CHAPTER 21

THE ALIAS METHOD FOR SAMPLING DISCRETE DISTRIBUTIONS

Chris Wyman
NVIDIA

ABSTRACT

The *alias method* is a well-known algorithm for constant-time sampling from arbitrary, discrete probability distributions that relies on a simple precomputed lookup table. We found many have never learned about this method, so we briefly introduce the concept and show that such lookup tables can easily be generated.

21.1 INTRODUCTION

When rendering, we often need to sample discrete probability distributions. For example, we may want to sample from an arbitrary environment map proportional to the incoming intensity. One widely used technique inverts the cumulative distribution function [2]. This is ideal for analytically invertible functions but requires an $O(\log N)$ binary search when inverting tabulated distributions like our environment map.

If you can afford to precompute a lookup table, the alias method [5] provides a simple, constant-time algorithm for sampling from arbitrary discrete distributions. However, precomputation makes it less desirable if taking just a few samples from a distribution or if the distribution continually changes.

21.2 BASIC INTUITION

Let's start by reviewing how we often generate samples without the alias method. Imagine a discrete distribution, as in Figure 21-1. Bins have arbitrary real-valued weights, and we want to randomly sample proportional to their relative weights.

Using traditional cumulative distribution function (CDF) inversion, we first build a discrete CDF, e.g., in Figure 21-1. We pick a uniform random number

A. Marrs, P. Shirley, I. Wald (eds.), *Ray Tracing Gems II*, https://doi.org/10.1007/978-1-4842-7185-8_21

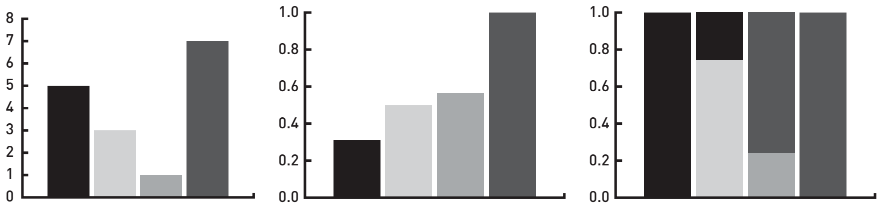

Figure 21-1. *Left: a simple discrete distribution. Center: its discrete cumulative distribution function. Right: an alias table for the distribution.*

$\xi \in [0 \ldots 1)$, place it on the y-axis, and select as our sample the first bin it falls into (moving left to right). Our example has weights summing to 16, so we choose the black bin if $\xi \leq \frac{5}{16}$, the light gray bin if $\frac{5}{16} < \xi \leq \frac{1}{2}$, etc. Here, finding a bin corresponding to ξ is straightforward, with at most three comparisons. But if sampling from a light probe, linearly searching each bin is costly. Even a fast binary search uses dozens of comparisons and walks memory incoherently.

Essentially, table inversion picks a random value then walks the tabulated CDF asking, "do you correspond to my sample?" Optimization involves designing a good search.

Instead, the alias method asks, "wouldn't it be easier with uniform weights?" You could easily pick a random bin without a search. Clearly, our weights are unequal. But imagine sampling bins as if uniformly weighted, say using the average weight. We would oversample those weighted below average and undersample those weighted above average.

Perhaps we could correct this error after selection? If we pick an oversampled bin, we could sometimes switch to an undersampled bin. This seems feasible; if sampling proportional to average weight, there's an equal amount of under- and oversampling.

Figure 21-1 also shows a table reorganizing weights from our distribution into equal-sized bins. We call this an *alias table* (though that term is a bit overloaded).

To sample, first choose a bin uniformly. Use random $\xi \in [0 \ldots 1)$ to check whether to switch due to oversampling, based on threshold τ. If $\xi \geq \tau$, switch to a different bin.