New coins from old: computing with unknown bias

Elchanan Mossel *
U.C. Berkeley
mossel@stat.berkeley.edu

Yuval Peres †
U.C. Berkeley
peres@stat.berkeley.edu

With an appendix by Christopher Hillar ‡
U.C. Berkeley
chillar@math.berkeley.edu

January 27, 2004

Abstract

Suppose that we are given a function $f : (0,1) \to (0,1)$ and, for some unknown $p \in (0,1)$, a sequence of independent tosses of a p-coin (i.e., a coin with probability p of "heads"). For which functions f is it possible to simulate an $f(p)$-coin? This question was raised by S. Asmussen and J. Propp. A simple simulation scheme for the constant function $f(p) \equiv 1/2$ was described by von Neumann (1951); this scheme can be easily implemented using a finite automaton. We prove that in general, an $f(p)$-coin can be simulated by a finite automaton for all $p \in (0,1)$, if and only if f is a rational function over $\mathbb{Q}$. We also show that if an $f(p)$-coin can be simulated by a pushdown automaton, then f is an algebraic function over $\mathbb{Q}$; however, pushdown automata can simulate $f(p)$-coins for certain non-rational functions such as $f(p) = \sqrt{p}$. These results complement the work of Keane and O'Brien (1994), who determined the functions f for which an $f(p)$-coin can be simulated when there are no computational restrictions on the simulation scheme.

1 Introduction

Fifty years ago, von Neumann [18] suggested a method to generate unbiased random bits from a sequence of i.i.d. biased bits. This method can be easily implemented using a finite automaton.

In this paper we study the following generalization. Let $\mathcal{D} \subset (0,1)$. Suppose that we are given a function $f : \mathcal{D} \to (0,1)$ and, for some unknown $p \in \mathcal{D}$, a sequence of independent tosses of a p-coin (i.e., $\{0,1\}$ valued random variables with mean p). For which functions f is it then possible to simulate an $f(p)$-coin?

The allowed simulation schemes apply a stopping rule to independent tosses of a p-coin, and then determine a $\{0,1\}$-valued variable with mean $f(p)$ as a function of the stopped sequence. We

*Supported by a Miller Fellowship
†Supported in part by NSF Grant DMS-0104073 and by a Miller Professorship
‡This work is supported under a National Science Foundation Graduate Research Fellowship.

emphasize that the scheme cannot depend on p. We are especially interested in simulation schemes that can be implemented by an automaton that receives the p-coin tosses as inputs, and outputs an $f(p)$-coin; see §1.2 for more formal definitions. A special case of this question was raised in 1991 by S. Asmussen (see [10]). We learned of the general problem from J. Propp (personal communication) who emphasized its computational aspects. The problem was considered in the context of Markov chain simulation by Glynn and Henderson [8].

Our main results follow; the second depends on the result of the appendix.

Theorem 1.1. *Let $\mathcal{D} \subset (0,1)$ and $f : \mathcal{D} \to (0,1)$. Then an $f(p)$-coin for $p \in \mathcal{D}$ can be simulated using a finite automaton from independent tosses of a p-coin, if and only if f is the restriction to $\mathcal{D}$ of a rational function F over $\mathbb{Q}$, such that $0 < F(x) < 1$ for $0 < x < 1$.*

Theorem 1.2. *Let $f : (0,1) \to (0,1)$. If an $f(p)$-coin can be simulated from tosses of a p-coin by a pushdown automaton for all $p \in (0,1)$, then f is an algebraic function over $\mathbb{Q}$; i.e., there exists a non-zero polynomial $P \in \mathbb{Q}[X,Y]$ such that $P(f(p),p) = 0$ for all $p \in [0,1]$.*

We don't know if every algebraic $f : (0,1) \to (0,1)$ can be simulated by a pushdown automaton.

Our results complement the work of Keane and O'Brien (1994), who considered the simulation problem without any computational restrictions. They showed that the functions $f : \mathcal{D} \to (0,1)$ for which an $f(p)$-coin can be simulated (in principle) are precisely the constants, and all continuous functions on $\mathcal{D}$ that satisfy $\min(f(x), 1 - f(x)) \geq \min(x, 1 - x)^n$ for some $n \geq 1$ and all $x \in \mathcal{D}$.

1.1 Examples

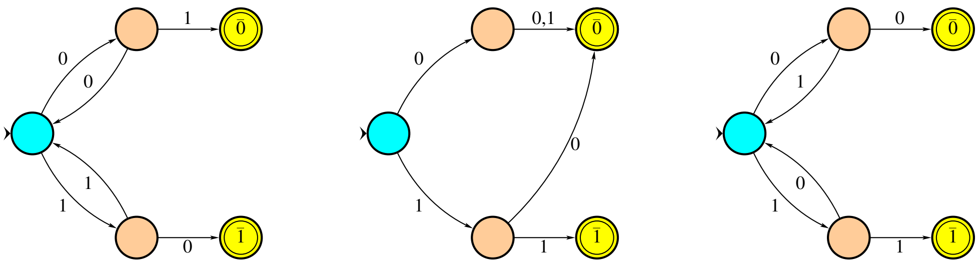


Figure 1: Simulating an unbiased coin, p^2 and $\frac{p^2}{p^2+(1-p)^2}$

- $f(p) = 1/2$. This is achievable by Von Neumann's trick: toss the p coin twice and let x and y be the outcome. If $xy = 01$ output 0, if $xy = 10$ declare 1; otherwise, do not declare and toss again. This is the leftmost automaton in Figure 1.
- $f(p) = p^2$. Toss the p-coin twice. If it's 11 declare 1, otherwise declare 0. This is the automaton in the middle.
- $f(p) = \frac{p^2}{p^2+(1-p)^2}$. Toss the p-coin twice until you get 00 or 11. In the first case, declare 0, in the second declare 1. This is the automaton on the right.

- $f(p) = \sqrt{p}$. Theorem 2.2 implies that this function cannot be simulated by a finite automaton. In Section 3 we construct a pushdown automaton which simulates f.

- Our main result implies that there is no infinite set $\mathcal{D}$ such that a $2p$-coin can be simulated by a finite automaton for all $p \in \mathcal{D}$ from tosses of a p-coin.

1.2 Formal definitions

Denote by $\{0,1\}^*$ the set of all finite binary strings, and call any subset of $\{0,1\}^*$ a *language*. A language L has the *prefix property* if for any $u, v \in L$ that are distinct, u is not a prefix of v.

For any binary string w, write $\mathbf{P}_p[w] = p^{n_1(w)}(1-p)^{n_0(w)}$, where $n_i(w)$ is the number of i's in w and let $\mathbf{P}_p(L) = \sum_{w \in L} \mathbf{P}_p(w)$ for any language L.

Definition 1.3. Let $f : \mathcal{D} \to [0,1]$ where $\mathcal{D} \subset (0,1)$. A *simulation of* f is a pair of disjoint languages $L_0, L_1 \subset \{0,1\}^*$ such that $L_0 \cup L_1$ has the prefix property and for all $p \in \mathcal{D}$ we have $\mathbf{P}_p[L_1] = f(p)$ and $\mathbf{P}_p[L_0 \cup L_1] = 1$ and $\mathbf{P}_p[L_1] = f(p)$. A *simulation of* f *by a finite automaton*, is a simulation (L_0, L_1) of f such that there exists a finite automaton which outputs 0 for the language L_0 and outputs 1 for the language L_1. An analogous definition applied to simulation by a pushdown automaton or a Turing machine.

It is easy to adapt the proofs of [10] in order to show that if $f : (0,1) \to (0,1)$ is computable [7, 13] and polynomially bounded from 0 and 1 at 0 and 1, then it possible to simulate the function $f(p)$ via a Turing machine.

Definition 1.4. *A finite automaton is defined by*

- *A set of states S, an initial state $s_0 \in S$ and an alphabet which we fix to be $\{0,1\}$.*

- *A transition function $\delta : S \times \{0,1\} \to S$, where $\delta(s,b)$ is the state the automaton moves to from state s if the current input symbol is b. For a string $w = w_1 \dots w_k \in \{0,1\}^*$, define $\delta(s,w)$ inductively by $\delta(s, w_1 \dots w_k) = \delta(\delta(s,w_1), w_2 \dots w_k)$.*

- *Two disjoint sets of final states S_0 and S_1, where the automaton stops. (Formally, $\delta(s,b) = s$ for all $s \in S_0 \cup S_1$ and $b \in \{0,1\}$). If the automaton stopped at S_0, then the output is 0; if it stopped in S_1, then the output is 1.*

Note that the input here is unbounded (cf. [6]).

Letting L_i be the strings in $\{0,1\}^*$ for which the automaton stops at S_i, the automaton will define a simulation of a function $f : (0,1) \to (0,1)$, if $\mathbf{P}_p[L_0 \cup L_1] = 1$. Note that L_0 and L_1 are both regular languages, and $L_0 \cup L_1$ has the prefix property. Moreover, for every pair of disjoint regular languages L_0 and L_1 with the prefix property, using a standard product construction, it easy to write down an automaton as in Definition 1.4 which outputs 0 for L_0 and 1 for L_1.

The definition below is a variation of the standard definition [6] of Pushdown automata.

Definition 1.5. *A* Pushdown automaton *over the alphabet* $\{0,1\}$ *is defined by*

- *A set of states S with a start state $s_0 \in Q$.*
- *A stack alphabet Λ. and a stack $T \in \Lambda^*$ which is initialized as the non-empty string τ.*
- *A transition function $(\delta, \eta) : S \times \{0,1\} \times \Lambda \to S \times \Lambda^*$. The first coordinate, $\delta(s, b, \tau)$, is the new state of the automaton given that it was in state s, the current input symbol is b and that the symbol at the top of the stack is τ. After this transition the symbol at the top of the stack is replaced by the string in the second coordinate of $\eta(s, b, \tau)$.*
- *Two disjoint sets of final states S_0 and S_1. The automaton stops when the stack is empty. If the automaton stops at a state of S_i, then the output is i.*

Note that if L_i is the language where the automaton output i, then L_0 and L_1 are context free languages, and $L_0 \cup L_1$ has the prefix property.

2 Rationality and finite automata

A sub-family of the coins which can be simulated via finite-automata are those which can be simulated via blocks. Von Neumann's trick consists of reading 2 bit blocks until 01 or 10 are reached, and then deciding 0 for 01 and 1 for 10. Block simulation is a generalization of this procedure defined as follows

Definition 2.1. *A* block simulation *of f, is a simulation of f of the following form. Let A_0 and A_1 be disjoint subsets of $\{0,1\}^k$, and $A' = \{0,1\}^k \setminus (A_0 \cup A_1)$. The simulation procedure has $L_0 = (\sum_{w \in A'} w)^*(\sum_{w \in A_0} w)$ and $L_1 = (\sum_{w \in A'} w)^*(\sum_{w \in A_1} w)$. In other words, the procedure reads a k bit string w. If $w \in A_0$, the procedure outputs 0, if $w \in A_1$ the procedure outputs 1; otherwise the procedure discards w and reads a new k bit string.*

Below we prove the following theorem which immediately implies Theorem 1.1.

Theorem 2.2. *Let $\mathcal{D} \subset (0,1)$. For $f : \mathcal{D} \to (0,1)$ the following are equivalent.*

I f can be block simulated.

II f can be simulated via a finite automaton.

III f is the restriction to $\mathcal{D}$ of a a rational function F over $\mathbb{Q}$ such that $0 < F(p) < 1$ for all $p \in (0,1)$.

Note that $I \Rightarrow II$ is trivial.

2.1 Finite automaton $\Rightarrow$ Rationality

Proposition 2.3. *Let $\mathcal{D} \subset (0,1)$. If a finite automaton Σ simulates $f : \mathcal{D} \to (0,1)$, then f is the restriction to $\mathcal{D}$ of a rational function F over $\mathbb{Q}$ such that $0 < F(p) < 1$ for all $p \in (0,1)$. Moreover if Σ has n states, then $F(p) = g(p)/h(p)$, where $g(p), h(p) \in \mathbb{Z}[p]$ and g and h are of degree at most n.*

The proof applies the maximum principle for harmonic functions on directed graphs.

Lemma 2.4. *Let Σ be a finite automaton and R a set of states such that for all $s \in S$, there exists $w \in \{0,1\}^*$, such that $\delta(s,w) \in R$. Let $0 < p < 1$ and $f : S \to \mathbb{R}$ be harmonic, so for all s it holds that $f(s) = pf(\delta(s,1)) + (1-p)f(\delta(s,0))$. Then f achieves its maximum and minimum in R. Moreover, given the values of f in R, the values of f in S are uniquely determined.*

Proof. Note first that the last assertion follows from the first one, as if f_1 and f_2 are two harmonic functions which have the same value on R, then $f_1 - f_2$ is harmonic and has the value 0 on R, which implies by the first assertion that $\max f_1 - f_2 = \min f_1 - f_2 = 0$, or $f_1 = f_2$.

In order to prove the first assertion, let $m = \max f$. Note that if $f(s) = m$, then $f(\delta(s,0)) = f(\delta(s,1)) = m$. Letting $w \in \{0,1\}^*$ be such that $\delta(s,w) \in R$, we obtain $f(\delta(s,w)) = m$ as needed. ■

Proof of Proposition 2.3: Suppose that f can is simulated by a finite automaton Σ. Let S' be the set of states s such that there exists $w \in \{0,1\}^*$ with $\delta(s_0,w) = s$. Clearly, we may remove from the automaton all the states not in S' (redefining δ by restriction) and still obtain a finite automaton which simulates $f(p)$. By the assumption that $\mathbf{P}_p[L_0 \cup L_1] = 1$, it follows that for all $s \in S'$, there exists $w \in \{0,1\}^*$ such that $\delta(s,w) \in S_0 \cup S_1$. From now on we assume that $S = S'$.

Note that since for all $s \in S$, there exists $w \in \{0,1\}^*$ such that $\delta(s,w) \in S_0 \cup S_1$, it follows that $\mathbf{P}_p[L_0 \cup L_1] = 1$ for all $0 < p < 1$. Therefore the function f is the restriction of the function $F(p) = \mathbf{P}_p[L_1]$ to $\mathcal{D}$.

Suppose that $F(s;p)$ satisfies $F(s;p) = 0$ for $s \in S_0$, and $F(s;p) = 1$ for $s \in S_1$. For all other s, assume that

$$F(s;p) = pF(\delta(s,1);p) + (1-p)F(\delta(s,0);p)\,.$$

By Lemma 2.4, the function $F(s;p)$ is uniquely determined by these equations. This implies that $F(s_0;p) = F(p)$. Since $F(p)$ is uniquely determined by a collection of linear equation with coefficients in $\mathbb{Z}[p]$, it follows by Cramer's rule that $F(p)$ may be written as the ratio of two determinants in $\mathbb{Z}[p]$, and therefore $F(p) = g(p)/h(p) \in \mathbb{Q}(p)$, where the degrees of g and h are at most the number of states of the automaton, as needed. ■

2.2 Block Simulation

Proposition 2.5. *f can be simulated using a block procedure if and only if f can be written as $D(p)/E(p)$ where*

$$D(p) = \textstyle\sum_{i=0}^{k} d_i p^i (1-p)^{k-i}, \quad E(p) = \sum_{i=0}^{k} e_i p^i (1-p)^{k-i}, \tag{1}$$

and for all i, the coefficients d_i and e_i are integers such that $0 \le d_i \le e_i$.

Proof. Suppose that f is block simulated. For a string $w \in \{0,1\}^*$, we write $n_1(w)$ for the number of 1s in w. Then

$$f(p) = \mathbf{P}_p[L_1] = \frac{\sum_{w \in A_1} \mathbf{P}_p[w]}{\sum_{w \in A_0 \cup A_1} \mathbf{P}_p[w]} = \frac{\sum_{i=0}^{k} d_i p^i (1-p)^{k-i}}{\sum_{i=0}^{k} e_i p^i (1-p)^{k-i}},$$

where $d_i = \#\{w \in A_1 : n_1(w) = i\}$, and $e_i = \#\{w \in A_1 \cup A_0 : n_1(w) = i\}$, satisfy that $0 \le d_i \le e_i$, as needed.

For the other direction, suppose that $f(p) = D(p)/E(p)$, where D and E satisfy (1). Let r be a number such that $e_i \le \binom{k}{i}\binom{2r}{r}$ for all i. For each i, fix a bijection B_i from

$$\{w \in \{0,1\}^k : n_1(w) = i\} \times \{v \in \{0,1\}^{2r} : n_1(v) = r\}$$

to $\{1, \ldots, \binom{k}{i}\binom{2r}{r}\}$.

The sets A_0 and A_1 are subsets of $\{0,1\}^k \times \{0,1\}^{2r}$, defined as follows. A_1 is defined as

$$\cup_{i=0}^{k}\{(v,w) : n_1(v) = i, n_1(w) = r, \text{ and } B_i(v,w) \le d_i\}$$

and A_0 as

$$\cup_{i=0}^{k}\{(v,w) : n_1(v) = i, n_1(w) = r, \text{ and } d_i < B_i(v,w) \le e_i\}.$$

So $\mathbf{P}_p[A_1] = p^r(1-p)^r \sum_{i=0}^{k} d_i p^i (1-p)^{k-i}$ and $\mathbf{P}_p[A_0 \cup A_1] = p^r(1-p)^r \sum_{i=0}^{k} e_i p^i (1-p)^{k-i}$. Therefore $\mathbf{P}[L_1] = \frac{\mathbf{P}_p[A_1]}{\mathbf{P}_p[A_0 \cup A_1]} = f(p)$ as needed. ∎

2.3 Rationality ⇒ Finite automaton

In this section we prove Theorem 2.2. The proof is based on a beautiful theorem by Pólya [11] (see [9], 57–59). We let Δ^s denote the open s-simplex of probability distributions,

$$\Delta^s = \{p \in (0,1)^{s+1} : \sum_{i=1}^{s+1} p_i = 1\}.$$

Theorem 2.6 (Pólya [11]). *Let $f : \Delta^{s-1} \to \mathbb{R}$ be homogeneous and positive polynomial in the variables $p_1, \ldots, p_s$. Then for all sufficiently large n, all the coefficients of $(p_1 + \ldots + p_s)^n f(p_1, \ldots, p_s)$ are positive.*

Lemma 2.7. *Let $f : (0,1) \to (0,1)$ be a rational function. Then there exist polynomials d and e*

$$d(p) = \sum_{i=0}^{k} d_i p^i (1-p)^{k-i}, \quad e(p) = \sum_{i=0}^{k} e_i p^i (1-p)^{k-i}, \tag{2}$$

where for all i, the coefficients d_i and e_i are integers such that $0 \le d_i \le e_i$, and $f(p) = d(p)/e(p)$.

Proof. As $f(p)$ is a rational function it may be written in the form $\overline{D}(p)/\overline{E}(p)$, where $\overline{D}(p) \in \mathbb{Z}[p]$ and $\overline{E}(p) \in \mathbb{Z}[p]$ are relatively prime polynomials. Since $0 < f(p)$ for all $0 < p < 1$, it follows that $\overline{D}(p)$ and $\overline{E}(p)$ do not change sign in the interval $(0,1)$. Without loss of generality we assume that $\overline{D}(p) > 0$ and $\overline{E}(p) > 0$ for all $p \in (0,1)$. Note furthermore that if $\overline{D}(p) = \sum_{i=0}^{k} a_i p^i$ and $\overline{E}(p) = \sum_{i=0}^{k} b_i p^i$, then we may define homogeneous polynomials $D(p,q)$ and $E(p,q)$ of degree k, by letting $D(p,q) = \sum_{i=0}^{k} a_i p^i (p+q)^{k-i}$, and $E(p,q) = \sum_{i=0}^{k} b_i p^i (p+q)^{k-i}$. Note that $\overline{D}(p) = D(p, 1-p)$ and $\overline{E}(p) = E(p, 1-p)$. Let us rewrite $D(p,q) = \sum_{i=0}^{k} d_i p^i q^{k-i}$ and $E(p,q) = \sum_{i=0}^{k} e_i p^i q^{k-i}$. The polynomials $D(p,q), E(p,q)$ and $E(p,q) - D(p,q)$ are all positive homogeneous polynomials. Therefore by Theorem 2.6, if follows that there exists an n such that letting $d(p,q) = (p+q)^n D(p,q)$ and $e(p,q) = (p+q)^n E(p,q)$, the polynomials d,e and $e-d$ all have positive coefficients as polynomials in p and q. Writing $f(p) = d(p, 1-p)/e(p, 1-p)$ we obtain the required result. ∎

Proof of Theorem 2.2: The implication $I \Rightarrow II$ is trivial, the implication $II \Rightarrow III$ follows from Proposition 2.3, while $III \Rightarrow I$ follows from Lemma 2.7 together with Proposition 2.5.

2.4 Extensions to dice and other k-sided coins

In this subsection we discuss generalizations of the problem to k-sided coins, such as dice. A *simulation* of $f = (f_1, \ldots, f_t) : \Delta^s \to \Delta^t$, is a collection of t disjoint languages property $(L_1, \ldots, L_t)$ over the alphabet $\Sigma = \{1, \ldots, s\}$, such that $\cup_{i=1}^t L_i$ has the prefix property and $\mathbf{P}_p[L_i] = f_i(p)$ for all $p \in \Delta^s$. The definition of simulation via finite/pushdown automata and Turing machines naturally extend to this setting. The continuity results of [10] extend to the more general setting as well.

Proposition 2.8. *If a finite automaton simulates $f : \Delta^s \to \Delta^t$, then f is a rational function over $\mathbb{Z}$ (i.e. $f_i(p)$ is a rational function over $\mathbb{Z}$ for all $1 \le i \le t+1$).*

Proof. Repeat the proof of Proposition 2.3 for each of the f_i's. ■.

Repeating the proof of Theorem 2.2, we see that

Theorem 2.9. *Any rational function $f : \Delta^s \to \Delta^1$ can be simulated via blocks.*

From which we conclude that

Corollary 2.10. *Any rational function $f : \Delta^s \to \Delta^t$ can be simulated via blocks.*

Proof. The proof is by induction on t. The case $t = 1$ is covered by Theorem 2.9. Suppose $t > 1$, and let $(f_1, \ldots, f_{t+1}) : \Delta^s \to \Delta^t$ be a rational function. By Theorem 2.9, there exists a block simulation $(A_1, A_2) \subset (\Sigma^k)^2$ for $(f_1, 1 - f_1)$, and by the induction hypothesis there exists a block simulation $(B_1, \ldots, B_t) \subset (\Sigma^r)^t$ for $(f_2/(1-f_1), \ldots, f_{t+1}/(1-f_1))$. Taking

$$(A_1 \times \Sigma^r, A_2 \times B_1, A_2 \times B_2, \ldots, A_2 \times B_t) \subset (\Sigma^{r+k})^{t+1},$$

we obtain a block simulation for $(f_1, \ldots, f_{t+1})$ as needed. □

3 Pushdown automata

We now prove Theorem 1.2. We begin by showing that if f is simulated by a pushdown automaton, then f is the unique solution of a set of polynomial equations. We then invoke the results of the appendix to deduce that f is an algebraic function.

3.1 Pushdown automata and algebraic functions

The Chomsky-Schützenberger theory [4] implies that if L_0 and L_1 are languages which are generated by unambiguous grammars and $f(p) = \mathbf{P}_p[L_1]$, then $f(p)$ is an algebraic function. However, it does not imply that coins tossed via pushdown automata are algebraic, as many of the context free languages are inherently ambiguous and for such languages, the non-commutative power series $P(L) = \sum_{w \in \{0,1\}^*} 1_{\{w \in L\}} w$ is not algebraic. (see e.g. [6]; as Larry Ruzzo kindly noted, there are also context free languages with the prefix property that are inherently ambiguous). In this subsection we aim to prove algebraic properties of f even when L_0 and L_1 are inherently ambiguous. Thus, while Proposition 2.3 could be obtained by projecting the Chomsky-Schützenberger results from the non-commutative setting to the commutative setting, an analogous result for pushdown automata cannot be obtained in a similar way.

Suppose Σ is a pushdown automaton which simulates a function f. Call $(b,s) \in \Lambda \times S$ *good*, if when the automaton is at state s and the stack is bw (where b is at the top), then with probability 1 at some point the stack will be w. Call (b,s) *bad* otherwise. By the assumption that $\mathbf{P}_p[L_0 \cup L_1] = 1$, it follows that starting at (s_0, τ) it is impossible for the automaton to reach a state s with bw at the top of the stack, where (b,s) is bad. Thus we can redefine all transitions $(b,s) \to (b'w', s')$, where (b', s') is bad, in an arbitrary manner, and still obtain $\mathbf{P}_p[L_0 \cup L_1] = 1$. Therefore, without loss of generality we may assume that all $(b,s) \in \Lambda \times S$ are good.

Let $\alpha(p; b, s, s')$, $\alpha : [0,1] \times \Lambda \times S \times S \to [0,1]$ be defined as follows. For $w \in \Lambda^*$, let $\alpha(p; b, s, s')$ be the probability that given that currently the automaton is at state s and has in its stack bw (where b is at the top), at the first time that the content of the stack will be w, it will be at state s'. It is easily seen that $\alpha(p; b, s, s')$ is well defined (does not depend on w). Moreover, by the assumption that all (b,s) are good, it follows that $\sum_{s' \in S_0 \cup S_1} \alpha(p; b, s, s') = 1$.

We extend the definition of α to $\tilde{\alpha}(p; u, s, s')$, $\tilde{\alpha} : \Lambda^* \times S \times S \to [0,1]$, where $\tilde{\alpha}(p; u, s, s')$ is the probability that given that currently the automaton is at state s and has in its stack uw (where u is above w), at the first time that the content of the stack will be w, it will be at state s'. Note that if $w = w_1 \dots w_r$, then

$$\tilde{\alpha}(p; w, s, s') = \sum \left\{ \prod_{i=1}^{r} \alpha(p; w_i, s_i, s_{i+1}) : (s_1, s_2, \dots, s_r, s_{r+1}) \in \{s\} \times S^{r-1} \times \{s'\} \right\}, \qquad (3)$$

and if ϵ denotes the empty word, then $\tilde{\alpha}(p; \epsilon, s, s') = 1_{\{s=s'\}}$ for all $s, s' \in Q$. Note that if τ is the initial word at the stack, then $\mathbf{P}_p[L_1] = \sum_{s' \in S_1} \tilde{\alpha}(p; \tau, s_0, s')$.

Therefore if we could prove algebraic properties of the functions $\alpha(p; b, s, s')$, we will deduce algebraic properties of f.

Claim 3.1. *For all $0 < p < 1$, $\tilde{\alpha}(p; \cdot, \cdot, \cdot) : \Lambda^* \times S \times S \to \mathbb{R}$ is the unique bounded solution of the equations*

$$\begin{aligned} \forall w \in \Lambda^*, \forall b \in \Lambda, \forall s, s' \in Q : \tilde{\alpha}(p; bw, s, s') &= p\tilde{\alpha}\left(p; \delta(s,1,b)w, \eta(s,1,b), s'\right) \\ &+ (1-p)\tilde{\alpha}\left(p; \delta(s,0,b)w, \eta(s,0,b), s'\right), \end{aligned} \qquad (4)$$

$$\forall s, s' \in Q : \tilde{\alpha}(p; \epsilon, s, s') = 1_{\{s=s'\}} \text{ (ϵ is the empty word).} \qquad (5)$$

Proof. By linearity, it suffices to prove that the zero function is the only bounded solution to (4) with the boundary conditions

$$\forall s, s' \in Q : \tilde{\alpha}(p; \epsilon, s, s') = 0. \qquad (6)$$

Fix $(w, s) \in \Lambda^* \times Q$. We will show that $\tilde{\alpha}(p; w, s, s') = 0$. Consider the random walk $(W_t, S_t)_{w,s}$ defined on the graph $\Lambda^* \times Q$, where $(W_0, S_0)_{w,s} = (w, s)$. Given $W_t = BU$, where $B \in \Lambda$, the conditional probabilities for (W_{t+1}, S_{t+1}) are given by

$$(W_{t+1}, S_{t+1}) = \begin{cases} (\delta(S_t, 1, B)U, \eta(S_t, 1, B)) & \text{with probability } p, \\ (\delta(S_t, 0, B)U, \eta(S_t, 0, B)) & \text{with probability } 1-p. \end{cases} \tag{7}$$

If W_t is the empty word, then we let $(W_{t+1}, S_{t+1}) = (W_t, S_t)$. By definition, for all $s' \in Q$, the process $\tilde{\alpha}(p; W_t, S_t, s')$ is a bounded martingale. The assumption that the pushdown automaton stops a.s. implies by (6) that the martingale converges to 0 a.s. (and therefore in L_1). We therefore conclude that $\tilde{\alpha}(p; W_t, S_t, s')$ is identically 0 as needed. ■

Recall that given that the current state is s, and the top of the stack is b, with probability p the automaton will move to state $s_1 = \delta(s, 1, b)$, and instead of b, the top of the stack will contain $\eta(s, 1, b) = c_1 \dots c_r$; with probability $1-p$ the automaton will move to state $\bar{s}_1 = \delta(s, 0, b)$, and instead of b, the top of the stack will contain $\eta(s, 0, b) = \bar{c}_1 \dots \bar{c}_{\bar{r}}$.

We can therefore write

$$\begin{aligned} \alpha(p; b, s, s') &= p\left(1_{\{r=0\}}1_{\{s_1=s'\}} + 1_{\{r>0\}}\tilde{\alpha}(p; c_1 \dots c_r, s_1, s')\right) \\ &+ (1-p)\left(1_{\{\bar{r}=0\}}1_{\{\bar{s}_1=s'\}} + 1_{\{\bar{r}>0\}}\tilde{\alpha}(p; \bar{c}_1 \dots \bar{c}_{\bar{r}}, s_1, s')\right), \end{aligned} \tag{8}$$

Or

$$\begin{aligned} \alpha(p; b, s, s') &= p\left(1_{\{r=0\}}1_{\{s_1=s'\}} + 1_{\{r>0\}}\sum\prod_{i=1}^{r}\alpha(p; c_i, s_i, s_{i+1})\right) \\ &+ (1-p)\left(1_{\{\bar{r}=0\}}1_{\{\bar{s}_1=s'\}} + 1_{\{\bar{r}>0\}}\sum\prod_{i=1}^{\bar{r}}\alpha(p; \bar{c}_i, \bar{s}_i, \bar{s}_{i+1})\right). \end{aligned} \tag{9}$$

where the first sum is taken over all $(s_1, \dots, s_{r+1}) \in \{s_1\} \times S^{r-1} \times \{s'\}$ (similarly for the second sum).

Note that (9) defines a set of algebraic equations in p and $\{\alpha(p; b, s, s')\}_{b,s,s'}$.

Claim 3.2. *For all $0 < p < 1$, there is a unique positive solution $\alpha(p; \cdot, \cdot, \cdot) : [0,1] \times \Lambda \times S \times S \to \mathbb{R}$, to equations (9) and*

$$\forall b \in \Lambda,\ \forall s \in Q : \sum_{s' \in Q} \alpha(p; b, s, s') = 1. \tag{10}$$

Proof. In order to prove the claim it suffices to show that each positive solution to (9) and (10) defines a positive bounded solution to (4) and (5) via (3)

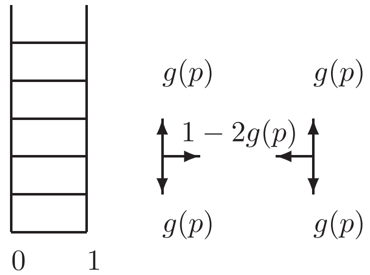
It is immediate to see that (9) implies (4). Moreover, (10) implies by (3) that for all $w \in \Lambda^*$ and $s \in Q$, it holds that $\sum_{s' \in Q} \tilde{\alpha}(p; w, s, s') = 1$. So $\tilde{\alpha}$ is a bounded function as needed. ■

Proof of Theorem 1.2: The result follows immediately from Claim 3.2 by Theorem A.1. ■

3.2 Pushdown automata which simulate non-rational functions

In this subsection we construct a pushdown automaton which simulates a non-rational function. Let $g : (0,1) \to (0, 1/2)$ be a rational function. We'll construct a pushdown automata which simulates the function $\gamma(p) = \frac{1-\sqrt{1-2g(p)}}{4g(p)}$.

Taking $g(p) = (1-p)/2$, we obtain $\gamma(p) = (1-\sqrt{p})/(1-p)$ - thus using a product construction, it is easy to construct a pushdown automaton simulating the function $f(p) = \sqrt{p}$.



Consider a random walk on the ladder graph $\mathbb{N} \times \{0,1\}$ where an edge $((x,y),(x',y'))$ is present if $x = x'$ and $|y-y'| = 1$, or $|x-x'| = 1$ and $y = y'$. The random walk moves to the left (right) with probability $1-2g(p)$, and up (down) with probability $g(p)$. Let γ be the probability that starting at $(0,1)$ the first hitting point of the random walk at level 0 ($\{(0,0),(1,0)\}$) is $(0,0)$. It is easy to see that

$$\gamma = g + (1-2g)(1-\gamma) + g(\gamma^2 + (1-\gamma)^2) = 1 - \gamma + 2g\gamma^2 \implies \gamma(p) = \frac{1-\sqrt{1-2g(p)}}{2g(p)}. \tag{11}$$

It is easy to simulate the random walk with a pushdown automaton. The stack alphabet is $\{x\}$ where x^n correspond to level n of the ladder – the initial word at the stack is x.

Assume first that the input alphabet is $\{0,1,2\}$ where the probability that letter 1 appears is $1-2g(p)$, and the probability that letter 0 or 2 appears is $g(p)$.

Let s_0 and s_1 be two states of the automaton corresponding the left ($\{0\}\times\mathbb{N}$) and right ($\{1\}\times\mathbb{N}$) of the ladder. Reading the symbol 1 will correspond to a transition from s_0 to s_1 or vice-versa without changing the content of the stack. Reading the symbol 2 at state s_i will result at staying at state s_i and pushing an x to the stack. Reading the symbol 0 at state s_i with x at the top of the stack, will result at staying at state s_i while popping x from the stack. In this way it is possible to simulate the random walk given an infinite sequence of $\{0,1,2\}$ symbols with $(g(p), 1-2g(p), g(p))$ bias - and therefore toss a coin with bias (11).

In the general case where we are given an infinite sequence of $(p, 1-p)$ bits, we use block constructions of Section 2 in order to generate a sequence of $(g(p), 1-2g(p), g(p))$ $\{0,1,2\}$ variables together with the above construction in order to obtain the required result.

Remark 3.3. *Similarly one may construct a pushdown automaton associated with a random walk on the ladder where the probabilities of going* (*up,left-right,down*) *are given by* $(g(p), 1-g(p)-h(p), h(p))$. *Note however that in this case* $\mathbf{P}_p[L_0 \cup L_1] = 1$ **iff** *the random walk is recurrent* **iff** $h(p) \geq g(p)$. *Thus, unlike finite automata, there exist pushdown automata which define a valid simulation only for a proper subset of the interval* $(0,1)$.

4 Exact sampling and unsolved problems.

The theory of exact sampling (see e.g. [2, 1, 15, 16]) deals with simulating a complicated probability measure using a simple one. In our setting, both probability measures of interest (the p coin and the $f(p)$ coin) are simple; the difficulty is that p is unknown. The following examples illustrate situations where one might want to simulate an $f(p)$ coin from tosses of a p coin.

- Suppose that some physical process produces percolation configuration on a grid Γ_1, where the probability of an open edge is p. We are interested in performing percolation on a grid Γ_2, where we want that the probability that an edge e is open in Γ_2 to equal the probability that two vertices at distance 2 in Γ_1 are connected by a path of length 2. Our results allow to use samples of the configuration Γ_1 in order to produce samples for the process on Γ_2.

- Let $\{x_i\}_{i\geq 1}$ be i.i.d. bits with unknown mean α. Suppose that we are given the products $\{y_i\}_{i\geq 1}$, where $y_i = x_{2i}x_{2i-1}$, and we want to simulate (one or more) i.i.d. bits $\{z_j\}$ with mean α. This can be done using the pushdown automaton in Subsection 3.2 that simulates the function $\sqrt{p}$.

We conclude with some unsolved problems.

Problem 4.1. *Let $f : (0,1) \to (0,1)$ be a rational function. What is the smallest size of an automaton that simulates $f(p)$? Is there an efficient algorithm for finding this automaton?*

The size of the automaton might depend on analytic and diophantine properties of f (see [14]).

Problem 4.2. *Let $f : (0,1) \to (0,1)$ be an algebraic function. Can f be simulated by a pushdown automaton?*

Acknowledgment: We thank Jim Propp for suggesting the problem. We are grateful to Omer Angel, Paul Beame and Larry Ruzzo for helpful remarks, and to Russ Lyons and Sergey Fomin for references. Most of this work was done while the first author was a postdoctoral researcher at Microsoft Research, and the second author was visiting there.

References

[1] D. J. Aldous (1990). A random walk construction of uniform spanning trees and uniform labeled trees. *SIAM Journal on Discrete Mathematics*, **3(4)**, 450–465.

[2] A. Broder (1989). Generating random spanning trees. In *30th Annual Symposium on Foundations of Computer Science*, 442–447.

[3] D. Cox, J. Little and D. O'Shea, Donal (1997). *Ideals, varieties, and algorithms. An introduction to computational algebraic geometry and commutative algebra.* Second edition. Undergraduate Texts in Mathematics. Springer-Verlag, New York, 1997.

[4] N. Chomsky and M. P. Schützenberger (1963). The algebraic theory of context-free languages. In *Computer programming and formal systems*, 118–161, North-Holland, Amsterdam.

[5] P. Elias (1972). The efficient construction of unbiased random sequence, *Ann. Math. Stat.* **43**, 865–870.

[6] J. E. Hopcroft and J. D. Ullman (1979). *Introduction to automata theory, languages, and computation.* Addison-Wesley Series in Computer Science. Addison-Wesley Publishing Co., Reading, Mass.

[7] A. Grzegorczyk (1955). Computable functionals. *Fund. Math.* **42**, 168–202.

[8] P.W. Glynn and S. Henderson. Nonexistence of a class of variate generation schemes. *Operations Research Letters*, to appear.

[9] G.H. Hardy, J.E. Littlewood, and G. Pólya (1959). *Inequalities*, Cambridge University Press, Cambridge.

[10] M. S. Keane , G. L. O'Brien (1994). A Bernoulli factory, *ACM Transactions on Modeling and Computer Simulation* **4**, Issue 2.

[11] G. Pólya (1928) Über positive Darstellung von Polynomen Vierteljschr, *Naturforsch. Gez. Žurich* **73**, 141–145. In *Collected papers* **2** (1974) MIT press, 309–313.

[12] Y. Peres (1992). Iterating von Neumann's Procedure for Extracting Random Bits, *Ann. Stat.*, **20**, 590–597.

[13] M. B. Pour-El and J. I. Richards (1988). *Computability in Analysis and Physics,* Springer-Verlag.

[14] V. Powers and B. Reznick (2002). A new bound for Polya's Theorem with applications to polynomials positive on polyhedra, to appear in *MEGA 2000 proceedings, J. Pure Applied Algebra.*

[15] J. G. Propp and D. B. Wilson (1996). Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures and Algorithms*, **9(1,2)**, 223–252.

[16] J. G. Propp and D. B. Wilson (1998). How to get a perfectly random sample from a generic Markov chain and generate a random spanning tree of a directed graph. *Journal of Algorithms* **27**, 170–217.

[17] A. Turing (1936). On computable numbers, with an application to the Entscheidungsproblem. *Proc. London Math. Soc.*, Ser. 2 **42**, 230–265.

[18] J. von Neumann (1951). Various techniques used in connection with random digits. *Applied Math Series* **12**, 36–38.

A Appendix on Algebraic functions

Christopher J. Hillar
chillar@math.berkeley.edu

The purpose of this appendix is to establish the following fact using techniques from real algebraic geometry. It will be a direct corollary of the more general Theorem A.2 below.

Theorem A.1. *Let* $\{F_i(p, x_1, \ldots, x_n)\}_{i=1}^m \subset \mathbb{Q}[p, x_1, \ldots, x_n]$*, and let* S *be the set in* $\mathbb{R}^{n+1}$ *defined by*

$$S = \left\{ (p, x_1, \ldots, x_n) : \; 0 < p < 1 \;\wedge\; \bigwedge_{j=1}^{n} 0 \leq x_j \leq 1 \;\wedge\; \bigwedge_{i=1}^{m} F_i(p, x_1, \ldots, x_n) = 0 \right\}.$$

Suppose that for each $p \in (0, 1)$*, there exist a unique* $(a_1, \ldots, a_n) \in \mathbb{R}^n$ *such that* $(p, a_1, \ldots, a_n) \in S$*; equivalently,* S *is given as the image of some function* $\phi : (0, 1) \to [0, 1]^{n+1}$ *with* $\phi(p) = (p, \phi_1(p), \ldots, \phi_n(p))$*. Then, there exist nonzero polynomials* $g_j \in \mathbb{Q}[X, Y]$ *such that for all* $p \in (0, 1)$*,*

$$g_j(p, \phi_j(p)) = 0.$$

We begin with an abstract setting. Let K be a field, and let P be a subset of K satisfying the following two properties:

1. If $x, y \in P$, then $x + y$ and $xy \in P$.
2. K is the disjoint union of P, $\{0\}$, and $-P$.

A subset P as above is called the set of *positive* elements, and we say that P is an *ordering* of K. A field K is then an *ordered* field if there exists an ordering of K. A *real field* is a field in which -1 is not a sum of squares, and a *real closed field* R is a real field such that such that any algebraic extension of R that is real must be equal to R. For example, both $\mathbb{R}$ and $\mathbb{R} \cap \mathbb{Q}^a$ are real closed fields (where $\mathbb{Q}^a$ is the algebraic closure of $\mathbb{Q}$). For ease of notation below, we set $\mathbb{Q}^* = \mathbb{R} \cap \mathbb{Q}^a$. From the definition, it is clear that any real field must have characteristic 0, and thus $\mathbb{Q}$ naturally embeds in any real field. It also follows that $\mathbb{Q}^*$ is a subset of every real closed field.

Any real closed field R has a unique ordering, and the positive elements are the squares of R. Moreover, every polynomial of odd degree in $R[x]$ has a root in R [2, p. 452]. In light of these observations, the axioms for the theory of real closed fields (RCF) consist of [1, p. 24]:

1. the axioms for ordered fields;
2. $\forall x > 0 \; \exists y \; y^2 = x$;
3. the axiom $\forall x_0 \ldots \forall x_{n-1} \; \exists y \; y^n + x_{n-1}y^{n-1} + \cdots + x_0 = 0$ for each odd $n > 0$.

The set S as in Theorem A.1 is an example of a set definable by a Boolean combination of a finite number of polynomial inequalities and equalities (over $\mathbb{Q}$). Such a set is called *semialgebraic* (over $\mathbb{Q}$), and a function is called *semialgebraic* if its graph is a semialgebraic set. We may now state the main theorem.

Theorem A.2. *Let R be a real closed field, and let A be a semialgebraic subset of R defined using polynomials $\{H_i(p)\}_{i=1}^k \subset \mathbb{Q}[p]$. Also, let S be a semialgebraic subset of R^{n+1} defined by $\{F_i(p, x_1, \ldots, x_n)\}_{i=1}^m \subset \mathbb{Q}[p, x_1, \ldots, x_n]$. Suppose that for each $p \in A$, there exist a unique point $(a_1, \ldots, a_n) \in R^n$ such that $(p, a_1, \ldots, a_n) \in S$; equivalently, S is given as the image of some function $\phi : A \to R^{n+1}$ with $\phi(p) = (p, \phi_1(p), \ldots, \phi_n(p))$. Then, there exist nonzero polynomials $g_j \in \mathbb{Q}[X, Y]$ such that $g_j(p, \phi_j(p)) = 0$ for all $p \in A$.*

A fundamental fact about RCF is that it is a *complete theory* [1, p. 19] in the sense that each first order sentence expressible in the theory of RCF is either true in every structure satisfying the RCF axioms or false in every such structure. For example, the sentence, $\forall x\ \exists y\ y^2 = -x$, evaluates to false for any structure (such as $\mathbb{R}$) satisfying the axioms of RCF. As a standard application of completeness, we present the following

Lemma A.3. *Assume the hypothesis of Theorem A.2. Then, for each $p \in A \cap \mathbb{Q}^*$, we have that $\phi_j(p) \in \mathbb{Q}^*$ for all $j \in \{1, \ldots, n\}$.*

Proof. To simplify notation, we write "$p \in A$", for example, in place of the Boolean combination of polynomial inequalities and equalities that defines A. As $\mathbb{Q}$ embeds in any real closed field, the sentence,

$$\forall p\ \exists x_1 \ldots \exists x_n\ \neg(p \in A)\ \vee\ (p, x_1, \ldots, x_n) \in S,$$

is a valid sentence in any structure satisfying the axioms of RCF. By completeness, it must have the same truth value in every real closed field. Since it is a true statement in R by assumption, it follows that it is also true in the real closed field $\mathbb{Q}^*$. Let $p \in A \cap \mathbb{Q}^*$. Then, there exists a tuple, $\mathbf{a} = (a_1, \ldots, a_n) \in (\mathbb{Q}^*)^n$, such that $(p, a_1, \ldots, a_n) \in S$. By the hypothesis, this $\mathbf{a}$ must be the unique tuple in R^n for this p, and hence for all j, $\phi_j(p) = a_j \in \mathbb{Q}^*$, completing the proof. ■

A key result in the theory of RCF is the Tarski-Seidenberg Theorem (which is essentially a restatement of the fact that RCF has quantifier elimination) [3, p. 92].

Theorem A.4. *The projection of a semialgebraic set is semialgebraic.*

Corollary A.5. *The functions ϕ_j are semialgebraic functions.*

Proof. Let $T = \{(p, x_1, \ldots, x_n)\ :\ p \in A\ \wedge\ (p, x_1, \ldots, x_m) \in S\}$, which is semialgebraic. Then, the image of the projection of T into R^2 given by $(p, a_1, \ldots, a_n) \mapsto (p, a_j)$ is also semialgebraic by the theorem. ■

Semialgebraic functions are well-behaved in the following sense [4, p. 17].

Theorem A.6. *Let R be a real closed field. If T is a semialgebraic subset of R^n and $h : T \to R$ is semialgebraic, then there is a nonzero polynomial $g(X_1, \ldots, X_n, Y) \in R[X_1, \ldots, X_n, Y]$ such that $g(\mathbf{x}, h(x)) = 0$ for all $\mathbf{x} \in T$.*

We remark that applying Theorem A.6 with Corollary A.5 already gives a result similar to Theorem A.2. The only subtlety is that we would like the polynomial g to be in $\mathbb{Q}[X, Y]$ instead of $R[X, Y]$. We are now ready prove Theorem A.2.

Proof of Theorem A.2. Fix $j \in \{1, \ldots, n\}$. We will apply Theorem A.6 with $R = \mathbb{Q}^*$ and $T = A \cap \mathbb{Q}^*$. From Lemma A.3, it follows that $\phi_j(T) \subseteq \mathbb{Q}^*$, and from Corollary A.5, we have that ϕ_j is semialgebraic. Therefore, from Theorem A.6, there is a nonzero polynomial $g(X, Y) \in \mathbb{Q}^*[X, Y]$ such that $g(p, \phi_j(p)) = 0$ for all $p \in A \cap \mathbb{Q}^*$. We will now produce a nonzero polynomial $\tilde{g} \in \mathbb{Q}[X, Y]$ with the same property.

Consider the field, $\mathbb{Q}^*(X)$, of rational functions in the variable X. View $g(X, Y)$ as a polynomial in the variable Y over $\mathbb{Q}^*(X)$, and, upon clearing denominators, let $q_i(X, Y) \in \mathbb{Q}^*[X, Y]$ ($i = 1, \ldots, r$) be the irreducible factors of $g(X, Y)$ (over $\mathbb{Q}^*(X)$). It is clear that for all $p \in A \cap \mathbb{Q}^*$, we have $\prod_{i=1}^r q_i(p, \phi_j(p)) = 0$.

Fix i and let $\alpha_1, \ldots, \alpha_t \in \mathbb{Q}^*$ be all of the coefficients in $q_i(X, Y)$. Extend $\mathbb{Q}(X)$ by these coefficients, so that $\mathbb{Q}(X)(\alpha_1, \ldots, \alpha_t)$ is a finite extension of $\mathbb{Q}(X)$. Also, let $\mathbb{Q}(X)(\alpha_1, \ldots, \alpha_t)(y)$ be a finite extension of $\mathbb{Q}(X)(\alpha_1, \ldots, \alpha_t)$ defined by the equation $q_i(X, y) = 0$. It follows that y is algebraic over $\mathbb{Q}(X)$, and upon clearing denominators, let $\tilde{q}_i(X, Y) \in \mathbb{Q}[X, Y]$ be such that $\tilde{q}_i(X, y) = 0$.

Since both $q_i(X, Y)$ and $\tilde{q}_i(X, Y)$ have y as a root and since $q_i(X, Y)$ is irreducible, it follows that $q_i(X, Y)$ divides $\tilde{q}_i(X, Y)$. As both $\tilde{q}_i(X, Y)$ and $q_i(X, Y)$ are in $\mathbb{Q}^*[X, Y]$, Gauss's Lemma [2, p. 181] gives us that,

$$h_i(X, Y) := \tilde{q}_i(X, Y)/q_i(X, Y) \in \mathbb{Q}^*[X, Y].$$

Let $\tilde{g}(X, Y) = \prod_{i=1}^r \tilde{q}_i(X, Y) \in \mathbb{Q}[X, Y]$. We claim that $\tilde{g}(X, Y)$ is our desired polynomial. But indeed, for all $p \in A \cap \mathbb{Q}^*$,

$$\tilde{g}(p, \phi_j(p)) = \prod_{i=1}^r q_i(p, \phi_j(p)) h_i(p, \phi_j(p)) = 0. \tag{12}$$

Finally, let A and R be as in the statement of Theorem A.2, and let W be the graph of $\phi_j : A \to R$, which is a semialgebraic set (over $\mathbb{Q}$). Consider the sentence,

$$\forall p\ \forall a\ \neg((p, a) \in W)\ \vee\ \tilde{g}(p, a) = 0,$$

which is valid in any structure satisfying the axioms of RCF (again, since $\mathbb{Q}$ embeds in any real closed field). By the above argument, it is a true statement for $\mathbb{Q}^*$, and therefore, by completeness, it is also true for R. This completes the proof. ■

As a final remark, we note that the fussiness in the proof of Theorem A.2 was necessary to avoid division by zero in (12) after applying the substitution homomorphism with $X \mapsto p$ and $Y \mapsto \phi_j(p)$.

References

[1] D. Haskell, A. Pillay, and C. Steinhorn, *Model Theory, Algebra, and Geometry*, Cambridge University Press, 2000.

[2] S. Lang, *Algebra -3rd ed*, Addison-Wesley Publishing Company, New York, 1993.

[3] D. Marker. *Model Theory: an Introduction*, Springer Verlag, 2002.

[4] D. Marker. *Introduction to the Model Theory of Fields*, http://www.math.uic.edu/~marker/.