Baltic J. Modern Computing, Vol. 4 (2016), No. 4, pp. 665–676
http://dx.doi.org/10.22364/bjmc.2016.4.4.04

Four Collaborations with Rūsiņš Freivalds

Andris AMBAINIS

Faculty of Computing, University of Latvia, Raina bulv. 19, Riga, LV-1586, Latvia

ambainis@lu.lv

Abstract. Rūsiņš Mārtiņš Freivalds (1942-2016) was one of European pioneers of theoretical computer science, making important contributions to the theory of probabilistic algorithms and other fields of theoretical computer science. He was also my first research supervisor at the University of Latvia and influenced my research career quite substantially. In this article, I describe some of my research experiences working together with him, from the first exercises in his undergraduate seminar to how we started working on quantum computing.

Keywords: theory of computation, randomized algorithms, communication complexity, inductive inference, quantum computing, quantum automata

1 Introduction

Rūsiņš Mārtiņš Freivalds was among the leading European theoretical computer scientists of his time. He was one of first to realize that probabilistic algorithms can be more efficient than deterministic, in a variety of contexts, from Turing machines (Freivalds, 1975, 1979c) to algorithms for verifying matrix multiplication (Freivalds, 1979b). Freivalds also made important contributions to other areas of theoretical computer science, from inductive inference (a recursion-theoretic model of learning) to quantum computing, and advised a number of graduate students.

Freivalds was my first research advisor. I started attending his seminar as a 1st year undergraduate and worked with him for the next 5 years. Working with Freivalds was a very important research experience which profoundly influenced me as a scientist. It was an important stepping stone that allowed me to be admitted into the Ph.D. program of University of California, Berkeley where I went on to doing research on quantum information.

Some of the topics that I learned with Rūsiņš Freivalds still fascinate me as a researcher. For example, under his supervision, I learned about deterministic, nondeterministic and probabilistic decision trees and relations between their complexities (for example, deterministic decision tree complexity $D(f)$ being at most $ND^2(f)$ where $ND(f)$ is the nondeterministic decision tree complexity, as described in a later survey by Buhrman and de Wolf (2002)). These interesting relations inspired me to study the quantum version of decision trees (known as *quantum query algorithms*) - a topic of many of my most highly valued research contributions, from the quantum adversary method for proving lower bounds on quantum algorithms (Ambainis, 2002) to the recent separations between quantum and deterministic and probabilistic and deterministic decision tree complexities (Ambainis et al., 2016).

In this article, I recall four important experiences as Freivalds' student and collaborator:

- The 1st Freivalds' seminar that I attended, the problem that we considered at the seminar, and how it got me started in theoretical computer science;
- The first open problem that I solved as Freivalds' student (communication complexity of equality function in the 3-party model (Yao, 1979, Ambainis, 1996b));
- The most difficult result that I obtained as Freivalds' student (the analysis of probability hierarchy for PFIN-type inductive inference (Ambainis, 1996a));
- Our first paper about quantum automata (Ambainis, Freivalds, 1998) which was a starting point in quantum computing for both of us.

2 The 1st Freivalds' seminar

As a high school student, I participated in mathematical olympiads, winning a gold medal at the International Mathematical Olympiad in 1991. Latvian mathematics olympiad team was coached by Agnis Andžāns, a professor at the University of Latvia. When I finished high school and entered university, Andžāns introduced me and two other mathematics olympiad competitors from my year to Rūsiņš Freivalds. Freivalds then started running an introductory seminar for the three of us. (One of the other two students, Ģirts Karnītis, is now a professor of Computer Science at the University of Latvia, specializing in databases and big data.)

During the first seminar, Freivalds gave the following problem to us. Consider computing a Boolean function $f(x_1, \ldots, x_n)$ using AND, OR and NOT gates. Each AND and OR gate takes two inputs (which can be either variables x_i or outputs of other gates), each NOT gate takes 1 input. The result of each gate can be used as an input to an arbitrary number of other gates. For what number of gates $M(n)$ is it true that any Boolean function $f(x_1, \ldots, x_n)$ can be computed with $M(n)$ gates?

The first result that we obtained during the seminar was that $n2^n$ gates were sufficient. Consider the set S consisting of all $(x_1, \ldots, x_n) \in \{0, 1\}^n$ with $f(x_1, \ldots, x_n) = 1$. If S contained all 2^n possible $(x_1, \ldots .x_n)$, then $f(x_1, \ldots, x_n)$ would always be equal to 1 and computing it would be trivial. Otherwise, S contains at most $2^n - 1$ elements. Then, we can compute f as follows:

1. We start by computing $NOT x_i$ for all $i \in \{1, \ldots, n\}$.
2. For every $(y_1, \ldots, y_n) \in S$ we create a gate that outputs 1 if $(x_1, \ldots, x_n) = (y_1, \ldots, y_n)$ and 0 otherwise. This can be done by taking AND of all x_i for $i : y_i = 1$ and $NOT x_i$ for $i : y_i = 0$.
3. We take OR of all gates from the previous step.

The first step requires n gates. In the second step, for every $(y_1, \ldots, y_n) \in S$, we take AND of n values. This can be done with $n-1$ AND gates. The overall number of gates in the second step is $(n-1)|S|$ where $|S|$ denotes the size of S. In the third step, we take OR of $|S|$ values which can be done with $|S|-1$ OR gates. Thus, the overall number of gates is

$$n + (n-1)|S| + (|S|-1) = n|S| + n - 1 < n2^n.$$

For example, if $f(x_1, x_2, x_3)$ is a function that is equal to 1 if (x_1, x_2, x_3) is equal to $(0, 0, 0)$ or $(1, 1, 1)$, the resulting circuit is shown in Figure 1.

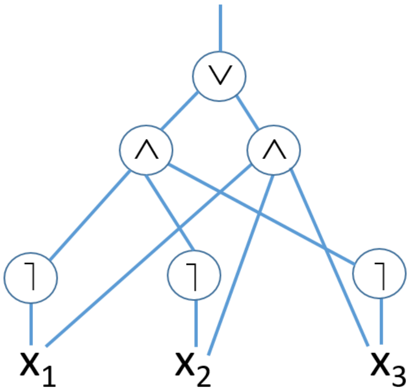

Fig. 1. Circuit for $f(x_1, x_2, x_3)$

The task was to come up with a better construction. Over the next week, I kept thinking about this problem and coming up with better and better constructions. The first conceptual improvement over the construction above was to express the function $f(x_1, \ldots, x_n)$ as

$$f(x_1, \ldots, x_n) = ((NOT x_1) AND f_0(x_2, \ldots, x_n)) OR (x_1 AND f_1(x_2, \ldots, x_n)).$$

Then, we can construct a circuit for f from circuits for f_0 and f_1 and 4 extra gates (one OR, two ANDs and one NOT). This shows that $M(n) \leq 2M(n-1) + 4$. Resolving this recurrence gives $M(n) \leq \frac{5}{2} 2^n$.

The next improvements involved identifying subfunctions that were frequently reused in the construction above and, instead of recomputing them every time, computing them once and reusing the result of the computation.

The best result that I achieved was $M(n) \leq (2 + o(1))\frac{2^n}{n}$. I also proved that $M(n) \geq (\frac{1}{2} - o(1))\frac{2^n}{n}$. The true answer was $M(n) = (1 + o(1))\frac{2^n}{n}$, as shown in the classical papers by Shannon (1949) and Lupanov (1958).

The process of coming up with better and better solutions was quite interesting and exciting for me. It is often said that student's interest depends on whether the lecturer manages to create interest in the subject during the first lecture. In my case, Freivalds created a long-lasting interest in theoretical computer science.

3 The first research problem

My results with the circuit problem impressed Freivalds quite thoroughly. He had actually expected substantially less than what I achieved.

Freivalds started thinking about open problems that he could give to me. On one hand, the problem had to be understandable to a 1st year undergraduate. On the other hand, he wanted to find something that would be interesting to other researchers if I solved it.

He came up with a problem in communication complexity, a research area invented by Yao (1979). Communication complexity studies computing in a setting where the input data is distributed among several parties. It is assumed that all parties have a substantial computational power and can carry out computation very quickly. The only bottleneck is the communication among the parties which is slow. The task is to optimize the computation so that it can be done with a minimum amount of communication. (A detailed review of the field of communication complexity can be found in the books by Kushilevitz and Nisan (2006) and Hromkovič (1997).)

The standard model of communication complexity consists of two parties (Alice and Bob) who want to compute a function $f(x, y)$ with x initially belonging to Alice and y initially belonging to Bob so that the amount of bits communicated between Alice and Bob is as small as possible. Alice and Bob can act either deterministically or probabilistically.

The equality problem is a very well known example where probabilistic communication protocols are more efficient than deterministic ones. In this case, Alice's and Bob's inputs x and y are strings of n bits and they would like to determine if $x = y$. Any deterministic protocol would require them to send n bits. Probabilistically, Alice can randomly choose a prime $p \in \{2, \ldots, cn\}$ for some constant c, interpret x as a number between 0 and $2^n - 1$ and send $(p, x \bmod p)$ to Bob. Bob also interprets y as a number and computes $y \bmod p$. If $x \bmod p = y \bmod p$, Bob concludes that $x = y$. Otherwise, he concludes that $x \neq y$.

This protocol allows to determine whether $x = y$ with Alice communicating $O(\log n)$ bits to Bob. In the first paper on communication complexity, Yao (1979) asked whether there is an efficient communication protocol in a model where Alice and Bob communicate to a third party (*a referee*). Namely, Alice holds an n-bit string x and communicates a message based on this string to the referee. Bob holds another n-bit string x and communicates a message based on this string to the referee. The referee then has to output his guess whether $x = y$.

At the time when I started working with Freivalds, this problem was still open and he gave it to me. After a few months (in Spring 1993), I came up with a probabilistic protocol that solves it with Alice and Bob communicating $O(\sqrt{n})$ bits. Thus, in the model with a referee, probabilistic protocols are still better than deterministic ones (which again need n bits of communication) but their advantage is smaller than in the standard model of communication.

The protocol is quite simple to describe:

1. Before running the protocol, Alice and Bob agree on an encoding scheme $E : \{0,1\}^n \to \{0,1\}^{Cn}$ such that any two encodings $E(x)$, $E(y)$ differ in at least 1/4 of all bits. (It is known from coding theory (McWilliams, Sloane, 1977) that such schemes exist.)
2. Alice computes $E(x)$, Bob computes $E(y)$. They both place their strings into a $\sqrt{Cn} \times \sqrt{Cn}$ table, with one bit in each cell of the table.
3. Alice chooses a random $i \in \{1, \ldots, \sqrt{Cn}\}$ and communicates i and the contents of the i^{th} row of the table.
4. Bob chooses a random $j \in \{1, \ldots, \sqrt{Cn}\}$ and communicates j and the contents of the j^{th} column of the table.
5. The referee compares the values of the entry that belongs to both the i^{th} row and the j^{th} column in Alice's and Bob's messages. If they differ, he concludes that $x \neq y$.

Since $E(x)$ and $E(y)$ differ in at least 1/4 of all bits, this protocol detects $x \neq y$ with a probability at least 1/4. For a higher probability of success, Alice, Bob and the referee can run this protocol several times, concluding that $x = y$ if none of the runs detects $x \neq y$.

Soon after I obtained this result, Freivalds went to several universities in United States to visit his colleagues. After his visit, he told me that at least two prominent scientists, Manuel Blum (Turing Award winner, then at the University of California, Berkeley) and Andrew Yao (Turing Award winner, then at Princeton University) were quite impressed by it.

This was my first research result but it took 2.5 years until it appeared in a journal (Ambainis, 1996b). First, Freivalds promised me to translate the paper into English if I wrote it in Latvian but it turned out that he was too busy with other matters. Secondly, the content of the paper kept changing. Freivalds gave me another problem, about the complexity of deciding whether $x > y$ in the same model of communication complexity. I solved it and started writing it down, only to discover that the solution to this problem follows from a known result (again, by Yao (1983)).Thirdly, since I did not know coding theory at that time, the first version of the paper contained a proof that an encoding scheme with the required properties exists. Later, I replaced it with a reference to a coding theory textbook.

While the paper kept changing, the news about the result had gotten out. The result rekindled interest in the model of communication with a referee (now called *Simultaneous messages* model) and, before my paper was published, several other scientists discovered the same protocol independently of me and also discovered a proof that it is optimal (Babai and Kimmel, 1997, Bourgain and Wigderson, 1996, Newman and Szegedy, 1996).

Interestingly, the result became useful for my current area of research, quantum information. It is a basis for a protocol called *quantum fingerprinting* which allows to encode long strings of classical bits into a small-dimensional quantum state (Buhrman et al., 2001).

After my first result in communication complexity I went on to work on several other questions in this area. From ICALP'1994, Freivalds brought me the conference proceedings containing a survey by Pudlák (1994), titled "Unexpected upper bounds in communication complexity" which contained communication protocols for several problems with an amount of communication that is smaller than one would expect. I then improved two of those protocols (Ambainis, 1996c, Ambainis and Lokam 2000).

4 My most difficult result with Freivalds

One of the main research interests of Freivalds was *inductive inference*, a model for machine learning based on computability theory. Inductive inference was invented in 1960s by Gold (1965, 1967) and is among the older theories of learning. Freivalds, together with other Latvian computer scientists (most notably, Jānis Bārzdiņš, Efim Kinber and Kārlis Podnieks) started working on inductive inference in early 1970s and their research left a substantial impact on this field (as described in later surveys by Freivalds (1991) and Freivalds et al. (1991)).

One of the basic models of inductive inference is finite learning of functions (abbreviated by FIN, first studied by Lindner (1972)):

1. The object to be learned is a computable function $f : \{1, 2, \ldots\} \to \{0, 1\}$ which belongs to some class of functions L.
2. The learner is a Turing machine M which receives values $f(1), f(2), \ldots$ and, at some point, may output a program P which is supposed to compute f.
3. M learns a class L if, for any $f \in L$, given $f(1), f(2), \ldots$, it outputs a program P which, given x, correcly computes f for all $x \in \{1, 2, \ldots\}$ (including x for which the machine M did not see $f(x)$).
4. FIN is the family of all L for which there exists M that learns L.

Freivalds was among the first researchers to study probabilistic algorithms in many contexts, from probabilistic Turing machines (Freivalds, 1975, 1979c) to algorithms for verifying matrix multiplication (Freivalds, 1979b). He also started the study of probabilistic algorithms in the context of inductive inference.

Let $FIN\left\langle p\right\rangle$ denote the class of all L for which there exists a probabilistic Turing machine M such that, for any $f \in L$, the probability that, given $f(1), f(2), \ldots$, M outputs a program P that computes f is at least p. (We note that a success probability p can be achieved either by outputting one correct program with probability p or more or by outputting one of several correct programs $P_1, \ldots, P_m$ with a total probability that is at least p.) Freivalds (1979a) showed that:

- For success probabilities $p > 2/3$, $FIN\left\langle p\right\rangle = FIN$.Thus, in this case, probabilistic machines are not more powerful than deterministic ones.
- For success probability $p = 2/3$, $FIN\left\langle \frac{2}{3}\right\rangle$ is strictly larger than FIN;

He then showed that

$$FIN\left\langle \frac{2}{3}\right\rangle \subset FIN\left\langle \frac{3}{5}\right\rangle \subset \ldots \subset FIN\left\langle \frac{k}{2k-1}\right\rangle \subset FIN\left\langle \frac{k+1}{2k+1}\right\rangle \subset \ldots \quad (1)$$

and, for any $p : \frac{k}{2k-1} < p < \frac{k+1}{2k+1}$, we have $FIN\left\langle \frac{k}{2k-1} \right\rangle = FIN\left\langle p \right\rangle$. Thus, decreasing the success probability increases the capabilities of probabilistic learning machines but this happens in discrete steps at certain probabilities.

For a long time, it remained open how the capabilities increase when the learning machine is required to succeed with probabaility $p < 1/2$ until Daley et al. (1995) and Daley and Kalyanasudaram (1997) discovered that

$$FIN\left\langle \frac{1}{2} \right\rangle \subset FIN\left\langle p_1 \right\rangle \subset \ldots \subset FIN\left\langle p_k \right\rangle \subset FIN\left\langle p_{k+1} \right\rangle \subset \ldots$$

and $FIN\left\langle p \right\rangle = FIN\left\langle p_i \right\rangle$ for $p : p_i < p < p_{i+1}$, for another sequence of probabilities defined by $p_1 = \frac{24}{49}$, $p_2 = \frac{20}{41}$, $\ldots$ and $p_m = \frac{12m-52}{25m-109}$ for $m \geq 11$. This sequence converges to $\lim_{m\to\infty} p_m = \frac{12}{25}$. Obtaining this result was quite difficult. It took the authors several years of work and the final proof was more than 100 pages long. Given that all this effort was just to analyze $FIN\left\langle p \right\rangle$ for probabilities p in a small interval $[\frac{12}{25}, \frac{1}{2}] = [0.48, 0.5]$, analyzing $FIN\left\langle p \right\rangle$ for smaller p looked infeasible.

When I heard about this problem from Freivalds, it fascinated me. Given the difficulty of analyzing FIN, I looked at a simpler model called $PFIN$ (introduced by Case and Ngo-Manguelle, 1979) where it is known that all programs P output by a learning machine M terminate on all inputs. (This allows to avoid some of the more nasty technical issues with analyzing FIN machines.) At that time, it was known that, for probabilities $p > 1/2$, we have

$$PFIN \subset PFIN\left\langle \frac{2}{3} \right\rangle \subset FIN\left\langle \frac{3}{5} \right\rangle \subset \ldots$$
$$\subset FIN\left\langle \frac{k}{2k-1} \right\rangle \subset FIN\left\langle \frac{k+1}{2k+1} \right\rangle \subset \ldots$$

with $PFIN\left\langle \frac{k}{2k-1} \right\rangle = PFIN\left\langle p \right\rangle$ for $p : \frac{k}{2k-1} < p < \frac{k+1}{2k+1}$ (Daley et al., 1992). For smaller p, two sequences $p_1 > p_2 > \ldots$ with

$$PFIN\left\langle p_1 \right\rangle \subset PFIN\left\langle p_2 \right\rangle \subset \ldots \subset PFIN\left\langle p_k \right\rangle \subset PFIN\left\langle p_{k+1} \right\rangle \subset \ldots$$

and $PFIN\left\langle p \right\rangle = PFIN\left\langle p_i \right\rangle$ for $p : p_i < p < p_{i+1}$ were found, with one sequence from $p_1 = \frac{1}{2}$ to $\lim_{i\to\infty} p_i = \frac{4}{9}$ and another sequence from $p_1 = \frac{4}{9}$ to $\lim_{i\to\infty} p_i = \frac{3}{7}$ (Daley et al., 1992, Daley and Kalyanasudaram, 1993). Even though $PFIN$ was simpler, it was still very difficult and Daley and Kalyanasudaram (1993) wrote that the prospects of fully analyzing the power of $PFIN\left\langle p \right\rangle$ even for $p \in [\frac{2}{5}, \frac{1}{2}]$ look quite bleak.

In my work (Ambainis, 1996a, 2008), I took a different approach to studying PFIN. Instead of trying to find out particular probabilities where the power of $PFIN\left\langle p \right\rangle$ changed, I looked at more general questions. The main technical result was

Theorem 1 *(Ambainis, 1996a, 2008) Let P_{PFIN} be the set of all p such that $PFIN\left\langle p \right\rangle$ is larger than $PFIN\left\langle p + \epsilon \right\rangle$ for any $\epsilon > 0$. Then P_{PFIN} is equal to the set A defined by the following rules:*

1. $1 \in A$;
2. *if* $p \in [0,1]$ *and there exist* $p_1, \ldots, p_s \in A$ *and* $q_1, \ldots, q_s \geq 0$ *such that* $p = q_1 + q_2 + \ldots + q_s$ *and* $p_i = \frac{p}{1-p+q_i}$, *then* $p \in A$.

This result has several consequences. By using it, I showed that there is an algorithm, which, given p and q, decides whether $PFIN\,\langle p \rangle = PFIN\,\langle q \rangle$. I also characterized the complexity of the set P_{PFIN}, by showing that its structure is equal to the ordinal number ϵ_0.

Ordinal numbers were another favorite of Freivalds. An ordinal is a set S whose elements are ordered so that, for every subset $S' \subseteq S$, S' has a smallest element (i.e. $x \in S'$ such that $x < x'$ for all other $x' \in S$). Some examples of ordinals are:

- the set of natural numbers, in an increasing order;
- the set of pairs of natural numbers (x, y) in the order defined by $(x, y) < (x', y')$ if $x < x'$ or if $x = x'$ and $y < y'$;
- the set of k-tuples of natural numbers $(x_1, \ldots, x_k)$ with $(x_1, \ldots, x_k) \leq (y_1, \ldots, y_k)$ if $x_1 = y_1$, ..., $x_{i-1} = y_{i-1}$ and $x_i < y_i$, for some $i \in \{1, \ldots, k\}$;
- the set of all $(x_1, \ldots, x_k)$ for all k with $(x_1, \ldots, x_k) \leq (y_1, \ldots, y_{k'})$ if $k < k'$ and the order for the $k = k'$ case as defined above.

The ordering types of these sets are denoted w, w^2, w^k and w^w. One can then define a sequence of even more complicated ordering types

$$w^w, w^{w^w}, w^{w^{w^w}}, \ldots,$$

with ϵ_0 being the limit of this sequence. I showed that ϵ_0 is also the ordering type of P_{PFIN} (when considered in decreasing order, with 1 as the smallest element and 0 as the largest). This shows that P_{PFIN} is vastly more complicated than the part which was explored before (which consisted of 3 sequences of type w each).

This work is still among most involved and mathematically most interesting things that I have done. The resulting paper, however, did not receive much attention in the theoretical computer science community. The focus of learning theory had shifted from abstract 1960s models like inductive inference (*which encompass any learning task but on a very abstract level*) to more concrete models (*which are less general but more suitable for obtaining algorithms for concrete learning tasks*).

5 Our first quantum collaboration

Quantum computing caught Freivalds' attention quite early. Seeing that the world was losing interest in inductive inference, he was looking for new research topics. In 1993, he came back from FOCS'1993 (*IEEE Conference on Foundations of Computer Science, one of two top theoretical computer science conferences in the world*) and gave me the conference proceedings with a paper on quantum computing (Yao, 1993), saying "This is the thing to study".

At this point, very few people were working on quantum computing. The basic models (such as quantum Turing machines and quantum circuits) were just defined a few years ago and the total number of papers in the field which one could call "quantum theoretical computer science" was around 10. The field would get a major boost a few months later when Peter Shor (1994) discovered his quantum algorithm for factoring but this had not yet happened. At that moment, only a few pioneers were looking at quantum computing at that moment.

I read that paper and a paper on quantum bit commitment (Brassard et al., 1993) in the same proceedings. I would read a few more papers on quantum algorithms in the next years. But, without anyone to guide me, it was difficult to do something on my own. So, I had no results on quantum computing until I left for my Ph.D. at University of California, Berkeley in 1997.

When I arrived at Berkeley, Umesh Vazirani was teaching a course on quantum computing in my first semester there. This was one of the first organized courses on quantum computing in the world. It gave me an opportunity to learn the subject in an organized way, including the things which I missed out while reading the research papers on my own. At the end of the course, Vazirani gave a list of open problems for course projects or future work. This was a great idea: it helped students to start their own research projects.

One of the problems on the list was developing a theory of quantum finite automata. It caught my attention because finite automata was one of topics on which I worked with Freivalds in Latvia. So, I knew something about the subject and could use my knowledge to build a theory in the quantum case. At the time, there were two papers on quantum automata, by Moore and Crutchfield (2000) and by Kondacs and Watrous (1997), with a more general model proposed by Kondacs and Watrous. One of my first results (obtained in December 1997 at Berkeley) was that the power of Kondacs-Watrous model increased if the required success probability was decreased. Namely:

Theorem 2 *(Ambainis and Freivalds, 1998)*

1. *If a language L is recognized by a 1-way quantum finite automaton (QFA) in the Kondacs-Watrous model with probability more than 7/9, it can be also recognized with probability 1.*
2. *There is a language L that is recognizable by a 1-way quantum finite automaton (QFA) in the Kondacs-Watrous model with probability 0.68... but not with probability 1.*

In its spirit, this result is similar to results about PFIN in the previous section but the reasons why the power of the model depends on the success probability p are completely different.

A few weeks later, I went on winter break to Latvia and met with Freivalds. To my surprise, he was now also studying quantum automata. In the previous summer, his student Juris Smotrovs (now a professor of computer science at the University of Latvia) went to a summer school in Finland on unconventional models of computation which included quantum computing. Following that, Freivalds started a seminar on quantum computing. Smotrovs taught what he had learned in Finland and they all tried to figure out: how would a quantum finite automaton look like and what would it be able to do?

Freivalds had a great guess for a problem where quantum finite automata would be better than classical. The problem was to recognize whether the length of the input word was divisible by p. To solve this problem, a classical automaton needs p states. Freivalds had an idea how to do counting modulo p with quantum states, by rotating a quantum state by an angle $\frac{2\pi k}{p}$ after reading every input letter (for some integer k). In this way, a quantum state would complete a full rotation by $2\pi k$ after reading every p input letters and return to the starting point.

The part that was not clear was: how do you choose k? Every fixed k would work in some cases but fail in other cases. Freivalds was trying to build a quantum automaton in which parts of the quantum states were rotated by different angles $\frac{2\pi k_i}{p}$, with all parts returned to the starting state after every p letters. He had a specific choice of parameters k_i in mind but the resulting trigonometric expression for the success probability was too complicated.

Seeing this, I proposed a simpler idea. At Berkeley, I had just finished a course on Randomized Algorithms (taught by Alistair Sinclair) and I proposed to choose k_i's randomly. Quite easily, I showed that a random choice worked, with a very high probability. We obtained a quantum automaton which solved the problem, by using a state-space with $O(\log p)$ dimensions, exponentially better than classical automata. (This result was also published in Ambainis and Freivalds, 1998).

Interestingly, this was the quantum counterpart of a problem which Freivalds had studied in the probabilistic context: given a language that requires n states for deterministic automata, how small can the best probabilistic automaton be? His first result was an example where probabilistic automata can solve the problem with $O(\frac{\log^2 n}{\log \log n})$ states but deterministic automata require n states (Freivalds, 1982) and he eventually improved the gap to $O(\log n)$ states for probabilistic vs. n states for deterministic automata (Freivalds, 2008).

This was the starting point in quantum computing for both of us. It started a collaboration in which I discussed research with Freivalds and gave talks in his seminar every time when I came to Riga for summer or winter breaks, for the next 4 or 5 years. Freivalds with his students visited me at Berkeley, Princeton and Waterloo. He was particularly fascinated by the result that the power of QFAs changed with the success probability and its parallels with his work on FIN and spent a lot of time with his students figuring out when and how exactly the power of QFAs changed, with me also contributing some ideas to this work (Ambainis et al., 1999, Ambainis et al., 2001, Ambainis and Kikusts 2003, Golovkins et al. 2011).

I went back to Berkeley in January 1998 and presented our construction of quantum automata to Umesh Vazirani and members of his group. This lead to more discussions about quantum computing, with members of his group describing their research problems to me. I then started to work on some of these problems, going from one topic in the theory of quantum computing to another.

The collaboration between myself and Freivalds also helped me to maintain a connection to Latvia in general and University of Latvia. In particular, this was one of the reasons which lead to my return to Latvia in 2007, ten years after I left for Berkeley.

References

Ambainis, A. (1996a). Probabilistic and Team PFIN-Type Learning: General Properties. *Proceedings of COLT 1996*, pp. 157-168.

Ambainis, A. (1996b). Communication Complexity in a 3-Computer Model. *Algorithmica*, 16(3): 298-301.

Ambainis, A. (1996c). Upper Bounds on Multiparty Communication Complexity of Shifts. *Proceedings of STACS 1996*, 631-642.

Ambainis, A. (2002). Quantum Lower Bounds by Quantum Arguments. *Journal of Computer and System Sciences*, 64(4): 750-767

Ambainis, A. (2008). Probabilistic and team PFIN-type learning: General properties. *Journal of Computer and System Sciences*, 74(4): 457-489

Ambainis, A., Balodis, K., Belovs, A., Lee, T., Santha, M. Smotrovs, J. (2016). Separations in query complexity based on pointer functions. *Proceedings of STOC 2016*, pp. 800-813.

Ambainis, A., Bonner, R.F., Freivalds, R., Kikusts, A. (1999). Probabilities to Accept Languages by Quantum Finite Automata. *Proceedings of COCOON 1999*, pp. 174-183.

Ambainis, A., Freivalds, R. (1998). 1-Way Quantum Finite Automata: Strengths, Weaknesses and Generalizations. *Proceedings of FOCS 1998*, pp. 332-341.

Ambainis, A., Ķikusts, A. (2003). Exact Results for Accepting Probabilities of Quantum Automata. *Theoretical Computer Science*, 295: 3-25

Ambainis, A., Ķikusts, A., Valdats, M. (2001). On the Class of Languages Recognizable by 1-Way Quantum Finite Automata. *Proceedings of STACS 2001*, pp. 75-86.

Ambainis, A., Lokam, S.L. (2000). Improved Upper Bounds on the Simultaneous Messages Complexity of the Generalized Addressing Function. *Proceedings of LATIN 2000*, pp. 207-216.

Babai, L., Kimmel, P.G. (1997). Randomized Simultaneous Messages: Solution of a Problem of Yao in Communication Complexity. *IEEE Conference on Computational Complexity*, pp. 239-246

Bourgain, J., Wigderson, A. (1996). Private communication by Avi Wigderson, March 1996, cited in Babai, Kimmel, 1997.

Brassard, G., Crépeau, C., Jozsa, R., Langlois, D. (1993). A Quantum Bit Commitment Scheme Provably Unbreakable by both Parties. *Proceedings of FOCS 1993*, pp. 362-371.

Buhrman, H., Cleve, R., Watrous, J., de Wolf, R. (2001). Quantum fingerprinting. *Physical Review Letters* ,87 (16), 167902.

Buhrman, H. and de Wolf, R. (2002). Complexity Measures and Decision Tree Complexity: A Survey. *Theoretical Computer Science*, 288(1):21-43.

Case, J., Jain, S., Ngo-Manguelle, S. (1994). Refinements of inductive inference by Popperian and reliable machines. *Kybernetika* 30(1): 23-52.

Case, J., Ngo- Manguelle, S. (1979) Refinements of inductive inference by Popperian machines. Technical Report, Department of Computer Science, State University of New York, Buffalo. Published in a journal in a revised form in 1994, as Case et al., 1994.

Daley, R., Kalyanasundaram, B. (1993). Use of reduction arguments in determining Popperian FIN-type learning capabilities. In K. Jantke, S. Kobayashi, E. Tomita, and T. Yokomori, editors, *Algorithmic Learning Theory: Fourth International Workshop (ALT '93)*, volume 744 of *Lecture Notes in Artificial Intelligence*, pp. 173–186.

Daley, R., Kalyanasundaram, B. (1997). FINite learning capabilities and their limits. In *Proceedings of the 10th Annual Conference on Computational Learning Theory*, pp. 81–89. ACM Press.

Daley, R., Kalyanasundaram, B., Velauthapillai, M. (1992). The power of probabilism in Popperian finite learning. In *Analogical and Inductive Inference, Proceedings of the Third International Workshop*, volume 642 of *Lecture Notes in Artificial Intelligence*, pp. 151–169.

Daley, R., Kalyanasundaram, B., Velauthapillai, M. (1995). Breaking the probability $\frac{1}{2}$ barrier in FIN-type learning. *Journal of Computer and System Sciences*, 50(3):574–599.

Freivalds, R. (1975). Fast computations by probabilistic Turing machines. *Theory of Algorithms and Programs*, Riga, University of Latvia, 233:201- 205 (in Russian)

Freivalds, R. (1979a). Finite identification of general recursive functions by probabilistic strategies. In *Proceedings of the Conference on Fundamentals of Computation Theory*, pages 138–145. Akademie-Verlag, Berlin.

Freivalds, R. (1979b). Fast Probabilistic Algorithms. *Proceedings of MFCS 1979*, pp. 57-69

Freivalds, R. (1979c). Speeding up recognition of certain sets by usage of random number generators. *Problemi kibernetiki*, 36:209-224 (in Russian)

Freivalds, R. (1982). On the growth of the number of states in result of determinization of probabilistic finite automata. *Automatic Control and Computer Sciences*, no. 3, 39-42

Freivalds, R. (1991). Inductive Inference of Recursive Functions: Qualitative Theory. *Baltic Computer Science*, Lecture Notes in Computer Science 502:77-110.

Freivalds, R. (2008). Non-Constructive Methods for Finite Probabilistic Automata. *International Journal of Foundations of Computer Science* 19(3): 565-580.

Freivalds, R., Barzdins, J., Podnieks, K. (1991). Inductive Inference of Recursive Functions: Complexity Bounds. *Baltic Computer Science*, Lecture Notes in Computer Science 502:111-155

Gold, E.M. (1965). Limiting Recursion. *Journal of Symbolic Logic*, 30(1): 28-48.

Gold, E.M. (1967). Language Identification in the Limit. *Information and Control*, 10(5): 447-474.

Golovkins, M., Kravtsev, M., Kravcevs, V. (2011). Quantum Finite Automata and Probabilistic Reversible Automata: R-trivial Idempotent Languages. *Proceedings of MFCS 2011*, pp. 351-363.

Hromkovič, J. (1997). *Communication complexity and parallel computing*. Springer.

Kondacs, A., Watrous, J. (1997). On the Power of Quantum Finite State Automata. *Proceedings of FOCS 1997*, pp. 66-75.

Kushilevitz, E., Nisan, N. (1996). *Communication Complexity.* Cambridge University Press.

Lindner, R. (1972). Algorithmische Erkennung, Ph.D. dissertation, Friedrich-Schiller-Universitat, Jena, G.D.R.

Lupanov, O. (1958). A method of circuit synthesis, *Izvestia V.U.Z. Radiofizika*, 1 :120-140.

MacWilliams, F.J., Sloane, N. J. A. (1977). *The Theory of Error-Correcting Codes.* North-Holland, Amsterdam.

Moore, C., Crutchfield, J.P. (2000). Quantum automata and quantum grammars. *Theoretical Computer Science* 237(1-2): 275-306, available as a preprint since 1997.

Newman, I., Szegedy, M. (1996). Public vs. Private Coin Flips in One Round Communication Games (Extended Abstract). *Proceedings of STOC 1996*, pp. 561-570.

Pudlák, P. (1994). Unexpected Upper Bounds on the Complexity of Some Communication Games. *Proceedings of ICALP 1994*, pp. 1-10.

Shannon, C. (1949). The synthesis of two-terminal switching circuits, *Bell Systems Technical Journal* 28:59-98.

Shor, P.W. (1994). Algorithms for Quantum Computation: Discrete Logarithms and Factoring. *Proceedings of FOCS 1994*, pp. 124-134.

Yao, A.C. (1979). Some Complexity Questions Related to Distributive Computing (Preliminary Report). *Proceedings of STOC 1979*, pp. 209-213.

Yao, A.C. (1983). Lower Bounds by Probabilistic Arguments (Extended Abstract). *Proceedings of FOCS 1983*, pp. 420-428.

Yao, A.C. (1993). Quantum Circuit Complexity. *Proceedings of FOCS 1993*, pp. 352-361

Received November 29, 2016 , accepted December 2, 2016