Simulating a dice with a dice

Benoît Kloeckner

October 29, 2008

In this note we solve the following problem: how can one simulate with the few possible rolls a k-sided dice when one only has a p-sided one ? The case $k = 4$ was first asked to me by Philippe Bruhat in order to program various randomness in befunge languages. In this fancy programming languages the source code lie on a torus and the only randomness available is the uniform choice of one of the four cardinal directions. It is also a frequent problem for rolists. As a less futile motivation some close problems were previously adressed : the simulation of a fair k-sided dice with several biased coins, in [FIN$^+$93] and [IM94] and the simulation of a fair coin with a biaised dice in [JJSH00]. After writting this note the paper [KY76], which solves a similar problem, was pointed out to us.

The first section is devoted to definitions and a criterion on k and p under which the simulation can be achieved in bounded time. In the second section we construct for all k and p a greedy simulation process. The third and last section is devoted to the proof of our main result, stated below. Note that only very elementary methods are needed.

Theorem A *The greedy simulation process is optimal for all values of p and k.*

1 Definitions

In all the paper, k and p denotes positive integers. We denote by A the set $\{1, \dots, p\}$, which shall be our alphabet. The set of words of finite length with letters in A is denoted by A^*, The set of infinite words by $A^{\mathbb{N}}$. The length of a finite word ω is denoted by $|\omega|$. Most of the time, we use greek letters for finite words and latin letter for infinite ones.

The uniform probability law on $\{1, \dots, k\}$ is denoted by $\mathscr{U}_k$; the probability law of a countable family of independent uniform random variables with value in A is denoted by $\mathscr{U}_p^{\otimes\mathbb{N}}$.

Given two words ω, ψ, we denote by $\omega\psi$ their concatenation. Note that while ψ can be finite or infinite, ω must be finite. We say that ω is a *prefix* of ϕ if there is a ψ such that $\phi = \omega\psi$. We denote by $\omega A^{\mathbb{N}}$ the set of infinite words having ω as a prefix.

We now give our definition of a simulation process.

Definition 1 *Let $f : A^{\mathbb{N}} \longrightarrow \{1, \ldots, k\}$ be a map. Define a* deciding word *for f to be any finite word ω such that f is constant on $\omega A^{\mathbb{N}}$ and for any prefix μ of ω, f is not constant on $\mu A^{\mathbb{N}}$. We say that f is a* simulation process *of a k-sided dice by a p-sided one, or briefly a (k,p)-simulation process, if f satisfies the following properties:*

1. *the image probability measure is uniform :*

$$f_* \mathscr{U}_p^{\otimes \mathbb{N}} = \mathscr{U}_k$$

2. *almost all word x in $A^{\mathbb{N}}$ has a prefix that is a deciding word.*

Such a deciding prefix must be unique, and its length is the stopping time *of the word x relative to f and is denoted by $T_f(x)$.*

Under condition 2 the process will almost surely stop in finite time; moreover it ensures that f is measurable. We can gather the deciding words in k possibly infinite collections $(\omega_i^\ell)_{i \in I_\ell}$ $(1 \leqslant \ell \leqslant k)$ satisfying (up to a null set)

$$f^{-1}(\ell) = \bigcup_{i \in I_\ell} \omega_i^\ell A^{\mathbb{N}}.$$

The balancing condition 1 asserts that the process simulates a fair dice. It can be reformulated as follows:

$$\sum_{i \in I_\ell} \frac{1}{p^{|\omega_i^\ell|}} = \frac{1}{k} \quad \forall \ell. \tag{1}$$

We are concerned with the stopping time of f which, while almost surely finite, do not need be uniformly (almost surely) bounded.

Definition 2 *Let X be a random variable with value in $\{1, \ldots, p\}^{\mathbb{N}}$ and law $\mathscr{U}_p^{\otimes \mathbb{N}}$. Given two (k,p)-simulation processes f and g, one say that f is* better *than g if $T_f(X)$ is stochastically dominated by $T_g(X)$.*

A (k,p)-simulation process f is said to be optimal *if it is better than any other.*

Last, a (k,p)-simulation process f is said to have bounded stopping time if there exists some integer T such that $T_f(X) \leqslant T$ almost surely. The least such T is then denoted by $T(f)$ and called the time bound *of f. If f does not have bounded stopping time, we write $T(f) = \infty$.*

Recall that if X and Y are two real valued random variables, one says that X is *stochastically dominated* by Y if for any $t \in \mathbb{R}$ one has

$$\mathbb{P}(Y \leqslant t) \leqslant \mathbb{P}(X \leqslant t)$$

thus the condition of optimality given above is the strongest we can reasonably ask for. In particular, it implies that the average time of decision is minimal.

Now we notice a conditions under which there is a (k, p)-simulation process having bounded time.

Lemma 3 *There exists a (k, p)-simulation process having bounded time if and only if every prime number dividing k also divides p. When this condition holds, let T_0 be the least integer such that k divides p^{T_0}. Then if f is a (k, p)-simulation process having bounded time, one has $T(f) \geqslant T_0$ and this bound is sharp.*

The proof is straightforward using equation (1).

2 Construction of the greedy simulation process

We now turn to the construction of the greedy simulation process, which will be denoted by g. We have to construct k collections of words $(\omega_i^\ell)_{i \in I_\ell}$ (on p letters) no one of them being a prefix of another and satisfying (1).

We proceed recursively. Start with k empty families and let Ω_1 be the set of one letter words : $\Omega_1 = \{1, \ldots, p\}$. Add to the first family I_1 some elements of Ω_1 while it does not imply $\sum_{i \in I_1} (1/p)^{|\omega_i^1|} > 1/k$. Proceed similarly with each other $\ell \leqslant k$. Denote by Ω_1' the set of remaining words of Ω_1.

Once constructed Ω_n', let Ω_{n+1} be the set of words of length $n+1$ having a prefix in Ω_n'. Add to each family words of Ω_{n+1} while it does not imply $\sum_{i \in I_\ell} (1/p)^{|\omega_i^\ell|} > 1/k$. Denote by Ω_{n+1}' the set of remaining words of Ω_{n+1}.

First note that at each step we add exactly the same number of words in each family, namely $\lfloor \log_p k' \rfloor$ where (at step n)

$$\frac{1}{k'} = \frac{1}{k} - \sum_{i \in I_\ell} \frac{1}{p^{|\omega_i^\ell|}}$$

is the "remaining room".

Second, if we must stop at one step, g has bounded time. If we never stop, each finite word becomes at one time or another a prefix of a word of one of the families. In both cases we get condition (1).

We can then state the following.

Proposition 4 *The application g that maps each $\omega_i^\ell A^{\mathbb{N}}$ to ℓ is a (k,p)-simulation process, called the* greedy simulation process.

Note that the choice of the words added at each step to the families allows us to construct several different such simulation processes, but they all are similar and we speak of "the" greedy simulation process.

Let us give a concrete exemple. If $p = 4$ and $k = 5$, the greedy simulation process leads to proceed as follow. Roll two times the 4-sided dice; if the rolls are :

1. $(1, 1)$, $(1, 2)$ or $(1, 3)$, the result is 1,
2. $(1, 4)$, $(2, 1)$ or $(2, 2)$, the result is 2,
3. $(2, 3)$, $(2, 4)$ or $(3, 1)$, the result is 3,
4. $(3, 2)$, $(3, 3)$ or $(3, 4)$, the result is 4,
5. $(4, 1)$, $(4, 2)$ or $(4, 3)$, the result is 5,
6. $(4, 4)$, reroll two times and reuse this table.

At the end of this paper, an appendix presents a source code in befunge-93 for the greedy simulation process (see the specifications in [Pre98]).

3 Proof of optimality

We now prove Theorem A.

Let g be the greedy simulation process and f be another (k, p)-simulation process. Let X be a random variable with value on $A^{\mathbb{N}}$ and law $\mathscr{U}_p^{\otimes\mathbb{N}}$. We have to prove that for any positive integer N,

$$\mathbb{P}(T_f(X) \leqslant N) \leqslant \mathbb{P}(T_g(X) \leqslant N).$$

Suppose that there exists some positive integer N such that

$$\mathbb{P}(T_f(X) \leqslant N) > \mathbb{P}(T_g(X) \leqslant N)$$

then there exists some $\ell \in \{1, \dots, k\}$ such that

$$\mathbb{P}(T_f(X) \leqslant N, f(X) = \ell) > \mathbb{P}(T_g(X) \leqslant N, f(X) = \ell).$$

But the two sides of this inequality are multiples of $1/p^N$, and by construction the left hand side is the greatest multiple of $1/p^N$ lesser or equal to $1/k$. Thus $\mathbb{P}(T_f(X) \leqslant N, f(X)) > 1/k$, a contradiction with the definition of a simulation process.

Last, we can deduce from Theorem A the following.

Corollary B *If there exists some (k, p)-simulation process having bounded time, then the greedy simulation process has bounded time and minimal time bound.*

Appendix : a befunge source code for the greedy $(5,4)$-simulation process

>	#	v	?		#	v	?	^		
						>	>		3	@
			>	#	v	?	v			
					>	>	>		4	@
					>	v				
		>	#	v	?	>			5	@
v				<	>	^				
						>	v			
			>	#	v	?	>		1	@
						>	^			
					>		>	>	2	@

To read the code, start from the upper left cell and start reading right. The boundary is glued so that the code lies on a torus. The following code character are used here:

^:	start moving up,	#:	skip next cell,
v:	start moving down,	@:	end program,
>:	start moving right,	0-9:	push this number on the stack,
<:	start moving left,	?:	start moving in a random cardinal direction

References

[FIN$^+$93] David Feldman, Russell Impagliazzo, Moni Naor, Noam Nisan, Steven Rudich, and Adi Shamir. On dice and coins: models of computation for random generation. *Inform. and Comput.*, 104(2):159–174, 1993.

[IM94] Toshiya Itoh and Takahiro Mochiduki. Simulating fair dice with a small set of rationally biased coins. *Sūrikaisekikenkyūsho Kōkyūroku*, (871):130–137, 1994. Computational complexity theory (Japanese) (Kyoto, 1994).

[JJSH00] Ari Juels, Markus Jakobsson, Elizabeth Shriver, and Bruce K. Hillyer. How to turn loaded dice into fair coins. *IEEE Trans. Inform. Theory*, 46(3):911–921, 2000.

[KY76] Donald E. Knuth and Andrew C. Yao. The complexity of nonuniform random number generation. In *Algorithms and complexity (Proc. Sympos., Carnegie-Mellon Univ., Pittsburgh, Pa., 1976)*, pages 357–428. Academic Press, New York, 1976.

[Pre98] Chris Pressey. Funge-98 final specification . `http://catseye.mine.nu:8080/projects/funge98/`, 1998.