Active Learning of Classes of Recursive Functions by Ultrametric Algorithms

Thomas Zeugmann and Rūsiņš Freivalds

Hokkaido University

WS on Computability Theory and the Foundations of Mathematics 2017,
Singapore

Active Learning I

The learning model considered in this part is an *active learning* one which is called *query learning*. In the query learning model the learner has access to a teacher that truthfully answers queries of a prespecified type. Here we only consider *value queries*. That is, the query is a natural number x, and the answer to the query is $f(x)$. A query learner is an algorithmic device that, depending on the answers already received, *either computes a new value query or it returns a hypothesis* i *and stops*.

Active Learning I

The learning model considered in this part is an *active learning* one which is called *query learning*. In the query learning model the learner has access to a teacher that truthfully answers queries of a prespecified type. Here we only consider *value queries*. That is, the query is a natural number x, and the answer to the query is $f(x)$. A query learner is an algorithmic device that, depending on the answers already received, *either computes a new value query or it returns a hypothesis* i *and stops*.

The hypothesis is interpreted with respect to a *fixed Gödel numbering* φ and it is required that the hypothesis returned satisfies $\varphi_i = f$.

Active Learning II

We are interested in active learners that can infer whole classes $\mathcal{U}$ of recursive functions. The complexity measure is then the *worst-case number of queries asked* to identify all the functions from the target class $\mathcal{U}$. We refer to any query learner as *query inference machine* (abbr. QIM).

Active Learning II

We are interested in active learners that can infer whole classes $\mathcal{U}$ of recursive functions. The complexity measure is then the *worst-case number of queries asked* to identify all the functions from the target class $\mathcal{U}$. We refer to any query learner as *query inference machine* (abbr. QIM).

Automata theory and complexity theory have considered several natural generalizations of deterministic algorithms, namely, *nondeterministic algorithm* and *probabilistic algorithms*. In many cases these generalized algorithms allow for computations having a complexity that is strictly less than their deterministic counterpart.

Active Learning III

Definition 1

We say that a nondeterministic QIM *learns a function* f if

(1) there is at least one computation path such that the QIM outputs a correct result on f, i.e., a program j s.t. $\varphi_j = f$;

(2) at no computation path the QIM produces an incorrect result on f.

Active Learning III

Definition 1

We say that a nondeterministic QIM *learns a function* f if

(1) there is at least one computation path such that the QIM outputs a correct result on f, i.e., a program j s.t. $\varphi_j = f$;

(2) at no computation path the QIM produces an incorrect result on f.

Probabilistic algorithms can be represented by rooted trees. The leaves of the tree are the output nodes and the probability to reach a leaf is computed in the usual way.

Definition 2

We say that a probabilistic QIM *produces a result m with a probability p* if the sum of the probabilities of all leaves which correctly produce the result m is no less than p.

Active Learning IV

Definition 3

We say that a probabilistic QIM *learns a function* f *with a probability* p if

(1) the sum of all probabilities of all leaves which produce a correct result on f, i.e., a number j such that $\varphi_j = f$, is no less than p.

(2) at no computation path the QIM produces an incorrect result on f.

Active Learning IV

Definition 3

We say that a probabilistic QIM *learns a function* f *with a probability* p if

(1) the sum of all probabilities of all leaves which produce a correct result on f, i.e., a number j such that $\varphi_j = f$, is no less than p.

(2) at no computation path the QIM produces an incorrect result on f.

Freivalds (2012) introduced a new type of indeterministic algorithms called *ultrametric* algorithms. So, ultrametric algorithms are a new concept and their potential still has to be explored. This is the first paper showing a problem, where ultrametric algorithms *have advantages* over nondeterministic algorithms.

Active Learning V

Ultrametric algorithms are very similar to probabilistic algorithms but while probabilistic algorithms use *real* numbers r with $0 \leqslant r \leqslant 1$ as parameters, ultrametric algorithms use *p-adic* numbers as their parameters.

Active Learning V

Ultrametric algorithms are very similar to probabilistic algorithms but while probabilistic algorithms use *real* numbers r with $0 \leqslant r \leqslant 1$ as parameters, ultrametric algorithms use *p-adic* numbers as their parameters.

As it turns out the usage of p-adic numbers as *amplitudes* and the ability to perform *measurements* to transform amplitudes into real numbers are inspired by quantum computations and allow for algorithms not possible in classical computations.

Active Learning V

Ultrametric algorithms are very similar to probabilistic algorithms but while probabilistic algorithms use *real* numbers r with $0 \leqslant r \leqslant 1$ as parameters, ultrametric algorithms use *p-adic* numbers as their parameters.

As it turns out the usage of p-adic numbers as *amplitudes* and the ability to perform *measurements* to transform amplitudes into real numbers are inspired by quantum computations and allow for algorithms not possible in classical computations.

Ostrowski (1916) showed that any non-trivial absolute value on the rational numbers $\mathbb{Q}$ is equivalent to either the usual real absolute value or a p-adic absolute value. So, using p-adic numbers is not one of many possibilities, but the only remaining one to generalize deterministic algorithms.

Ultrametric Algorithms I

Let p be an arbitrary prime number. A number $a \in \mathbb{N}$ with $0 \leqslant a \leqslant p - 1$ is called a *p-adic digit*. A *p-adic integer* is by definition a sequence $(a_i)_{i \in \mathbb{N}}$ of p-adic digits. We write this conventionally as $\cdots a_i \cdots a_2 a_1 a_0$, i.e., the a_i are written from left to right.

Ultrametric Algorithms I

Let p be an arbitrary prime number. A number $a \in \mathbb{N}$ with $0 \leqslant a \leqslant p-1$ is called a *p-adic digit*. A *p-adic integer* is by definition a sequence $(a_i)_{i \in \mathbb{N}}$ of p-adic digits. We write this conventionally as $\cdots a_i \cdots a_2 a_1 a_0$, i.e., the a_i are written from left to right.

If n is a natural number, and $n = \overline{a_{k-1} a_{k-2} \cdots a_1 a_0}$ is its p-adic representation, i.e., $n = \sum_{i=0}^{k-1} a_i p^i$, where each a_i is a p-adic digit, then we identify n with the p-adic integer (a_i), where $a_i = 0$ for all $i \geqslant k$. This means that the natural numbers can be identified with the p-adic integers $(a_i)_{i \in \mathbb{N}}$ for which all but finitely many digits are 0. In particular, the number 0 is the p-adic integer all of whose digits are 0, and 1 is the p-adic integer all of whose digits are 0 except the right-most digit a_0 which is 1.

Ultrametric Algorithms II

To obtain p-adic representations of all rational numbers, $\frac{1}{p}$ is represented as $\cdots 00.1$, the number $\frac{1}{p^2}$ as $\cdots 00.01$, and so on. For any p-adic number it is allowed to have infinitely many (!) digits to the left of the "p-adic" point but only a finite number of digits to the right of it.

Ultrametric Algorithms II

To obtain p-adic representations of all rational numbers, $\frac{1}{p}$ is represented as $\cdots 00.1$, the number $\frac{1}{p^2}$ as $\cdots 00.01$, and so on. For any p-adic number it is allowed to have infinitely many (!) digits to the left of the "p-adic" point but only a finite number of digits to the right of it.

However, p-adic numbers are not merely a generalization of rational numbers. They are related to the notion of *absolute value* (or *norm*) of numbers.
A norm is called *ultrametric* if the triangle inequality can be replaced by the stronger statement: $\|x + y\| \leqslant \max\{\|x\|, \|y\|\}$.

Ultrametric Algorithms III

Definition 4

Let $p \in \{2, 3, 5, 7, 11, 13, \ldots\}$ be any prime number. For any nonzero integer a, let the p-adic *ordinal* (or *valuation*) of a, denoted $\operatorname{ord}_p a$, be the highest power of p which divides a, i.e., the greatest number $m \in \mathbb{N}$ such that $a \equiv 0 \bmod p^m$. For any rational number $x = a/b$ we define $\operatorname{ord}_p x =_{df} \operatorname{ord}_p a - \operatorname{ord}_p b$. Additionally, $\operatorname{ord}_p x =_{df} \infty$ if and only if $x = 0$.

Ultrametric Algorithms III

Definition 4

Let $p \in \{2, 3, 5, 7, 11, 13, \ldots\}$ be any prime number. For any nonzero integer a, let the p-adic *ordinal* (or *valuation*) of a, denoted $\operatorname{ord}_p a$, be the highest power of p which divides a, i.e., the greatest number $m \in \mathbb{N}$ such that $a \equiv 0 \bmod p^m$. For any rational number $x = a/b$ we define $\operatorname{ord}_p x =_{df} \operatorname{ord}_p a - \operatorname{ord}_p b$. Additionally, $\operatorname{ord}_p x =_{df} \infty$ if and only if $x = 0$.

Let $x = 63/550 = 2^{-1} \cdot 3^2 \cdot 5^{-2} \cdot 7^1 \cdot 11^{-1}$. Thus, we have

$$\begin{aligned} \operatorname{ord}_2 x &= -1 & \operatorname{ord}_7 x &= +1 \\ \operatorname{ord}_3 x &= +2 & \operatorname{ord}_{11} x &= -1 \\ \operatorname{ord}_5 x &= -2 & \operatorname{ord}_p x &= 0 \ \text{ for every } p \notin \{2, 3, 5, 7, 11\} . \end{aligned}$$

Ultrametric Algorithms IV

Definition 5

Let $p \in \{2, 3, 5, 7, 11, 13, \ldots\}$ be any prime number. For any rational number x, we define its p-*norm* as $p^{-\mathrm{ord}_p x}$, and we set $\|0\|_p =_{df} 0$.

Ultrametric Algorithms IV

Definition 5

Let $p \in \{2, 3, 5, 7, 11, 13, \ldots\}$ be any prime number. For any rational number x, we define its p-*norm* as $p^{-\mathrm{ord}_p x}$, and we set $\|0\|_p =_{df} 0$.

For example, with $x = 63/550 = 2^{-1} \cdot 3^2 \cdot 5^{-2} \cdot 7^1 \cdot 11^{-1}$ we obtain:

$$\begin{array}{rclcrcll} \|x\|_2 & = & 2 & & \|x\|_7 & = & 1/7 & \\ \|x\|_3 & = & 1/9 & & \|x\|_{11} & = & 11 & \\ \|x\|_5 & = & 25 & & \|x\|_p & = & 1 & \text{for every } p \notin \{2,3,5,7,11\}\,. \end{array}$$

Rational numbers are p-adic integers for all prime numbers p. Since the definitions given above are all we need, we finish our exposition of p-adic numbers here.

Ultrametric Algorithms V

We continue with *ultrametric algorithms*. In the following, p always denotes a prime number.

Ultrametric algorithms are described by finite directed acyclic graphs (abbr. DAG), where exactly one node is marked as root. As usual, the root does not have any incoming edge. Furthermore, every node having outdegree zero is said to be a *leaf*. The leaves are the output nodes of the DAG.

Ultrametric Algorithms V

We continue with *ultrametric algorithms*. In the following, p always denotes a prime number.
Ultrametric algorithms are described by finite directed acyclic graphs (abbr. DAG), where exactly one node is marked as root. As usual, the root does not have any incoming edge. Furthermore, every node having outdegree zero is said to be a *leaf*. The leaves are the output nodes of the DAG.

Let v be a node in such a graph. Then each outgoing edge is labeled by a p-adic number which we call *amplitude*. We require that the sum of all amplitudes that correspond to v sums up to 1. In order to determine the *total amplitude* along a computation path, we need the following definition:

Ultrametric Algorithms VI

Definition 6

The *total amplitude* of the root is defined to be 1. Furthermore, let v be a node at depth d in the DAG, let α be its total amplitude, and let $\beta_1, \beta_2, \cdots, \beta_k$ be the amplitudes corresponding to the outgoing edges $e_1, \ldots, e_k$ of v. Let $v_1, \ldots, v_k$ be the nodes where the edges $e_1, \ldots, e_k$ point to. Then the total amplitude of v_ℓ, $\ell \in \{1, \ldots, k\}$, is defined as follows:

(1) If the indegree of v_ℓ is one, then its *total amplitude* is $\alpha\beta_\ell$.

Ultrametric Algorithms VI

Definition 6

The *total amplitude* of the root is defined to be 1. Furthermore, let v be a node at depth d in the DAG, let α be its total amplitude, and let $\beta_1, \beta_2, \cdots, \beta_k$ be the amplitudes corresponding to the outgoing edges $e_1, \ldots, e_k$ of v. Let $v_1, \ldots, v_k$ be the nodes where the edges $e_1, \ldots, e_k$ point to. Then the total amplitude of v_ℓ, $\ell \in \{1, \ldots, k\}$, is defined as follows:

(1) If the indegree of v_ℓ is one, then its *total amplitude* is $\alpha\beta_\ell$.

(2) If the indegree of v_ℓ is bigger than one, i.e., if two or more computation paths are joined, say m paths, then let $\alpha, \gamma_2, \ldots, \gamma_m$ be the corresponding total amplitudes of the predecessors of v_ℓ and let $\beta_\ell, \delta_2, \ldots, \delta_m$ be the amplitudes of the incoming edges The *total amplitude of the node v_ℓ* is then defined to be $\alpha\beta_\ell + \gamma_2\delta_2 + \cdots + \delta_m\gamma_m$.

Ultrametric Algorithms VII

Note that the total amplitude is a p-adic integer.

Ultrametric Algorithms VII

Note that the total amplitude is a p-adic integer.

It remains to define what is meant by saying that a p-ultrametric algorithm produces a result with a certain probability. This is specified by performing a so-called *measurement* at the leaves of the corresponding DAG. Here by measurement we mean that we transform the total amplitude β of each leaf to $\|\beta\|_p$. We refer to $\|\beta\|_p$ as the p-*probability* of the corresponding computation path.

Ultrametric Algorithms VIII

Definition 7

We say that a p-ultrametric algorithm *produces a result* m *with a probability* q if the sum of the p-probabilities of all leaves which correctly produce the result m is no less than q.

Ultrametric Algorithms VIII

Definition 7

We say that a p-ultrametric algorithm *produces a result* m *with a probability* q if the sum of the p-probabilities of all leaves which correctly produce the result m is no less than q.

Definition 8

We say that a p-ultrametric QIM *learns a function* f *with a* p*-probability* q if

(1) the sum of the p-*probabilities* of all leaves which produce a correct result on f, i.e., a number j such that $\varphi_j = f$ is no less than q,

(2) at no computation path the QIM produces an incorrect result on f.

Results I

To show our results we use a combinatorial structure called the *Fano plane*. It is one of *finite geometries* (see Meserve (1983)). The Fano plane consists of seven *points* $0, 1, 2, 3, 4, 5, 6$ and seven *lines* $(0, 1, 3)$, $(1, 2, 4)$, $(2, 3, 5)$, $(3, 4, 6)$, $(4, 5, 0)$, $(5, 6, 1)$, $(6, 0, 2)$. For any two points i, j with $i \neq j$, in this geometry there is exactly one line that contains these points (cf. Figure 1). For any two different lines in this geometry there is exactly one point contained in these two lines. In our construction the points $0, 1, 2, 3, 4, 5, 6$ are interpreted as colored in two colors RED and BLUE, respectively.

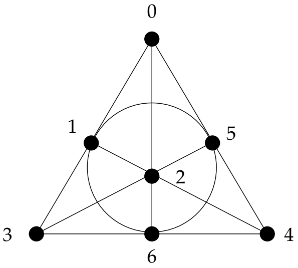

Figure 1: The Fano Plane

Results II

Let φ be a Gödel numbering of $\mathcal{P}$. We consider the following class $\mathcal{U}_7$ of recursive functions: Each function $f \in \mathcal{U}_7$ is such that $f \in \mathcal{R}$ and

(1) every $f(x)$ where $0 \leqslant x \leqslant 6$ equals either 2^s or 3^t, where $s, t \in \mathbb{N}$, $s, t \geqslant 1$,

Results II

Let φ be a Gödel numbering of $\mathcal{P}$. We consider the following class $\mathcal{U}_7$ of recursive functions: Each function $f \in \mathcal{U}_7$ is such that $f \in \mathcal{R}$ and

(1) every $f(x)$ where $0 \leqslant x \leqslant 6$ equals either 2^s or 3^t, where $s, t \in \mathbb{N}$, $s, t \geqslant 1$,

(2) if $0 \leqslant x_1 < x_2 \leqslant 6$, $f(x_1) = 2^s$ and $f(x_2) = 2^t$, then $f(x_1) = f(x_2)$,

(3) if $0 \leqslant x_1 < x_2 \leqslant 6$, $f(x_1) = 3^s$ and $f(x_2) = 3^t$, then $f(x_1) = f(x_2)$,

Results II

Let φ be a Gödel numbering of $\mathcal{P}$. We consider the following class $\mathcal{U}_7$ of recursive functions: Each function $f \in \mathcal{U}_7$ is such that $f \in \mathcal{R}$ and

(1) every $f(x)$ where $0 \leqslant x \leqslant 6$ equals either 2^s or 3^t, where $s, t \in \mathbb{N}$, $s, t \geqslant 1$,

(2) if $0 \leqslant x_1 < x_2 \leqslant 6$, $f(x_1) = 2^s$ and $f(x_2) = 2^t$, then $f(x_1) = f(x_2)$,

(3) if $0 \leqslant x_1 < x_2 \leqslant 6$, $f(x_1) = 3^s$ and $f(x_2) = 3^t$, then $f(x_1) = f(x_2)$,

(4) there is a line (i, j, k) in the Fano plane such that $f(i) = f(j) = f(k) = 2^s$ and $\varphi_s = f$ or there exists a line (i, j, k) in the Fano plane such that $f(i) = f(j) = f(k) = 3^t$ and $\varphi_t = f$.

Results III

Comment. By construction of $\mathcal{U}_7$ the points $0, 1, 2, 3, 4, 5, 6$ can be interpreted as colored in two colors. Some points $f(i)$ are such that $f(i) = 2^s$ (described as RED) while some other points j are such that $f(j) = 3^t$ (described as BLUE). By the properties of the Fano plane, for every such coloring in two colors there is a line such that the 3 points on it are colored in the same color, and there cannot exist two lines colored in opposite colors.

Results III

Comment. By construction of $\mathcal{U}_7$ the points $0, 1, 2, 3, 4, 5, 6$ can be interpreted as colored in two colors. Some points $f(i)$ are such that $f(i) = 2^s$ (described as RED) while some other points j are such that $f(j) = 3^t$ (described as BLUE). By the properties of the Fano plane, for every such coloring in two colors there is a line such that the 3 points on it are colored in the same color, and there cannot exist two lines colored in opposite colors.

Theorem 1

There is a deterministic QIM M that learns the class $\mathcal{U}_7$ with 7 queries.
There is no deterministic QIM M that learns the class $\mathcal{U}_7$ with 6 queries.

Results IV

Theorem 2

There is a nondeterministic QIM M learning $\mathfrak{U}_7$ *with 3 queries.*

Results IV

Theorem 2

There is a nondeterministic QIM M learning $\mathfrak{U}_7$ with 3 queries.

Theorem 3

There is no nondeterministic QIM learning $\mathfrak{U}_7$ with 2 queries.

Results IV

Theorem 2

There is a nondeterministic QIM M learning $\mathfrak{U}_7$ with 3 queries.

Theorem 3

There is no nondeterministic QIM learning $\mathfrak{U}_7$ with 2 queries.

Theorem 4

There is a probabilistic QIM M learning $\mathfrak{U}_7$ with probability $\frac{1}{7}$ with 3 queries.

Results V

Theorem 5

For every prime number p, there is a p-ultrametric QIM M learning the class $\mathcal{U}_7$ with p-probability 1 with 2 queries.

Results V

Theorem 5

For every prime number p, there is a p-ultrametric QIM M learning the class $\mathcal{U}_7$ with p-probability 1 with 2 queries.

Proof. The desired QIM M branches its computation path into 7 branches at the root, where each branch corresponds to exactly one line of the Fano plane. We assign to each edge the amplitude $1/7$. At the second level, each of these branches is branched into 3 subbranches each of which is assigned the amplitude $1/3$. So far we have at level three 21 nodes denoted by $v_1, \ldots, v_{21}$ (cf. Figure 2).

Proof I

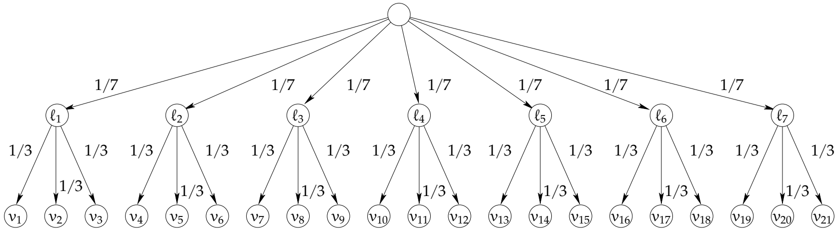

Figure 2: The first three levels of the DAG representing the computation of the QIM M

Proof II

For each of these nodes we formulate two queries. Let v be such that its father node corresponds to the line containing the point i, j, k of the Fano plane, where we order these points such that $i < j < k$. If v is the leftmost node then we query (i, j), if v is the middle node then we query (j, k) and if v is the rightmost node then we query (i, k). Every triple of nodes having the same father share a register, say r_{ijk}. Initially, the register contains the value $\uparrow$ which stands for "no output."

Proof II

For each of these nodes we formulate two queries. Let v be such that its father node corresponds to the line containing the point i, j, k of the Fano plane, where we order these points such that $i < j < k$. If v is the leftmost node then we query (i, j), if v is the middle node then we query (j, k) and if v is the rightmost node then we query (i, k). Every triple of nodes having the same father share a register, say r_{ijk}. Initially, the register contains the value $\uparrow$ which stands for "no output."

The node activated when reached in the computation path sends the following value to r_{ijk}: After having received the answer to its queries, e.g., $f(i) = 2^s$ and $f(j) = 3^t$ then it writes 0 in r_{ijk}, and if the values coincide, e.g., $f(i) = 3^t$ and $f(j) = 3^t$, then it writes t in r_{ijk}.

Proof III

Looking at any triple of nodes having a common father at the third level, we note that the following 8 cases may occur as answer: We use again the corresponding colors, where R and B are shortcuts for RED and BLUE, respectively.

Proof III

Looking at any triple of nodes having a common father at the third level, we note that the following 8 cases may occur as answer: We use again the corresponding colors, where R and B are shortcuts for RED and BLUE, respectively.

(i,j)	(j,k)	(i,k)	(i,j)	(j,k)	(i,k)
(B,B)	(B,R)	(B,R)	(R,R)	(R,B)	(R,B)
(B,B)	(B,B)	(B,B)	(R,R)	(R,R)	(R,R)
(B,R)	(R,R)	(B,R)	(R,B)	(B,B)	(R,B)
(B,R)	(R,B)	(B,B)	(R,B)	(B,R)	(R,R)

Proof III

Looking at any triple of nodes having a common father at the third level, we note that the following 8 cases may occur as answer: We use again the corresponding colors, where R and B are shortcuts for RED and BLUE, respectively.

(i, j)	(j, k)	(i, k)	(i, j)	(j, k)	(i, k)
(B, B)	(B, R)	(B, R)	(R, R)	(R, B)	(R, B)
(B, B)	(B, B)	(B, B)	(R, R)	(R, R)	(R, R)
(B, R)	(R, R)	(B, R)	(R, B)	(B, B)	(R, B)
(B, R)	(R, B)	(B, B)	(R, B)	(B, R)	(R, R)

Thus, we need for each node at the third level 8 outgoing edges as the table above shows. If the edge corresponds to a pair (R, R) or (B, B) then we assign the amplitude $1/2$ and otherwise the amplitude $-1/4$. Note that sum of these amplitudes is again 1.

Proof IV

Finally, we join each triple as shown in table above into one node, e.g., the edges corresponding to (B, B), (B, R), and (B, R) are joined. If the total amplitude of such a node at the third level is different from zero, then the node produces as output the value stored in register r_{ijk}.

Proof IV

Finally, we join each triple as shown in table above into one node, e.g., the edges corresponding to (B, B), (B, R), and (B, R) are joined. If the total amplitude of such a node at the third level is different from zero, then the node produces as output the value stored in register r_{ijk}.

Figure 3 shows the part of the DAG for the queries performed for the first line of the Fano plane, i.e., for the line $(0, 1, 3)$. So this part starts at the nodes v_1, v_2 and v_3 shown in Figure 2. We show the queries asked at each node, i.e., $(0, 1)$ at node v_1, $(1, 3)$ at node v_2, and $(0, 3)$ at node v_3. A blue edge denotes the case that both answers to the queries asked at the corresponding vertex returned a value of f indicating that the related nodes of the first line of the Fano plane are blue. Analogously, the red case is handled. Otherwise, the edge is drawn in black.

Proof V

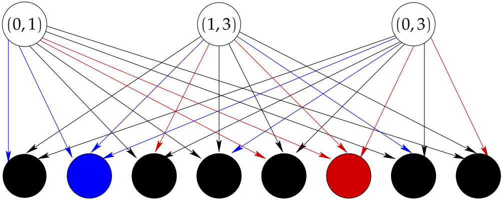

Figure 3: The part of the DAG representing the computation of the QIM M for the line $(0,1,3)$ starting at the nodes of the third level

Proof VI

It remains to show that the QIM M has the desired properties. By construction, at every computation path exactly two queries are asked.

Proof VI

It remains to show that the QIM M has the desired properties. By construction, at every computation path exactly two queries are asked.

Next, by Definition 6 it is obvious that the total amplitude of each node at the second level is $1/21$. Next, we consider any node at the third level. If a triple (B, B), (B, B), and (B, B) is joined then the total amplitude is

$$\frac{1}{21} \cdot \frac{1}{2} + \frac{1}{21} \cdot \frac{1}{2} + \frac{1}{21} \cdot \frac{1}{2} \quad = \quad \frac{1}{2 \cdot 7} \,.$$

The same holds for (R, R), (R, R), and (R, R) (cf. Definition 6). Figure 3 shows the corresponding leaves in blue and red, respectively.

Proof VII

If a triple has a different form than considered above, e.g., (B, B), (B, R), and (B, R) then, again by Definition 6, we have for the total amplitude

$$\frac{1}{21} \cdot \frac{1}{2} - \frac{1}{21} \cdot \frac{1}{4} - \frac{1}{21} \cdot \frac{1}{4} \quad = \quad 0\,.$$

Proof VII

If a triple has a different form than considered above, e.g., (B, B), (B, R), and (B, R) then, again by Definition 6, we have for the total amplitude

$$\frac{1}{21} \cdot \frac{1}{2} - \frac{1}{21} \cdot \frac{1}{4} - \frac{1}{21} \cdot \frac{1}{4} \quad = \quad 0\,.$$

One easily verifies that all remaining total amplitudes are also 0. The corresponding leaves are drawn in black in Figure 3.

Proof VII

If a triple has a different form than considered above, e.g., (B, B), (B, R), and (B, R) then, again by Definition 6, we have for the total amplitude

$$\frac{1}{21} \cdot \frac{1}{2} - \frac{1}{21} \cdot \frac{1}{4} - \frac{1}{21} \cdot \frac{1}{4} \quad = \quad 0 \,.$$

One easily verifies that all remaining total amplitudes are also 0. The corresponding leaves are drawn in black in Figure 3.

Finally, we perform the measurement. For each leaf which has a total amplitude 0 the measurement results in $\|0\|_p = 0$. For the remaining nodes we obtain $\|\frac{1}{2\cdot 7}\|_p$ which is 1 for every prime p such that $p \notin \{2, 7\}$. If $p = 2$ then we have $\|\frac{1}{2\cdot 7}\|_2 = 2$ and for $p = 7$ we directly get $\|\frac{1}{2\cdot 7}\|_7 = 7$.

Proof VIII

By the properties of the Fano plane, there must be at least one line such that all nodes have the same color, and it is not possible to have a line colored in RED and a line colored in BLUE simultaneously. So at least one node has p-probability at least 1, and the result output is correct in accordance with the definition of the class $\mathcal{U}_7$.

Proof VIII

By the properties of the Fano plane, there must be at least one line such that all nodes have the same color, and it is not possible to have a line colored in RED and a line colored in BLUE simultaneously. So at least one node has p-probability at least 1, and the result output is correct in accordance with the definition of the class $\mathcal{U}_7$.

If there are several lines colored in the same color then distinct but correct results may be produced, since any two lines share exactly one point. Thus, the resulting p-probability is always no less than 1. $\blacksquare$

Proof VIII

By the properties of the Fano plane, there must be at least one line such that all nodes have the same color, and it is not possible to have a line colored in RED and a line colored in BLUE simultaneously. So at least one node has p-probability at least 1, and the result output is correct in accordance with the definition of the class $\mathcal{U}_7$.

If there are several lines colored in the same color then distinct but correct results may be produced, since any two lines share exactly one point. Thus, the resulting p-probability is always no less than 1. ∎

The ideas of this paper can be extended to obtain even more spectacular advantages of ultrametric algorithms over nondeterministic ones.

Conclusions

We have studied active learning of classes of recursive functions from value queries. We compared the query complexity of deterministic, nondeterministic, probabilistic, and ultrametric QIM and showed the somehow unexpected result that p-ultrametric QIM can learn classes of recursive function with *significantly fewer queries* than nondeterministic QIM, and probabilistic QIM can do.

Conclusions

We have studied active learning of classes of recursive functions from value queries. We compared the query complexity of deterministic, nondeterministic, probabilistic, and ultrametric QIM and showed the somehow unexpected result that p-ultrametric QIM can learn classes of recursive function with *significantly fewer queries* than nondeterministic QIM, and probabilistic QIM can do.

The situation resembles quantum computation. In quantum computation we also find the usage of amplitudes and of measurements to transform amplitudes into real numbers. Quantum computation is famous for algorithms unimaginable in classical implementation.

Conclusions II

For instance, there exists a quantum query algorithm computing the Boolean function $x_1 \oplus x_2$ asking only one query but computing the binary xor-function function with probability 1. The essence of this algorithm is the quantum parallelism of the computation process (cf. Bernstein and Vazirani (1997)). One computation path queries the value of x_1, the other computation path queries the value of x_2, and the addition of the amplitudes results in the probability 1 for the correct result and in the probability 0 for the wrong result.

Conclusions II

For instance, there exists a quantum query algorithm computing the Boolean function $x_1 \oplus x_2$ asking only one query but computing the binary xor-function function with probability 1. The essence of this algorithm is the quantum parallelism of the computation process (cf. Bernstein and Vazirani (1997)). One computation path queries the value of x_1, the other computation path queries the value of x_2, and the addition of the amplitudes results in the probability 1 for the correct result and in the probability 0 for the wrong result.

Our main result (Theorem 5) is based on the same idea. So, it remains to explore further problems for which ultrametric algorithms can achieve a substantial advantage over nondeterministic algorithm or probabilistic algorithms.

Thank you!

Extensions I

It is proved that there exist finite projective geometries with $n^2 + n + 1$ points and $n^2 + n + 1$ lines such that any two lines have exactly one common point and any two points lie on a common line. This allows us to construct a class $\mathcal{U}_m$ of recursive functions similar to the class $\mathcal{U}_7$ above, where $m =_{df} q^2 + q + 1$ for any prime power q.

The counterpart of the properties of the Fano plane that the must be always a monochromatic line does not hold but this demands only an additional requirement for the function in the class to have a line colored in one color.

Extensions II

Theorem 6

Let q *be any prime power, let* $m = q^2 + q + 1$, *and let* $\mathcal{U}_m \subseteq \mathcal{R}$ *be the corresponding class of recursive functions. Then there is a nondeterministic QIM* M *learning* $\mathcal{U}_m$ *with* $q + 1$ *queries.*

Extensions II

Theorem 6

Let q be any prime power, let $m = q^2 + q + 1$, and let $\mathcal{U}_m \subseteq \mathcal{R}$ be the corresponding class of recursive functions. Then there is a nondeterministic QIM M learning $\mathcal{U}_m$ with $q + 1$ queries.

Theorem 7

Let q be any prime power, let $m = q^2 + q + 1$, and let $\mathcal{U}_m \subseteq \mathcal{R}$ be the corresponding class of recursive functions. Then there is no nondeterministic QIM M learning $\mathcal{U}_m$ with q queries.

Extensions II

Theorem 6

Let q be any prime power, let $m = q^2 + q + 1$, and let $\mathcal{U}_m \subseteq \mathcal{R}$ be the corresponding class of recursive functions. Then there is a nondeterministic QIM M learning $\mathcal{U}_m$ with $q + 1$ queries.

Theorem 7

Let q be any prime power, let $m = q^2 + q + 1$, and let $\mathcal{U}_m \subseteq \mathcal{R}$ be the corresponding class of recursive functions. Then there is no nondeterministic QIM M learning $\mathcal{U}_m$ with q queries.

Theorem 8

For every prime number p, there is a p-ultrametric QIM M learning the class $\mathcal{U}_{13}$ with p-probability 1 with 2 queries.