Optimal Coin Flipping

Dexter Kozen

Department of Computer Science
Cornell University
Ithaca, New York 14853-7501, USA
kozen@cs.cornell.edu
http://www.cs.cornell.edu/~kozen

In honor of Prakash Panangaden on the occasion of his sixtieth birthday

Abstract. This paper studies the problem of simulating a coin of arbitrary real bias q with a coin of arbitrary real bias p with minimum loss of entropy. We establish a lower bound that is strictly greater than the information-theoretic bound. We show that as a function of q, it is an everywhere-discontinuous self-similar fractal. We provide efficient protocols that achieve the lower bound to within any desired accuracy for $(3-\sqrt{5})/2 < p < 1/2$ and achieve it exactly for $p = 1/2$.

Keywords: probabilistic protocols, randomness, entropy.

1 Introduction

A *discrete simulation protocol* is any procedure that maps a stream of digits from one alphabet to a stream of digits from another alphabet. If the input sequence comes from a random process, then the statistical properties of the input stream impart statistical properties to the output stream, and we can think of the protocol as a reduction from one random source to another.

The *efficiency* of the simulation is the rate of entropy produced per unit of entropy consumed [4,7]. The efficiency measures the amount of randomness lost in the conversion. By general information-theoretic considerations, this value cannot exceed unity [1,6]. In general, the efficiency may not exist, or it may exist but vary with time.

A paradigmatic example is the simulation of a coin of arbitrary real bias q with a coin of arbitrary real bias p. Here, both the input and output alphabets are binary, the input is a sequence of i.i.d. bias-p coin flips, $0 < p < 1$, and the output is a sequence of i.i.d. bias-q coin flips, $0 \le q \le 1$. We call this a *p, q-simulation protocol*. For such protocols, the efficiency is

$$\frac{H(q) \cdot E_{\text{prod}}}{H(p) \cdot E_{\text{cons}}},$$

where H is the Shannon entropy

$$H(p) = -p \log p - (1-p) \log(1-p)$$

F. van Breugel et al. (Eds.): Panangaden Festschrift, LNCS 8464, pp. 407–426, 2014.

and E_{prod} and E_{cons} are, respectively, the expected number of output digits produced and the expected number of input digits consumed in one round of the protocol. If $E_{\text{prod}} = 1$, this gives an information-theoretic lower bound

$$E_{\text{cons}} \geq \frac{H(q)}{H(p)}$$

on the number of bias-p coin flips required by the protocol to produce one output digit. To maximize the efficiency of the simulation, we should minimize this quantity.

A classical example of a $p, \frac{1}{2}$-simulation protocol is the *von Neumann trick* [9]. The bias-p coin is flipped twice. If the outcome is HT, the protocol halts and declares H for the fair coin. If the outcome is TH, the protocol halts and declares T. On any other outcome, the process is repeated. This protocol has the advantage that it is oblivious to the bias of the input coin, but its efficiency is quite poor even for p close to $1/2$. For example, for $p = 1/3$, the von Neumann trick consumes 4.5 input digits per output digit, whereas the Shannon bound is only $1/(\log 3 - 2/3) \approx 1.083\cdots$.

More efficient simulations and enhancements have been studied in [4,7,13]. It is known that any discrete i.i.d. process can simulate any other discrete i.i.d. process with efficiency asymptotically approaching 1, provided the protocol is allowed unbounded *latency*; that is, it may wait and produce arbitrarily long strings of output digits at once. Unbounded latency is exploited in [7] to simulate a fair coin with an arbitrary coin with asymptotically optimal efficiency. The technique is a generalization of the von Neumann trick. In the other direction, [6, Theorem 5.12.3] shows that a fair coin can in principle generate one output digit of an arbitrary coin with expected consumption at most two more than the entropy. In conjunction with [6, Theorem 5.4.2], this yields a method for generating a sequence of i.i.d. bias-q coins from a fair coin with efficiency asymptotically approaching 1, again allowing unbounded latency.

In this paper we consider *non-oblivious*, *one-bit output* protocols: those that output exactly one output digit in each round but take advantage of the knowledge of p. For fixed $0 < p < 1$, let $E_{\text{opt}}(q)$ be the infimum of E_{cons} over all one-bit output p, q-simulation protocols. We show:

- The function $E_{\text{opt}}(q)$ is an everywhere-discontinuous self-similar fractal. For all but finitely many points, it is strictly larger than the Shannon bound $H(q)/H(p)$. A graph of E_{opt} compared to the Shannon bound for $p = 1/3$ is shown in Fig. 1.
- For all $0 \leq q \leq 1$, there exists a p, q-simulation protocol that achieves $E_{\text{opt}}(q)$. Previously, this was known only for $p = 1/2$ [6].
- There exists a single *residual probability protocol* that is optimal for all q. A *residual probability protocol* is a protocol whose state set is the closed unit interval $[0, 1]$ and the probability of halting and reporting heads (respectively, tails) starting from state q is q (respectively, $1 - q$). It is optimal for all q in the sense that $E_{\text{cons}}(q) = E_{\text{opt}}(q)$. The protocol is nondeterministic, and it is not known whether it can be made deterministic in general, even for rational p and q.

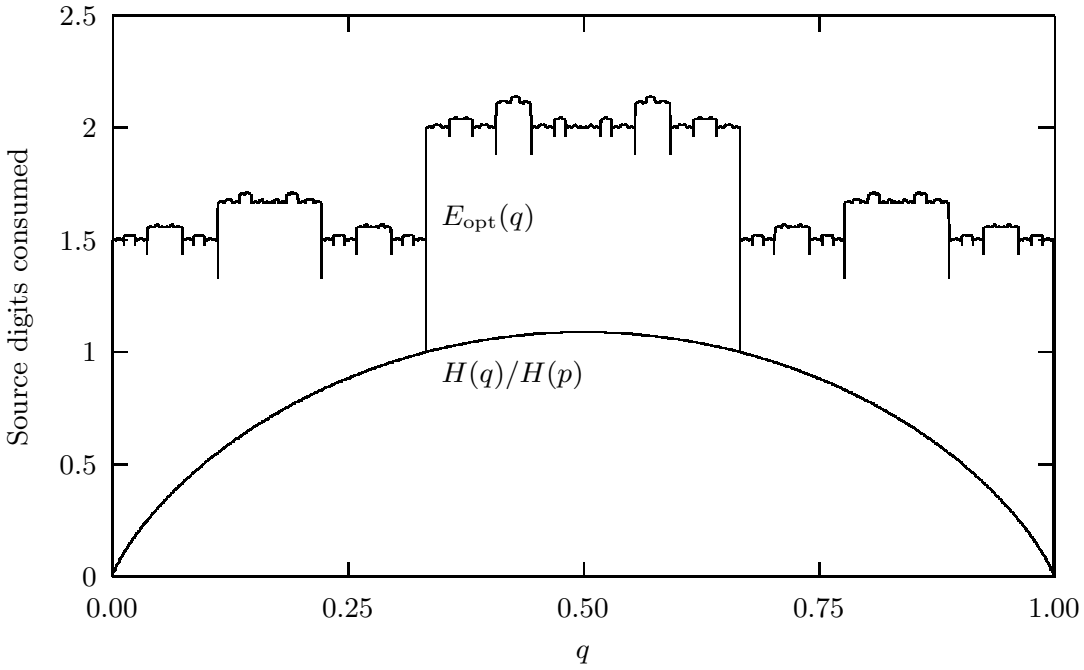

Fig. 1. Comparison of $E_{\mathrm{opt}}(q)$ and the Shannon bound $H(q)/H(p)$ for $p = 1/3$

- For $(3-\sqrt{5})/2 < p \leq 1/2$, we exhibit a family of deterministic, efficiently computable[1] residual probability protocols that achieve $E_{\mathrm{opt}}(q) + \varepsilon$ for any desired degree of accuracy $\varepsilon > 0$ and all q.
- For a fair input coin ($p = 1/2$), we show that the optimal residual probability protocol is computable, and determine the values of $E_{\mathrm{opt}}(q)$ exactly. A similar protocol for $p = 1/2$ was proposed in [6] but without proof, and the values of E_{opt} were not established.

Some of the proof techniques we use are somewhat nonstandard. One particular innovation is the coalgebraic formulation of stochastic simulation protocols introduced in Section 2. In contrast to the usual formulation of stochastic processes as sequences of random variables, this approach gives a powerful technique for reasoning about various functions defined as fixpoints of recursive equations.

1.1 Other Related Work

There is a large body of interesting work on extracting randomness from weak random sources (e.g. [10,11,14,15]). These models typically work with imperfect knowledge of the input source and provide only approximate guarantees on the quality of the output. In this paper, however, we assume that the statistical properties of the input and output are known completely, and simulations must be exact.

[1] As we are computing with real numbers, we assume unit-time real arithmetic and comparison of real numbers. These assumptions are not necessary if computation is restricted to the rationals.

The fractal nature of certain residual probability protocols was observed in [8], but the existence of optimal protocols was left unresolved.

2 Simulation Protocols

Let $0 < p \leq 1/2$ and $0 \leq q \leq 1$. To simulate a bias-q coin with a bias-p coin, we would ordinarily define the input to the simulation to be a Bernoulli process consisting of a sequence of i.i.d. random variables $X_0, X_1, \ldots$ with success probability p. The simulation would be specified by a function that decides, given a finite history $X_0, X_1, \ldots, X_{n-1}$ of previous bias-p coin flips, whether to halt and declare heads, halt and declare tails, or flip again. The process must halt with probability 1 and must declare heads with probability q and tails with probability $1 - q$.

However, it is technically convenient to specify protocols in terms of more general state sets. We thus define a *protocol* to be a triple (S, β, s_0) consisting of a coalgebra (S, β), where

$$\beta : S \;\to\; \{\text{H}, \text{T}\} + (\{\mathsf{0}, \mathsf{1}\} \to S), \tag{1}$$

and a distinguished *start state* $s_0 \in S$.[2] Intuitively, depending on the current state, the protocol decides either

- to halt immediately and return H or T, thereby declaring the result of the bias-q coin flip to be heads or tails, respectively; or
- to consume a random bias-p coin flip ($\mathsf{0}$ or $\mathsf{1}$), and based on that information, enter a new state.

A protocol is a *p, q-simulation protocol* if, when it is started in its start state s_0 with the input stream generated by a Bernoulli process with success probability p, it halts with probability 1, declaring H with probability q and T with probability $1 - q$.

The protocol is *computable* if the function β is.

Example 1. A traditional choice for the state set would be $\{\mathsf{0}, \mathsf{1}\}^*$, the history of outcomes of previous bias-p coin flips. The transition function would be

$$\beta : \{\mathsf{0}, \mathsf{1}\}^* \to \{\text{H}, \text{T}\} + (\{\mathsf{0}, \mathsf{1}\} \to \{\mathsf{0}, \mathsf{1}\}^*),$$

and the start state would be the empty history $\varepsilon \in \{\mathsf{0}, \mathsf{1}\}^*$. The next step of the protocol is determined by the previous history. If this history is $X_0, \ldots, X_{n-1}$ and the protocol decides to halt and declare heads or tails, then $\beta(X_0, \ldots, X_{n-1})$ would be H or T, respectively. If on the other hand the protocol decides not to halt, and the result of the next bias-p coin flip is X_n, then $\beta(X_0, \ldots, X_{n-1})(X_n) = X_0, \ldots, X_n$.

[2] For clarity, we are using different symbols to distinguish the input coin (heads = $\mathsf{0}$, tails = $\mathsf{1}$) from the output coin (heads = H, tails = T).

Example 2. The following example is a slight modification of one from [8]. The state set is the closed real interval $[0,1]$. If $q \in \{0,1\}$, then $\beta(q) \in \{\text{H},\text{T}\}$, otherwise $\beta(q) \in \{0,1\} \to [0,1]$. The values are

$$\beta(q) = \begin{cases} \text{H} & \text{if } q = 1 \\ \text{T} & \text{if } q = 0 \end{cases}$$

$$\beta(q)_X = \begin{cases} 0 & \text{if } 0 < q \le p \text{ and } X = 1 \\ \dfrac{q}{p} & \text{if } 0 < q \le p \text{ and } X = 0 \\ \dfrac{q-p}{1-p} & \text{if } p < q < 1 \text{ and } X = 1 \\ 1 & \text{if } p < q < 1 \text{ and } X = 0. \end{cases}$$

Intuitively, if $p < q < 1$ and the bias-p coin flip returns heads (0), which occurs with probability p, then we halt and output heads; this gives a fraction p/q of the desired probability q of heads of the simulated bias-q coin. If the bias-p coin returns tails (1), which occurs with probability $1-p$, we rescale the problem to condition on that outcome, setting the state to $(q-p)/(1-p)$ because that is the residual probability of heads, and repeat. Similarly, if $0 < q \le p$ and the bias-p coin returns tails, then we halt and output tails; and if not, we rescale appropriately and repeat.

Example 3. The final coalgebra $(\mathcal{C}, \delta)$ of the type (1) is the set of binary prefix codes for the two-element alphabet $\{\text{H},\text{T}\}$. Each such code consists of a pair of disjoint sets $H, T \subseteq \{0,1\}^*$ such that the elements of $H \cup T$ are pairwise prefix-incomparable. The operation δ is defined by

$$\delta(H,T) = \begin{cases} \text{H} & \text{if } \varepsilon \in H \\ \text{T} & \text{if } \varepsilon \in T \\ \lambda a \in \{0,1\}.(D_a(H), D_a(T)) & \text{otherwise,} \end{cases}$$

where D_a is the Brzozowski derivative $D_a(A) = \{x \mid ax \in A\}$.

The coalgebra $(\mathcal{C}, \delta)$ is *final* in the sense that from any other coalgebra (S, β), there is a unique coalgebra homomorphism $\mathsf{code} : (S,\beta) \to (\mathcal{C},\delta)$, defined by: $\mathsf{code}(s) = (H_s, T_s)$, where H_s (respectively, T_s) is the set of strings $x \in \{0,1\}^*$ such that running the protocol starting from s results in output H (respectively, T) after consuming input digits x. The function code is a coalgebra homomorphism in that $\delta(\mathsf{code}(s)) = \beta(s)$ if $\beta(s) \in \{\text{H},\text{T}\}$, otherwise $\beta(s) : \{0,1\} \to S$ and $\delta(\mathsf{code}(s)) = \mathsf{code} \circ (\beta(s)) : \{0,1\} \to \mathcal{C}$.

In the definition of $\mathcal{C}$, the sets H, T must be disjoint to ensure that δ is well-defined. They need not be nonempty; in fact, if $\beta(s) = \text{H}$, then $h(s) = (\{\varepsilon\}, \varnothing)$. There is no other possible choice for $h(s)$ due to the requirement that H, T be disjoint and elements of $H \cup T$ be pairwise prefix-incomparable. The single-state subcoalgebra $(\varnothing, \varnothing)$ represents protocols that never halt.

2.1 Residual Probability Protocols

Intuitively, the *residual probability* of a state s of a p,q-simulation protocol is the probability r that the protocol halts and declare heads when started in state s. In order to halt with probability 1 from that state, it should also halt and declare tails with probability $1-r$. It is conceivable that a protocol might want to take different actions in two different states, even if the residual probabilities are the same.

Formally, a *residual probability protocol* is a protocol whose state space is the closed unit interval $[0,1]$ and whose probability of halting and declaring heads (resp., tails) when started in state q is q (resp., $1-q$). Thus the next action of the protocol depends only on the residual probability. Example 2 is an example of a residual probability protocol. Theorem 4 below says that when searching for an optimal protocol, we can restrict our attention to residual probability protocols without loss of generality.

2.2 Impatient Protocols

A protocol (S,β) is *impatient* if in every state s, the probability of halting in at most one step is nonzero; that is, either $\beta(s)$, $\beta(\beta(s)_0)$, or $\beta(\beta(s)_1 \in \{\mathrm{H},\mathrm{T}\})$. Assuming computable real arithmetic and comparison of real numbers[3], every p,q has a computable impatient protocol; for example, the protocol of Example 2, as well as others described in [8], are computable and impatient. Every impatient protocol has at most one infinite computation path starting from any state, which occurs with probability 0.

Impatient strategies are not necessarily optimal. Example 2 is not: in that example, $\beta(1-p)_0 = 1$ and $\beta(1-p)_1 = (1-2p)/(1-p)$, whereas a better choice would be $\beta(1-p)_0 = 0$ and $\beta(1-p)_1 = 1$.

2.3 Greedy Protocols

Greedy protocols are a special class of impatient residual probability protocols. Intuitively, a protocol is locally greedy at a state if it attempts to optimize in the next step by halting as early as possible with the maximum allowable probability. To define this formally, we start with the special case

$$(1-p)^2 \le p \le 1-p; \tag{2}$$

equivalently, $(3-\sqrt{5})/2 \le p \le 1/2$. In this case, let us define the *ambiguous region* as the open interval $(p,1-p)$. A greedy protocol must halt immediately when $q \in \{0,1\}$, declaring heads for the q-coin if $q=1$ and tails if $q=0$. Otherwise, if q is not in the ambiguous region, it must flip the p-coin and halt if the outcome is tails, which occurs with probability $1-p$, declaring either tails or heads for the q-coin, depending on whether $q \le p$ or $q \ge 1-p$, respectively. If q is in the ambiguous region, it must flip the p-coin and halt if the outcome is heads,

[3] If p and q are rational, this assumption is not needed.

which occurs with probability p, but there is a choice whether to declare heads or tails for the q-coin, leading to two possible greedy strategies. If it declares heads when the p-coin returns heads, then it must rescale to $(q-p)/(1-p)$ when the p-coin returns tails. If it declares tails when the p-coin returns heads, then it must rescale to $q/(1-p)$ when the p-coin returns tails. It is not immediately clear which action will ultimately be better.

The significance of the restriction (2) is that the protocol exits the ambiguous region after only one step, and that is the case that we will focus on in this paper. More generally, let $k = \lfloor -1/\log_2(1-p) \rfloor$, the least positive integer such that $(1-p)^{k+1} < 1/2$. The *ambiguous region* for p is the open interval $(b, 1-b)$, where b is either $1-(1-p)^k$ or $(1-p)^{k+1}$, depending on which interval is smaller. Under the restriction (2), $k = 1$. In this more general situation, a protocol is *greedy* if it moves so as to enter one of the regions $q \le p$ or $q \ge 1-p$ as quickly as possible; this is determined except when q is in the ambiguous region.

Greedy strategies are not necessarily optimal. For example, let p be a transcendental number satisfying (2). There is an uncountable nowhere-dense set of points on which the greedy strategy achieves its best running time $1/(1-p)$; that is, the protocol never enters the ambiguous region. It can be shown that these are exactly finite and infinite alternating sums of increasing integer powers of p:

$$J = \{p^{k_0} - p^{k_1} + p^{k_2} - p^{k_3} + \cdots \mid k_i \in \mathbb{Z},\ 0 \le k_0 < k_1 < \cdots\}.$$

Consider $q = 2p(1-p)$. Then $p < q < 1-p$, so q is in the ambiguous region. After one greedy step in either direction, it is easily checked that the resulting image of q is not in J. Moreover, there must subsequently be an infinite computation path, because otherwise p would be algebraic. Thus the expectation of any greedy protocol is strictly larger than $p+(1-p)(1+1/(1-p)) = 2$. A better strategy is to flip the p-coin twice, declaring heads if the outcome is $\mathsf{10}$ or $\mathsf{01}$, tails otherwise. The expectation is 2, and this is optimal.

3 Coalgebras and Fixpoint Induction

Technically, coalgebras of type (1) are F-coalgebras, where $F : \mathsf{Set} \to \mathsf{Set}$ is the polynomial functor $FX = \mathbb{1} + \mathbb{1} + X^2$. Given an F-coalgebra (S, β), many interesting functions can be specified by providing an F-algebra (A, α) with some extra order structure allowing for the existence of least fixpoints. The function defined is the least fixpoint of the map

$$f \ \mapsto\ \alpha \circ Ff \circ \beta, \tag{3}$$

that is, the least f such that the following diagram commutes:

$$\begin{array}{ccc} S & \xrightarrow{\ f\ } & A \\ {\scriptstyle\beta}\downarrow & & \uparrow{\scriptstyle\alpha} \\ FS & \xrightarrow{\ Ff\ } & FA \end{array}$$

Intuitively, the destructor $\beta : S \to FS$ computes the arguments to a recursive call, the map $Ff : FS \to FA$ is the recursive call, and the constructor $\alpha : FA \to A$ is the construction applied to the returned element. This general scheme for recursively defined functions has been previously studied in [2,3,5].

If A is a chain-complete partially ordered set and α order-continuous, then the map (3) is monotone and order-continuous on functions $S \to A$ under the pointwise order, therefore by the Knaster–Tarski theorem has a unique least fixpoint.

Example 4. The *outcome* $O(s)$ of the simulation starting from state s is a random variable defined on the probability space $\{0,1\}^\omega$ taking values in $\{\text{H}, \text{T}, \bot\}$. The value $\bot$ signifies nonhalting. Formally,

$$O : S \to \{0,1\}^\omega \to \{\text{H}, \text{T}, \bot\}$$

is the least fixpoint of the equation

$$O(s)(X \cdot \sigma) = \begin{cases} \beta(s) & \text{if } \beta(s) \in \{\text{H}, \text{T}\} \\ O(\beta(s)_X)(\sigma) & \text{if } \beta(s) \in \{0,1\} \to S. \end{cases}$$

This would be specified by the F-algebra (A, α), where

$$A = \{0,1\}^\omega \to \{\text{H}, \text{T}, \bot\}$$

$$\alpha(f) = \begin{cases} \lambda\sigma \in \{0,1\}^\omega.f & \text{if } f \in \{\text{H}, \text{T}\} \\ \lambda\sigma \in \{0,1\}^\omega.f(\mathsf{head}\,\sigma)(\mathsf{tail}\,\sigma) & \text{if } f \in \{0,1\} \to A \end{cases}$$

under the pointwise ordering on A induced by $\bot < \text{H}$ and $\bot < \text{T}$.

Example 5. Define $P(s) = \Pr(O(s) = \text{H})$, the probability that the outcome is heads starting in state s. This is specified by the F-algebra on $[0,1]$ with constructor

$$X \mapsto \begin{cases} 1 & \text{if } X = \text{H} \\ 0 & \text{if } X = \text{T} \\ p \cdot X(0) + (1-p) \cdot X(1) & \text{if } X \in \{0,1\} \to [0,1] \end{cases}$$

and the natural order on $[0,1]$.

Example 6. The expected consumption of input digits starting from state s satisfies the equation

$$E(s) = \begin{cases} 0 & \text{if } \beta(s) \in \{\text{H}, \text{T}\} \\ 1 + p \cdot E(\beta(s)_0) + (1-p) \cdot E(\beta(s)_1) & \text{if } \beta(s) \in \{0,1\} \to S. \end{cases}$$

The function E is specified by the F-algebra on $\mathbb{R}^+ = \{x \in \mathbb{R} \mid x \geq 0\} \cup \{\infty\}$ with constructor

$$X \mapsto \begin{cases} 0 & \text{if } X \in \{\text{H}, \text{T}\} \\ 1 + p \cdot X(0) + (1-p) \cdot X(1) & \text{if } X \in \{0,1\} \to \mathbb{R}^+ \end{cases}$$

and the natural order on $\mathbb{R}^+$.

An important property for our purposes, due to Adámek, Milius, and Velebil [3], is that the least fixpoint construction is *natural* in S the sense that if f and f' are the least solutions of (3) in the F-coalgebras S and S', respectively, and if $h : S \to S'$ is an F-coalgebra morphism, then $f = f' \circ h$ (Theorem 1 below). The significance of this property is that a function defined by (3), such as the probability of heads or the expected comsumption of input digits, is the same whether measured in S or any quotient of S by a bisimulation. In particular, if $s \in S$ is a start state of a protocol and $\mathsf{code}(s) \in \mathcal{C}$ is its image in the final coalgebra, then the expected consumption of input digits starting in state s is just the expected codeword length $\sum_{x \in \mathsf{code}(s)} \Pr(x) \cdot |x|$ if $P(s) = 1$, or ∞ if $P(s) < 1$.[4]

Theorem 1 ([3], Proposition 3.5). *Let (A, α) be an ordered F-algebra such that A is a chain-complete and α order-continuous. The construction of the least fixpoint of (3) is natural in S; that is, if $h : S \to S'$ is an F-coalgebra morphism, then $f_S = f_{S'} \circ h$.*

Proof. Let τ_S be the map (3) on functions $S \to A$. The assumptions on A and α imply that τ_S is monotone and order-continuous under the pointwise order on $S \to A$. Let $\bot$ be the bottom element of A. The map $\lambda s \in S.\bot$ is the bottom element of $S \to A$. If $h : S \to S'$ is an F-coalgebra morphism, then clearly $\lambda s \in S.\bot = (\lambda s \in S'.\bot) \circ h$, therefore the selection of $\lambda s \in S.\bot$ is natural in S. Moreover, it is easily argued that τ_S is also natural in S. By induction, $\tau_S^n(\lambda s \in S.\bot)$ is natural in S for all n. By continuity, the least fixpoint is $\sup_n \tau_S^n(\lambda s \in S.\bot)$, and the result follows from the observation that suprema are preserved by composition with h on the right. □

3.1 Fixpoint Induction

The construction of the least fixpoint of the monotone map τ_S admits the use of the following *fixpoint induction rule* [12]: If $f : S \to A$ is the least fixpoint of τ_S, and if $\tau_S(g) \leq g$, then $f \leq g$.

3.2 Two Metrics

A popular metric on streams defines the distance between two streams to be 2^{-n} if n is the length of their maximal common prefix, or 0 if the streams are equal. There is an analogous metric on codes. We say that binary codes $s = (H, T)$ and $t = (H', T')$ *agree to length* n if for all words $x \in \{\mathsf{0}, \mathsf{1}\}^*$ of length n or less, $x \in H$ iff $x \in H'$ and $x \in T$ iff $x \in T'$. We define $d'(s, t) = p^n$ if n is the maximum number such that s and t agree to length n, or 0 if they are equal. We use p^n

[4] Here $\Pr(x) = p^{\#0(x)}(1-p)^{\#1(x)}$, where $\#a(x)$ is the number of occurrences of a in x for $a \in \{\mathsf{0}, \mathsf{1}\}$ and $x \in \{\mathsf{0}, \mathsf{1}\}^*$. We write $x \in \mathsf{code}(s)$ for $x \in H \cup T$, where $\mathsf{code}(s) = (H, T)$ is the image of state s under the unique F-coalgebra morphism to the final F-coalgebra $\mathcal{C}$.

instead of 2^{-n} for technical reasons, but the difference is of no consequence, as the same topology is generated. The metric d' satisfies the recurrence

$$d'(s,t) = \begin{cases} 1 & \text{if either } \delta(s), \delta(t) \in \{\text{H}, \text{T}\} \text{ and } \delta(s) \neq \delta(t) \\ 0 & \text{if both } \delta(s), \delta(t) \in \{\text{H}, \text{T}\} \text{ and } \delta(s) = \delta(t) \\ p \cdot \max(d'(\delta(s)_0, \delta(t)_0),\ d'(\delta(s)_1, \delta(t)_1)) & \\ & \text{if } \delta(s), \delta(t) \in \{0,1\} \to \mathcal{C}, \end{cases}$$

and in fact this can be taken as a formal definition according to (3). A similar map d' is induced on the states of any protocol by $d'(s,t) = d'(\mathsf{code}(s), \mathsf{code}(t))$, where code is the unique F-coalgebra morphism to $\mathcal{C}$. On arbitrary protocols, the map d' is not a metric in general, but only a pseudometric.

Alternatively, we might consider two protocols similar if, when run simultaneously, they halt at the same time and produce the same output with high probability. Thus we define $d : S \times T \to [0,1]$ to be the least solution of the equation

$$d(s,t) = \begin{cases} 1 & \text{if either } \beta(s), \beta(t) \in \{\text{H}, \text{T}\} \text{ and } \beta(s) \neq \beta(t) \\ 0 & \text{if both } \beta(s), \beta(t) \in \{\text{H}, \text{T}\} \text{ and } \beta(s) = \beta(t) \\ p \cdot d(\beta(s)_0, \beta(t)_0) + (1-p) \cdot d(\beta(s)_1, \beta(t)_1) & \\ & \text{if } \beta(s), \beta(t) \in \{0,1\} \to S. \end{cases}$$

Formally, d can be specified in curried form $d(s,t) = d(s)(t)$ by an F-algebra on $T \to [0,1]$ as above. We could also define an F-coalgebra on $S \times T$ with

$$(s,t) \mapsto \begin{cases} \text{H} & \text{if either } \beta(s), \beta(t) \in \{\text{H}, \text{T}\} \text{ and } \beta(s) \neq \beta(t) \\ \text{T} & \text{if both } \beta(s), \beta(t) \in \{\text{H}, \text{T}\} \text{ and } \beta(s) = \beta(t) \\ \lambda a \in \{0,1\}.(\beta(s)_a, \beta(t)_a) & \text{if } \beta(s), \beta(t) \in \{0,1\} \to S \end{cases}$$

and take $d(s,t) = \Pr(O(s,t) = \text{H})$.

Symmetry and the triangle inequality are easy to verify, thus any protocol S is a pseudometric space under the distance functions d and d'.

Lemma 1. *Let S and T be F-coalgebras, $s \in S$, and $t \in T$. The following are equivalent:*

1. $d(s,t) = 0$
2. $d'(s,t) = 0$
3. *s and t are bisimilar.*

Proof. The states s and t are bisimilar iff they have the same image in the final coalgebra, and d and d' are also preserved. Thus if s and t are bisimilar, then $d(s,t) = d'(s,t) = 0$. Conversely, any two distinct prefix codes must differ on some codeword $x \in \{0,1\}^*$, in which case both $d(s,t), d'(s,t) \geq p^{|x|}$. □

Lemma 2. *Every d'-open set is d-open. If $E(s) < \infty$, then every d-open neighborhood of s is d'-open.*

Proof. If s and t disagree on x, then the probability of disagreement is at least $p^{|x|}$, thus $d(s,t) \geq d'(s,t)$, so every basic d'-open set $\{t \mid d'(s,t) < \varepsilon\}$ contains the basic d-open set $\{t \mid d(s,t) < \varepsilon\}$, thus is also d-open.

Conversely, suppose $E(s) < \infty$. If $d'(s,t) \leq p^n$, then the codes s and t agree to length n, thus s and t differ with probability at most $\Pr(|x| > n) \leq E(s)/n$ by the Markov inequality. Thus $d(s,t) \leq E(s)/n$. We conclude that $d(s,t) \leq E(s)/\log_p d'(s,t)$. □

Lemma 2 says that d generates a finer topology than d' on $\mathcal{C}$. They are not the same: an example of a d-open set that is not d'-open is the ε-neighborhood of $s = (\varnothing, \varnothing)$ in the d-metric for any $0 < \varepsilon < 1$. For $s_n = (\{0,1\}^n, \varnothing)$, $d(s, s_n) = 1$ but $d'(s, s_n) = p^n$.

In the final F-coalgebra $\mathcal{C}$, $d(s,t) = 0$ implies $s = t$, since bisimilar states of $\mathcal{C}$ are equal. Thus $\mathcal{C}$ is a metric space under d. However, it is not complete, even restricted to points with finite expectation. For example, the sequence $(\{0,1\}^n, \varnothing)$ has no limit point. However, the subspace of points with expected running time bounded by any constant b is compact, thus complete, as we will now show.

Theorem 2. *Let $\mathcal{C}_b$ be the subspace of points $s \in \mathcal{C}$ such that $E(s) \leq b$. Then $\mathcal{C}_b$ is a compact, hence complete, metric space under d.*

Proof. We have argued that $\mathcal{C}_b$ is a metric space, thus it remains to show compactness. Certainly $\mathcal{C}_b$ is compact under d'. By Lemma 2, d and d' generate the same topology on $\mathcal{C}_b$, therefore $\mathcal{C}_b$ is also compact under d. □

Recall that $P(s) = \Pr(O(s) = \text{H})$.

Lemma 3. *The map P is continuous with respect to d on $\mathcal{C}$.*

Proof.

$$\begin{aligned} |P(s) - P(t)| &= |\Pr(O(s) = \text{H} \wedge O(t) \neq \text{H}) - \Pr(O(t) = \text{H} \wedge O(s) \neq \text{H})| \\ &\leq \Pr(O(s) = \text{H} \wedge O(t) \neq \text{H}) + \Pr(O(t) = \text{H} \wedge O(s) \neq \text{H}) \\ &\leq \Pr(O(s) \neq O(t)) \\ &= d(s,t). \end{aligned}$$

□

The map E is not continuous at any point in either metric, not even restricted to $\mathcal{C}_b$. However, we have the following.

Lemma 4. *Let $A \subseteq \mathcal{C}$ and let $\text{cl}'(A)$ denote the closure of A under the d' metric. Then $\sup\{E(s) \mid s \in \text{cl}'(A)\} \leq \sup\{E(t) \mid t \in A\}$.*

Proof. Recall that for points s in the final coalgebra, $E(s) = \sum_{x \in s} \Pr(x) \cdot |x|$ if $\sum_{x \in s} \Pr(x) = 1$, and ∞ if $\sum_{x \in s} \Pr(x) < 1$. Let $s \in \text{cl}'(A)$. If $\sum_{x \in s} \Pr(x) < 1$, then that is also true for some $t \in A$, so in that case both suprema are ∞; so assume that $\sum_{x \in s} \Pr(x) < 1$.

For $\varepsilon > 0$, let n be large enough that

$$\sum_{\substack{x \in s \\ |x| \le n}} \Pr(x) \cdot |x| \ge \begin{cases} E(s) - \varepsilon & \text{if } E(s) < \infty, \\ 1/\varepsilon & \text{if } E(s) = \infty \end{cases}$$

and choose a point $t \in A$ such that s and t agree to length n. Then

$$E(t) \ge \sum_{\substack{x \in t \\ |x| \le n}} \Pr(x) \cdot |x| = \sum_{\substack{x \in s \\ |x| \le n}} \Pr(x) \cdot |x| \ge \begin{cases} E(s) - \varepsilon & \text{if } E(s) < \infty, \\ 1/\varepsilon & \text{if } E(s) = \infty, \end{cases}$$

thus $\sup\{E(t) \mid t \in A\} \ge E(s)$. As s was arbitrary, the conclusion follows. □

4 Residual Probability Protocols Are Optimal

Let $E_{\mathrm{opt}}(q)$ be the infimum of expectations of all p, q-simulation protocols. There exist protocols with expectation at most $1/p$ (e.g., Example 2), so $E_{\mathrm{opt}}(q) \le 1/p$. A p, q-simulation protocol with start state s is *optimal* if $E_{\mathrm{cons}}(s) = E_{\mathrm{opt}}(q)$.

Theorem 3. *For every p, q such that $0 < p \le 1/2$ and $0 \le q \le 1$, there exists an optimal p, q-simulation protocol.*

Proof. We show that $E_{\mathrm{opt}}(q)$ is attained at a state in the final F-coalgebra $\mathcal{C}$. Let $s_0, s_1, \ldots$ be a sequence of start states of p, q-protocols such that $E(s_n)$ is decreasing and $\lim_n E(s_n) = E_{\mathrm{opt}}(q)$. Since $E(s)$ and $P(s)$ are preserved under morphisms, the images of these states in $\mathcal{C}$ are also start states of p, q-protocols in $\mathcal{C}$ and their expectations are the same, thus we can assume without loss of generality that the s_n are states of $\mathcal{C}_{1/p}$. Since $\mathcal{C}_{1/p}$ is compact, there exists a convergent subsequence with limit $u_q \in \mathcal{C}_{1/p} \in \mathcal{C}_{1/p}$. Since P is continuous (Lemma 3), $P(u_q) = q$, thus u_q is the start state of a p, q-protocol. By Lemma 4, $E(u_q) = E_{\mathrm{opt}}(q)$. □

Theorem 4. *For every p, there is a residual probability protocol U_p that is optimal for every q.*

Proof. Let u_q be the optimal p, q-protocol constructed in Theorem 3. Consider the coalgebra $U_p = (\mathtt{[}0,1\mathtt{]}, \upsilon)$, where

$$\upsilon(q) = \begin{cases} \delta(u_q) & \text{if } \delta(u_q) \in \{\mathrm{H}, \mathrm{T}\} \\ \lambda X \in \{\mathtt{0},\mathtt{1}\}.\Pr(O(\delta(u_q)_X) = \mathrm{H}) & \text{if } \delta(u_q) \in \{\mathtt{0},\mathtt{1}\} \to \mathcal{C}. \end{cases}$$

We claim that for all q,

$$E_U(q) = E_{\mathrm{opt}}(q) \qquad \Pr(O(q) = \mathrm{H}) = q \qquad \Pr(O(q) = \mathrm{T}) = 1 - q, \tag{4}$$

thus U_p with start state q is an optimal p,q-simulation protocol. We first show that

$$E_U(q) \le E_{\mathrm{opt}}(q) \qquad \Pr(O(q) = \mathrm{H}) \le q \qquad \Pr(O(q) = \mathrm{T}) \le 1 - q \tag{5}$$

by fixpoint induction.

For the first inequality of (5), define a property φ on S to be *hereditary* if $\varphi(\beta(s)_0)$ and $\varphi(\beta(s)_1)$ whenever $\beta(s) \in \{0,1\} \to S$ and $\varphi(s)$. The property

$$E(s) = E_{\mathrm{opt}}(P(s)) \tag{6}$$

is hereditary, because it says that s is an optimal protocol for its residual probability. But if s is, then so must be its successors; if not, then we could replace them by a better protocol and thereby improve $E(s)$ as well.

Now we proceed by fixpoint induction to show that $E_U(q) \le E_{\mathrm{opt}}(q)$. It suffices to show that E_{opt} is a fixpoint of the defining equation τ for E_U.

$$\begin{aligned}
&\tau(E_{\mathrm{opt}})(q)\\
&= \begin{cases} 0 & \text{if } \upsilon(q) \in \{\mathrm{H},\mathrm{T}\} \\ 1 + p \cdot E_{\mathrm{opt}}(\upsilon(q)_0) + (1-p) \cdot E_{\mathrm{opt}}(\upsilon(q)_1) & \text{if } \upsilon(q) \in [0,1]^2 \end{cases} && (7)\\
&= \begin{cases} 0 & \text{if } \delta(u_q) \in \{\mathrm{H},\mathrm{T}\} \\ 1 + p \cdot E_{\mathrm{opt}}(P(\delta(u_q)_0)) + (1-p) \cdot E_{\mathrm{opt}}(P(\delta(u_q)_1)) & \text{if } \delta(u_q) \in \mathcal{C}^2 \end{cases} && (8)\\
&= \begin{cases} 0 & \text{if } \delta(u_q) \in \{\mathrm{H},\mathrm{T}\} \\ 1 + p \cdot E_{\mathcal{C}}(u_{P(\delta(u_q)_0)}) + (1-p) \cdot E_{\mathcal{C}}(u_{P(\delta(u_q)_1)}) & \text{if } \delta(u_q) \in \mathcal{C}^2 \end{cases} && (9)\\
&= \begin{cases} 0 & \text{if } \delta(u_q) \in \{\mathrm{H},\mathrm{T}\} \\ 1 + p \cdot E_{\mathcal{C}}(\delta(u_q)_0) + (1-p) \cdot E_{\mathcal{C}}(\delta(u_q)_1) & \text{if } \delta(u_q) \in \mathcal{C}^2 \end{cases} && (10)\\
&= E_{\mathcal{C}}(u_q) && (11)\\
&= E_{\mathrm{opt}}(q). && (12)
\end{aligned}$$

Inference (7) is by the definition of τ. Inference (8) is by the definition of $\upsilon(q)$. Inference (9) is from the construction of Theorem 3. Inference (10) is by the fact that $\delta(u_q)_1$ and $\delta(u_q)_0$ satisfy property (6), since u_q does and the property is hereditary, therefore

$$E_{\mathcal{C}}(\delta(u_q)_X) = E_{\mathrm{opt}}(P(\delta(u_q)_X)) = E_{\mathcal{C}}(u_{P(\delta(u_q)_X)})$$

for $X \in \{0,1\}$. Inference (11) is by the definition of $E_{\mathcal{C}}$. Inference (12) is by Theorem 3.

For the second inequality of (5), writing $P(q)$ for $\Pr(O(q) = \mathrm{H})$, it suffices to show that the identity function on $[0,1]$ is a fixpoint of the defining equation τ

for P.

$$\tau(\lambda q.q)(q) = \begin{cases} 1 & \text{if } \upsilon(q) = \text{H} \\ 0 & \text{if } \upsilon(q) = \text{T} \\ p \cdot (\lambda q.q)(\upsilon(q)_0) + (1-p) \cdot (\lambda q.q)(\upsilon(q)_1) & \text{if } \upsilon(q) \in [0,1]^2 \end{cases} \tag{13}$$

$$= \begin{cases} 1 & \text{if } \upsilon(q) = \text{H} \\ 0 & \text{if } \upsilon(q) = \text{T} \\ p \cdot \upsilon(q)_0 + (1-p) \cdot \upsilon(q)_1 & \text{if } \upsilon(q) \in [0,1]^2 \end{cases} \tag{14}$$

$$= \begin{cases} 1 & \text{if } \delta(u_q) = \text{H} \\ 0 & \text{if } \delta(u_q) = \text{T} \\ p \cdot P(\delta(u_q)_0) + (1-p) \cdot P(\delta(u_q)_1) & \text{if } \delta(u_q) \in \mathcal{C}^2 \end{cases} \tag{15}$$

$$= P(u_q) \tag{16}$$

$$= q. \tag{17}$$

Inference (13) is by definition of τ. Inference (14) is by the application of the identity function. Inference (15) is by definition of υ. Inference (16) is by definition of $P(u_q)$. Inference (17) is by the fact that u_q is the start state of a p, q-protocol.

The proof of the third inequality of (5) is symmetric.

Now we argue that all the inequalities (5) are actually equalities (4). By the first inequality, the probability of halting is 1, since E_U is finite. Since the last two inequalities hold and the left-hand sides sum to 1, the last two inequalities must be equalities. Then U with start state q is a p, q-simulation protocol, thus $E_{\text{opt}}(q) \leq E_U(q)$, therefore the first inequality of (5) is an equality as well. □

5 Properties of E_{opt}

We assume throughout this section and the next that $(3-\sqrt{5})/2 \leq p \leq 1/2$; equivalently, $(1-p)^2 \leq p \leq 1-p$.

For fixed p, a real number $q \in [0,1]$ is *exceptional of degree* d if it has a finite binary prefix code with probabilities $p, 1-p$ whose longest codeword is of length d. The number q is *exceptional* if it is exceptional of some finite degree.

If q is exceptional of degree d, then so is $1-q$, and the pair of codes form a finite loop-free p, q-protocol with maximum running time d. In this case q and $1-q$ are polynomial functions of p of degree d. The twelve exceptional values of degree at most 2 are shown in Table 1.

Some rows of Table 1 collapse for certain degenerate values of p. For $p = 1/2$, rows (iii), (iv), and (v) collapse and rows (ii) and (vi) collapse. For $p = (3-\sqrt{5})/2$, rows (ii) and (v) collapse. These are the only two degenerate values that cause collapse. Rows (v) and (vii) would collapse for $p = 1/3$, but this case is ruled out by the assumption $p \geq (3-\sqrt{5})/2 \approx .382$.

Table 1. Exceptional values of degree at most 2

	q	$1-q$	degree	E_{opt}
(i)	0	1	0	$=0$
(ii)	p	$1-p$	1	$=1$
(iii)	$p(1-p)$	$1-p+p^2$	2	$=1+p$
(iv)	p^2	$1-p^2$	2	$=1+p$
(v)	$(1-p)^2$	$p+p(1-p)$	2	$=2-p$
(vi)	$2p(1-p)$	$p^2+(1-p)^2$	2	≤ 2

The exceptional points form a countable dense subset of the unit interval. The set is countable because there are countably many polynomials in p with integer coefficients. It is dense because for any $0 \leq a < b \leq 1$, for sufficiently large n (viz., $n > \log_{1-p} b-a$), $\Pr(x) \leq (1-p)^n < b-a$ for all binary strings x of length n, therefore $a \leq \sum_{x\in A} \Pr(x) \leq b$ for some $A \subseteq \{0,1\}^n$.

Lemma 5. *Let $([0,1],\beta)$ be a greedy residual probability protocol with expectation E. If $(3-\sqrt{5})/2 \leq p < 1/2$, then*

1. *For $q \leq p$ or $1-p \leq q$, $E(q) < 2$.*
2. *For $p < q < 1-p$, $E(q) < (2-p)/(1-p+p^2)$.*

If $p = 1/2$, then $E(q) \leq 2$.

Proof. For $q \in [0,p] \cup [1-p,1]$, either $\beta(q) \in \{\text{H},\text{T}\}$ or $\beta(q)_1 \in \{\text{H},\text{T}\}$, thus the protocol takes at most one step with probability at least $1-p$. For $q \in (p,1-p)$, either $\beta(q)_0 = \text{H}$ and $\beta(q)_1 = (q-p)/(1-p)$ or $\beta(q)_0 = \text{T}$ and $\beta(q)_1 = q/(1-p)$. In the former case, $q < 1-p \leq 1-(1-p)^2$ so $\beta(q)_1 = (q-p)/(1-p) < p$. In the latter case, $(1-p)^2 \leq p < q$ so $\beta(q)_1 = q/(1-p) > 1-p$. In either case, the protocol reenters the region $[0,p] \cup [1-p,1]$ in the next step. Thus $E(q)$ is bounded by M for $q \in [0,p] \cup [1-p,1]$ and by N for $(p,1-p)$, where M and N satisfy the system of recurrences

$$\begin{aligned} M &= (1-p) + p(1+N) = 1 + pN \\ N &= p + (1-p)(1+M) = 1 + (1-p)M. \end{aligned} \tag{18}$$

The unique bounded solution is

$$M = \frac{1+p}{1-p+p^2} \qquad\qquad N = \frac{2-p}{1-p+p^2},$$

thus

$$E(q) \leq \begin{cases} \dfrac{1+p}{1-p+p^2} & \text{if } q \leq p \text{ or } 1-p \leq q \\ \dfrac{2-p}{1-p+p^2} & \text{if } p < q < 1-p. \end{cases}$$

In the case $q \leq p$ or $1-p \leq q$, the value is 2 for $p = 1/2$ and strictly less than 2 if $p < 1/2$. The inequality is also strict in the case $p < q < 1-p$ if $p < 1/2$, since it is governed by the system (18). □

We show that for $p < 1/2$, the function $E_{\rm opt}$ has a dense set of discontinuities on the unit interval, and the function is self-similar but for a discrete set of exceptions.

Lemma 6. *For all non-exceptional q, $E_{\rm opt}(q) \geq 1/(1-p)$, and for $p < q < 1-p$, $E_{\rm opt} \geq 2$.*

Proof. We will show in Lemma 8 that greedy is optimal on non-exceptional q, and non-exceptionality is preserved by greedy steps. Thus the optimal protocol is purely greedy on non-exceptional q. The remainder of the proof is similar to the proof of the corresponding inequalities (3.14) and (3.15) of [8].

The first inequality follows from the observation that a greedy protocol can do no better than to halt with probability $1-p$ in every step, giving the same expectation as a Bernoulli process with success probability $1-p$.

For the second, if $p < q < 1-p$, then after one greedy step, the residual probability is either $q' = q/(1-p) > 1-p$ or $q' = (q-p)/(1-p) < p$. In either case, by the previous argument, $E_{\rm opt}(q') \geq 1/(1-p)$. Thus

$$E_{\rm opt}(q) = 1 + (1-p)E_{\rm opt}(q') \geq 1 + (1-p)\frac{1}{1-p} = 2.$$

□

Theorem 5. *For $p < 1/2$, the function $E_{\rm opt}$ is everywhere discontinuous; that is, every open subinterval of the closed unit interval contains a discontinuity.*

Proof. The argument is very similar to one given in [8], with minor modifications to account for exceptional points.

It follows from Lemmas 5 and 6 that $E_{\rm opt}$ has discontinuities at p and $1-p$. By Lemma 6, all non-exceptional q approaching p from above have $E_{\rm opt}(q) \geq 2$; by Lemma 5, all non-exceptional q approaching p from below have $E_{\rm opt}(q) \leq (1+p)/(1-p+p^2) < 2$; and $E_{\rm opt}(p) = 1$.

Now we show that every nonempty open interval (a,b) contains a discontinuity. If the interval (a,b) is entirely contained in one of the three regions $(0,p)$, $(p,1-p)$, or $(1-p,1)$, then a greedy step maps the non-exceptional elements of (a,b) conformally to a larger subinterval. For example, if $(a,b) \subseteq (0,p)$, then

$$E_{\rm opt}(q) = 1 + pE_{\rm opt}(q/p)$$

for non-exceptional $a < q < b$, thus

$$E_{\rm opt}(q/p) = (E_{\rm opt}(q) - 1)/p$$

for $a/p < q/p < b/p$, so the non-exceptional elements of (a,b) are mapped conformally onto the interval $(a/p, b/p)$. But the length of this interval is $(b-a)/p$, thus we have produced a longer interval.

A similar argument holds if (a,b) is contained in one of the intervals $(p,1-p)$ or $(1-p,1)$. In each of these three cases, we can produce an interval of continuity that is longer than (a,b) by a factor of at least $1/(1-p)$. This process can be repeated at most $\log_{1-p}(b-a)$ steps before the interval must contain one of the discontinuities p or $1-p$. As the mappings were conformal on non-exceptional points, the original interval (a,b) must have contained a discontinuity. □

6 Algorithms

Throughout this section, as in the last, we assume that $(3-\sqrt{5})/2 \le p \le 1/2$.

Lemma 7. *For residual probability protocols, a greedy step is optimal at all but finitely many exceptional q.*

Proof. Suppose first that $p < 1/2$. By Lemma 5, $E_{\text{opt}} \le 2-\varepsilon$ for some $\varepsilon > 0$. Suppose we have a residual probability protocol that is not greedy at q for some $0 < q \le p$ or $1-p \le q < 1$. If the protocol generates an infinite computation path from q, then

$$E(q) \ge p + (1-p)(1+\frac{1}{1-p}) = 2.$$

This is the minimum possible expectation with at least one an infinite path if the protocol does not halt with probability at least $1-p$ in the first step. Truncating at depth k, the running time would be

$$2 - p^{k-1}(1-p)(k+\frac{1}{1-p}) = 2 - p^{k-1}(k(1-p)+1),$$

and this is greater than any $2-\varepsilon$ for sufficiently large k. By Lemma 5, any protocol that is not greedy in the first step and generates a computation path of length at least k cannot be optimal. But the only q that can generate computation trees of depth k or less are the exceptional q of degree at most k, and there are only finitely many of these.

If $p = 1/2$, the situation is even simpler. By Lemma 5, $E_{\text{opt}} \le 2$. In this case, however, any impatient protocol is greedy. If the protocol is not impatient at q, then all computation paths are of length at least 2. The only way this can be optimal is if q is exceptional of degree 2, and all computation paths are of length exactly 2. But according to Table 1, this is impossible: row (vi) collapses to row (ii) for $p = 1/2$, so there is no such optimal computation.

Now let us consider the case $p < q < 1-p$. Any strategy that is not greedy in the first step must take at least 2 steps in all instances; it cannot halt in one step with probability $1-p$, because that probability is too big to assign either H or T. If the protocol generates an infinite computation path from q, then it takes time at least

$$2 + p^2(2+\frac{1}{1-p}).$$

But N is less than this for $p \ge (3-\sqrt{5})/2$:

$$\frac{2-p}{1-p+p^2} \le 2 + p^2(2+\frac{1}{1-p}).$$

This can be shown by comparing derivatives. The derivative of the left-hand side is negative for all points greater than $2-\sqrt{3}$, and $2-\sqrt{3} < (3-\sqrt{5})/2 \le p$.

The derivative of the right-hand side is positive for all p. The inequality holds at $3/8$, where the values are $104/49$ and $401/160$, respectively, and $2-\sqrt{3} < 3/8 < (3-\sqrt{5})/2$.

As above, by Lemma 5, any protocol that is not greedy but generates a computation path of sufficient length k cannot be optimal. So if the optimal protocol is not greedy at q, then q must be exceptional of degree at most k. □

Lemma 8. *Assume* $(3-\sqrt{5})/2 \leq p \leq 1/2$. *At all non-exceptional points, greedy is globally optimal.*

Proof. By Lemma 7, the optimal local strategy at all but finitely many exceptional points is greedy. But it is not difficult to show that a greedy step preserves non-exceptionality, therefore for non-exceptional points, greedy is globally optimal as well. □

Theorem 6. *For* $p = 1/2$, $E_{\mathrm{opt}}(q) = 2$ *but for the following exceptional values:* $E_{\mathrm{opt}}(k/2^n) = (2^n-1)/2^{n-1}$, $k \leq 2^n$ *odd. Greedy is optimal for all* q.

Proof. Lemmas 5 and 7 establish that $E_{\mathrm{opt}}(q) \leq 2$ for all q and that $E_{\mathrm{opt}}(q) = 2$ for all nonexceptional q. Any non-greedy strategy takes at least two steps on all computation paths, thus greedy is optimal for all q. For the exceptional points mentioned in the statement of the theorem, it is easily checked inductively that the greedy strategy behaves as stated. Moreover, all exceptional points are of this form. □

6.1 An Approximation Algorithm

Were it not for the ambiguous region $(p, 1-p)$, we would be done. We could check in each step whether q is one of finitely many exceptional values; if so, obtain the optimal strategy by table lookup, and if not, take a greedy step. Note that this gives an optimal protocol for $p = 1/2$, as the ambiguous region is empty.

Unfortunately, for q in the ambiguous region $(p, 1-p)$, there are always two choices, and we do not know which will ultimately be the better choice. To approximate the globally optimal expectation E_{opt} to within any desired $\varepsilon > 0$, we will simulate all possible greedy choices down to a fixed depth k depending on ε.

Let d be a bound on the degree of those exceptional points for which a local greedy step is not optimal, as guaranteed by Lemma 7. Let G be the set of exceptional points of degree at most $d+k$. As G is a finite set, whenever $q \in G$ during the execution of the protocol, we can obtain the optimal local action by table lookup and take that action.

Otherwise, on input $q \notin G$, if q is not in the ambiguous region $(p, 1-p)$, we take the unique possible greedy step. This is optimal, by Lemma 7. If $q \in (p, 1-p)$, we have two greedy choices. We know that one of them is optimal, but we do not know which. In this case we simulate all possible greedy paths down to depth k. This involves branching when q is in the ambiguous region $(p, 1-p)$ to

simulate the two possible greedy steps. No greedy path ever encounters a $q \in G$ by choice of G, so we know that some greedy path is optimal down to depth k.

At depth k, we have several paths x that are currently being simulated. One of them is optimal. For each such x, let E_x be the expected time to halt before reaching the end of x, given that the path x is taken; that is, E_x is the expected length of a shortest path prefix-incomparable to x. Let $f_x(q) \in [0,1]$ be the residual probability after following path x if the computation has not halted by then. Then

$$\begin{aligned} E_{\text{opt}}(q) &= \min_x(E_x + \Pr(x) \cdot (k + E_{\text{opt}}(f_x(q)))) \\ &\geq \min_x(E_x + \Pr(x) \cdot k). \end{aligned}$$

But for any such x, continuing from x with a purely greedy strategy yields an expectation no worse than

$$E_x + \Pr(x) \cdot (k+2) \tag{19}$$

by Lemma 5, and

$$\begin{aligned} \min_x(E_x + \Pr(x) \cdot (k+2)) &\leq \min_x(E_x + \Pr(x) \cdot k) + 2(1-p)^k \\ &\leq E_{\text{opt}}(q) + \varepsilon, \end{aligned}$$

provided k is large enough that $(1-p)^k \leq \varepsilon/2$, that is, $k \geq \log_{1-p}(\varepsilon/2)$. Thus the greedy strategy x that minimizes (19) will be within ε of optimal.

6.2 Analysis

The algorithm constructs a tree with $2^{k/2}$ nodes in the worst case, where $k = \log_{1-p}(\varepsilon/2)$. It is $2^{k/2}$ and not 2^k because branching occurs at most once every two steps. The algorithm thus runs in time bounded by $2^{k/2} \leq (\varepsilon/2)^{1/\log(1-p)^2}$. The exponent $1/\log(1-p)^2$ ranges between $-.72$ and $-.5$ for p in the range $(3-\sqrt{5})/2 \leq p \leq 1/2$, thus the algorithm is better than linear in $1/\varepsilon$.

7 Conclusion

Several questions present themselves for further investigation.

Our analysis gives a worst-case time bound less than linear in $1/\varepsilon$, but empirical evidence suggests that the true time bound is exponentially better and that we actually achieve the optimal on all but a very sparse set. In the many experiments we have tried, the size of the set of candidate greedy paths x does not grow beyond two if demonstrably suboptimal paths are pruned along the way, and the algorithm invariably exits the loop with one candidate, which must be optimal.

The restriction $p \geq (3-\sqrt{5})/2$ was made to simplify many of the proofs, but it should be possible to eliminate it.

Most importantly, it would be nice to know whether the optimal protocol is computable for all rational p and q.

Acknowledgments. Much of this work was conducted while the author was visiting the University of Warsaw, the University of Copenhagen, and the University of Aarhus during April and May 2009. Special thanks to Mikołaj Bojańczyk, Fritz Henglein, Bobby Kleinberg, Eryk Kopczyński, Peter Bro Miltersen, Damian Niwiński, Nicholas Ruozzi, Michael Schwartzbach, Anna Talarczyk, Paweł Urzyczyn, Aaron Wagner, Anna Zdunik, and the anonymous referees. This work was supported by NSF grant CCF-0635028.

References

1. Adamek, J.: Foundations of Coding. Wiley (1991)
2. Adámek, J., Lücke, D., Milius, S.: Recursive coalgebras of finitary functors. Theoretical Informatics and Applications 41, 447–462 (2007)
3. Adámek, J., Milius, S., Velebil, J.: Elgot algebras. Log. Methods Comput. Sci. 2(5:4), 1–31 (2006)
4. Blum, M.: Independent unbiased coin flips from a correlated biased source: a finite state Markov chain. Combinatorica 6(2), 97–108 (1986)
5. Capretta, V., Uustalu, T., Vene, V.: Corecursive algebras: A study of general structured corecursion. In: Oliveira, M.V.M., Woodcock, J. (eds.) SBMF 2009. LNCS, vol. 5902, pp. 84–100. Springer, Heidelberg (2009)
6. Cover, T.M., Thomas, J.A.: Elements of Information Theory. Wiley-Interscience (August 1991)
7. Elias, P.: The efficient construction of an unbiased random sequence. Ann. Math. Stat. 43(3), 865–870 (1992)
8. Kozen, D.: Coinductive proof principles for stochastic processes. Logical Methods in Computer Science 3(4:8) (2007)
9. von Neumann, J.: Various techniques used in connection with random digits. In: Forsythe, G.E. (ed.) National Bureau of Standards. Applied Math Series, vol. 12, pp. 36–38 (1951); reprinted in: von Neumann's Collected Works, vol. 5, pp. 768–770. Pergamon Press (1963)
10. Nisan, N., Ta-shma, A.: Extracting randomness: A survey and new constructions. Journal of Computer and System Sciences 58, 148–173 (1999)
11. Nisan, N., Zuckerman, D.: Randomness is linear in space. Journal of Computer and System Sciences 52, 43–52 (1996)
12. Park, D.M.R.: Fixpoint induction and proofs of program properties. In: Meltzer, B., Michie, D. (eds.) Machine Intelligence, vol. 5, pp. 59–78. Edinburgh University Press (1969)
13. Peres, Y.: Iterating von Neumann's procedure for extracting random bits. Ann. Stat. 20(1), 590–597 (1992)
14. Srinivasan, A., Zuckerman, D.: Computing with very weak random sources. SIAM J. Computing 28, 264–275 (1999)
15. Ta-shma, A.: On extracting randomness from weak random sources. In: Proc. 28th ACM Symp. Theory of Computing, pp. 276–285 (1996)