Computational Methods Assignment 3 - Random Numbers

Martik Aghajanian

1 Question 1

The random number generator used for this was found in *Numerical Recipes* page 342, which is a *struct* called *Ran*. This was selected because it is not based on any kind of linear congruential generator, and has a period of $3.138 \times 10^{57} \approx 2^{191} \approx 10^{57}$ which is neither too small to produce feasibly "psuedorandom" numbers nor too large so to be "over-engineered". It combines a 64bit-XORshift method ($x_1 = 17$, $x_2 = 31$, $x_3 = 8$), and 64-bit Multiply-with-carry (MWC) method with base $b = 32$ and multiplier $a = 4294957665$. This combination of unrelated methods essentially makes the "randomness" higher, since they do not share a state throughout the generation. The method works using three updatable integers u_i, v_i, w_i, which are independently updated via unrelated generation methods and combined to give results. For each call of the function the v_i are updated through the 64-bit XOR shift, whilst the w_i parameters are updated via the MWC method. The u_i vary with each call through $u_i \rightarrow u_{i+1} = au_i + b$ and then this is used to produce a temporary value h which is obtained through a different 64-bit XORshift method with $x'_1 = 21, x'_2 = 35, x'_3 = 4$. These three random numbers generated separately and independently are combined to output the used random number $(h + v_i)\text{XOR}w_i$, and makes this generator reliable.

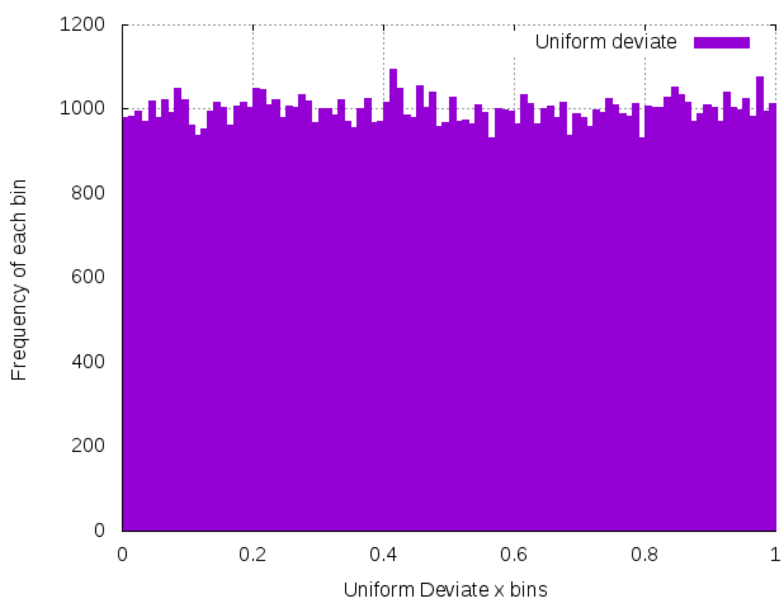

Figure 1: Histogram of uniform deviate x between $0.0 \leq x < 1.0$ plotted frequency bins, from Ran random number generator with seed 42.

Figure 1 shows the distribution of the double precision floating point variable uniform deviate portrayed in a histogram.

2 Question 2

To obtain the correct transformation $y(x)$ on uniform deviate $x \in [0,1]$ with probability distribution $\text{pdf}(x) = U(x)$ (where U is the uniform distribution) such that the corresponding values follow a probability distribution function $\text{pdf}(y) = \frac{1}{2}\sin(x)$, the cummulative distribution function needs to be calculated and inverted. For this specific case this can be derived through:

$$\begin{aligned}
\text{pdf}(y)|\text{d}y| &= \frac{1}{2}\sin(y)|\text{d}y| = \text{pdf}(x)|\text{d}x| = U(x)|\text{d}x| \\
\text{pdf}(y) &= \frac{1}{2}\sin(y) = \frac{|\text{d}x|}{|\text{d}y|} \implies \text{cdf}(y) = \int_0^y \text{d}y'\frac{1}{2}\sin(y') = \left[-\frac{1}{2}\cos(y')\right]_0^y = \frac{1}{2}(1-\cos(y)) \\
&= \int_0^y \frac{|\text{d}x|}{|\text{y}y'|}\text{d}y' = x(y) \\
&\implies y(x) = \cos^{-1}(1-2x)
\end{aligned}$$

which means that $0 \leq y \leq \pi$. This transformation is applie to 10^5 samples of a uniform deviate x and the resulting histogram is plotted in Figure 2.

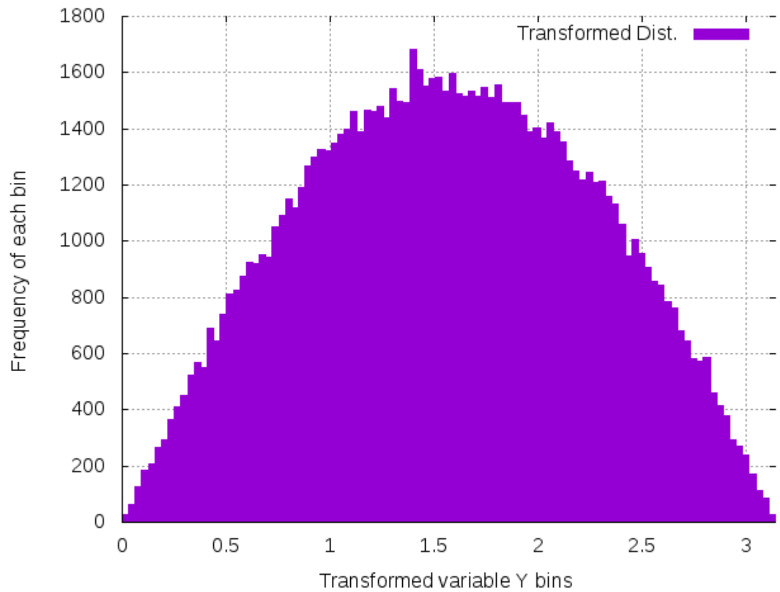

Figure 2: Histogram of transformed random variable $y(x) = \cos^{-1}(1-2x)$ between $0.0 \leq y < \pi$ plotted frequency bins, from a uniform deviate x generated from Ran with seed 42.

3 Question 3

For implementing the rejection method, a good comparison function $c(y)$ to obtain a $\text{pdf}(y) = \frac{2}{\pi}\sin^2(y)$ such that $c(y) \geq \text{pdf}(y), \forall y$, that was chose was $c(y) = \frac{2}{\pi}\sin(y)$ since:

$$c(y) - \text{pdf}(y) = \frac{2}{\pi}\sin(y)(1-\sin(y)) \geq 0 \ \forall y \in [0,\pi]$$

Using this comparison function the rejection method produced this probability distribution function from a uniform deviate x generated by Ran with seed 42. The resulting histogram is shown in Figure 3.

To time the generation of random numbers from arbitrary distributions for both the rejection method an the transformation method, the *ctime* library was used. This showed that, for the same seed, that the

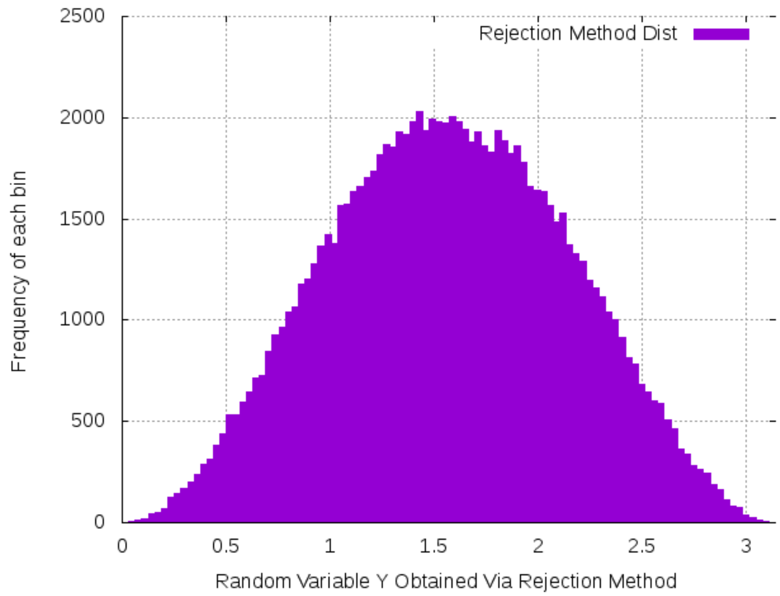

Figure 3: Histogram of random variable y between $0.0 \leq y < \pi$ following probability distribution $\mathrm{pdf}(y) = \frac{2}{\pi}\sin^2(y)$ using the rejection method. The comparison function used was $c(y) = \frac{2}{\pi}\sin(y)$ with samples obtained from a uniform deviate x generated from Ran with seed 42.

transformation method took approximately 0.005s whilst the rejection method took approximately $0.041ss$, which is 8.2 times slower. The theoretical ratio expected is determined by considering that for every random number generated by the rejection method, two calls to the uniform deviate random number generator have to be made, and the percentage of extra function calls that have to be made to account for the rejected entries, is the inverse of the fraction of accepted numbers. This fraction of accepted numbers is integral over the product of the probability of choosing a particular y from the comparison function $c(y)$, and the probability of being accepted at that particular y

$$\begin{aligned}\int_0^\pi \mathrm{d}y' \left(\left[\frac{c(y')}{4/\pi}\right] \left[\frac{\mathrm{pdf}(y')}{(y')}\right] \right) &= \frac{4}{\pi}\int_0^\pi \mathrm{d}y'\mathrm{pdf}(y') = \frac{4}{\pi}\frac{2}{\pi}\int_0^\pi \mathrm{d}y'\sin^2(y') \\ &= \frac{4}{\pi}\frac{1}{\pi}\int_0^{'} \mathrm{d}y'(1-\cos(2y')) = \frac{4}{\pi^2}\left[y' - \frac{\sin(2y')}{2}\right]_0^\pi \\ &= \frac{4}{\pi^2}\pi = \frac{4}{\pi}\end{aligned}$$

This is also the same as the ratio of the two areas covered by $c(y)$ and $\mathrm{pdf}(y)$. Hence the expected theoretical ratio is twice the inverse of this fraction, giving $\frac{4}{\pi} \times 2/approx 2.54$ which is around a factor of 3 less than what is seen in the simulation.