Discrete-Event Simulation:

A First Course

Section 2.3: Monte Carlo Simulation

Section 2.3: Monte Carlo Simulation

- With *Empirical Probability*, we perform an experiment many times n and count the number of occurrences n_a of an event $\mathcal{A}$
  - The *relative frequency* of occurrence of event $\mathcal{A}$ is n_a/n
  - The *frequency theory of probability* asserts that the relative frequency converges as $n \longrightarrow \infty$

$$\Pr(\mathcal{A}) = \lim_{n \longrightarrow \infty} \frac{n_a}{n}$$

- *Axiomatic Probability* is a formal, set-theoretic approach
  - Mathematically construct the sample space and calculate the number of events $\mathcal{A}$
- The two are complementary!

Example 2.3.1

- Roll two dice and observe the up faces

$(1,1)$	$(1,2)$	$(1,3)$	$(1,4)$	$(1,5)$	$(1,6)$
$(2,1)$	$(2,2)$	$(2,3)$	$(2,4)$	$(2,5)$	$(2,6)$
$(3,1)$	$(3,2)$	$(3,3)$	$(3,4)$	$(3,5)$	$(3,6)$
$(4,1)$	$(4,2)$	$(4,3)$	$(4,4)$	$(4,5)$	$(4,6)$
$(5,1)$	$(5,2)$	$(5,3)$	$(5,4)$	$(5,5)$	$(5,6)$
$(6,1)$	$(6,2)$	$(6,3)$	$(6,4)$	$(6,5)$	$(6,6)$

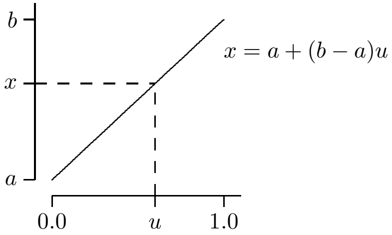
- If the two up faces are summed, an integer-valued random variable, say X, is defined with possible values 2 through 12 inclusive

sum, x :	2	3	4	5	6	7	8	9	10	11	12
$\Pr(X = x)$:	$\frac{1}{36}$	$\frac{2}{36}$	$\frac{3}{36}$	$\frac{4}{36}$	$\frac{5}{36}$	$\frac{6}{36}$	$\frac{5}{36}$	$\frac{4}{36}$	$\frac{3}{36}$	$\frac{2}{36}$	$\frac{1}{36}$

- $\Pr(X = 7)$ could be estimated by replicating the experiment many times and calculating the relative frequency of occurrence of 7's

Random Variates

- A *Random Variate* is an algorithmically generated realization of a random variable
- u = `Random()` generates a *Uniform*(0, 1) random variate
- How can we generate a *Uniform*(a, b) variate?



Generating a Uniform Random Variate

```
double Uniform(double a, double b)      /* use a < b */ {
    return (a + (b - a) * Random());
}
```

Equilikely Random Variates

- *Uniform*$(0, 1)$ random variates can also be used to generate an *Equilikely*(a, b) random variate

$$
\begin{aligned}
0 < u < 1 &\iff 0 < (b - a + 1)u < b - a + 1 \\
&\iff 0 \le \lfloor (b - a + 1)u \rfloor \le b - a \\
&\iff a \le a + \lfloor (b - a + 1)u \rfloor \le b \\
&\iff a \le x \le b
\end{aligned}
$$

- Specifically, $x = a + \lfloor (b - a + 1)\, u \rfloor$

Generating an Equilikely Random Variate

```
long Equilikely(long a, long b)     /* use a < b */ {
    return (a + (long) ((b - a + 1) * Random()));
}
```

Examples

- **Example 2.3.3** To generate a random variate x that simulates rolling two fair dice and summing the resulting up faces, use
  `x = Equilikely(1, 6) + Equilikely(1, 6);`
  Note that this is *not* equivalent to
  `x = Equilikely(2, 12);`
- **Example 2.3.4** To select an element x at random from the array $a[0], a[1], \ldots, a[n-1]$ use
  `i = Equilikely(0, n - 1); x = a[i];`

Galileo's Dice

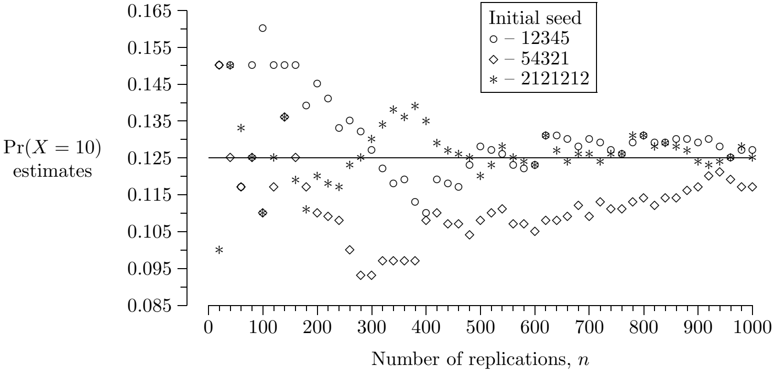
- If three fair dice are rolled, which sum is more likely, a 9 or a 10?
  - There are $6^3 = 216$ possible outcomes

$$\Pr(X = 9) = \frac{25}{216} \cong 0.116 \qquad \text{and} \qquad \Pr(X = 10) = \frac{27}{216} = 0.125$$

- Program `galileo` calculates the probability of each possible sum between 3 and 18
- The drawback of Monte Carlo simulation is that it only produces an estimate
  - Larger n does not guarantee a more accurate estimate

Example 2.3.6

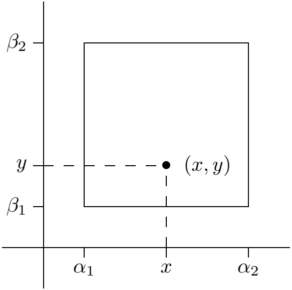
- Frequency probability estimates converge slowly and somewhat erratically



- You should *always* run a Monte Carlo simulation with multiple initial seeds

Geometric Applications

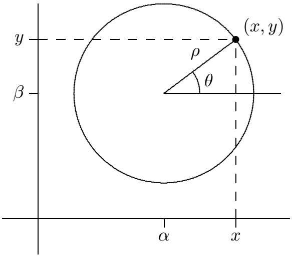
- Generate a point at random inside a rectangle with opposite corners at (α_1, β_1) and (α_2, β_2)




```
x = Uniform(α1, α2); y = Uniform(β1, β2);
```

Geometric Applications

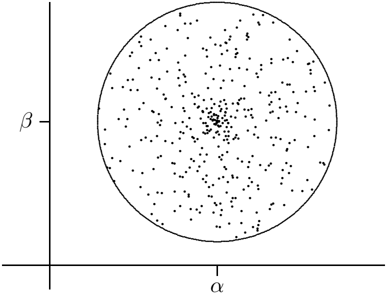
- Generate a point (x, y) at random on the circumference of a circle with radius ρ and center (α, β)



θ = Uniform($-\pi$, π); x = α + ρ * cos(θ); y = β + ρ * sin(θ);

Example 2.3.8

- Generate a point (x, y) at random *interior* to the circle of radius ρ centered at (α, β)



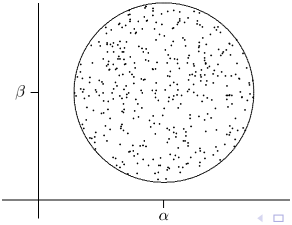
θ = Uniform($-\pi$, π); r = Uniform(0, ρ);
INCORRECT! $x = \alpha + r *$ cos(θ); $y = \beta + r *$ sin(θ);

Acceptance/Rejection

- Generate a point at random within a circumscribed square and then either accept or reject the point

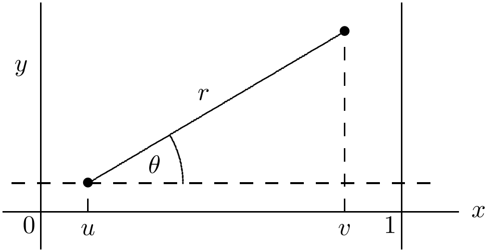
Generating a Random Point

```
do {
    x = Uniform(-ρ, ρ);
    y = Uniform(-ρ, ρ); } while (x * x + y * y >= ρ * ρ);
x = α + x;
y = β + y;
return (x, y);
```



Buffon's Needle

- Suppose that an infinite family of infinitely long vertical lines are spaced one unit apart in the (x, y) plane. If a needle of length $r > 0$ is dropped at random onto the plane, what is the probability that it will land crossing at least one line?



- u is the x-coordinate of the left-hand endpoint
- v is the x-coordinate of the right-hand endpoint, $v = u + r\cos\theta$
- The needle crosses at least one line if and only if $v > 1$

Program buffon

- Program `buffon` is a Monte Carlo simulation
  - The random number library can be used to automatically generate an initial seed

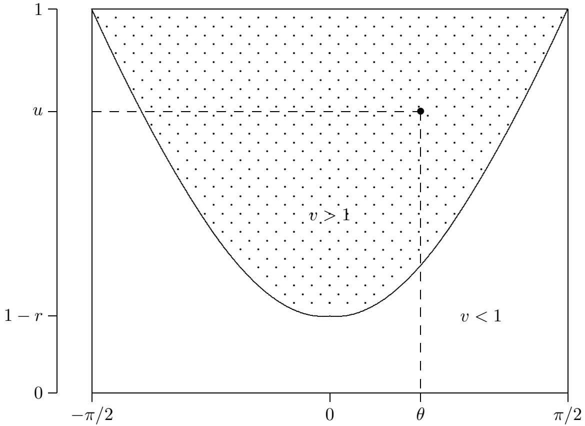
Random Seeding

```
PutSeed(-1);     /* any negative integer will do */
GetSeed(&seed);     /* trap the value of the initial seed */
     .
     .
     .
printf("with an initial seed of %ld", seed);
```

- Inspection of the program `buffon` illustrates how to solve the problem axiomatically

Axiomatic Approach to Buffon's Needle

- "Dropped at random" is interpreted (modeled) to mean that u and θ are independent *Uniform*$(0, 1)$ and *Uniform*$(-\pi/2, \pi/2)$ r.v.s



Axiomatic Approach to Buffon's Needle

- The shaded region has a curved boundary defined by the equation $u = 1 - r\cos(\theta)$
- If $0 < r \leq 1$, the area of the shaded region is

$$\pi - \int_{-\pi/2}^{\pi/2} \left(1 - r\cos\theta\right) d\theta = r\int_{-\pi/2}^{\pi/2} \cos\theta \, d\theta = \cdots = 2r$$

- Therefore, because the area of the rectangle is π the probability that the needle will cross at least one line is $2r/\pi$

