

Simulating an n -Sided Coin with One Fair Coin

Or How to Choose a Dinner [incomplete]

Ross Girshick*

1 Introduction

Sometimes life is hard. Imagine that you have to make a decision between two options, but you're paralyzed by ambivalence. Lucky for you, never was there a problem that money could not solve. Faced with such a situation, naturally you would remove a perfectly fair coin from your pocket and, in exactly one toss, let it decide your fate. But sometimes life is *really* hard. What if you have more than two options and you're still ambivalent? By extension you'd want to use a coin to decide between the $n > 2$ options with uniform probability. Is it possible to do this for all values of $n > 2$ with exactly one fair coin? If so, what is the asymptotic performance (in terms of number of tosses) with respect to the number of options? Here we present one algorithm for deciding between n equally weighted options and prove that the expected number of coin tosses to make a decision is $\Theta(\log n)$.

2 The N-Choose-Dinner Algorithm

Algorithm 1 N-Choose-Dinner

```
1:  $l \leftarrow \lceil \log n \rceil$ 
2: Fix a permutation of the  $n$  options:  $\pi \leftarrow (o_0, \dots, o_{n-1})$ 
3: for option  $o_i$  in  $\pi$  do
4:    $code_i \leftarrow i$  left zero-padded to length  $l$  (e.g., if  $l = 3$  and  $i = 1$ , then  $code \leftarrow 001$ )
5:    $code_i \leftarrow code_i$  with 0 replaced by H and 1 by T (e.g.,  $code \leftarrow HHT$ )
6:   /*  $code_i$  represents the heads-tails sequence required to select option  $o_i$ . */
7: end for
8: while true do
9:    $code \leftarrow$  sequence of  $l$  fair coin tosses (e.g., THT)
10:  if  $\exists choice \in \{0, \dots, n-1\}$  such that  $code_{choice} = code$  then
11:    return  $o_{choice}$ 
12:  end if
13: end while
```

3 Expected Number of Tosses

Algorithm **N-Choose-Dinner** may never terminate, however, it *almost always* will. I.e., the probability of terminating goes to 1 as $n \rightarrow \infty$.

Figure 1 illustrates the expected number of tosses, $E[X_n]$. It's interesting to note that $E[X_n]$ reaches a local minimum when $n = 2^x$ and a local maximum when $n = 1 + 2^y$ for any positive integers x, y .

*rbg@cs.uchicago.edu

3.1 Algebraic Analysis and Asymptotic Behavior

3.2 Geometric Distribution Analysis

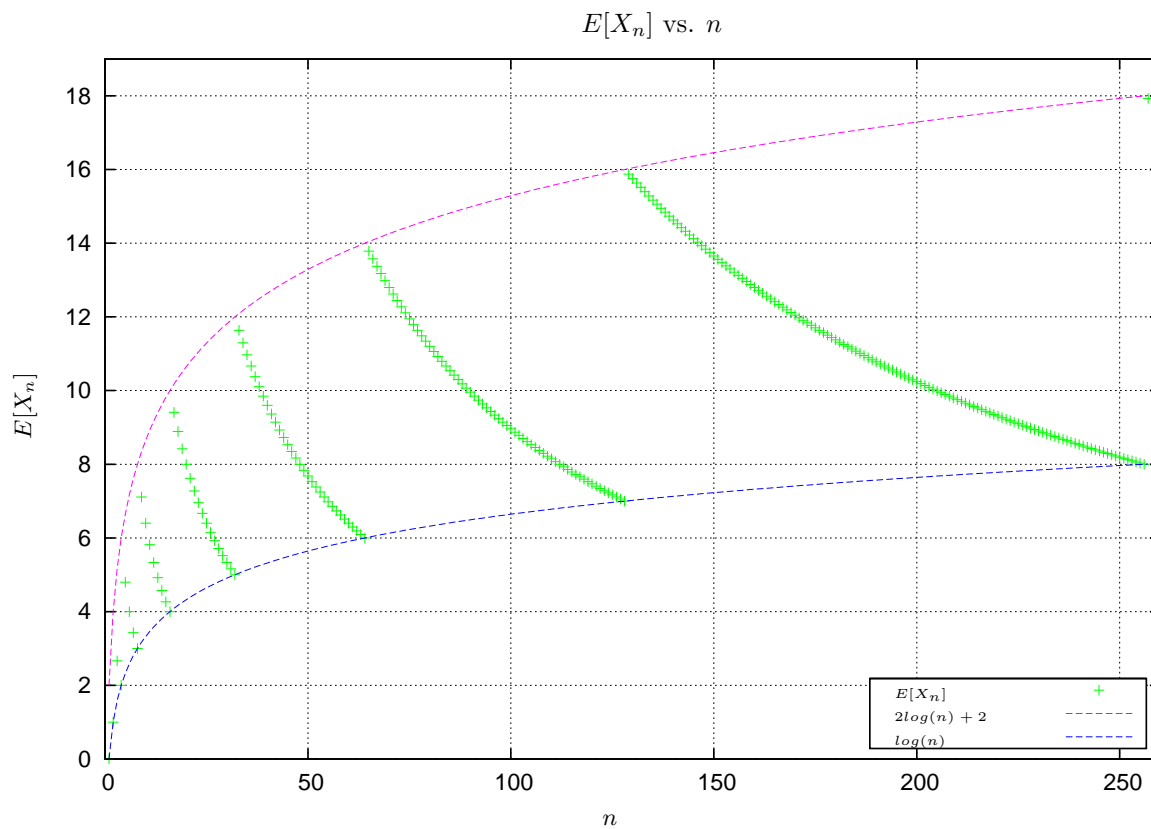


Figure 1: Expected number of coin tosses for deciding amongst n options.

4 Optimizations