CDMTCS Research Report Series

Deterministic Incomplete Automata: Simulation, Universality and Complementarity

Elena Calude

Marjo Lipponen
Department of Computer Science
University of Auckland
Auckland, New Zealand

CDMTCS-040
June 1997

Centre for Discrete Mathematics and
Theoretical Computer Science

Deterministic Incomplete Automata: Simulation, Universality and Complementarity*

Elena Calude[†] and **Marjo Lipponen**[‡]

Abstract

We study finite deterministic incomplete automata *without initial states.* This means that at any stage of a computation there is at most one transition to the next state. We will first investigate how two incomplete automata can simulate each other. Further on we construct an incomplete automaton which simulates a given automaton S and has the minimum number of states compared to any other automaton simulating S. Finally, we study Moore's uncertainty principles for incomplete automata. In contrast with the case of complete automata, it is possible to construct incomplete three-state automata displaying both types of complementarity.

1 Introduction

The theory of relativity and quantum mechanics have altered the classical concept of physical objectivity: the experimenter is situated *in* the universe and can be modeled as a "sturdy, classical entity" composed of a macroscopic number of microscopic objects. The experimenter is bound by complementarity: he experiences either a certain type of observation or a different, complementary one. This complementarity is tied up with measurement, making it a highly controversial matter (see, for example, Wigner [15], Wheeler [13], and Bell [1]). As Greenberg [10] mentions, in certain instances it is even possible to "reconstruct" the quantum wave function after its so-called "collapse", where not a single quantum bit of information remains available from the "measurement".

Moore [12] was the first to study some experiments on finite deterministic automata in an attempt to understand what kind of conclusions about the internal conditions of a finite machine it is possible to draw from input-output experiments.[1] A Moore experiment can be described as follows: a copy of the machine will be experimentally observed, i.e. the experimenter will input a finite sequence of input symbols to the machine and will observe the sequence of output symbols. The correspondence between input and output symbols depends on the particular chosen machine and on its state at the beginning of the experiment. The experimenter will study sequences of input and output symbols and will try to conclude that "the machine being experimented on was in state q at the beginning of the experiment". Moore's experiments have been studied from a mathematical point of view by various researchers, notably by Ginsburg [8], Gill [7], Chaitin [5], Conway [6], Brauer [2], Calude, Calude, Svozil and Yu [4]. The main conclusion of these studies is that *it is impossible to determine the initial state of an automaton* and, consequently, the classical

*The first author was partially supported by AURC Grant A18/62090/F3414070. The second author is supported by Alfred Kordellin Foundation.

†Computer Science Department, The University of Auckland, Private Bag 92109, Auckland, New Zealand, e-mail: elena@cs.auckland.ac.nz.

‡Computer Science Department, The University of Auckland, Private Bag 92109, Auckland, New Zealand; on leave from the Department of Mathematics, University of Turku, Finland; e-mail: marjo@cs.auckland.ac.nz.

[1] From a physical point of view, the working hypothesis is that *a particle is a deterministic finite automaton*, i.e. acts as a machine having a finite number of states, a finite number of input symbols, and a finite number of output symbols.

theory of finite deterministic automata – which considers automata with initial states – is not adequate. The first step to remedy this situation was to study finite deterministic automata *without initial states*, see Calude, Calude and Khoussainov [3]. The second natural step is to take into consideration the fact that some internal transitions of the automaton *cannot be known* or *measured*, leading to the notion of *incomplete automata without initial states.*

We will first define and study the complexity of various types of simulations between incomplete automata. Minimal incomplete automata will be constructed and proven to be unique up to an isomorphism; this situation parallels and extends the theory of complete automata without initial states (see Calude, Calude, Khoussainov[3]), but heavily contrasts with the case of incomplete Mealy automata *without* initial states where "... there frequently is more than one minimal–state machine of an incomplete machine S" (see Ginsburg [8, p. 46]).[2]

As in the case of complete automata, we will build our results on an extension of Myhill–Nerode technique; all constructions will only make use of "automata responses" to simple experiments, i.e., no information about the internal machinery will be considered available.

In the second part of the paper we will study the notions of complementarity CI and CII, introduced in Calude, Calude, Svozil and Yu [4]. Both of these phenomena are closely related to Heisenberg's Uncertainty Principle in Physics, as was noticed by F.E. Moore [12] already in the late 50's. In Physics it is impossible to measure both the velocity and the position of an electron without affecting the other one; in automata theory the information of knowing that any two given states are distinguishable is of no use when we try to find for each state a single experiment which distinguishes it from the other states.

In contrast to the case of complete automata, some experiments performed on incomplete automata may not be relevant; this makes these devices even better models for physical reality. Two major differences between complete and incomplete automata found in this paper are the following:

A) Complementarity properties CI/CII cannot hold for a complete automaton with less than four states; however, there exist three-state incomplete automata having CI/CII.

B) The case of having only one element in an output alphabet is trivial and not relevant for complete automata; on the other hand, we are able to construct, for any given $n \geq 3$, an incomplete automaton having n states with the same output and still have properties CI/CII.

1.1 Notations

If S is a finite set, then $|S|$ denotes the cardinality of S. A **partial** function $f : A \stackrel{\circ}{\to} B$ is a function defined for some elements from A. In case f is not defined on $a \in A$ we write $f(a) = \infty$. Let $D(f) = \{a \in A \mid f(a) \neq \infty\}$ denote the domain of f. If $D(f) = A$, we say that f is **total**. Two partial functions f and g are equal, when $D(f) = D(g)$ and $f(a) = g(a)$, for every $a \in D(f)$. If Σ is a finite set, called alphabet, then Σ^* stands for the set of all finite words over Σ; the empty word, denoted by λ. By w^+ we mean all nonempty powers w^i, $i > 0$, of the word $w \in \Sigma^*$ whereas w^* includes also w^0, the empty word. The length of a word w is denoted by $|w|$.

We fix two finite, nonempty alphabets Σ and O: Σ contains **input** symbols, and O contains **output** symbols. A **deterministic (finite) incomplete automaton** over the alphabets Σ and O is a system $A = (S_A, \Delta_A, F_A)$, where the **set of states** S_A is a finite, nonempty set, the **transition table** Δ_A is a partial function from $S_A \times \Sigma$ to the set of states S_A, and the **output function** F_A is a total mapping from the set of states S_A into output alphabet O.

Since Δ_A is a partial function, $D(\Delta_A)$ denotes the domain of Δ_A. Thus $D(\Delta_A)$ is a subset of $S_A \times \Sigma$ and needs not include every state–input pair. For a current state $q \in S_A$ and a current input

[2] Ginsburg continues by saying: "While it is easy to find a solution to the minimalization problem when the machine is complete, it is vastly more difficult when the machine is incomplete. In fact, there is still no complete satisfactory procedure for finding a solution to the incomplete case".

$\sigma \in \Sigma$, $\Delta_A(q, \sigma)$ is the next state of A if $(q, \sigma) \in D(\Delta_A)$; otherwise, the next state is undefined. Consequently, such automata are usually qualified as being incompletely specified; hence the name **incomplete**, see Ginsburg [11]. We will say that a finite automaton $A = (S_A, \Delta_A, F_A)$ is **complete** when Δ_A is a total function, so every complete automaton is a special case of an incomplete automaton. To make the difference clear, an incomplete automaton which has at least one state $s \in S_A$ and one letter $\sigma \in \Sigma$ such that $\Delta_A(s, \sigma) = \infty$ is called a **proper incomplete** automaton.

In this paper we will deal only with deterministic, incomplete or complete, automata; for this reason we will omit the word deterministic in the sequel.

Let $A = (S_A, \Delta_A, F_A)$ be an incomplete automaton. We will extend the transition diagram Δ_A to a partial function, also denoted by Δ_A, $\Delta_A : S_A \times \Sigma^* \stackrel{\circ}{\to} S_A$, as follows: for every $s \in S_A$, $w \in \Sigma^*$ and $\sigma \in \Sigma$,

$$\Delta_A(s, \lambda) = s, \text{ and}$$
$$\Delta_A(s, \sigma w) = \begin{cases} \Delta_A(\Delta_A(s, \sigma), w), & \text{if } \Delta_A(s, \sigma) \neq \infty, \\ \infty, & \text{otherwise.} \end{cases}$$

Furthermore, for all $p \in S_A$, the set $W_A(p) = \{w \in \Sigma^* \mid \Delta_A(p, w) \neq \infty\}$ consists of all words leading to complete computations on state p. Following Ginsburg [9], we say that a word u is **applicable to** the state p if $u \in W_A(p)$.

The following remarks are straight consequences of definitions.

Lemma 1.1 *Let $A = (S_A, \Delta_A, F_A)$ be an incomplete automaton. Then*

1) *For any word $w = \sigma_1\sigma_2 \ldots \sigma_n \in \Sigma^*$, $\sigma_i \in \Sigma$, $1 \leq i \leq n$, and a state $s \in S_A$, $\Delta_A(s, w) \neq \infty$ iff $\Delta_A(s, \sigma_1) \neq \infty$, $\Delta_A(s, \sigma_1\sigma_2) \neq \infty, \ldots,$ $\Delta_A(s, \sigma_1\sigma_2 \ldots \sigma_n) \neq \infty$.*

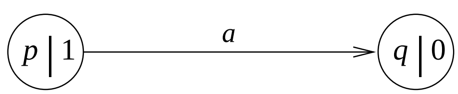
2) *For all $p \in S_A$ and $u, v \in \Sigma^*$, $vu \in W_A(p)$ iff $v \in W_A(p)$ and $u \in W_A(\Delta_A(p, v))$.*

3) *If $vu \in W_A(p)$, then $v \in W_A(p)$.*

4) *For all $p \in S_A$, $\lambda \in W_A(p)$.*

5) *A is complete iff for all $s \in S_A$, $W_A(s) = \Sigma^*$.*

In drawing graph representations of incomplete automata, we denote states by circles and label them with symbols from the output alphabet. For example, in the figure below there is a transition from p to q, labeled by a, that is $\Delta(p, a) = q$. The state p emits output 1, $F_A(p) = 1$, and q emits output 0, $F_A(q) = 0$.



We will next define, following Calude, Calude, Khoussainov [3], the response of an incomplete automaton $A = (S_A, \Delta_A, F_A)$ to an **input signal** $w \in \Sigma^*$.

- The **total response** of A is the partial function $R_A : S_A \times \Sigma^* \stackrel{\circ}{\to} O^*$,

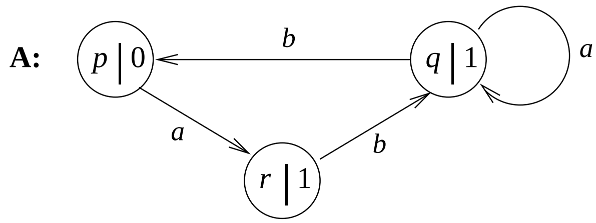
$$R_A(s, \lambda) = F_A(s), \text{ and}$$
$$R_A(s, \sigma_1 \ldots \sigma_n) = F_A(s)F_A(\Delta_A(s, \sigma_1))F_A(\Delta_A(s, \sigma_1\sigma_2)) \ldots F_A(\Delta_A(s, \sigma_1 \ldots \sigma_n)),$$

  for $s \in S_A$, $\sigma_1 \ldots \sigma_n \in W_A(s)$, $\sigma_i \in \Sigma$, $n \geq 1$ and $1 \leq i \leq n$.

- The **final response** of A is the partial function $f_A : S_A \times \Sigma^* \stackrel{\circ}{\to} O$, $f_A(s, w) = F_A(\Delta_A(s, w))$, for all $s \in S_A$ and $w \in W_A(s)$.

Thus, the total response is a sequence of outputs emitted by all the states that are visited in the complete computation of the input, whereas the final response is the output emitted only by the last state. Notice that $D(R_A) = D(f_A)$.

Example 1.1 Let $\Sigma = \{a, b\}$, $O = \{0, 1\}$ and consider the three-state incomplete automaton A presented below.



The output function is defined by $F_A(p) = 0$ and $F_A(q) = F_A(r) = 1$. Clearly, $R_A(p, aba) = 0111$, $R_A(q, aba) = 1101$, and $R_A(r, aba) = \infty$. We also have $f_A(p, aba) = 1$, $f_A(q, aba) = 1$, and $f_A(r, aba) = \infty$.

2 Simulations

We say that an incomplete automaton B simulates another incomplete automaton A if B can perform all computations performed by A *in the same way.* It turns out that there are various possibilities to model this intuitive notion.

2.1 Strong and Weak Simulations

Let $A = (S_A, \Delta_A, F_A)$ and $B = (S_B, \Delta_B, F_B)$ be incomplete automata, and fix a mapping $h : S_A \to S_B$. Consider the following conditions:

(i) For all $s \in S_A$, $W_A(s) \subseteq W_B(h(s))$.

(ii) For all $s \in S_A$, $W_A(s) = W_B(h(s))$.

(iii) For all $s \in S_A$ and $\sigma \in \Sigma \cap W_A(s)$, $h(\Delta_A(s, \sigma)) = \Delta_B(h(s), \sigma)$.

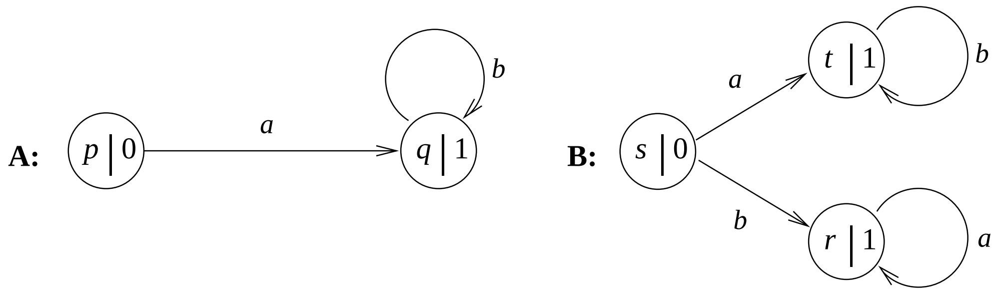
(iv) For all $s \in S_A$ and $w \in W_A(s)$, $R_A(s, w) = R_B(h(s), w)$.

(v) For all $s \in S_A$ and $w \in W_A(s)$, $f_A(s, w) = f_B(h(s), w)$.

We define two types of simulations: weak and strong. In the weak case, for every state $s \in S_A$ there is a state $h(s) \in S_B$ such that $h(s)$ does everything that s does (and possibly more) whereas in the strong simulation $h(s)$ does exactly what s does (and nothing more). Formally, we say that

- A is **weakly simulated** by B, $A \prec B$, if the mapping h satisfies (i), (iii) and (iv);
- A is **weakly f–simulated** by B, $A \prec_f B$, if the mapping h satisfies (i), (iii) and (v);
- A is **strongly simulated** by B, $A \ll B$, if the mapping h satisfies (ii), (iii) and (iv);
- A is **strongly f–simulated** by B, $A \ll_f B$, if the mapping h satisfies (ii), (iii) and (v).

Clearly, strong simulation implies weak simulation, but the converse implication is false as the following example shows.



Example 2.1 The incomplete automaton A is weakly (but not strongly) simulated by the incomplete automaton B above via the mapping $h : S_A \to S_B$, $h(p) = s$, $h(q) = t$; however, the converse implication fails to be true: there is no mapping from S_B to S_A which preserves the computational power of the state r.

We will next study the connection between the strong/weak simulation and f–strong/f–weak simulation. It turns out that it makes no difference whether we define simulation by total or final response. This is the case also for complete automata, see Calude, Calude, Khoussainov [3], and since the proofs are quite similar (we only have to restrict ourselves to applicable words of the given states) we omit them here.

Lemma 2.1 *If $h : S_A \to S_B$ and B strongly/weakly (or f-strongly/f-weakly) simulates A via h, then $h(\Delta_A(s,w)) = \Delta_B(h(s),w)$, for all $s \in S_A$, $w \in W_A(s)$.*

Note that for strong simulations the equality $h(\Delta_A(s,w)) = \Delta_B(h(s),w)$ holds actually true for all $s \in S_A$ and $w \in \Sigma^*$, since $\Delta_A(s,w) = \Delta_B(h(s),w) = \infty$, for $w \notin W_A(s)$. For weak simulations this is not the case; for instance, in Example 2.1, $\Delta_A(p,b) = \infty$ but $\Delta_B(h(p),b) = \Delta_B(s,b) \neq \infty$.

Theorem 2.1 *For any incomplete automata A and B:*

1) *A is strongly simulated by B iff A is strongly f–simulated by B.*

2) *A is weakly simulated by B iff A is weakly f–simulated by B.*

The following result shows that in some sense all strong simulations preserve the completeness of automata.

Theorem 2.2 *Let A and B be incomplete automata.*

1) *If $A \ll B$ and B is complete, then A is also complete.*

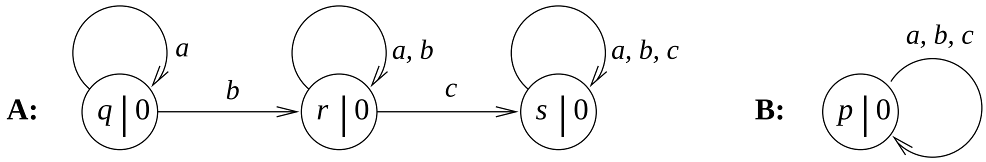
2) *If $A \ll B$ and A is properly incomplete, then B is also properly incomplete.*

Proof. Let $h : S_A \to S_B$ be the mapping verifying properties (ii), (iii) and (iv).

To prove 1) assume that B is a complete automaton. By Lemma 1.1, this is equivalent to the fact that $W_B(q) = \Sigma^*$, for any $q \in S_B$. Hence also $W_B(h(p)) = \Sigma^*$, for any $p \in S_A$, and since $W_A(p) = W_B(h(p))$ by property (ii), the automaton A is also complete.

To prove 2) assume that A is a proper incomplete automaton. This means that for some state $s \in S_A$, $W_A(s)$ is a proper subset of Σ^*. Again by property (ii), $W_B(h(s))$ is also a proper subset of Σ^*, which proves that B is properly incomplete. □

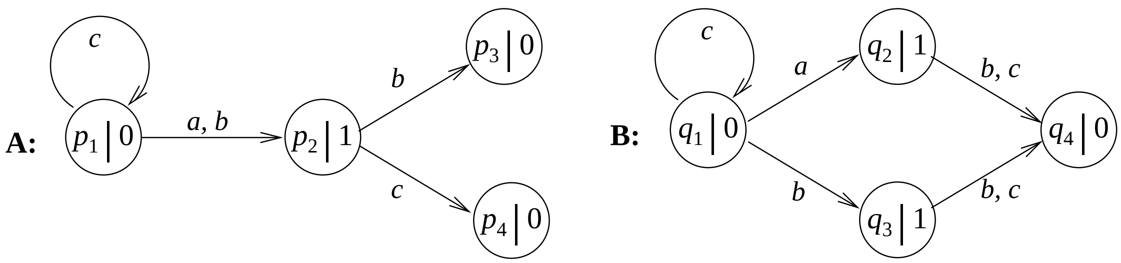
Corollary 2.1 *Let A and B be incomplete automata. If $A \ll B$ and $B \ll A$, then A and B are both either complete or proper incomplete automata.*



Theorem 2.2 and Corollary 2.1 are equally valid for strongly f–simulation because of Theorem 2.1. For weak simulations there is no corresponding result. In particular, Corollary 2.1 does not hold for weak simulations. For example, A and B above are weakly simulating each other, even though A is properly incomplete and B is complete.

2.2 Behavioral Simulations

In this section we discuss another notion of simulation, the behavioral simulation, which is weaker than all previous simulations; it makes use only of the outputs produced by the automaton, but not of the transition Δ_A (which cannot be measured under the physical interpretation of automata).



To motivate the following formalization, consider the incomplete automata A and B above. Neither A nor B is strongly (nor weakly) simulating the other one; nevertheless, if we consider the mapping $h_1 : S_A \to S_B$,

$$h_1(p_1) = q_1, \ h_1(p_2) = q_2, \ h_1(p_3) = q_4, \ h_1(p_4) = q_4,$$

we notice that for any state $x \in S_A$ and any word $w \in W_A(x)$, there is a state $h_1(x) \in S_B$ such that starting from this state, B responds to w in the same way as A starting from the state x. There is also a mapping $h_2 : S_B \to S_A$ having a similar behavior.

We are now ready to give formal definitions for behavioral simulations (called in what follows β-simulations). Let $A = (S_A, \Delta_A, F_A)$ and $B = (S_B, \Delta_B, F_B)$ be incomplete automata. We say that

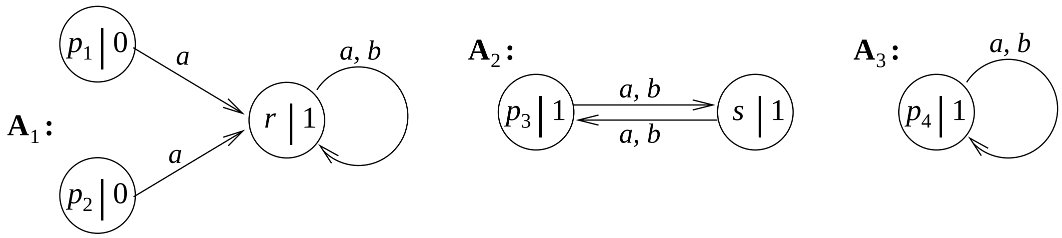
- A is **strongly β-simulated by** B, $A \ll_\beta B$, if there is a mapping $h : S_A \to S_B$ which satisfies conditions (ii) and (iv) in Section 2.1.
- A is **weakly β-simulated** by B, $A \prec_\beta B$, if there is a mapping $h : S_A \to S_B$ which satisfies conditions (i) and (iv) in Section 2.1.

We are not going to define behavioral f-simulation since the result analogue of Theorem 2.1 shows that this definition brings nothing new.

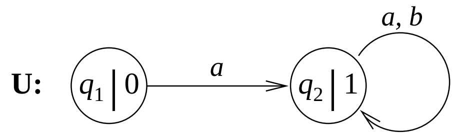
We also notice that Theorem 2.2 (and hence also Corollary 2.1) is valid for strong β-simulation.

Finally, notice that strong simulation implies both weak simulation and strong β-simulation and weak simulation implies weak β-simulation, but weak simulation and strong β-simulation are incomparable; for instance, in Example 2.1 automaton A is weakly simulated but not strongly β-simulated by B, whereas in the example above A and B are strongly β-simulating but not weakly simulating each other.

3 Universal Incomplete Automata



Consider the following class of incomplete automata with initial states: $\mathcal{C} = \{(A_1, p_1),\ (A_1, p_2),\ (A_2, p_3),\ (A_3, p_4)\}$, where A_1, A_2 and A_3 are given above.



The incomplete automaton U has the following properties:

1. Each incomplete automaton A_i is strongly β–simulated by U with an adequately chosen initial state from S_U:
   - $W_{A_1}(p_1) = W_U(q_1)$ and $R_{A_1}(p_1, w) = R_U(q_1, w)$, for all $w \in W_{A_1}(p_1)$,
   - $W_{A_1}(p_2) = W_U(q_1)$ and $R_{A_1}(p_2, w) = R_U(q_1, w)$, for all $w \in W_{A_1}(p_2)$,
   - $W_{A_2}(p_3) = W_U(q_2)$ and $R_{A_2}(p_3, w) = R_U(q_2, w)$, for all $w \in W_{A_2}(p_3)$,
   - $W_{A_3}(p_3) = W_U(q_2)$ and $R_{A_3}(p_4, w) = R_U(q_2, w)$, for all $w \in W_{A_3}(p_3)$.
2. The automaton U, starting from any of its states, is strongly β–simulated by some automaton (A_i, p_j) from $\mathcal{C}$:
   - $R_U(q_1, w) = R_{A_1}(p_1, w)$, for all $w \in W_U(q_1)$,
   - $R_U(q_2, w) = R_{A_3}(p_4, w)$, for all $w \in W_U(q_2)$.

We say that U is a "universal incomplete automaton" for the class $\mathcal{C}$. Our aim in the sequel is to define the notion of universal incomplete automaton and to prove that every finite class $\mathcal{C}$ can be embedded into a class having a universal incomplete automaton which is unique up to an isomorphism; finally, we show that universality is related to minimality.

Suppose that we have a finite class $\mathcal{C}$ containing pairs (A_i, q_i) of incomplete automata $A_i = (S_i, \Delta_i, F_i)$ and initial states $q_i \in S_i$, $i = 1, \ldots, n$. An incomplete automaton $U_C = (S_{U_C}, \Delta_{U_C}, F_{U_C})$ is **universal** for the class $\mathcal{C}$ if the following two conditions hold:

1. For any $1 \leq i \leq n$, there is a state $s \in S_{U_C}$ such that $W_{U_C}(s) = W_{A_i}(q_i)$ and $R_{U_C}(s, w) = R_{A_i}(q_i, w)$, for all $w \in W_{U_C}(s)$.
2. For any $s \in S_{U_C}$, there is an i, $1 \leq i \leq n$, such that $W_{U_C}(s) = W_{A_i}(q_i)$ and $R_{U_C}(s, w) = R_{A_i}(q_i, w)$, for all $w \in W_{A_i}(q_i)$.

It is not hard to see that every incomplete automaton V (with no initial state) naturally defines a class $\mathcal{C}(V)$ for which V itself is universal. Indeed, let $q_1, \ldots, q_n \in S_V$ be all the states of V, and for each i, define $A_i = V$. Clearly V is universal for the class $\mathcal{C}(V) = \{(A_1, q_1), \ldots, (A_n, q_n)\}$.

Not every finite class of finite incomplete automata with initial states has a universal incomplete automaton; nevertheless, we can always enlarge it to one which has this property.

Proposition 3.1 *Every finite class of pairs of incomplete automata and initial states can be embedded into a finite class which has at least one universal incomplete automaton.*

Proof. Let $\Gamma = \{(A_i, q_i) \mid 1 \leq i \leq n\}$, where $A_i = (S_i, \Delta_i, F_i)$. Assume that all the states of these incomplete automata are pairwise disjoint. Consider the incomplete automaton U,

$$U = (\cup_i^n S_i,\ \cup_i^n \Delta_i,\ \cup_i^n F_i).$$

Let $\mathcal{C}(U)$ be the class as defined above and take $B = \Gamma \cup \mathcal{C}(U)$. It is easy to see that Γ is contained in B and U is universal for B. □

Theorem 3.1 *The incomplete automata A and B strongly β–simulate each other iff A and B are universal for the same class.*

Proof. Suppose that A and B strongly β–simulate each other via $h_1 : S_A \to S_B$ and $h_2 : S_B \to S_A$. Consider the class $\mathcal{C}(A)$, for which the incomplete automaton A is universal. We show that B is universal for $\mathcal{C}(A)$. Suppose that (A_1, q_1) belongs to $\mathcal{C}(A)$. Then for all $w \in W_{A_1}(q_1)$, we have $R_{A_1}(q_1, w) = R_B(h_1(q_1), w)$. For every $q \in S_B$ there exists a state $q' = h_2(q) \in S_A$ such that for the pair (A, q') we have $R_A(q', w) = R_B(q, w)$, for all $w \in W_A(q')$. Hence B is also universal for $\mathcal{C}(A)$.

Now assume that A and B are both universal for the class $\mathcal{C} = \{(A_1, q_1), (A_2, q_2), \ldots, (A_n, q_n)\}$. For every $q \in S_A$ there exists $i \in \{1, 2, \ldots, n\}$, such that $R_A(q, w) = R_{A_i}(q_i, w)$, for all $w \in W_A(q)$. Since $(A_i, q_i) \in \mathcal{C}$ and B is universal for $\mathcal{C}$ there is a state $p_i \in S_B$ – say, the minimal one according to a fixed linear order defined on the set of all states – such that $R_{A_i}(q_i, w) = R_B(p_i, w)$, for all $w \in W_{A_i}(q_i)$. Hence A is strongly β–simulated by B via mapping $q \mapsto p_i$. Similarly, B can be strongly β–simulated by A. □

An incomplete automaton A which is universal for the class $\mathcal{C}$ is said to be **minimal** if it has the least number of states compared with all other incomplete automata universal for the same class.

From this definition and Theorem 3.1 above we obtain:

Corollary 3.1 *The following statements are equivalent:*

1) *The incomplete automaton A is a minimal universal automaton for a class $\mathcal{C}$.*

2) *For every incomplete automaton B universal for $\mathcal{C}$, if $A \ll_\beta B$ and $B \ll_\beta A$, then $|S_A| \leq |S_B|$.*

We will now show how to construct a minimal incomplete automaton for each class $\mathcal{C}(A)$ where A is an incomplete automaton, using the generalized Myhill–Nerode equivalence relation. Let K be one of the response functions on A, $K \in \{R_A, f_A\}$. Two states p and q from S_A are K–**equivalent** if

$$W_A(p) = W_A(q) \quad \text{and} \quad K(p, w) = K(q, w), \text{ for all } w \in W_A(p).$$

If p and q are K–equivalent we denote this fact by $p \equiv_K q$. Intuitively, p and q are K–equivalent, when all computations of A which begin from p cannot be K–distinguished by computations of A which begin from q and vice versa. It follows immediately that $\equiv_K$ is an equivalence relation on S_A. We also obtain the following results similar to complete automata, see Calude, Calude, Khouissanov [3].

Lemma 3.1 *Let p and q be any states of an incomplete automaton $A = (S_A, \Delta_A, F_A)$. Then*

1) $p \equiv_{R_A} q$ *iff* $p \equiv_{f_A} q$.

2) $p \equiv_{f_A} q$ *implies* $\Delta_A(p, w) \equiv_{f_A} \Delta_A(q, w)$, *for all* $w \in W_A(p)$.

3) $p \equiv_{f_A} q$ *implies* $F_A(p) = F_A(q)$.

Since $\equiv_{R_A}$ and $\equiv_{f_A}$ are equivalent by the lemma above, we will simply use $\equiv$ in the sequel. For any state $s \in S_A$, let $[s]$ denote the equivalence class of s under $\equiv$, that is, $[s] = \{p \in S_A \mid s \equiv p\}$. Define a new automaton $M(A) = (S_{M(A)}, \Delta_{M(A)}, F_{M(A)})$ such that $S_{M(A)} = \{[s] \mid s \in S_A\}$, and for all $[s] \in S_{M(A)}$,

$$F_{M(A)}([s]) = F_A(s), \text{ and}$$
$$\Delta_{M(A)}([s], \sigma) = \begin{cases} [\Delta_A(s,\sigma)], & \text{if } \sigma \in \Sigma \cap W_A(s), \\ \infty, & \text{otherwise.} \end{cases}$$

Because of Lemma 3.1, we have indeed a well-defined automaton and it turns out to be unique up to an "isomorphism". An incomplete automaton $A = (S_A, \Delta_A, F_A)$ is said to be **isomorphic** to $B = (S_B, \Delta_B, F_B)$ if there is a one-to-one onto mapping $h : S_A \to S_B$ such that for all $s \in S_A$, $F_A(s) = F_B(h(s))$, $W_A(s) = W_B(h(s))$, and $h(\Delta_A(s,\sigma)) = \Delta_B(h(s),\sigma)$, for any $\sigma \in W_A(s) \cap \Sigma$. Clearly, if A is isomorphic to B, then A and B strongly simulate (hence strongly β–simulate) each other. However, the converse implication is not obviously always true.

The construction of minimal automata follows Calude, Calude, Khoussainov [3], and we easily obtain the following results using essentially the same argument.

Theorem 3.2 *For every incomplete automaton A,*

1) *$M(A)$ and $M(M(A))$ are isomorphic.*

2) *$M(A)$ and A strongly β-simulate each other.*

3) *$M(A)$ is minimal.*

4) *If B and A strongly β-simulate each other and B is minimal, then $M(A)$ and B are isomorphic.*

To summarize:

Corollary 3.2 *For any two minimal incomplete automata A and B the following are equivalent:*

1) *A and B strongly simulate each other.*

2) *A and B strongly f-simulate each other.*

3) *A and B strongly β-simulate each other.*

4) *A and B are isomorphic.*

Corollary 3.3 *Let $\mathcal{C}$ be a finite class of incomplete automata with initial states and M the unique (up to an isomorphism) universal minimal incomplete automaton for $\mathcal{C}$. Then M is a complete automaton iff all automata in $\mathcal{C}$ are complete.*

4 Complementarity

4.1 Indistinguishability

Following the study initiated in Moore [12], think of an incomplete automaton as a black box. Assume that we want to "distinguish" between two states p and q of the automaton A by means of a "measurable experiment", i.e. by the responses of the automaton to an input $w \in \Sigma^*$. There are two basic possibilities:

(i) The experiment is *not relevant* in case $\Delta_A(p,w) = \Delta_A(q,w) = \infty$ (so $R_A(p,w) = R_A(q,w) = \infty$); hence, another experiment is required.

(ii) The experiment is *relevant* in case w is applicable to at least one state p or q; in this case we have two further possibilities:

(a) w *distinguishes* between p and q in case w is applicable to both of them and $R_A(p,w) \neq R_A(q,w)$, or w is applicable to either p or q, but not to both.

(b) w *does not distinguish* between p and q in case w is applicable to both of them and $R_A(p,w) = R_A(q,w)$.

To summarise, w distinguishes between p and q if $R_A(p,w) \neq R_A(q,w)$ (meaning that either w is applicable to both p and q and the responses are different or w is applicable to only one of the states). In the remaining cases, w may not distinguish or may not be relevant for distinguishing between p and q. The above facts motivate the introduction of the following notions.

Let $A = (S_A, \Delta_A, F_A)$ be an incomplete automaton. The states $p, q \in S_A$ are **indistinguishable** iff

$$W_A(p) = W_A(q) \quad \text{and} \quad R_A(p,w) = R_A(q,w), \text{ for all } w \in W_A(p).$$

We notice that p and q are indistinguishable iff they are R_A-equivalent as in Section 3. Hence, because of Lemma 3.1, we can define indistinguishability by final response f_A instead of total response R_A.

If the states p and q are not indistinguishable, we say that they are **distinguishable**, and every word from the set

$$\{w \in W_A(p) \cup W_A(q) \mid R_A(p,w) \neq R_A(q,w)\} \tag{1}$$

is said to **distinguish** between p and q. In the same way, a word w **cannot distinguish** between p and q if $R_A(p,w) = R_A(q,w)$ or $w \notin W_A(p) \cup W_A(q)$.

4.2 Computational Complementarity

Let $A = (S_A, \Delta_A, F_A)$ be an incomplete automaton. Following the terminology of Calude, Calude, Svozil, Yu [4], we define the properties **A**, **B** and **C** as follows.

A Every pair of the distinct states of A are distinguishable.

B For every state p of A there exists a word $w \in \Sigma^*$ which distinguishes p from all the other states.

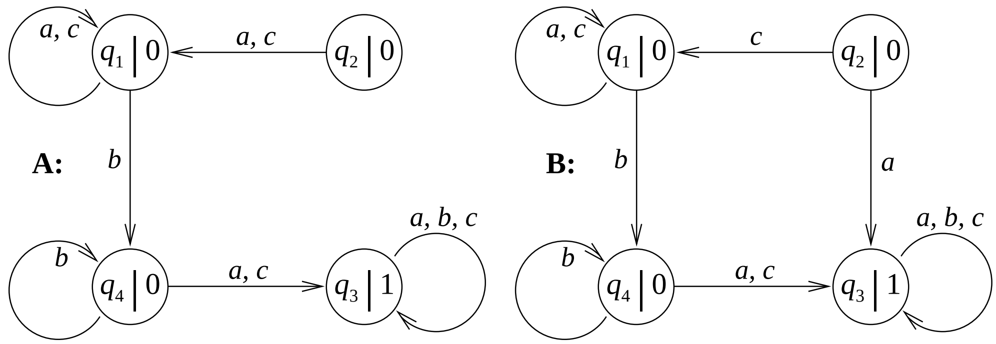
C There exists a word w which distinguishes between any two distinct states of A.

The complementarity properties CI (property **A** but not **B**) and CII (property **B** but not **C**) can be stated as follows.

CI Every state is distinguishable from the others but there is some state p for which there is no single experiment that distinguishes p from the other states.

CII For every state there is a word which distinguishes it from the other states but there is no single experiment which can distinguish between any two given states of the automaton.

Example 4.1 Automaton A below has property **A** since any two states are distinguishable. For instance, the word $w = b$ distinguishes between q_1 and q_2 since $w \in W_A(q_1) \setminus W_A(q_2)$. But A does not have property **B**. If we choose $w = ax$ or $w = cx$, $x \in \Sigma^*$, then the states q_1 and q_2 are indistinguishable and if we choose $w = bx$ then q_1 and q_4 are indistinguishable. Hence there is no experiment which distinguishes the state q_1 from the other states.



On the other hand, automaton B has property **B** since by choosing the words a, b, c, and a for the states q_1, q_2, q_3, and q_4, respectively, each of them can be distinguished from the other states. There is, however, no single experiment which can distinguish between any two states. For $w = ax$, where $x \in \Sigma^*$, the states q_2 and q_4, for $w = bx$ the states q_1 and q_4, and finally for $w = cx$, the states q_1 and q_2, respectively, are indistinguishable.

4.3 Deciding Properties A, B, C

For complete automata the properties **A**, **B** and **C** were shown to be decidable in Calude, Calude, Svozil, Yu [4]. In this section we will construct an algorithm which is more general than the one presented in Calude, Calude, Svozil, Yu [4], since it is applicable also for complete automata.

Let $A = (S_A, \Delta_A, F_A)$ be an incomplete automaton. Having property **A** means that for each pair of its distinct states p and q, there is a word $w \in W_A(p) \cup W_A(q)$ for which $R_A(p,w) \neq R_A(q,w)$. In other words the distinguishing sets

$$R_{p,q} = \{w \in W_A(p) \cup W_A(q) \mid R_A(p,w) \neq R_A(q,w)\}$$

are nonempty for all $p, q \in S_A$, $p \neq q$.

Consider, for each state $q \in S_A$, the finite deterministic automaton

$$M_q = (S', \Sigma' \times O, q, \delta, \Lambda)$$

where $S' = S_A \cup \{\#\} \cup \{\Lambda\}$ and $\Sigma' = \Sigma \cup \{\lambda\}$, q is the initial and Λ the final state. The transition function δ is defined, for $p \in S_A$, $\sigma \in \Sigma$, $\tau \in O$,

$$\delta(p,(\sigma,\tau)) = \begin{cases} \Delta_A(p,\sigma) & \text{if } F_A(p) = \tau,\ \sigma \neq \lambda, \text{ and } \Delta_A(p,\sigma) \neq \infty, \\ \Lambda & \text{if } F_A(p) = \tau, \text{ and } \sigma = \lambda, \\ \# & \text{otherwise,} \end{cases}$$

and, for $p \in \{\#, \Lambda\}$, $\sigma \in \Sigma'$, $\tau \in O$,

$$\delta(p,(\sigma,\tau)) = \#.$$

It follows immediately that for $p \in S_A$ and $(\sigma_1,\tau_1)\cdots(\sigma_n,\tau_n) \in (\Sigma' \times O)^*$,

$$\overline{\delta}(p,(\sigma_1,\tau_1)\cdots(\sigma_n,\tau_n)) = \Lambda \text{ iff } \sigma_1,\ldots,\sigma_{n-1} \in \Sigma, \sigma_n = \lambda, \text{ and } R_A(p,\sigma_1\ldots\sigma_{n-1}) = \tau_1\ldots\tau_n,$$

and, hence,

$$L(M_q) = \{(w\lambda, R_A(q,w)) \mid w \in W_A(q)\}.$$

But this means that properties **A**, **B** and **C** are algoritmically decidable. Indeed, we have

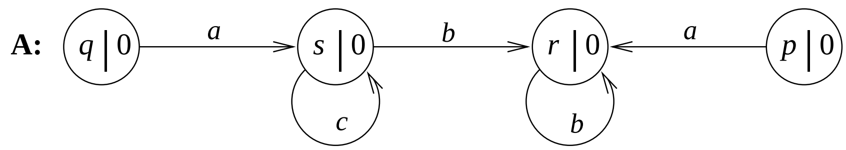
1. A has **A** iff $R_{p,q} \neq \emptyset$, for all $p, q \in S_A$, $p \neq q$,

2. A has **B** iff for all $q \in S_A$, $\bigcap_{p \neq q} R_{p,q} \neq \emptyset$,

3. A has **C** iff $\bigcap_{q \in S_A} \bigcap_{p \neq q} R_{p,q} \neq \emptyset$,

and, for any states p and q,

$$R_{p,q} \neq \emptyset \quad \text{iff} \quad (L(M_p) \cup L(M_q)) \setminus (L(M_p) \cap L(M_q)) \neq \emptyset.$$



Example 4.2 Consider the automaton A above. We notice that A has property CI since all states are distinguishable but for the states p and s there is no single experiment which distinguishes them from all the other states. The applicable words for each state are the following

$$W(q) = \{\lambda, ac^*b^*\},\ W(r) = \{c^*b^*\},\ W(s) = \{b^*\},\ W(p) = \{\lambda, ab^*\},$$

and the distinguishing sets

$$\begin{array}{lll} R_{q,r} = \{ac^*b^*, c^+b^*, c^*b^+\}, & R_{q,s} = \{ac^*b^*, b^+\}, & R_{q,p} = \{ac^+b^*\}, \\ R_{r,p} = \{ab^*, c^+b^*, c^*b^+\}, & R_{r,s} = \{c^+b^*\}, & R_{s,p} = \{b^+, ab^*\}. \end{array}$$

Hence A cannot have property **B** since

$$R_{q,p} \cap R_{r,p} \cap R_{s,p} = \emptyset = R_{q,s} \cap R_{r,s} \cap R_{s,p},$$

but it has **A** since every distinguishing set is nonempty.

We end this section by showing that the decidability of property **A** in incomplete case can be drawn also from the known result of the complete case, see Conway [6].

Let $A = (S_A, \Delta_A, F_A)$ be an incomplete automaton. We will turn it complete by adding an extra state, denoted by X, which takes care of all undefined transitions of A. Formally, define $A^\alpha = (S_{A^\alpha}, \Delta_{A^\alpha}, F_{A^\alpha})$ such that $S_{A^\alpha} = S_A \cup \{X\}$. The transition function Δ_{A^α} is the same as Δ_A, except that $\Delta_{A^\alpha}(p, \sigma) = X$ if $\Delta_A(p, \sigma) = \infty$, and $\Delta_{A^\alpha}(X, \sigma) = X$ for all $\sigma \in \Sigma$. In order to distinguish X from the other states we have to make sure that it produces a different output, $F_{A^\alpha}(X) = x$ where $x \notin O$.

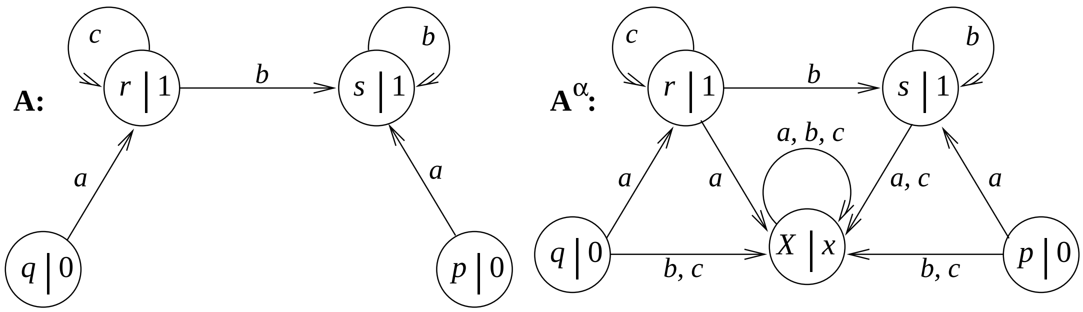
Lemma 4.1 *An incomplete automaton A has property* **A** *iff its completion automaton A^α has property* **A**.

Proof. Assume first that A has property **A**, i.e., every pair of its distinct states are distinguishable. Since the new state X produces a different output, it can be distinguished from the other states by the empty word. On the other hand, for any pair of distinct states of A there is a word which is either applicable to both of these states or to one of them. In both of these cases w can distinguish the states also in A^α.

Assume now that A does not have **A** and let p and q be indistinguishable, i.e., $W_A(p) = W_A(q)$ and $R_A(p, w) = R_A(q, w)$ for all $w \in W_A(p)$. Take any word $w \in \Sigma^*$. We can write it in such a way that $w = u\sigma v$ where $u \in W_A(p)$ (u may also be empty) and $u\sigma \notin W_A(p)$. But this means that $\Delta_{A^\alpha}(p, u\sigma) = X = \Delta_{A^\alpha}(q, u\sigma)$ and, hence, $R_{A^\alpha}(p, w) = R_{A^\alpha}(q, w)$. So p and q must be indistinguishable also in A^α. □

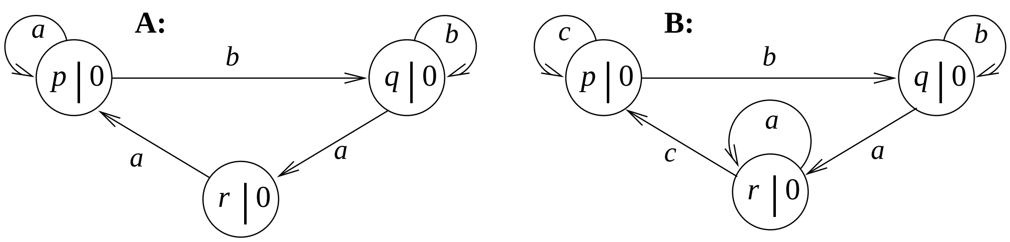
The following example shows that the completion automaton A^α does not preserve properties **B** and **C**. This is due to the fact that the states p and q can be distinguished in A by the word w which is not applicable to both of these states in A but will become applicable in A^α.

Example 4.3 Automaton A below has property CI ($\mathbf{A}\backslash\mathbf{B}$) since there is no word which distinguishes p or s from the other states. But the new automaton A^{α} has $\mathbf{B}$ since now the word $w = ac$ distinguishes p and $w = c$ distinguishes s, respectively, from the other states.



4.4 Complexity Issues

We start this section with an interesting observation about incomplete automata. The following example shows that, contrary to the case of complete automata (see Conway [6]), complementary properties CI and CII can be found already for three-state automata. We also notice that because of the definition of indistinguishability these properties may exist even though the output alphabet has only one letter.



Example 4.4 Consider automaton A above. It has clearly property $\mathbf{A}$ since, for instance, $w = b$ distinguishes between p and r ($R_A(p, w) = 00 \neq \infty = R_A(r, w)$). But A does not have property $\mathbf{B}$ since the state p does not have any single experiment which distinguishes it from the other states. Indeed, for $w = ax$, the states p and q, and for $w = bx$, the states p and r, respectively, are indistinguishable. Thus A has CI.

In the same way automaton B has CII, since now the words b, a, and c can distinguish the states p, q, and r, respectively, from the other states, but there is no single experiment which can distinguish between any two given states. For $w = ax$ the states r and q, for $w = bx$ the states p and q, and finally for $w = cx$ the states p and r, respectively, are indistinguishable.

Another interesting problem is to consider the length of the shortest experiment needed to decide properties $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$. So far only property $\mathbf{A}$ has been throughly studied, see for instance Chaitin [5], Conway [6]. A short overview of this problem for properties $\mathbf{B}$ and $\mathbf{C}$ can be found in Calude, Calude, Svozil, Yu [4].

Proposition 4.1 (Conway) *Let $K = (S_K, \Delta_K, F_K)$ be a complete automaton in a binary alphabet, $|\Sigma| = 2$. To test property $\mathbf{A}$ for K it is sufficient to test the condition*

$$R_K(q, w) \neq R_K(p, w) \tag{2}$$

for all words of length $|S_K| - 2$.

Using Lemma 4.1 we can extend this result to incomplete automata though we loose the original bound.

Theorem 4.1 *The length of the shortest experiment needed to decide property* **A** *for an incomplete automaton* $K = (S_K, \Delta_K, F_K)$ *in a binary alphabet is* $|S_K| - 1$.

Proof. Consider the complete automaton $K^\alpha = (S_{K^\alpha}, \Delta_{K^\alpha}, F_{K^\alpha})$. By Proposition 4.1 it is sufficient to test condition (2) for all words of length $|S_{K^\alpha}| - 2$ in order to decide whether K^α has property **A**. On the other hand, by Lemma 4.1 this is equivalent for testing whether K has property **A**. Since $|S_{K^\alpha}| = |S_K| + 1$, this gives us the bound $|S_K| - 1$ for an automaton K. □

Notice that the bound $|S_A| - 1$ cannot be improved. In Example 4.4 the shortest experiment distinguishing between the states r and q in automaton A is $w = ab$.

References

[1] Bell, J. S. Against "Measurement". *Physics World 3* (1990).

[2] Brauer, W. *Automatentheorie.* Teubner, Stuttgart, 1984.

[3] Calude, C., Calude, E., and Khoussainov, B. Deterministic automata: simulation, universality and minimality. *Research Report CDMTCS 15* (1996).

[4] Calude, C., Calude, E., Svozil, K., and Yu, S. Physical versus computational complementarity I. *International Journal of Theoretical Physics* (1997). in press.

[5] Chaitin, G. J. An improvement on a theorem by E. F. Moore. *IEEE Transactions on Electronic Computers EC-14* (1965), 466–467.

[6] Conway, J. H. *Regular Algebra and Finite Machines.* Chapman and Hall Ltd., London, 1971.

[7] Gill, A. State-identification experiments in finite automata. *Information and Control 4* (1961), 132–154.

[8] Ginsburg, S. On the length of the smallest uniform experiment which distinguishes the terminal states of the machine. *Journal of the Association for Computing Machinery 5* (1958), 266–280.

[9] Ginsburg, S. *An Introduction to Mathematical Machine Theory.* Addison Wesley, Reading Massachusetts, 1962.

[10] Greenberger, D. B., and YaSin, A. "Haunted" measurements in quantum theory. *Foundation of Physics 19*, 6 (1989), 679–704.

[11] Izbicki, H. Die entropie η–transformierter gerichteter graphen, sitzungsber. Österr. Akad. Wiss. Math. Naturwiss. Klasse Abt. *177* (1969), 227–235.

[12] Moore, E. F. Gedanken-experiments on sequential machines. In *Automata Studies*, C. E. Shannon and J. McCarthy, Eds. Princeton University Press, Princeton, 1956.

[13] Wheeler, J. A. Law without law. In *Quantum Theory and Measurement*, J. A. Wheeler and W. H. Zurek, Eds. Princeton University Press, Princeton, 1983, pp. 182–213. [14].

[14] Wheeler, J. A., and Zurek, W. H. *Quantum Theory and Measurement.* Princeton University Press, Princeton, 1983.

[15] WIGNER, E. P. Remarks on the mind-body question. In *The Scientist Speculates*, I. J. Good, Ed. Heinemann and Basic Books, London and New York, 1961, pp. 284–302. Reprinted in [14, pp. 168-181].