Design and Analysis of Algorithms: Lecture 6

Ben Chaplin

Contents

1 Randomization

A **randomized** or **probabilistic** algorithm makes some decisions based on a random number generated at runtime. On the same input, a randomized algorithm may:

- return different outputs
- run differently, but return the same output

Definition. Monte Carlo algorithms are probably fast, but may not be correct.

Definition. Las Vegas algorithms are certainly fast, but only probably correct.

Definition. Atlantic City algorithms are both probably fast and probably correct.

1.1 Matrix Products

Example. Consider the problem of verifying an $n \times n$ matrix product. The standard method of multiplying matricies gives a runtime of $O(n^3)$. Better algorithms have been developed, however they only offer runtimes of around $O(n^{\log 7})$. We seek to use an $O(n^2)$ randomized algorithm where:

- if $A \times B = C$, then $P(\text{returns } \textbf{True}) = 1$
- else, if $A \times B \neq C$, then $P(\text{returns } \textbf{True}) \leq \frac{1}{2}$

Then, running the algorithm on matrices (A, B, C) k times, and getting a result of **True** every time gives a probability of incorrectness:

$$P(\text{incorrect}) \leq \frac{1}{2^k}$$

In other words, we can ensure a given level of certainty by running the algorithm a constant number of times. Thus, the overall runtime is still $O(n^2)$.

1.2 Frievold's algorithm

Algorithm 1 Frievold's algorithm

Input

A, B, C where each is an $n \times n$ matrix

Output

{True, False} whether $AB = C$

```
r = random binary vector of length n
if A(Br) = Cr then
    return True
else
    return False
end if
```

1.3 Runtime Analysis

The runtime of Frievold's is:

- $O(n)$ to choose a random vector
- $O(n^2)$ to compute Br, $O(n^2)$ to compute $A(Br)$ and $O(n^2)$ to compute Cr
- $O(n)$ to compare $A(Br)$ to Cr

Overall: $O(n^2)$.

1.4 Correctness

Claim 1. *If $AB = C$, Frievold's always returns **True**.*

Proof. Assume $AB = C$, and r is some vector of length n.

$$\begin{aligned} Cr &= (AB)r && \text{by matrix associativity} \\ &= A(Br) \end{aligned}$$

□

Claim 2. *If $AB \neq C$, $P(\textbf{True}) = \frac{1}{2}$.*

Proof. Let $D = AB - C$. Suppose Frievold's generates a random vector r for which it returns **True**. Then:

$$\begin{aligned} Dr &= (AB - C)r \\ &= (AB)r - Cr \\ &= A(Br) - Cr \\ &= 0 \end{aligned}$$

$D = AB - C \neq 0$, so we can take $i, j \in \{1, \ldots, n\}$ such that $D_{ij} = 1$. Let v be the vector:

$$v[k] = \begin{cases} 0 & \text{if } k \neq j \\ 1 & \text{if } k = j \end{cases}$$

Notice that $Dv \neq 0$, because its j^{th} entry will be 1.

Let $r' = r + v$. Then:

$$\begin{aligned} Dr' &= D(r + v) \\ &= 0 + Dv \\ &\neq 0 \end{aligned}$$

If $Dr' \neq 0$, $(AB - C)r' = ABr' - Cr' \neq 0$, so Frievold's would return **False**. This defines a one-to-one mapping between vectors r that make Frievold's return **True**, and r' that make Frievold's return **False**.

Therefore, Frievold's returns **True** for half of all possible binary vectors of length n, and **False** for the rest. $P(\textbf{True}) = \frac{1}{2}$. □