

Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

Information and Computation 190 (2004) 179–195

Information
and
Computationwww.elsevier.com/locate/ic

Automata theory based on quantum logic: some characterizations[☆]

Daowen Qiu

Department of Computer Science, Zhongshan University, Guangzhou, 510275, PR China

Received 30 April 2003; revised 10 November 2003

Abstract

Automata theory based on quantum logic (abbr. l -valued automata theory) may be viewed as a logical approach of quantum computation. In this paper, we characterize some fundamental properties of l -valued automata theory, and discover that some properties of the truth-value lattices of the underlying logic are equivalent to certain properties of automata. More specifically (i) the transition relations of l -valued automata are extended to describe the transitions enabled by strings of input symbols, and particularly, these extensions depend on the distributivity of the truth-value lattices (Proposition 3.1); (ii) some properties of the l -valued *successor* and *source* operators and l -valued subautomata are demonstrated to be equivalent to a property of the truth-value lattices which is exactly equivalent to the distributive law (Proposition 4.3 and Corollary 4.4). This is a new characterization of Boolean algebras in the framework of l -valued automata theory; (iii) we verify that the intersection of two l -valued subautomata is still an l -valued subautomaton if and only if the multiplication ($\&$) is distributive over the union in the truth-value lattices (Proposition 4.5), which is strictly weaker than the usual distributivity; (iv) we show that some topological characterizations in terms of the l -valued *successor* and *source* operators also rely on the distributivity of truth-value lattices (Theorem 5.6). Finally, we address some related topics for further study.

© 2004 Elsevier Inc. All rights reserved.

[☆]This research is supported by the National Natural Science Foundation (No. 90303024) and the Natural Science Foundation of Guangdong Province (No. 020146, 031541) of China.

E-mail address: issqdw@zsu.edu.cn.

1. Introduction

With a desire to circumvent the fundamental limits on current computing technologies, some nontraditional models of computation such as analog, molecular, and quantum computation have been greatly interested in both physics and computer science (e.g., [1,8,12,20,24,43]). In particular, quantum computation has become a highly active research area (e.g., [17,25,46]). To a certain extent, this originated from Shor's findings of quantum algorithms for factoring prime integers in polynomial time [41] and Grover's algorithm [16] for searching through an unstructured database which could also be sped up on a quantum computer.

The idea of quantum computation came from the studies of connections between physics and computation. The first step toward it was the understanding of the thermodynamics of classical computation. In 1973, Bennett [5] noted that a logically reversible operation need not dissipate any energy and found that a logically reversible Turing machine is a theoretical possibility. Quantum computers were first conceived by Benioff [4] and Feynman [15] in the early of 1980s. In [12], Deutsch elaborated and formalized Benioff and Feynman's idea. In particular, he proposed the so-called Church–Turing principle as a physical principle underlying the Church-Turing hypothesis in computing theory, and introduced the notion of quantum Turing machine. After Shor's impressive discovery [41] of a polynomial-time algorithm on quantum computers for prime factorization, quantum computation has become a very active research area in both quantum physics and computer science.

Roughly speaking, current studies of quantum computation may be divided into four strata: (1) physical implementations, (2) physical models, (3) mathematical models, and (4) logical foundations [10,17,25,46]. Almost all pioneer works of quantum computation such as [4,12,15,48] were devoted to establishing physical models of quantum computing. In 1990s, a great attention was paid to the physical implementations of quantum computation. For example, Lloyd [21] considered the practical implementation of quantum computing by using electromagnetic pulses and Cirac and Zoller [11] used laser manipulations of cold trapped ions to implement quantum computation. In classical computing theory, automata are simple mathematical models of computers. Correspondingly, the notion of quantum automata was introduced as a mathematical model of quantum computer; see [2,6,7,18,20,24,29,30] and [17, pp. 151–215] for example. Quantum automata are generalizations of probabilistic automata. In a probabilistic automaton [26,38], each transition is equipped with a number in the unit interval which indicates the probability of the occurrence of the transition; by contrast in a quantum automaton we associate with each transition a vector in a Hilbert space which is interpreted as the probability amplitude of the transition. In a sense, mathematical models of quantum computation may be seen as abstractions of its physical models.

Quantum automata are usually more powerful than classical ones [45] (for example, two-way quantum finite automata can recognize nonregular language [20]), but the unitarity (retrievability of computation) of quantum physics results in some restrictions and limitations (for example, the class of languages recognized by one-way quantum finite automata is a proper subclass of regular languages [7]). Also, in quantum information processing [23], due to the unitarity and linearity of quantum physics, there exist some restrictions and limitations such as *no-cloning theorem* [47], *no-deleting principle* [27], and a number of probabilistic and approximate quantum cloning and deleting machines [14,19,31–35]. So, clarifying those essential differences between traditional and quantum computation is very significant.

Quantum automata indicated above may be viewed as some computing models based on quantum mechanics. As is well known, quantum logic was introduced by Birkhoff and von Neumann [9] as the logic of quantum mechanics, and it stemmed from von Neumann's Hilbert space formalism of quantum mechanics in which the behavior of a quantum mechanical system is described by a closed subspace of a Hilbert space. By noticing that the set of closed subspaces of a Hilbert space is an orthomodular lattice, Birkhoff and von Neumann [9] suggested to use orthomodular lattice as the algebraic version of the logic of quantum mechanics, just like Boolean algebra acting as an algebraic counterpart of classical logic.

Recently, the author [49,50] primarily and very significantly considered automata theory based on quantum logic (l -valued automata), in which quantum logic is understood as a logic whose truth-value set is an orthomodular lattice, and an element of an orthomodular lattice is assigned to each transition of an automaton and it is considered to be the truth value of the proposition describing the transition. This is a logical approach to quantum computation, and it should be treated as a further abstraction of mathematical models of quantum computation.

With this approach, the author [49,50] dealt with some operations on l -valued automata, and interestingly established corresponding pumping lemma. In this paper, we discover some new phenomena in l -valued automata theory, which reveal to some extent some intrinsic distinctions of quantum computation from classical one, and may stimulate us to further exploration of quantum computing. We now state our results in detail. (i) We find that some basic properties of transition functions of l -valued automata heavily rely on the distributivity of the truth-value lattices of the underlying logic (Proposition 3.1). As is known, orthomodular lattices have weaker condition than distributivity, and any orthomodular lattice satisfying distributivity reduces to a Boolean algebra, so, this point we found implies that those properties of transition functions do not hold in the framework of l -valued automata theory. (ii) Successor and source operators as well as subautomata introduced by Bavel [3] are a fundamental tool in classical automata theory and have been applied to topological characterization of automata [44], so we define l -valued successor and source operators and l -valued subautomata, especially discover that some properties of these operators are equivalent to some properties of the truth-value lattices which are exactly equivalent to distributive law (Proposition 4.3). This is a new characterization of Boolean algebras in the framework of l -valued automata theory. (iii) We show that the meet of any two l -valued subautomata still is an l -valued subautomaton iff the distributivity of $\&$ over \vee of truth-value lattice holds (Proposition 4.5). It is interesting to note that the distributivity of $\&$ over \vee is strictly weaker than the distributivity of \wedge over \vee , so we discover a new characterization of orthomodular lattices in terms of automata. (iv) We present topological characterizations in terms of the l -valued *successor* and *source* operators, and find that they also depend on the distributivity of truth-value lattices.

So, these findings imply that some fundamental properties of classical automata theory do not hold in l -valued automata, unless some conditions are imposed on the truth-value lattices, some of which are equivalent to distributive law, while another conditions are strictly weaker than distributivity. This leads naturally us to guess that there may be a spectrum of properties of orthomodular lattices which is pointwise equivalent to a spectrum of properties of automata. Furthermore, These results display some essential distinctions of quantum computation from classical one, and show that systematically establishing computational models based on quantum logic needs new idea and method, and is worthy of further exploration.

2. Preliminaries

Here we briefly review some notions and terminology in quantum logic. For details, we refer to [13,22,28,42]. A 7-tuple $l = \langle L, \leq, \wedge, \vee, \perp, 0, 1 \rangle$ is called a complete orthomodular lattice, if it satisfies the following conditions:

- (1) $\langle L, \leq, \wedge, \vee, \perp, 0, 1 \rangle$ is a complete lattice, 0 and 1 are the least and greatest elements of L , respectively, \leq is the partial ordering in L , and for any $M \subseteq L$, $\wedge M$, and $\vee M$ stand for the greatest lower bound and the least upper bound of M , respectively.
- (2) \perp is a unary operation on L , called orthocomplement, and it is required to satisfy the following conditions: for any $a, b \in L$,
 - (2.1) $a \wedge a^\perp = 0, a \vee a^\perp = 1$.
 - (2.2) $a^{\perp\perp} = a$.
 - (2.3) $a \leq b$ implies $b^\perp \leq a^\perp$.
 - (2.4) $a \geq b$ implies $a \wedge (a^\perp \vee b) = b$.

The condition (2.4) may be restated as follows:

$$(2.4)' \text{ For any } a, b \in L, a \wedge (a^\perp \vee (a \wedge b)) \leq b.$$

A quantum logic is a complete orthomodular lattice-valued logic. In this paper, we mainly use the Sasaki arrow as the implication operator. The Sasaki arrow is defined as follows: for any $a, b \in L$,

$$(3) a \rightarrow b \stackrel{def}{=} a^\perp \vee (a \wedge b).$$

The conjunction in a quantum logic is usually interpreted as the meet operation of the truth-value lattices. It is easy to see that the meet operation is not conjugate to the Sasaki arrow. Thus, Román and Rumbos [39] introduced a new conjunction operator, namely, the multiplication. The multiplication is defined as follows: for all $a, b \in L$,

$$(4) a \& b \stackrel{def}{=} (a \vee b^\perp) \wedge b.$$

For the sake of convenience, we give here some of properties of the Sasaki arrow and the multiplication:

$$(4.1) a \& b \leq c \text{ iff } a \leq b \rightarrow c.$$

$$(4.2) a \leq b \text{ iff } a \rightarrow b = 1.$$

$$(4.3) (a \rightarrow b) \& a \leq b.$$

$$(4.4) a \& (a \rightarrow b) \leq b.$$

$$(4.5) 0 \rightarrow a = 1 = a \rightarrow 1.$$

$$(4.6) a \& b \leq b.$$

$$(4.7) a \wedge b \leq a \& b.$$

(4.8) Let $l = \langle L, \leq, \wedge, \vee, \perp, 0, 1 \rangle$ be a complete orthomodular lattice. Then L is a Boolean algebra, i.e., it satisfies the distributive law: $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$ for all $a, b, c \in L$, if and only if any one of the following holds:

(i) $\&$ is commutative, i.e., $a \& b = b \& a$ for any $a, b \in L$.

(ii) $b \leq c \Rightarrow a \& b \leq a \& c$ for any $a, b, c \in L$.

The bi-implication operator corresponding to the Sasaki arrow is defined as follows:

for all $a, b \in L$,

$$(5) a \leftrightarrow b \stackrel{def}{=} (a \rightarrow b) \wedge (b \rightarrow a).$$

Let $l = \langle L, \leq, \wedge, \vee, \perp, 0, 1 \rangle$ be a complete orthomodular lattice and \rightarrow the Sasaki arrow. The syntax of l -valued logic is similar to that of classical first-order logic. We have three primitive connectives \neg (negation), \wedge (conjunction), and \rightarrow (implication) and a primitive quantifier \forall (universal quantifier). The connectives \vee (disjunction) and \leftrightarrow (bi-implication) and the existential quantifier \exists are defined in terms of \neg , \wedge , \rightarrow , and \forall in the usual way.

In addition, we need to use some set-theoretical formulas. Let \in (membership) be a binary (primitive) predicate symbol. Then \subseteq (inclusion) and \equiv (equality) can be defined with \in as usual. The semantics of l -valued logic is given by interpreting the connectives \neg , \wedge , and \rightarrow as the operations \perp , \wedge , and \rightarrow , respectively, on L and by interpreting the quantifier \forall as the greatest lower bound in L . In addition, the truth value of set-theoretical formula $x \in A$ is $\lceil x \in A \rceil = A(x)$. It is worth indicating that in this paper the set A and its characteristic function are identified. In the l -valued logic, 1 is the unique designated truth value. In other words, a formula φ is valid iff its truth value $\lceil \varphi \rceil$ is 1.

3. Characterizations of transition relations

Let $\langle L, \leq, \wedge, \vee, \perp, 0, 1 \rangle$ be a complete orthomodular lattice, let \rightarrow be an implication operator on l , and let Σ be a finite alphabet. An l -valued (quantum) automata over Σ is a quadruple $\mathfrak{R} = \langle Q, I, T, \delta \rangle$, where:

1. Q is a finite set of states.
2. $I \subseteq Q$ is the set of initial states.
3. $T \subseteq Q$ is the set of terminal states.
4. δ is an l -valued subset of $Q \times \Sigma \times Q$, i.e., a mapping from $Q \times \Sigma \times Q$ into L , called the l -valued (quantum) transition relation of \mathfrak{R} . Intuitively, for any $p, q \in Q$ and $\sigma \in \Sigma$, $\delta(p, \sigma, q)$ indicates the truth value of the proposition that input σ causes state p to become q .

The transition relation δ represents the transition of states induced by a single input symbol. To depict the transitions enabled by a string of input symbols, δ may be naturally extended to $\delta^* : Q \times \Sigma^* \times Q \rightarrow L$ as follows: For any $p, q \in Q$, if $q = p$, then $\delta^*(p, \epsilon, q) = 1$; otherwise, $\delta^*(p, \epsilon, q) = 0$, and

$$\delta^*(p, xa, q) = \vee \{ \delta^*(p, x, r) \wedge \delta^*(r, a, q) : r \in Q \},$$

for any $x \in \Sigma^*$ and $a \in \Sigma$, where $\Sigma^* = \bigcup_{k=0}^{\infty} \Sigma^k$, $\Sigma^{(0)} = \{\epsilon\}$ represents an empty string, and $\Sigma^k = \{\sigma_1 \sigma_2 \cdots \sigma_k : \sigma_i \in \Sigma, i = 1, 2, \dots, k\}$. On the other hand, δ^* may be treated as an l -valued predicate on $Q \times \Sigma^* \times Q$. For any $p, q \in Q$ and $x \in \Sigma^*$, $\delta^*(p, x, q)$ may be seen as the proposition that state p becomes state q after inputting the string x of symbols, and its truth value is $\lceil \delta^*(p, x, q) \rceil \stackrel{def}{=} \delta^*(p, x, q)$. Intuitively, we may expect that the concatenation of two strings of input symbols causes a state p becoming another state q iff the first string enables the state p becoming some state r and then r becomes q after inputting the second string. In other words, for any $p, q \in Q$ and for any $x, y \in \Sigma^*$,

$$\stackrel{l}{\models} \delta^*(p, xy, q) \leftrightarrow (\exists r \in Q) (\delta^*(p, x, r) \wedge \delta^*(r, y, q)).$$

However, this conclusion appeals to the distributivity of the lattice of truth values. For convenience, denote by $\mathbf{a}(\Sigma, l)$ the class of all l -valued automata over Σ .

Proposition 3.1. *The following three statements are equivalent.*

- (i) L satisfies the distributivity: $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$ for any $a, b, c \in L$.
- (ii) For any $\mathfrak{R} = (Q, I, T, \delta) \in \mathbf{A}(\Sigma, l)$, for any $p, q \in Q$ and for any $x, y \in \Sigma^*$,

$$\models^l \delta^*(p, xy, q) \rightarrow (\exists r \in Q)(\delta^*(p, x, r) \wedge \delta^*(r, y, q)).$$

- (iii) For any $\mathfrak{R} = (Q, I, T, \delta) \in \mathbf{A}(\Sigma, l)$, for any $p, q \in Q$ and for any $x, y \in \Sigma^*$,

$$\models^l (\exists r \in Q)(\delta^*(p, x, r) \wedge \delta^*(r, y, q)) \rightarrow \delta^*(p, xy, q).$$

Proof. (i) \Rightarrow (ii) and (i) \Rightarrow (iii): It suffices to show that $\delta^*(p, xy, q) = \vee\{\delta^*(p, x, r) \wedge \delta^*(r, y, q) : r \in Q\}$. For the case of $y \in \Sigma \cup \{\epsilon\}$, it is clear by the definition of δ^* . Suppose that it holds for any $y \in \Sigma^*$ with $|y| \leq k-1$, where $|y|$ denotes the length of word y . Let $y = \sigma_1 \cdots \sigma_k \in \Sigma^*$. Then from the definition of δ^* and the condition (i) that \wedge is distributive over \vee one has

$$\begin{aligned} \delta^*(p, xy, q) &= \vee\{\delta^*(p, x\sigma_1 \cdots \sigma_{k-1}, r) \wedge \delta(r, \sigma_k, q) : r \in Q\} \\ &= \vee\{\vee\{\delta^*(p, x, r') \wedge \delta^*(r', \sigma_1 \cdots \sigma_{k-1}, r) : r' \in Q\} \wedge \delta(r, \sigma_k, q) : r \in Q\} \\ &= \vee\{\delta^*(p, x, r') \wedge \delta^*(r', \sigma_1 \cdots \sigma_{k-1}, r) \wedge \delta(r, \sigma_k, q) : r', r \in Q\} \\ &= \vee\{\vee\{\delta^*(r', \sigma_1 \cdots \sigma_{k-1}, r) \wedge \delta(r', \sigma_k, q) : r \in Q\} \wedge \delta^*(p, x, r') : r' \in Q\} \\ &= \vee\{\delta^*(r', \sigma_1 \cdots \sigma_k, q) \wedge \delta^*(p, x, r') : r' \in Q\} \\ &= \vee\{\delta^*(p, x, r') \wedge \delta^*(r', y, q) : r' \in Q\}. \end{aligned}$$

(ii) \Rightarrow (i): Given $a, b, c \in L$, then the purpose is to show $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$ for any $a, b, c \in L$. We take $\mathfrak{R} = (Q, I, T, \delta) \in \mathbf{A}(\Sigma, l)$, where $Q = \{p_0, p_1, p_2, p_3, p_4\}$, $I = \{p_0\}$, $T = \{p_4\}$, and $\delta(p_0, \sigma_1, p_1) = \delta(p_1, \sigma_2, p_3) = b$, $\delta(p_0, \sigma_1, p_2) = \delta(p_2, \sigma_2, p_3) = c$, $\delta(p_3, \sigma_3, p_4) = a$ for some $\sigma_1, \sigma_2, \sigma_3 \in \Sigma$. Let $x = \sigma_1$ and $y = \sigma_2\sigma_3$. Then we have

$$\begin{aligned} \delta^*(p_0, xy, p_4) &= \vee\{\delta^*(p_0, \sigma_1\sigma_2, r) \wedge \delta(r, \sigma_3, p_4) : r \in Q\} \\ &= \delta^*(p_0, \sigma_1\sigma_2, p_3) \wedge \delta(p_3, \sigma_3, p_4) \\ &= (b \vee c) \wedge a, \end{aligned}$$

and

$$\begin{aligned} &\vee\{\delta^*(p_0, \sigma_1, r) \wedge \delta^*(r, \sigma_2\sigma_3, p_4) : r \in Q\} \\ &= (b \wedge (b \wedge a)) \vee (c \wedge (c \wedge a)) \\ &= (b \wedge a) \vee (c \wedge a). \end{aligned}$$

Therefore, it follows from (ii) that $a \wedge (b \vee c) \leq (a \wedge b) \vee (a \wedge c)$. However, $a \wedge (b \vee c) \geq (a \wedge b) \vee (a \wedge c)$ always holds. So $(a \wedge b) \vee (a \wedge c) \leq a \wedge (b \vee c)$ holds.

(iii) \Rightarrow (i): Let $a, b, c \in L$. Take $\mathfrak{R} = (Q, I, T, \delta) \in \mathbf{A}(\Sigma, l)$, where $Q = \{p_0, p_1, p_2, p_3, p_4\}$, $I = \{p_0\}$, $T = \{p_4\}$, and $\delta(p_0, \sigma_1, p_1) = a$, $\delta(p_1, \sigma_2, p_2) = \delta(p_2, \sigma_3, p_4)$, $\delta(p_1, \sigma_2, p_3) = \delta(p_3, \sigma_3, p_4) = c$ for some $\sigma_1, \sigma_2, \sigma_3 \in \Sigma$. Let $x = \sigma_1$ and $y = \sigma_2\sigma_3$. Then we have

$$\begin{aligned} \delta^*(p_0, xy, p_4) &= \vee \{ \delta^*(p_0, \sigma_1\sigma_2, r) \wedge \delta(r, \sigma_3, p_4) : r \in Q \} \\ &= (\delta^*(p_0, \sigma_1\sigma_2, p_2) \wedge \delta(p_2, \sigma_3, p_4)) \vee (\delta^*(p_0, \sigma_1\sigma_2, p_3) \wedge \delta(p_3, \sigma_3, p_4)) \\ &= ((a \wedge b) \wedge b) \vee ((a \wedge c) \wedge c) \\ &= (a \wedge b) \vee (a \wedge c), \end{aligned}$$

and

$$\begin{aligned} &\vee \{ \delta^*(p_0, \sigma_1, r) \wedge \sigma^*(r, \sigma_2\sigma_3, p_4) : r \in Q \} \\ &= \delta(p_0, \sigma_1, p_1) \wedge \delta^*(p_1, \sigma_2\sigma_3, p_4) \\ &= a \wedge (b \vee c). \end{aligned}$$

Therefore it follows from (ii) that $a \wedge (b \vee c) \leq (a \wedge b) \vee (a \wedge c)$. So $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$ holds. This completes the proof. \square

4. l -Valued successor and source operators

The *successor* and *source* operators introduced by Bavel [3] are a fundamental tool in classical automata theory. For any state p , the *successor* operator describes all states which may be reached from p by a sequence x of inputs; conversely, the *source* operator presents all predecessors of p , i.e., states that may get at p by inputting a string. In this section, we will establish some of their basic properties in l -valued automata theory.

Definition 4.1. Let $\mathcal{M} = (Q, \Sigma, \delta)$ be an l -valued automaton. Then we define the *successor* and *source* operators S and R from L^Q to L^Q as follows: for any $A \in L^Q$ and $q \in Q$,

$$S(A)(q) \stackrel{def}{=} \vee \{ A(p) \wedge \delta^*(p, x, q) : p \in Q, x \in \Sigma^* \},$$

$$R(A)(q) \stackrel{def}{=} \vee \{ A(p) \wedge \delta^*(q, y, p) : p \in Q, y \in \Sigma^* \}.$$

From the above definitions it follows that for any $p \in Q$,

$$S(A)(p) \geq A(p) \text{ and } R(A)(p) \geq A(p).$$

Thus $\stackrel{l}{\models} A \subseteq S(A)$ and $\stackrel{l}{\models} A \subseteq R(A)$ always holds.

A subautomaton of an automaton means a subset of the set of states satisfying the closure property under the *successor* operator. Formally, we have:

Definition 4.2. Let $\mathcal{M} = (Q, \Sigma, \delta)$ be an l -valued automaton. Then for any $A \in L^Q$, we call A an l -valued subautomaton of \mathcal{M} if

$$\models^l (\forall q \in Q)(q \in A \rightarrow (\forall p \in Q)(\forall x \in \Sigma^*)((q, x, p) \in \delta^* \rightarrow p \in A)),$$

equivalently, for any $q \in Q$,

$$A(q) \leq \bigwedge \{ \delta^*(q, x, p) \rightarrow A(p) : x \in \Sigma^*, p \in Q \}.$$

As we know, in classical automata theory the operators S and R may be described in terms of each other, and both of them can be used to give a characterization of subautomata. Of course, we hope to generalize these results to l -valued automata theory. However, the following proposition shows that these results heavily depends upon the distributivity of the underlying logic, and so they are invalid unless our quantum logic degenerates to classical Boolean logic.

Proposition 4.3. *The following three statements are equivalent.*

- (i) For any $a, b \in L$, $b^\perp \vee (b \wedge a) \geq a$.
- (ii) For any l -valued automaton $\mathcal{M} = (Q, \Sigma, l)$ and $A \in L^Q$, $\models^l S(A) \equiv A$ if and only if $\models^l R(A^\perp) \equiv A^\perp$.
- (iii) For any l -valued automaton $\mathcal{M} = (Q, \Sigma, l)$ and $A \in L^Q$, $\models^l S(A) \equiv A$ if and only if A is an l -subautomaton of \mathcal{M} .

Proof. (i) \Rightarrow (ii): Since for any $A \in L^Q$ and $q \in Q$, $A(q) \leq S(A)(q)$ and $A(q) \leq R(A)(q)$ always hold, it suffices to prove that $\models^l S(A) \subseteq A$ if and only if $\models^l R(A^\perp) \subseteq A^\perp$. Suppose that $S(A) \subseteq A$, then for any $q \in Q$,

$$\bigvee \{ A(p) \wedge \delta^*(p, x, q) : p \in Q, x \in \Sigma^* \} \leq A(q). \quad (1)$$

For any $p \in Q$, our purpose is to show that $R(A^\perp)(p) \leq A^\perp(p)$, i.e.,

$$\bigvee \{ A(r)^\perp \wedge \delta^*(p, y, r) : r \in Q, y \in \Sigma^* \} \leq A(p)^\perp. \quad (2)$$

For any $r \in Q$ and $y \in \Sigma^*$, from (1) we have

$$A(p) \wedge \delta^*(p, y, r) \leq A(r).$$

Consequently,

$$\begin{aligned} A(r) \vee \delta^*(p, y, r)^\perp &\geq (A(p) \wedge \delta^*(p, y, r)) \vee \delta^*(p, y, r)^\perp \\ &\geq A(p). \end{aligned}$$

So $A(r)^\perp \wedge \delta^*(p, y, r) \leq A(p)^\perp$, and (2) holds. With a similar argument we can obtain the converse implication.

(ii) \Rightarrow (i): Given $a, b \in L$, we take an l -valued automata (Q, Σ, δ) as follows: $Q = \{p, q\}$, $\Sigma = \{\sigma\}$, $\delta(p, \sigma, q) = b$. Furthermore, suppose that $A \in L^Q$, $A(p) = a$ and $A(q) = b \wedge a$. Then we have $S(A)(p) = A(p)$ and $S(A)(q) = a \wedge b = A(q)$. It follows that $\models^l S(A) \equiv A$. Thus with (ii) $\models^l R(A^\perp) \equiv A^\perp$ holds. By this we have $A(p) \leq A(q) \vee \delta^*(p, \sigma, q)^\perp$, and therefore $a \leq (b \wedge a) \vee b^\perp$.

(i) \Rightarrow (iii): Suppose $\models^l S(A) \equiv A$. The aim is to show that for any $p, q \in Q, x \in \Sigma^*, A(q) \leq \delta^*(q, x, p) \rightarrow A(p)$, i.e.,

$$A(q) \leq \delta^*(q, x, p)^\perp \vee (\delta^*(q, x, p) \wedge A(p)). \quad (3)$$

Because $S(A)(p) \leq A(p)$, it holds that $A(q) \wedge \delta^*(q, x, p) \leq A(p)$. Therefore we have

$$\begin{aligned} & \delta^*(q, x, p)^\perp \vee (\delta^*(q, x, p) \wedge A(p)) \\ & \geq \delta^*(q, x, p)^\perp \vee (\delta^*(q, x, p) \wedge (A(q) \wedge \delta^*(q, x, p))) \\ & = \delta^*(q, x, p)^\perp \vee (\delta^*(q, x, p) \wedge A(q)) \\ & \geq A(q). \end{aligned}$$

Thus inequality (3) holds.

(iii) \Rightarrow (i): Let $a, b \in L$. We consider the l -valued automata (Q, Σ, δ) defined as follows: $Q = \{p, q\}$, $\Sigma = \{\sigma\}$, $\delta(p, \sigma, q) = b$. Suppose that $A \in L^Q$ satisfies that $A(q) = A(p) = a$. We have shown that $\models^l S(A) \equiv A$, and thus A is an l -subautomaton of \mathcal{M} . Consequently, $A(q) \leq \delta^*(q, \sigma, p)^\perp \vee (\delta^*(q, \sigma, p) \wedge A(p))$, i.e., $a \leq (b \wedge a) \vee b^\perp$. We complete the proof. \square

Indeed, from the proof of Proposition 4.3 it follows easily that the following Corollary 4.4 holds.

Corollary 4.4. *The following six statements are equivalent:*

- (i) L satisfies the distributive law: $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$ for all $a, b, c \in L$.
- (ii) For any $a, b \in L$, $b \wedge (b^\perp \vee a) \leq a$, i.e., $a \&b \leq a$.
- (iii) For any $a, b \in L$, $b^\perp \vee (b \wedge a) \geq a$.
- (iv) For any L -valued automaton $\mathcal{M} = (Q, \Sigma, \delta)$ and for any $A \in L^Q$, if $\models^l S(A) \equiv A$, then $\models^l R(A^\perp) \equiv A^\perp$.
- (v) For any L -valued automaton $\mathcal{M} = (Q, \Sigma, \delta)$ and for any $A \in L^Q$, if $\models^l R(A^\perp) \equiv A^\perp$, then $\models^l S(A) \equiv A$.
- (vi) For any L -valued automaton $\mathcal{M} = (Q, \Sigma, \delta)$ and for any $A \in L^Q$, if $\models^l S(A) \equiv A$, then A is an l -valued subautomaton of \mathcal{M} .

Proof. (i) \Rightarrow (ii): Clear.

(ii) \Rightarrow (i): From (ii) it is easy to see that $a \wedge b \leq a \&b \leq a \wedge b$ and $a \wedge b \leq b \&a \leq a \wedge b$, and hence $a \&b = a \wedge b = b \&a$. So by (4.8) in Section 2 it yields (i).

(ii) \Leftrightarrow (iii): Immediate.

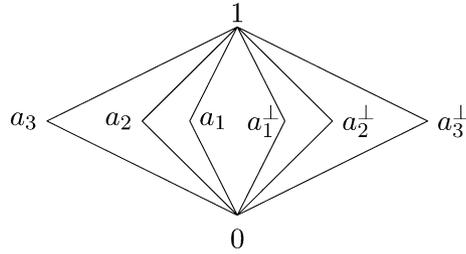


Fig. 1.

(v) \Rightarrow (ii): Given $a, b \in L$, and let $Q = \{p, q\}$, $\Sigma = \{\delta\}$, and $\delta(q, \sigma, p) = b$. Suppose that $A \in L^Q$ with $A(p) = a$ and $A(q) = b^\perp \vee a$, then $R(A^\perp)(p) = A^\perp(p)$ and $R(A^\perp)(q) = \delta(q, \sigma, p) \wedge A^\perp(p) = b \wedge a^\perp = A(q)^\perp$. Hence $\models R(A^\perp) \equiv A^\perp$ holds. By (v) $S(A)(p) \leq A(p)$ follows. Thus we have

$$S(A)(p) = \delta(q, \delta, p) \wedge A(q) = b \wedge (b^\perp \vee a) \leq a.$$

(ii) \Rightarrow (v) and (iii) \Leftrightarrow (iv) and (iii) \Leftrightarrow (vi): From Proposition 4.3. \square

In Propositions 3.1, 4.3, and Corollary 4.4 we showed that some properties of *successor* and *source* operators are equivalent to the distributivity of \wedge over \vee in the lattice of truth values. The following proposition indicates that a certain property of l -valued subautomata is equivalent to the distributivity of $\&$ over \vee . It is interesting to note that the distributivity of $\&$ over \vee is strictly weaker than the distributivity of \wedge over \vee . Clearly, the distributivity of \wedge over \vee implies the distributivity of $\&$ over \vee , but in general the converse implication does not hold. This may be seen from the example visualized by Fig. 1. Indeed, it is easy to check that the lattice depicted by Fig. 1 enjoys the distributivity of $\&$ over \vee , but obviously \wedge is not distributive over \vee .

Proposition 4.5. *The following three statements are equivalent:*

- (i) For any $a, b, c \in L$, $(b\&a) \vee (c\&a) = (b \vee c)\&a$.
- (ii) For any $a, b, c \in L$, $(a \vee (a^\perp \wedge b)) \wedge (a \vee (a^\perp \wedge c)) = a \vee (a^\perp \wedge b \wedge c)$.
- (iii) For any l -valued automaton $\mathcal{M} = (Q, \Sigma, \delta)$, if A and B are l -valued subautomata of \mathcal{M} , then so is $A \cap B$.

Proof. (i) \Leftrightarrow (ii): Straightforward from the definition of $\&$.

(ii) \Rightarrow (iii): For any $p, q \in Q$ and for any $x \in \Sigma^*$, since A and B are l -valued subautomata of \mathcal{M} , we have

$$A(p) \leq \delta^*(p, x, q) \rightarrow A(q),$$

and

$$B(p) \leq \delta^*(p, x, q) \rightarrow B(q).$$

Therefore

$$\begin{aligned} A(p) \wedge B(p) &\leq (\delta^*(p, x, q) \rightarrow A(q)) \wedge (\delta^*(p, x, q) \rightarrow B(q)) \\ &= ((\delta^*(p, x, q) \wedge A(q)) \vee \delta^*(p, x, q)^\perp) \wedge ((\delta^*(p, x, q) \wedge B(q)) \vee \delta^*(p, x, q)^\perp) \\ &= \delta^*(p, x, q) \rightarrow A(q) \wedge B(q). \end{aligned}$$

From definition 4.2 it follows that $A \cap B$ is an l -valued subautomaton of \mathcal{M} .

(iii) \Leftrightarrow (ii): Given $a, b, c \in L$, let us take $\mathcal{M} = \{p, q\}$, $\Sigma = \{\sigma\}$, $\delta(p, \delta, q) = a^\perp$. If $A \in L^Q$ satisfies $A(q) = b$, $A(p) = (a^\perp \wedge b) \vee a$, $B(q) = c$, $B(p) = (a^\perp \wedge c) \vee a$, then obviously A and B are l -valued subautomata of \mathcal{M} . So

$$A(p) \wedge B(p) \leq \delta^*(p, x, q) \rightarrow A(q) \wedge B(q),$$

i.e.,

$$((a^\perp \wedge b) \vee a) \wedge ((a^\perp \wedge c) \vee a) \leq a^\perp \rightarrow b \wedge c = (a^\perp \wedge b \wedge c) \vee a.$$

Therefore it yields (ii) since $((a^\perp \wedge b) \vee a) \wedge ((a^\perp \wedge c) \vee a) \geq (a^\perp \wedge b \wedge c) \vee a$ always holds. \square

5. Topological characterizations

To study the topological structures induced from l -valued automata, we first need to present some of the fundamental properties of S and R as well as l -valued subautomata.

Proposition 5.1. *Let $\mathcal{M} = (Q, \Sigma, \delta)$ be an l -valued automaton and let $A, B \in L^Q$. Then*

- (i) $\models^l S(\emptyset) \equiv \emptyset \wedge S(Q) \equiv Q \wedge R(\emptyset) \equiv \emptyset \wedge R(Q) \equiv Q$,
and \emptyset and Q are l -valued subautomata of \mathcal{M} .
- (ii) $\models^l A \subseteq S(A)$ and $\models^l A \subseteq R(A)$.
- (iii) $\models^l S(\mathbf{a}) \equiv \mathbf{a}$ and $\models^l R(\mathbf{a}) \equiv \mathbf{a}$
for all $a \in L$, where $\mathbf{a} \in L^Q$ is the constant l -valued subset of Q with height a , i.e., $\mathbf{a}(q) = a$ for any $q \in Q$.
- (iv) If $\models^l S(A) \equiv A$ and $\models^l S(B) \equiv B$, then $\models^l S(A \cap B) \equiv A \cap B$.
- (v) If $\models^l R(A) \equiv A$ and $\models^l R(B) \equiv B$, then $\models^l R(A \cap B) \equiv A \cap B$.
- (vi) $\models^l \bigcup_{i \in J} S(A_i) \subseteq S(\bigcup_{i \in J} A_i)$ and $\models^l \bigcup_{i \in J} R(A_i) \subseteq R(\bigcup_{i \in J} A_i)$
for any $A_i \in L^Q$, $i \in J$, where J denotes any indexing set.
- (vii) If A and B are l -valued subautomata of \mathcal{M} , then so is $A \cup B$.

Proof. (i) Immediate.

(ii) For any $p \in Q$, we have

$$\begin{aligned} S(A)(p) &= \bigvee_{q \in Q} \bigvee_{x \in \Sigma^*} (\delta^*(q, \delta, p) \wedge A(q)) \\ &\geq \delta^*(p, \epsilon, p) \wedge A(p) = A(p), \end{aligned}$$

and

$$\begin{aligned} R(A)(p) &= \bigvee_{q \in Q} \bigvee_{x \in \Sigma^*} (\delta^*(p, \delta, q) \wedge A(q)) \\ &\geq \delta^*(p, \epsilon, p) \wedge A(p) = A(p). \end{aligned}$$

So (ii) holds.

(iii) For any $p \in Q$ and for any $a \in L$, we have

$$\begin{aligned} S(\mathbf{a})(p) &= \bigvee_{q \in Q} \bigvee_{x \in \Sigma^*} (\delta^*(q, x, p) \wedge \mathbf{a}(q)) \\ &= \bigvee_{q \in Q} \bigvee_{x \in \Sigma^*} (\delta^*(q, x, p) \wedge a) \\ &= \delta^*(p, \epsilon, p) \wedge a \\ &= \mathbf{a}(p), \end{aligned}$$

and similarly we can obtain that $R(\mathbf{a})(p) = \mathbf{a}(p)$ for any $p \in Q$ and for any $a \in L$. Thus we have justified (iii).

(iv) It is enough to prove that for any $p \in Q$,

$$S(A \cap B)(p) \leq A(p) \wedge B(p), \quad (4)$$

i.e.,

$$\bigvee_{q \in Q} \bigvee_{x \in \Sigma^*} (\delta^*(q, x, p) \wedge A(q) \wedge B(q)) \leq A(p) \wedge B(p). \quad (5)$$

For any $q \in Q$ and for any $x \in \Sigma^*$, from $\models^l S(A) \equiv A$ and $\models^l S(B) \equiv B$ it follows that $\delta^*(q, x, p) \wedge A(q) \leq A(p)$ and $\delta^*(q, x, p) \wedge B(q) \leq B(p)$. Clearly, it follows that

$$\delta^*(q, x, p) \wedge A(q) \wedge B(q) \leq A(p) \wedge B(p).$$

So (5) holds and hence (4) follows.

(vi) For any $p \in Q$, one has

$$\begin{aligned} S\left(\bigcup_{i \in J} A_i\right)(p) &= \bigvee_{q \in Q} \bigvee_{x \in \Sigma^*} (\delta^*(q, x, p) \wedge (\bigvee_{i \in J} A_i(p))) \\ &\geq \bigvee_{i \in J} \bigvee_{q \in Q} \bigvee_{x \in \Sigma^*} (\delta^*(q, x, p) \wedge A_i(p)) \\ &= \bigcup_{i \in J} S(A_i)(p). \end{aligned}$$

This yields that $\models^l \bigcup_{i \in J} S(A_i) \subseteq S(\bigcup_{i \in J} A_i)$. Similarly we can obtain that $\models^l \bigcup_{i \in J} R(A_i) \subseteq R(\bigcup_{i \in J} A_i)$ for any $A_i \in L^Q$ and $i \in J$.

(vii) Let A and B are l -valued subautomata of \mathcal{M} . Then for any $p, q \in Q$ and $x \in \Sigma^*$, one has

$$A(p) \leq \delta^*(p, x, q) \rightarrow A(q),$$

and

$$B(p) \leq \delta^*(p, x, q) \rightarrow B(q).$$

Therefore

$$\begin{aligned}
A(p) \vee B(p) &\leq (\delta^*(p, x, q) \rightarrow A(q)) \vee (\delta^*(p, x, q) \rightarrow B(q)) \\
&= ((\delta^*(p, x, q) \wedge A(q)) \vee \delta^*(p, x, q)^\perp) \\
&\quad \vee ((\delta^*(p, x, q) \wedge B(q)) \vee \delta^*(p, x, q)^\perp) \\
&\leq (\delta^*(p, x, q) \wedge (A(q) \vee B(q))) \vee \delta^*(p, x, q)^\perp \\
&= \delta^*(p, x, q) \rightarrow A(q) \vee B(q).
\end{aligned}$$

So $A \cup B$ is an l -valued subautomaton of \mathcal{M} , and we complete the proof. \square

Let $\mathcal{M} = (Q, \Sigma, \delta)$ be an l -valued automaton. We set

$$\mathcal{J}_S = \{A \in L^Q \mid \stackrel{l}{\models} S(A) \equiv A\},$$

$$\mathcal{J}_R = \{A \in L^Q \mid \stackrel{l}{\models} R(A^\perp) \equiv A^\perp\}, \text{ and}$$

$$\mathcal{J}_I = \{A \in L^Q \mid A \text{ is an } l\text{-subautomaton of } \mathcal{M}\}.$$

Proposition 5.2. *Let $\mathcal{M} = (Q, \Sigma, \delta)$ be an l -valued automaton. If $A \in L^Q$ is an l -valued subautomaton of \mathcal{M} , then $\stackrel{l}{\models} S(A) \equiv A$.*

Proof. For any $x \in \Sigma^*$ and for any $p, q \in Q$, we have

$$A(q) \leq \delta^*(q, x, p) \rightarrow A(p)$$

since A is an l -valued subautomaton of \mathcal{M} . Thus

$$\begin{aligned}
\delta^*(q, x, p) \wedge A(q) &\leq \delta^*(q, x, p) \wedge (\delta^*(q, x, p) \rightarrow A(p)) \\
&\leq \delta^*(q, x, p) \& (\delta^*(q, x, p) \rightarrow A(p)) \\
&\leq A(p).
\end{aligned}$$

So $S(A)(p) \leq A(p)$ for all $p \in Q$, and with $\stackrel{l}{\models} A \subseteq S(A)$ it yields that $\stackrel{l}{\models} S(A) \equiv A$. \square

Proposition 5.3. *Let $\mathcal{M} = (Q, \Sigma, \delta)$ be an l -valued automaton. Then $\mathcal{J}_I \subset \mathcal{J}_S$.*

Proof. Straightforward from Proposition 5.2. \square

In classical automata theory, for any sets A and B of states, if A and B are equal to their *successor* (*source*) sets, respectively, then the union of A and B is also equal to the *successor* (*source*) of the union of A and B . In the framework of many-valued logics, however, this conclusion appeals to the distributivity of the underlying logic. This is similar to Propositions 3.1, 4.3, and Corollary 4.4.

Proposition 5.4. *The following three statements are equivalent:*

- (i) L satisfies the distributive law: $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$ for all $a, b, c \in L$.

- (ii) If $\models^l S(A) \equiv A$ and $\models^l S(B) \equiv B$, then $\models^l S(A \cup B) \equiv A \cup B$.
 (iii) If $\models^l R(A) \equiv A$ and $\models^l R(B) \equiv B$, then $\models^l R(A \cup B) \equiv A \cup B$.

Proof. (i) \Rightarrow (ii): For any $q \in Q$ and $x \in \Sigma^*$, from $\models^l S(A) \equiv A$ and $\models^l S(B) \equiv B$ we have

$$A(q) \wedge \delta^*(q, x, p) \leq A(p)$$

and

$$B(q) \wedge \delta^*(q, x, p) \leq B(p).$$

So

$$\begin{aligned} & (A(q) \vee B(q)) \wedge \delta^*(q, x, p) \\ &= (A(q) \wedge \delta^*(q, x, p)) \vee (B(q) \wedge \delta^*(q, x, p)) \\ &\leq A(p) \vee B(p). \end{aligned}$$

Therefore $S(A \cup B)(p) \leq (A \cup B)(p)$, and further we have $\models^l S(A \cup B) \equiv A \cup B$ since $\models^l A \subseteq S(A)$ and $\models^l A \subseteq R(A)$ always hold (Proposition 5.1 (ii)).

(ii) \Rightarrow (i): Given $a, b, c \in L$, let us set $Q = \{p, q\}$, $\Sigma = \{\sigma\}$, $\delta(q, \sigma, p) = c$, and take $A, B \in L^Q$ with $A(p) = a \wedge c$, $A(q) = a$, $B(p) = b \wedge c$, $B(q) = b$. Then it is easy to check that $\models^l S(A) \equiv A$ and $\models^l S(B) \equiv B$. Thus $S(A \cup B)(p) \leq (A \cup B)(p) = A(p) \vee B(p)$, where

$$S(A \cup B)(p) \geq (A(q) \vee B(q)) \wedge \delta(q, \sigma, p) = (a \vee b) \wedge c.$$

So by (ii) we obtain $(a \wedge c) \vee (b \wedge c) \geq (a \vee b) \wedge c$. Therefore $(a \wedge c) \vee (b \wedge c) = (a \vee b) \wedge c$ since $(a \wedge c) \vee (b \wedge c) \leq (a \vee b) \wedge c$ always holds.

(i) \Leftrightarrow (iii): Similar to the proof of (i) \Leftrightarrow (ii). Thus the proof is completed. \square

In order to present topological characterizations in terms of the l -valued *successor* and *source* operators, we introduce the notion of l -valued topology, a natural generalization of general topology.

Definition 5.5. Let X be a non-empty set, and let $\mathcal{T} \subseteq L^X$. Then we call \mathcal{T} an l -valued topology over X if it satisfies:

- (i) $\emptyset, X \in \mathcal{T}$.
 (ii) If $A, B \in \mathcal{T}$, then $A \cap B \in \mathcal{T}$.
 (iii) If I is an indexing set and for every $i \in I$, $A_i \in \mathcal{T}$, then $\bigcup_{i \in I} A_i \in \mathcal{T}$.

Theorem 5.6. The following six statements are equivalent:

- (i) L satisfies the distributive law of \wedge over \vee .
 (ii) For any l -valued automaton $\mathcal{M} = (Q, \Sigma, \delta)$, $\mathcal{J}_R = \mathcal{J}_S$.
 (iii) For any l -valued automaton $\mathcal{M} = (Q, \Sigma, \delta)$, $\mathcal{J}_R = \mathcal{J}_l$.
 (iv) For any l -valued automaton $\mathcal{M} = (Q, \Sigma, \delta)$, $\mathcal{J}_S = \mathcal{J}_l$.
 (v) For any l -valued automaton $\mathcal{M} = (Q, \Sigma, \delta)$, \mathcal{J}_S is an l -valued topology on Q .
 (vi) For any l -valued automaton $\mathcal{M} = (Q, \Sigma, \delta)$, \mathcal{J}_R is an l -valued topology on Q .

Proof. Immediate from Corollary 4.4, Propositions 5.1 and 5.4. \square

6. Concluding remarks

In this paper, we have characterized some fundamental properties of l -valued automata theory. The most interesting thing that we found is that some properties of the truth-value lattices of the underlying logic are equivalent to certain properties of automata, which implies that these conclusions in l -valued automata theory may not hold, and therefore need to be reconsidered.

More specifically, (i) the transition relations of l -valued automata have been extended to describe the transitions enabled by strings of input symbols, and particularly, these extensions depend on the distributivity of orthomodular lattices of the underlying logic; (ii) Proposition 4.3 and Corollary 4.4 show that some properties of the l -valued *successor* and *source* operators and l -valued subautomata are equivalent to a property of the truth-value lattices which is exactly equivalent to the distributive law. This is a new characterization of Boolean algebra in the framework of l -valued automata theory; (iii) Proposition 4.5 indicates that the intersection of two subautomata is still a subautomaton if and only if the multiplication ($\&$) is distributive over the union in the truth-value lattice, which is strictly weaker than the usual distributivity; (iv) and Theorem 5.6 presents that some l -valued topological characterizations in terms of the l -valued *successor* and *source* operators also rely on the distributivity of truth-value lattices.

It is quite interesting to note that the distributivity of multiplication over union is strictly weaker than that of meet over union. This leads naturally us to guess that there may be a spectrum of properties of orthomodular lattices which is pointwise equivalent to a spectrum of properties of automata. In [36,37], we established a basic framework of automata theory based on residuated lattice-valued logic. We note that some basic properties in such a framework were verified to be valid (for example Proposition 4.3 (ii) and (iii)), but they do not hold in l -valued automata theory. Furthermore, these results suggest us to examine systematically the logical laws underlying various theorems in mathematics and computer science and to find the weakest logic which guarantees the validity of these theorems. This is a very interesting problem in mathematical logic. In 1952, Rosser and Turquette [40] raised the following problem: if there are many-valued theories beyond the level of predicate calculus, then what are the details of such theories? In a sense, our problem may be seen as an inverse problem of Rosser and Turquette's one. Instead of asking what can be deduced based on a fixed logic? we want to ask how strong logic is needed to derive a certain theorem? We would like to emphasize it here because we believe that it is a promising programme to re-investigate some fundamental issues in mathematics and computer science from the standpoint of non-classical logics.

Acknowledgments

The author is very grateful to Professor Arto Salomaa, Editor, and the referee for invaluable suggestions and comments. Also, the author wishes to thank Professor Ying Mingsheng for very useful discussion and giving me many good ideas.

References

- [1] L.M. Adleman, Molecular computation of solutions to combinatorial problems, *Science* 266 (1994) 1021–1023.
- [2] A. Ambainis, R. Freivalds, One-way quantum finite automata: strengths, weaknesses and generalizations, in: *Proceedings of the 39th Annual Symposium on Foundations of Computer Science*, 1998, pp. 332–341.
- [3] Z. Bavel, The source as a tool in automata, *Inf. Control* 18 (1971) 140–155.
- [4] P. Benioff, The computer as a physical system: a microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines, *J. Stat. Phys.* 22 (1980) 563–591.
- [5] C. Bennett, Logical reversibility of computation, *IBM J. Res. Development* 17 (1979) 525–532.
- [6] A. Bertoni, M. Carpentieri, Regular languages accepted by quantum automata, *Inform. Comput.* 165 (2001) 174–182.
- [7] A. Broadsky, N. Pippenger, Characterizations of 1-way quantum finite automata, *SIAM J. Comp.* 31 (2002) 1456–1478.
- [8] E. Bernstein, U. Vazirani, Quantum complexity theory, *SIAM J. Comp.* 26 (1997) 1411–1473.
- [9] G. Birkhoff, J. von Neumann, The logic of quantum mechanics, *Ann. Math.* 37 (1936) 823–843.
- [10] G.D. Berman, G.D. Doolen, R. Mainieri, V.I. Tsifrinovich, *Introduction to Quantum Computers*, World Scientific, Singapore, 1998.
- [11] J.I. Cirac, P. Zoller, Quantum computations with cold trapped ions, *Phys. Rev. Lett.* 74 (1995) 4091–4094.
- [12] D. Deutsch, Quantum theory, the Church–Turing principle and the universal quantum computer, *Proc. R. Soc. Lond. A* 400 (1985) 97–117.
- [13] M.L. Dalla Chiara, Quantum logic, in: in: D.M. Gabbay, F. Guenther (Eds.), *Handbook of Philosophical Logic*, vol. III, D. Reidel, Dordrecht, 1986, pp. 427–469.
- [14] L.M. Duan, G.C. Guo, Probabilistic cloning and identification of linearly independent quantum states, *Phys. Rev. Lett.* 80 (1998) 4999–5002.
- [15] R.P. Feynman, Simulating physics with computers, *Int. J. Theor. Phys.* 21 (1982) 467–488.
- [16] L. Grover, A fast quantum mechanical algorithms for database search, in: *Proceedings of the 28th Annual ACM Symposium on the Theory of Computing*, 1996, pp. 212–219.
- [17] J. Gruska, *Quantum Computing*, McGraw-Hill, London, 1999.
- [18] S. Gudder, Quantum Computers, *Int. J. Theor. Phys.* 39 (2000) 2151–2177.
- [19] N. Gisin, S. Massar, Optimal quantum cloning machines, *Phys. Rev. Lett.* 79 (1997) 2153–2156.
- [20] A. Kondacs, J. Watrous, On the power of finite state automata, in: *Proceedings of the 38th FOCS*, 1997, pp. 66–75.
- [21] S. Lloyd, A potentially realizable quantum computer, *Science* 261 (1993) 1569–1571.
- [22] P. Mittelstaedt, *Quantum Logic*, D. Reidel, Dordrecht, 1978.
- [23] C. Monroe, Quantum information processing with atoms and photons, *Nature* 416 (2002) 238–246.
- [24] C. Moore, J.P. Crutchfield, Quantum automata and quantum grammars, *Theor. Comput. Sci.* 237 (2000) 275–306.
- [25] M.A. Nielsen, I.L. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press, Cambridge, 2000.
- [26] A. Paz, *Introduction to Probabilistic Automata*, Academic Press, New York, 1971.
- [27] A.K. Pati, S.L. Braunstein, Impossibility of deleting an unknown quantum states, *Nature* 404 (2000) 164–165.
- [28] P. Pták, S. Pulmannová, *Orthomodular Structures as Quantum Logics*, Kluwer, Dordrecht, 1991.
- [29] D.W. Qiu, Characterization of sequential quantum machines, *Int. J. Theor. Phys.* 41 (2002) 811–822.
- [30] D.W. Qiu, Quantum pushdown automata, *Int. J. Theor. Phys.* 41 (2002) 1627–1639.
- [31] D.W. Qiu, Some analogies between quantum cloning and quantum deleting, *Phys. Rev. A* 65 (2002) 052303.
- [32] D.W. Qiu, Combinations of probabilistic and approximate quantum cloning and deleting, *Phys. Rev. A* 65 (2002) 052329.
- [33] D.W. Qiu, Upper bound on the success probability of separation among quantum states, *J. Phys. A* 35 (2002) 6931–6937.
- [34] D.W. Qiu, Non-optimal universal quantum deleting machine, *Phys. Lett. A* 301 (2002) 112–116.
- [35] D.W. Qiu, Some general probabilistic quantum cloning and deleting machines, *Phys. Lett. A* 308 (2003) 335–342.
- [36] D.W. Qiu, Automata theory based on complete residuated lattice-valued logic(I), *Sci. China (F)* 44 (2002) 419–429.
- [37] D.W. Qiu, Automata theory based on complete residuated lattice-valued logic(II), *Sci. China (F)* 45 (2002) 442–452.
- [38] M.O. Rabin, Probabilistic automata, *Inf. Control* 6 (1963) 230–244.
- [39] L. Román, B. Rumbos, Quantum logic revisited, *Found. Phys.* 21 (1991) 727–734.

- [40] J.B. Rosser, A.R. Turquette, *Many-Valued Logics*, North-Holland, Amsterdam, 1952.
- [41] P.W. Shor, Algorithm for quantum computation: discrete logarithms and factoring, in: *Proceedings of the 37th IEEE Annular Symposium on Foundations of Computer science*, 1994, pp. 124–134.
- [42] K. Svozil, *Quantum Logic*, Springer-verlag, Berlin, 1998.
- [43] H. Siegelmann, E.D. Sontag, Analog computation via neural networks, *Theor. Comput. Sci.* 131 (1994) 331–360.
- [44] W. Shukla, A.K. Srivastava, A topology for automata: a note, *Inf. Control* 32 (1976) 163–168.
- [45] J.E. Hopcroft, J.D. Ullman, *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley, Reading, MA, 1979.
- [46] C.P. Williams, S.H. Clearwater, *Explorations in Quantum Computing*, Springer-Verlag, New York, 1998.
- [47] W.K. Wootters, W.H. Zurek, A single quantum cannot be cloned, *Nature* 299 (1982) 802–803.
- [48] A.C. Yao, Quantum circuit complexity, in: *Proceedings of the 34th IEEE Symposium on Foundations of Computer science*, 1993, pp. 352–361.
- [49] M.S. Ying, Automata theory based on quantum logic(I), *Int. J. Theor. Phys.* 39 (2000) 981–991.
- [50] M.S. Ying, Automata theory based on quantum logic(II), *Int. J. Theor. Phys.* 39 (2000) 2545–2557.