is equivalent to another state, it may be more efficient to use an *adaptive experiment* instead. This is an experiment that provides input symbols one at a time. The next input symbol depends on the output symbols that resulted from the previous input symbols.

Definition 5 (Adaptive Experiment)**.** *An* adaptive experiment *is a tuple* $\mathcal{E} = (S, s_0, \mu, \nu)$*, where*

- S *is a finite set of* states *and* $s_0 \in S$ *is the* initial state*,*
- $\mu\colon S \rightharpoonup I$ *is an* input function
- $\nu\colon S \times O \rightharpoonup S$ *is a* transition function

We generalize the transition function to output words of length n *by composing* ν *with itself* n *times. Whenever it is clear from the context, we also use* ν *for words. We require that each adaptive experiment satisfies the following two properties:*

1. *The underlying digraph is acyclic, that is, for all* $s \in S$ *and* $\sigma \in O^*$*,*

$$\nu(s, \sigma) = s \Leftrightarrow \sigma = \epsilon$$

2. *The input function is defined for all states except for the leaves:*

$$\mu(s) \uparrow \Leftrightarrow s \text{ is a leaf}$$

where a leaf *is a state* $s \in S$ *without outgoing transitions, that is, for all* $o \in O$*,* $\nu(s, o)\uparrow$*.*

We use superscript $\mathcal{E}$ *to disambiguate to which experiment we refer, e.g.* $S^{\mathcal{E}}$*,* $s_0^{\mathcal{E}}$*,* $\mu^{\mathcal{E}}$*, and* $\nu^{\mathcal{E}}$*. We write* $s \xrightarrow{o} s'$*, with* $s, s' \in S$*,* $o \in O$*, to denote* $\nu(s, o) = s'$*. For each state* $s \in S$ *we write* $\mathcal{E} \dagger s$ *to denote the adaptive experiment obtained from* $\mathcal{E}$ *by replacing its initial state with* s*, that is,* $\mathcal{E} \dagger s = (S, s, \mu, \nu)$*.*

Fig. 3.2b shows an adaptive experiment. Each circle represents a state. The top circle marked with an arrow is the initial state. The result of the input function μ is shown inside each circle. Each arrow between circles represents a transition in the transition function ν.

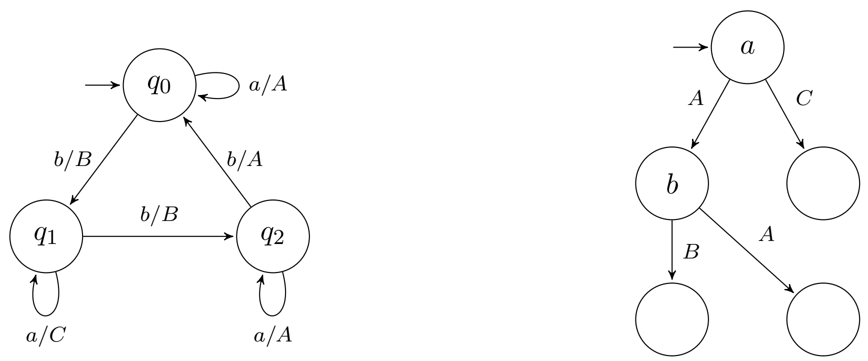
We may apply an adaptive experiment $\mathcal{E}$ to a Mealy machine, resulting in a sequence of outputs. We first apply the input label of the initial state of $\mathcal{E}$ to $\mathcal{M}$. If this input is undefined in the current state of $\mathcal{M}$ the experiment stops. Otherwise, $\mathcal{M}$ will update its state and return an output. If this output is undefined in the current state of $\mathcal{E}$ the experiment stops. Otherwise, also the initial state of the experiment is updated by applying the output obtained from $\mathcal{M}$. The input of the new initial state of the experiment is applied to the Mealy machine and all steps are repeated until a leaf state of the experiment is reached, the input symbol is not defined in the current state of the Mealy machine, or the Mealy machine generates an unexpected output. The resulting sequence of outputs obtained from $\mathcal{M}$ is called the *outcome of running experiment* $\mathcal{E}$ *on* $\mathcal{M}$.

Definition 6 (Outcome)**.** *Let $\mathcal{M} = (Q, q_0, \delta, \lambda)$ be a Mealy machine and let $\mathcal{E} = (S, s_0, \mu, \nu)$ be an adaptive experiment. Then* $\mathsf{outcome}(\mathcal{E}, \mathcal{M})$*, the* outcome of running experiment $\mathcal{E}$ on $\mathcal{M}$*, is the output word defined as follows (by induction on the length of the longest path in $\mathcal{E}$ from the initial state to a leaf):*

$$\mathsf{outcome}(\mathcal{E}, \mathcal{M}) = \begin{cases} \epsilon & \text{if } s_0 \text{ is a leaf, or if } \delta(q_0, i)\uparrow \\ \textit{undefined} & \text{if } \nu(s_0, o)\uparrow \\ o \cdot \mathsf{outcome}(\mathcal{E} \dagger \nu(s_0, o), \mathcal{M} \dagger \delta(q_0, i)) & \text{otherwise} \end{cases}$$

where $i = \mu(s_0)$, $o = \lambda(q_0, i)$, and $\mathcal{M} \dagger q$ denotes the Mealy machine obtained from $\mathcal{M}$ by replacing its initial state by q. The cases should be considered in order, i.e. the second and third case should only be considered when s_0 is not a leaf. If $x = \textit{undefined}$, then $o \cdot x = \textit{undefined}$ as well, where o is a symbol.

For an example of $\mathsf{outcome}$, let $\mathcal{M}$ be the Mealy machine shown in Fig. 3.2a and let $\mathcal{E}$ be the adaptive experiment shown in Fig. 3.2b. Then when computing $\mathsf{outcome}(\mathcal{E}, \mathcal{M})$ we first get the third case. We apply the input from the initial state of $\mathcal{E}$ to $\mathcal{M}$ and get $\lambda(q_0, \mu(s_0)) = \lambda(q_0, a) = A$ as response. Now $\mathcal{M}$ is in the state $\delta(q_0, a) = q_0$. Feeding the response into the transition function, we get that $\mathcal{E}$ is now in the left child of the initial state. Repeating this process one more time, we get that our next input symbol is b, which gives us B as response from $\mathcal{M}$. After this we are in a leaf of $\mathcal{E}$ and the experiment stops, so in full, $\mathsf{outcome}(\mathcal{E}, \mathcal{M}) = AB$.



(a) A three state Mealy machine.

(b) An adaptive experiment

Figure 3.2: Visualization of an adaptive experiment and its application to a Mealy machine.

By running the experiment $\mathcal{E}$ from different states of $\mathcal{M}$, we can obtain different outcomes. The set of all possible outcomes of an experiment $\mathcal{E}$ for a set of states U is called the *set of possible outcome words of E for U*.

Definition 7. *Let $\mathcal{E}$ be an adaptive experiment and $\mathcal{M}$ a Mealy machine. Let U be a subset of $Q^{\mathcal{M}}$. Then* $\mathsf{words}(\mathcal{E}, \mathcal{M}, U)$*, the* set of possible outcome words of E for U*, is defined as*

$$\mathsf{words}(\mathcal{E}, \mathcal{M}, U) = \{\, \sigma \mid \exists q \in U \colon \mathsf{outcome}(\mathcal{E}, \mathcal{M} \dagger q) = \sigma \text{ and } \sigma \neq \textit{undefined} \}.$$

For example, let $\mathcal{M}$ again be the Mealy machine shown in Fig. 3.2a and let $\mathcal{E}$ again be the adaptive experiment shown in Fig. 3.2b. There are three states in $\mathcal{M}$ which we can run the experiment from:

- $\mathsf{outcome}(\mathcal{E}, \mathcal{M} \dagger q_0) = AB$,
- $\mathsf{outcome}(\mathcal{E}, \mathcal{M} \dagger q_1) = C$, and
- $\mathsf{outcome}(\mathcal{E}, \mathcal{M} \dagger q_2) = AA$.

If U is the full set of states $Q^{\mathcal{M}}$, then $\mathsf{words}(\mathcal{E}, \mathcal{M}, U) = \{AA, AB, C\}$. If we then remove a state from U, the corresponding outcome word disappears as well, so for instance $\mathsf{words}(\mathcal{E}, \mathcal{M}, \{q_0, q_2\}) = \{AA, AB\}$.

We also define the inverse of the outcome function. This gives for an outcome sequence the states that have that outcome.

Definition 8. *Let $\mathcal{E}$ be an adaptive experiment, $\mathcal{M}$ a Mealy machine, and σ a sequence of symbols. Then* $\mathsf{outcome}^{-1}(\mathcal{E}, \mathcal{M}, \sigma)$*, the* set of states in $\mathcal{M}$ that have σ as outcome from $\mathcal{E}$*, is defined as*

$$\mathsf{outcome}^{-1}(\mathcal{E}, \mathcal{M}, \sigma) = \{\, q \in Q^{\mathcal{M}} \mid \mathsf{outcome}(\mathcal{E}, \mathcal{M} \dagger q) = \sigma \,\}.$$

With $\mathcal{E}$ and $\mathcal{M}$ as defined above, $\mathsf{outcome}^{-1}(\mathcal{E}, \mathcal{M}, A) = q_1$ and $\mathsf{outcome}^{-1}(\mathcal{E}, \mathcal{M}, AB) = q_0$.

Not all adaptive experiments are equally useful. For instance, the trivial adaptive experiment, where S contains just a single state which is a leaf, does not provide any information at all. In the ideal case, given a Mealy machine $\mathcal{M}$, an adaptive experiment will tell us that all states in $Q^{\mathcal{M}}$ are pairwise apart. Especially with observation trees, this will not be the case. Alternatively, we can see if an experiment can tell use that all states in a subset $U \subseteq Q^{\mathcal{M}}$ are pairwise apart. If it can, the experiment is called an *adaptive distinguishing experiment for U in $\mathcal{M}$*.

Definition 9 (Adaptive Distinguishing Experiment)**.** *If an adaptive experiment $\mathcal{E}$ perfectly identifies the states in $U \subseteq Q^{\mathcal{M}}$ with some Mealy machine $\mathcal{M}$, that is,* $\mathsf{outcome}(\mathcal{E}, \mathcal{M} \dagger q)$ *gives a unique result for each $q \in U$, then $\mathcal{E}$ is an* adaptive distinguishing experiment for U in $\mathcal{M}$*.*

For some Mealy machines, no adaptive distinguishing experiment exists for the complete set of states $Q^{\mathcal{M}}$. This is the case for instance for the Mealy machine in Fig. 2.1. From any initial state, the only valid inputs are a and b. However, after giving a as input, q_0 and q_1 both go to q_0, meaning they cannot be distinguished anymore afterwards. The same goes for b as input and states q_1 and q_2. This means no adaptive experiment can perfectly distinguish the states of this Mealy machine.

Nevertheless, if an adaptive distinguishing experiment does exist, it can be used to efficiently show that all states in $Q^{\mathcal{M}}$ are pairwise apart, namely by just running it from each state once. Additionally, [Lee and Yannakakis, 1994] show that for each state, at most n inputs are needed, making a total of n^2 inputs. This means adaptive experiments have the potential to be significantly more efficient than separating sequences.

If an experiment is used on an observation tree, it is possible that two states have a different outcome at first, but turn out to be equivalent after the observation tree is extended. In this case, the difference in the outcomes is solely caused by one outcome being longer than the other, as opposed to there being an i-th symbol where the outcomes differ. If the outcomes differ not only in length but actually contain different symbols at at least one index, we say that they *exclude* each other.

Definition 10. *Let $\mathcal{E}$ be an adaptive experiment, let $\mathcal{M}$ be a Mealy machine, and let U be a subset of $Q^{\mathcal{M}}$. Let $\sigma \in O^*$ be a sequence of output symbols. Then* $\mathsf{excluded}(\mathcal{E}, \mathcal{M}, \sigma)$*, the* states of $\mathcal{M}$ excluded by σ in $\mathcal{E}$*, is defined as*

$$\mathsf{excluded}(\mathcal{E}, \mathcal{M}, \sigma) = \{\, q \in Q^{\mathcal{M}} \mid \sigma \perp \tau \textit{ and } \tau \perp \sigma \textit{ with } \tau = \mathsf{outcome}(\mathcal{E}, \mathcal{M} \dagger q) \,\}$$

where $a \perp b$ if a and b are sequences of symbols and a is not a prefix of b.

For example with the $\mathcal{E}$ and $\mathcal{M}$ as defined above, $\mathsf{excluded}(\mathcal{E}, \mathcal{M}, A) = \{C\}$ and $\mathsf{excluded}(\mathcal{E}, \mathcal{M}, C) = \{AA, AB\}$.

For an observation tree $\mathcal{T}$, a way of describing exclusion is that $\mathsf{excluded}(\mathcal{E}, \mathcal{T}, \sigma)$ is the set of states in $\mathcal{T}$ that will never have σ as outcome from $\mathcal{E}$, no matter how $\mathcal{T}$ is extended.

3.3 Observation Tree Experiment

When working with an observation tree, it is relevant to be able to tell which states are apart and which are not. For instance in the context of the $L^{\#}$ algorithm, which uses observation trees, an adaptive experiment could help us decide what a good next query could be. To this end, [Vaandrager et al., 2021] define a reward function, which tells us for a set of states U how well they can be distinguished from one another using an adaptive experiment. The following definition is based on their *expected reward* function E.

Definition 11 (Reward)**.** *Let $\mathcal{T}$ be an observation tree and let U be a subset of $Q^{\mathcal{T}}$. Then* $\mathsf{reward}(U, \mathcal{T})$*, the* reward of U in $\mathcal{T}$*, is defined as follows (by induction on the length of the longest path in $\mathcal{T}$ from the initial state to a leaf):*

$$\mathsf{reward}(U, \mathcal{T}) = \max_{I_U} \left(\sum_{o \in O} \frac{|\,\Delta(U, i, o)\,| \cdot (|\,U_i\,| - |\,\Delta(U, i, o)\,| + \mathsf{reward}(\Delta(U, i, o), \mathcal{T}))}{|\,U_i\,|} \right)$$

where

- *$I_U \subseteq I$ is the set of input symbols accepted by at least one state in U,*
- *U_i are the states in U that accept i as input,*
- *$\Delta(U, i, o) = \{\, \delta(u, i) \mid u \in U \text{ and } \lambda(u, i) = o \,\}$, the set of states that are reached from a state in U with input i and output o, and*
- *$\mathsf{maxsymbol}(U, \mathcal{T})$ refers to the input symbol that provides the maximum value in $\max_{I_U}$. There may be several that achieve the maximum and we nondeterministically select one. If $I_U = \emptyset$, then $\mathsf{maxsymbol}(U, \mathcal{T}) \uparrow$ and $\mathsf{reward}(U, \mathcal{T}) = 0$.*

Intuitively, $\mathsf{reward}(U, \mathcal{T})$ gives us the maximal expected number of apartness pairs in U using $\mathcal{T}$. Working out reward by hand is quite cumbersome, but as an example, let $\mathcal{T}$ be the observation tree shown in Fig. 3.3a and let $U = \{q_0, q_1\}$. Then $\mathsf{maxsymbol}(U, \mathcal{T}) = a$ and the reward of U in $\mathcal{T}$ is

$$\begin{aligned}\mathsf{reward}(U,\mathcal{T}) &= \sum_{o\in O} \frac{|\,\Delta(U,a,o)\,|\cdot(|\,U_a\,| - |\,\Delta(U,a,o)\,| + \mathsf{reward}(\Delta(U,a,o),\mathcal{T}))}{|\,U_a\,|} \\ &= \frac{|\,\Delta(U,a,A)\,|\cdot(|\,U_a\,| - |\,\Delta(U,a,A)\,| + \mathsf{reward}(\Delta(U,a,A),\mathcal{T}))}{|\,U_a\,|} \\ &\quad + \frac{|\,\Delta(U,a,B)\,|\cdot(|\,U_a\,| - |\,\Delta(U,a,B)\,| + \mathsf{reward}(\Delta(U,a,B),\mathcal{T}))}{|\,U_a\,|} \\ &= \frac{1\cdot(2-1+\mathsf{reward}(q_1,\mathcal{T}))}{2} + \frac{1\cdot(2-1+\mathsf{reward}(q_2,\mathcal{T}))}{2} \\ &= \frac{1}{2}+\frac{1}{2} = 1\end{aligned}$$

So $\mathsf{reward}(U,\mathcal{T}) = 1$. This means that we can expect one apartness pair in the states of U. Since U only contains two states, and a state cannot be apart from itself, this means that the two states in U should be apart from each other. We can see that this is indeed the case: q_0 and q_1 have different output symbols for the input symbol a.

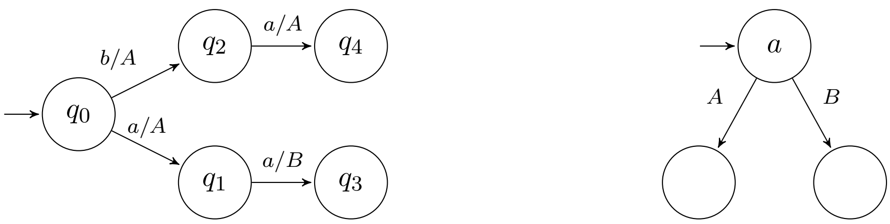
Using Def. 11, we can construct an adaptive experiment which distinguishes the states in U as well as possible.

Definition 12. *Let $\mathcal{T}$ be an observation tree and let U be a subset of $Q^{\mathcal{T}}$. Then an adaptive experiment $\mathcal{E} = (S, s_0, \mu, \nu)$ can be* derived *from $\mathcal{T}$ for U as follows:*

1. *Let S equal the set of all non-empty subsets of U.*
2. *The initial state s_0 is the full set U.*
3. *For all $s \in S$, the input function can be defined as $\mu(s) = \mathsf{maxsymbol}(s,\mathcal{T})$.*
4. *Lastly, for the transition function, for all $s \in S, o \in O$ with $\mu(s)\downarrow$, let $D = \Delta(s,\mu(s),o)$. If $D \neq \emptyset$, then $\nu(s,o) = D$.*

Clearly, most states in S are actually not used. For simplicity, all states that are unreachable from s_0 can be omitted.

An intuitive way to think about the states in S being defined as subsets of U is that this represents the possible initial states. That is, when $\mathcal{E}$ is the experiment derived from $\mathcal{T}$ for U, then after running $\mathcal{E}$ from some state q in $\mathcal{T}$, we know q is one of the states in the subset that $\mathcal{E}$ ends in. Also note that the subsets that appear as U in the recursive calls of reward are exactly the same subsets that are reachable from s_0 in the derived experiment.



(a) An observation tree. (b) An adaptive experiment derived from it.

Figure 3.3: An observation tree and an adaptive experiment derived from it.

Fig. 3.3b shows the experiment derived from the observation tree in Fig. 3.3a for $U = \{q_0, q_1\}$. Sec. 3.4 defines *identification power*, a measure for the effectiveness of an adaptive experiment. The identification power of this experiment is 1 and it is an adaptive distinguishing experiment. If we extend U to $U = \{q_0, q_1, q_2\}$, the derived experiment remains the same. The identification power goes up a bit though, to $\frac{4}{3}$. This is explained by the fact that q_2 cannot be perfectly distinguished, but it can be distinguished from q_1, as they have different outputs for the input symbol a.

Item 4 of Def. 12 guarantees that for every output symbol obtained from U with input symbol $\mu(U) = \mathsf{maxsymbol}(U, \mathcal{T})$ (item 3), there is a corresponding transition. For this reason, we can be sure that for this experiment, $\mathsf{outcome}$ is defined for every state in U.

Lemma 1. *Let $\mathcal{T}$ be an observation tree and let U be a subset of $Q^{\mathcal{T}}$. Let $\mathcal{E}$ be an adaptive experiment derived from $\mathcal{T}$ for U. Then $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger u) \downarrow$ for all $u \in U$.*

Proof. If $U = \emptyset$, this is true trivially. If $U \neq \emptyset$, we distinguish between two cases.

Case 1: $I_U = \emptyset$. In this case, the states in U do not accept any input symbol in $\mathcal{T}$. Consequently, $\mathsf{outcome}$ will end up in its first case, since $\delta(u, i) \uparrow$ for all $i \in I$. More precisely, $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger u) = \epsilon$ with u any state in U and hence $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger u) \downarrow$ for all $u \in U$.

Case 2: $I_U \neq \emptyset$. In this case, there is at least one state in U that accepts an input symbol and hence there is also an input symbol that elicits the highest reward, i.e. $\mathsf{maxsymbol}(U) \downarrow$. By item 2 of Def. 12, $s_0 = U$. Then it follows from item 4 of Def. 12 that for any output o resulting from $\delta(q, \mu(s_0)$ with q any state in s_0, the transition function $\nu(s_0, o)$ is defined. Namely if and only if o is a result of $\delta(q, \mu(s_0)$ for some $q \in s_0$, then $\Delta(s, \mu(s), o) \neq \emptyset$. Hence also in this case $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger u) \downarrow$ for all $u \in U$.

When Case 2 applies, this approach can be applied recursively to $\mathcal{T} \dagger \delta(q_0, i)$ and $\mathcal{E} \dagger \nu(s_0, \lambda(q_0, i))$ with $i = \mu(s_0)$ until Case 1 applies. □

Proving that this experiment is indeed the best possible experiment for U is a topic that needs further research. This will be discussed in further detail in Sec. 4.2.

3.4 Identification power

To compare adaptive experiments, we define the notion of identification power (IP). Intuitively, for an experiment $\mathcal{E}$, a Mealy machine $\mathcal{M}$, and a set of states $U \subseteq Q^{\mathcal{M}}$, the identification power represents the average number of states in U that a state in U can be distinguished from by applying $\mathcal{E}$ to $\mathcal{M}$.

Definition 13 (identification power)**.** *Let $\mathcal{E}$ be an adaptive experiment and $\mathcal{M}$ a Mealy machine. Let U be a subset of $Q^{\mathcal{M}}$. Then $\mathsf{IP}(\mathcal{E}, U, \mathcal{M})$, the* identification power of $\mathcal{E}$ for U with $\mathcal{M}$*, is defined as*

$$\mathsf{IP}(\mathcal{E}, U, \mathcal{M}) = \sum_{q \in U} \frac{1}{|\,U\,|} \cdot |\,\mathsf{excluded}(\mathcal{E}, \mathcal{M}, \mathsf{outcome}(\mathcal{E}, \mathcal{M} \dagger q)) \cap U\,|$$

The part inside the summation can be described in words as the probability of ending up in some state of $\mathcal{E}$, multiplied by the number of states of U that leaf 'excludes'. Since some of states in U may have the same $\mathsf{outcome}$, the formula for identification power can be simplified somewhat:

$$\begin{aligned}\mathsf{IP}(\mathcal{E}, U, \mathcal{M}) &= \sum_{\sigma \in \mathsf{words}(\mathcal{E},\mathcal{M},U)} \frac{|\,\mathsf{outcome}^{-1}(\mathcal{E}, \mathcal{M}, \sigma) \cap U\,|}{|\,U\,|} \cdot |\,\mathsf{excluded}(\mathcal{E}, \mathcal{M}, \sigma) \cap U\,| \\ &= \sum_{\sigma \in \mathsf{words}(\mathcal{E},\mathcal{M},U)} \frac{|\,\mathsf{outcome}^{-1}(\mathcal{E}, \mathcal{M}, \sigma) \cap U\,| \cdot |\,\mathsf{excluded}(\mathcal{E}, \mathcal{M}, \sigma) \cap U\,|}{|\,U\,|}\end{aligned}$$

For example, let $\mathcal{M}$ be the Mealy machine shown in Fig. 3.2a, let $U = Q$, and let $\mathcal{E}$ be the adaptive experiment shown in Fig. 3.2b. Then as demonstrated in Sec. 3.2, $\mathsf{words}(\mathcal{E}, \mathcal{M}, U) = \{AA, AB, C\}$ The identification power of $\mathcal{E}$ for U with $\mathcal{M}$ is

$$\begin{aligned}\mathsf{IP}(\mathcal{E}, U, \mathcal{M}) &= \sum_{\sigma \in \mathsf{words}(\mathcal{E},\mathcal{M},U)} \frac{|\,\mathsf{outcome}^{-1}(\mathcal{E}, \mathcal{M}, \sigma) \cap U\,| \cdot |\,\mathsf{excluded}(\mathcal{E}, \mathcal{M}, \sigma) \cap U\,|}{|\,U\,|} \\ &= \frac{1 \cdot 2}{3} + \frac{1 \cdot 2}{3} + \frac{1 \cdot 2}{3} \\ &= 2\end{aligned}$$

There is a clear upper bound for the identification power of an experiment for a certain set of states U. Namely, only the states in U can count towards the identification power. Additionally, a state cannot be apart from itself, hence the identification power for a set U is at most $|U| - 1$.

Lemma 2. *If an adaptive experiment $\mathcal{E}$ is an* adaptive distinguishing experiment *for U in $\mathcal{M}$, with $\mathcal{M}$ a Mealy machine and $U \subset Q^{\mathcal{M}}$, then $\mathsf{IP}(\mathcal{E}, U, \mathcal{M}) = |\,U\,| - 1$.*

Proof. Since $\mathcal{E}$ is an *adaptive distinguishing experiment* for U in $\mathcal{M}$, it is the case that $|\,\mathsf{outcome}^{-1}(\mathcal{E}, \mathcal{M}, \sigma) \cap U\,| = 1$ for each $\sigma \in \mathsf{words}(\mathcal{E}, \mathcal{M}, U)$. Consequently, $|\,\mathsf{words}(\mathcal{E}, \mathcal{M}, U)\,|$ equals $|\,U\,|$. Furthermore, since all other states in U have a different $\mathsf{outcome}$, the degree of $\mathsf{excluded}(\mathcal{E}, \mathcal{M}, \sigma) \cap U$ is $|\,U\,| - 1$ for all $\sigma \in \mathsf{words}(\mathcal{E}, \mathcal{M}, U)$. The identification power then boils down to

$$\begin{aligned}\mathsf{IP}(\mathcal{E}, U, \mathcal{M}) &= \sum_{\sigma \in \mathsf{words}(\mathcal{E},\mathcal{M},U)} \frac{|\,\mathsf{outcome}^{-1}(\mathcal{E}, \mathcal{M}, \sigma) \cap U\,| \cdot |\,\mathsf{excluded}(\mathcal{E}, \mathcal{M}, \sigma) \cap U\,|}{|\,U\,|} \\ &= |\,U\,| \cdot \frac{1 \cdot |\,U\,| - 1}{|\,U\,|} \\ &= |\,U\,| - 1\end{aligned}$$

□

There is also a clear lower bound for identification power, namely it cannot be lower than 0. This occurs when the experiment offers no information.

Lemma 3. *If an Adaptive Experiment $\mathcal{E}$ is trivial such that it gives the same outcome word for all states in a set $U \subseteq Q^{\mathcal{M}}$ with $\mathcal{M}$ a Mealy machine, then the identification power of that Adaptive Experiment for that set in that Mealy machine is $\mathsf{IP}(\mathcal{E}, U, \mathcal{M}) = 0$.*

Proof. Since $\mathcal{E}$ gives the same outcome for all $q \in U$, there is only one possible outcome word, such that $|\,\mathsf{words}(\mathcal{E}, \mathcal{M}, U)\,| = 1$. Let r be this word. All states in U have r as outcome word, so $\mathsf{excluded}(\mathcal{E}, \mathcal{M}, \sigma) \cap U = \emptyset$. The identification power then boils down to

$$\begin{aligned}
\mathsf{IP}(\mathcal{E}, U, \mathcal{M}) &= \sum_{\sigma \in \mathsf{words}(\mathcal{E}, \mathcal{M}, U)} \frac{|\,\mathsf{outcome}^{-1}(\mathcal{E}, \mathcal{M}, \sigma) \cap U\,| \cdot |\,\mathsf{excluded}(\mathcal{E}, \mathcal{M}, \sigma) \cap U\,|}{|\,U\,|} \\
&= \sum_{\sigma \in \mathsf{words}(\mathcal{E}, \mathcal{M}, U)} \frac{|\,\mathsf{outcome}^{-1}(\mathcal{E}, \mathcal{M}, \sigma) \cap U\,| \cdot 0}{|\,U\,|} \\
&= 0
\end{aligned}$$

□

3.5 Identification Power of Extended Observation Trees

Take an observation tree $\mathcal{T}$ and an adaptive experiment $\mathcal{E}$. As $\mathcal{T}$ is extended to $\mathcal{T}'$, due to the nature of adaptive experiments, the outcome of running $\mathcal{E}$ from a state in $\mathcal{T}$ is not really different from running $\mathcal{E}$ from the same state in $\mathcal{T}'$. Rather, the outcome is extended.

Lemma 4. *Let $\mathcal{E}$ be an adaptive experiment. Let $\mathcal{T}$ be an observation tree, let $\mathcal{T}'$ be an extension of $\mathcal{T}$, and let q be any state in $Q^{\mathcal{T}}$ for which $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger q)\downarrow$ and $\mathsf{outcome}(\mathcal{E}, \mathcal{T}' \dagger q)\downarrow$. Lastly, let $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger q) = \sigma$ and let $\mathsf{outcome}(\mathcal{E}, \mathcal{T}' \dagger q) = \sigma'$. Then σ is a prefix of σ'.*

Proof. Since $\mathcal{T}'$ is an extension of $\mathcal{T}$, the computation of $\mathsf{outcome}(\mathcal{E}, \mathcal{T}' \dagger q)$ will first follow the exact same cases of $\mathsf{outcome}$ with the exact same results as $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger q)$ before reaching any different results. Since σ and σ' are both not undefined, σ must be a prefix of σ'. □

This also means that if the outcome for a state is undefined, the outcome for this state will also be undefined for an extended version of $\mathcal{T}$.

Lemma 5. *Let $\mathcal{E}$ be an adaptive experiment. Let $\mathcal{T}$ be an observation tree, let $\mathcal{T}'$ be an extension of $\mathcal{T}$, and let q be any state in $Q^{\mathcal{T}}$. Then $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger q)\uparrow \rightarrow \mathsf{outcome}(\mathcal{E}, \mathcal{T}' \dagger q)\uparrow$.*

Proof. By Def. 6, if undefined is reached, it is the end of the outcome. By Lemma 4, $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger q)$ is a prefix of $\mathsf{outcome}(\mathcal{E}, \mathcal{T}' \dagger q)$. Hence $\mathsf{outcome}(\mathcal{E}, \mathcal{T}' \dagger q)$ will also end at undefined and so if $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger q)\uparrow$, then also $\mathsf{outcome}(\mathcal{E}, \mathcal{T}' \dagger q)\uparrow$. □

Furthermore, this means that states that exclude each other, also exclude each other in an extended observation tree. As soon as there is a position at which two outcomes have different symbols, these different symbols will be preserved as prefix of the outcomes in an extended observation tree.

Lemma 6. *Let $\mathcal{E}$ be an adaptive experiment. Let $\mathcal{T}$ be an observation tree, let $\mathcal{T}'$ be an extension of $\mathcal{T}$, and let q, r be states in $Q^{\mathcal{T}}$, such that $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger q)\downarrow$, $\mathsf{outcome}(\mathcal{E}, \mathcal{T}' \dagger q)\downarrow$, $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger r)\downarrow$, and $\mathsf{outcome}(\mathcal{E}, \mathcal{T}' \dagger r)\downarrow$. Let $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger q) = \sigma$, $\mathsf{outcome}(\mathcal{E}, \mathcal{T}' \dagger q) = \sigma'$, $\mathsf{outcome}(\mathcal{E}, \mathcal{T} \dagger r) = \tau$, and $\mathsf{outcome}(\mathcal{E}, \mathcal{T}' \dagger r) = \tau'$. If $r \in \mathsf{excluded}(\mathcal{E}, \mathcal{T}, \sigma)$, then also $r' \in \mathsf{excluded}(\mathcal{E}, \mathcal{T}', \sigma')$.*

Proof. If $r \in \mathsf{excluded}(\mathcal{E}, \mathcal{M}, \sigma)$, then τ is not a prefix of σ and vice versa. Since τ and σ are both not undefined, they must both be a sequence of symbols. Since τ and σ are not prefixes of each other, there must be an index i with $i < \mathsf{min}(|\sigma|, |\tau|)$, such that σ and τ both have an i-th symbol and such that these i-th symbols are different.

By Lemma 4, σ is a prefix of σ' and τ is a prefix of τ' and since $i < \mathsf{min}(|\sigma|, |\tau|)$, also the i-th symbols in σ' and τ' are different as well. Hence $r' \in \mathsf{excluded}(\mathcal{E}, \mathcal{T}', \sigma')$. □

This means that as the observation tree is extended, each state can exclude more and more states. The more other states a state excludes, the better it can be identified, meaning the identification power increases as well. In fact, the identification power increases monotonically as the observation tree is extended.

Theorem 1. *Let $\mathcal{T}$ be an observation tree, let $\mathcal{T}'$ be an extension of $\mathcal{T}$, and let U be a subset of $Q^{\mathcal{T}}$. Let $\mathcal{E}$ be an adaptive experiment such that* $\mathsf{outcome}(\mathcal{E}, \mathcal{T}' \dagger q) \downarrow$ *for all $q \in Q^{\mathcal{T}'}$. Then* $\mathsf{IP}(\mathcal{E}, U, \mathcal{T}') \geq \mathsf{IP}(\mathcal{E}, U, \mathcal{T})$.

Proof. By Lemma 6, $\mathsf{excluded}(\mathcal{E}, \mathcal{T}, \sigma)$ grows monotonically for any σ as $\mathcal{T}$ is extended. This means that in the formula for identification power, $|\,\mathsf{excluded}(\mathcal{E}, \mathcal{M}, \mathsf{outcome}(\mathcal{E}, \mathcal{M} \dagger q)) \cap U\,|$ will increase monotonically as $\mathcal{T}$ is extended. Therefore, the identification power will be higher as $\mathcal{T}'$ is extended and hence $\mathsf{IP}(\mathcal{E}, U, \mathcal{T}') \geq \mathsf{IP}(\mathcal{E}, U, \mathcal{T})$. □

Chapter 4

Conclusions

This chapter summarizes our main contributions and provides some suggestions for future research.

4.1 Main contributions

This thesis studies adaptive experiments and various related concepts in a setting of Mealy machines and observation trees. In particular, we formally defined the extension of observation trees, adaptive experiments, adaptive distinguishing experiments, and identification power. Furthermore we formulated definitions for applying an adaptive experiment to a Mealy machine and the result of this operation, as well as for deriving an adaptive experiment from an observation tree.

We found that our method for deriving an experiment from an observation tree provides useful experiments, i.e. the derived experiment produces an output for the states that it was based on. We derived the lower and upper bound for the notion of identification power, and showed that an adaptive distinguishing experiment has maximal identification power. Lastly, we found that adaptive experiments gain more identification power as the observation tree they are applied to is extended.

4.2 Suggestions for future work

As mentioned in Sec. 3.3, a particularly interesting use for adaptive experiments could be in the $L^{\#}$ algorithm. At several points during its executing, the $L^{\#}$ algorithm will consider a hypothesis $\mathcal{H}$, which is its best approximation for the target Mealy machine $\mathcal{M}$. One could construct an adaptive experiment $\mathcal{E}$ for this $\mathcal{H}$, for instance by using the method described by [Lee and Yannakakis, 1994]. If the identification power of $\mathcal{E}$ is high, it is possible that it could be used to pose some useful output queries.

If for instance the identification power of $\mathcal{E}$ is x, this implies that there are x apartness pairs that can be derived from $\mathcal{E}$. If $\mathcal{E}$ is then applied to the basis states of the observation tree, one would expect that afterwards these same apartness pairs are also contained in the observation tree. This would mean that the identification power of the experiment derived from the observation tree is now at least x as well.

Theorem 1 shows that the identification power of the experiment derived from the observation tree will at least be higher, but whether its identification power would indeed rise to at least x is a topic that needs further research.

As pointed out in Sec. 3.3, it would be useful to show that the experiment obtained by the method described in Def. 12 is the best possible adaptive experiment for that observation tree and that set of states. This seems plausible, as it is based on reward and maxsymbol which are designed to select the best possible input symbol, but it would be valuable to prove this formally as well.

Sec. 3.4 defined the notion of identification power and Sec. 3.3 defined the reward function. These concepts are very similar and it appears that they are in fact equal. More formally, for an observation tree $\mathcal{T}$, a set of states $U \subseteq Q^{\mathcal{T}}$, and $\mathcal{E}$ the experiment derived from $\mathcal{T}$ for U, it seems to be the case that $\mathsf{IP}(\mathcal{E}, \mathcal{T}, U) = \mathsf{reward}(U, \mathcal{T})$. A formal proof of this remains to be constructed however.

This paper has only considered Mealy machines. However, many other types of Finite State Machines exist, for instance pushdown automata, probabilistic automata, and timed automata. Future reseach could look into whether the concepts explored here apply to these FSMs as well.

Chapter 5

Acknowledgements

First of all, I owe a huge thanks to my supervisor prof. dr. Frits Vaandrager and second reader Bharat Garhewal. Their patience with me as I tried to grasp the working of their algorithm was near infinite and I learned incredibly much thanks to the time and advice they offered me. They also helped with the writing style of this report and pointed out many errors and possible improvements in the first version. I should also mention that Def. 5 and the basis for Def. 6 were provided by Frits.

I would like to also thank my parents, sister, girlfriend, and my friend Aron, for enduring very long answers to the question “so what’s your thesis about?” A question that for longest time I did not know the answer to myself, let alone could answer in an effective way. Trying to condense the explanation of my thesis topic so that it did not take several hours was a very useful exercise and I would have gotten much more stuck without them offering their attention.

Bibliography

Angluin, D. (1987). Learning regular sets from queries and counterexamples. *Information and computation*, *75*(2), 87–106.

Lee, D., & Yannakakis, M. (1994). Testing finite-state machines: State identification and verification. *IEEE Transactions on computers*, *43*(3), 306–320.

Mealy, G. H. (1955). A method for synthesizing sequential circuits. *The Bell System Technical Journal*, *34*(5), 1045–1079.

Moore, E. F. (1956). Gedanken-experiments on sequential machines. *Automata Studies*, (34), 129–153.

Vaandrager, F., Garhewal, B., Rot, J., & Wißmann, T. (2021). A new approach for active automata learning based on apartness. *arXiv preprint arXiv:2107.05419*.

van den Bos, P., & Vaandrager, F. (2021). State identification for labeled transition systems with inputs and outputs. *Science of Computer Programming*, *209*, 102678. https://doi.org/https://doi.org/10.1016/j.scico.2021.102678

Wikipedia contributors. (2021). Mealy machine — Wikipedia, the free encyclopedia [[Online; accessed 11-October-2021]].