Computer simulation of physical systems I

Task V: Basic Monte-Carlo integration

1 Calculation of π

1.1 Rejection sampling

"Throw and count" is a very straightforward technique to calculate π. We generate random numbers and count how many fall inside the circle. Fig. 1 shows how the error of the calculated π depends on the number of sampling points: even for a large number of the sampling points the error remains substantial.

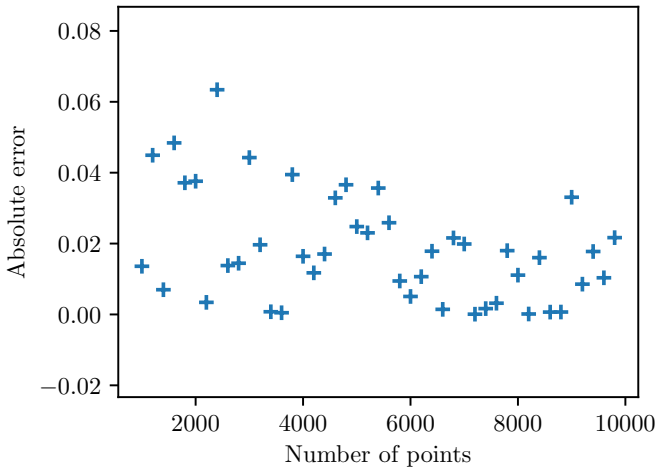

Figure 1: Throw and Count

1.2 Direct

Alternative approach for computing π is based on integration:

$$A = \int_0^1 \frac{1}{1+x^2} dx \tag{1}$$

As shown in the description of the task, the error estimate of basic Monte-Carlo integration scales $\propto 1/\sqrt{N}$. The standard error as a function of the number of sampling points is show in Fig. 3. The absolute error of the direct Monte-Carlo integration is shown in Fig. 2. The errors are moderately improved compared to the "throw and count" method.

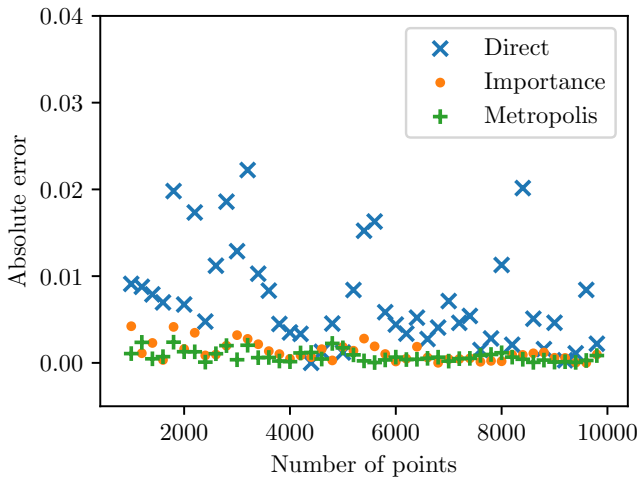

Figure 2: Absolute error

1.3 Importance Sampling

Importance sampling requires the integration of a smoothened function $g(x) = \frac{f(x)}{w(x)}$:

$$\tilde{A} = \int_0^1 g(x)w(x)dx = \int_0^1 \frac{3}{[1+x(y)^2]\,[4-2x(y)]}dy \tag{2}$$

As shown in Fig. 2 and 3, the smoothening of the function significantly reduced the standard error as well as the absolute error.

1.4 Metropolis

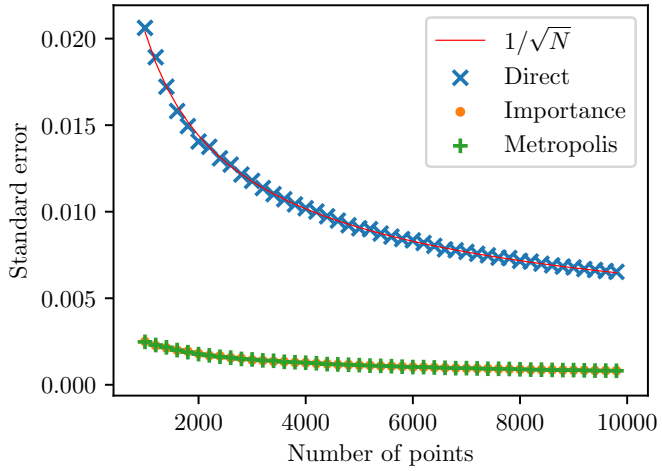

Figure 3: Direct Monte-Carlo

Metropolis algorithm requires compliance with the detailed balance equation:

$$\frac{T\left(x \to x'\right)}{T\left(x' \to x\right)} = \frac{w\left(x'\right)}{w(x)} \geq w_i \tag{3}$$

In order to use the Metropolis algorithm in the most efficient way the trial step size h has to be chosen such that the accepting rate is $\approx 50\%$. If the accepting rate is too high, the Metropolis algorithm behaves just like a basic importance sampling scheme. On the other hand, when the accepting rate is very low, much of the time is wasted in searching the new configuration. In Fig. 4 the accepting rate is shown as a function of the trial step size. Thus, the best choice for the trial step is $h = 0.9$.

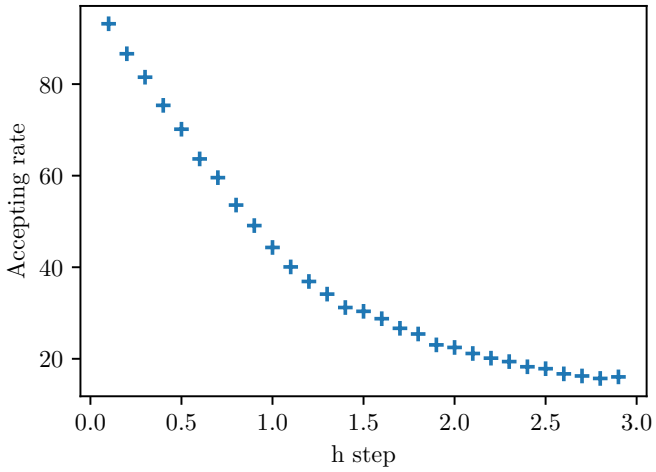

Figure 4: Accepting rate

In the Metropolis algorithm, it is important to make sure that the sampled configurations are uncorrelated. The correlation between data points can be analyzed by the *autocorrelation* function. Fig. 5 shows the autocorrelation function. The correlation is almost fully suppressed when $l \geq 8$. Thus, we can safely use $l = 10$ in our calculations.

As shown in Fig. 2 and Fig. 3, the errors of the Metropolis method are similar to those of the importance sampling. Thus, by using the Metropolis algorithm, we were able to achieve the same accuracy without using $x(y)$, which is important as in most cases $x(y)$ is too complex to derive.

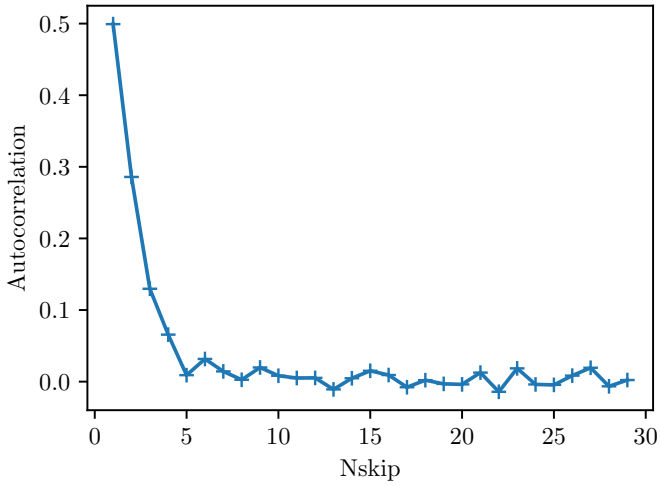

Figure 5: Autocorrleation function