Obviating the bin width effect of the $1/t$ algorithm for multidimensional numerical integration

CrossMark

Wanyok Atisattapong *, Pasin Maruphanton

Department of Mathematics and Statistics, Faculty of Science and Technology, Thammasat University, Pathum Thani 12120, Thailand

A B S T R A C T

In this work we improve the accuracy and the convergence of the $1/t$ algorithm for multidimensional numerical integration. The proposed strategy is to introduce a new approximation method which obviates the bin width effect of the conventional $1/t$ algorithm by using the average of y values, which varies as the number of Monte Carlo trials changes, instead of the fixed value of y. The non-convergence of the $1/t$ algorithm and the convergence of the new method are proved by theoretical analysis. The potential of the method is illustrated by the evaluation of one-, two- and multi- dimensional integrals up to six dimensions. Our results show that the numerical estimates from our method converge to their exact values without either error saturation or the bin with effect, in contrast with the conventional $1/t$ algorithm.

© 2015 IMACS. Published by Elsevier B.V. All rights reserved.

1. Introduction

Integration is a mathematical method which is an important concept in calculus. Most integration cannot be performed by analytical methods using the basic formulas, therefore an approximation method is required. Numerical integration methods such as the trapezoidal method, Simpson's method and the Gauss quadrature [7] are efficient for approximating the integral in low dimensions, but they lack efficiency in high dimensions. The Monte Carlo method is one of the most efficient algorithms for approximating the integrals in high dimensions [5,7].

The Wang–Landau algorithm was originally proposed as a technique for approximating the density of states of discrete spin models in statistical physics [9]. The density of states $g(E)$ denotes the number of configurations corresponding to energy E. It is used to calculate the partition function Z defined by:

$$Z = \sum_{configurations} e^{-E/k_B T} = \sum_E g(E)e^{-E/k_B T}. \tag{1}$$

The important quantities in statistical physics such as internal energy, free energy and entropy can be obtained by the partition function.

In the Wang–Landau algorithm, the density of states is *a priori* unknown. Initially, we set $g(E) = 1$ for all possible energies E. A random walk is performed in energy space according to a Metropolis acceptance probability (the probability for changing from current state to other state) which is proportional to $\frac{1}{g(E)}$. In general, the acceptance probability from energy level E_1 to E_2 is:

* Corresponding author.
  E-mail addresses: wanyok@mathstat.sci.tu.ac.th (W. Atisattapong), oporkabbb@hotmail.com (P. Maruphanton).

$$p(E_1 \rightarrow E_2) = \min\left(1, \frac{g(E_1)}{g(E_2)}\right). \tag{2}$$

Each time an energy level E is visited, it is updated by a modification factor $f > 1$, i.e. $g(E) \rightarrow g(E) * f$. We also update a histogram of energy level E by one, i.e. $H(E) \rightarrow H(E) + 1$. If the random walk rejects a move to the new state, then the current $g(E)$ and $H(E)$ are updated again. Initially, $f_0 = e^1 \simeq 2.71828\ldots$ and $H(E) = 0$ for all energy levels E. The process is repeated until the entry histogram is flat, i.e. $H(E) > p\langle H(E)\rangle$ for all possible energies E, where p is a flatness criterion. It is usually set between 0.8 and 0.9. $\langle H(E)\rangle$ is an average of histograms over possible energies. The flatness is generally checked for each fixed Monte Carlo trial. Then, the modification factor f is reduced $f \rightarrow \sqrt{f}$ and $H(E)$ is reset to 0. The process is terminated when f is close to 1 (such as $f_{final} = \exp(10^{-8}) \simeq 1.00000001$). The parameters of the algorithm can be modified according to the type, size and complexity of the system.

The $1/t$ algorithm [2] was proposed as an improved version of the Wang–Landau algorithm. Belardinelli and Pereyra [3] found that the non-convergence of the Wang–Landau algorithm results from the way of decreasing the modification factor, while the $1/t$ algorithm provides an efficient way to avoid the error saturation by choosing a refinement parameter that depend on time (or the number of Monte Carlo trials). There are two differences between the algorithms. First, the $1/t$ algorithm does not use the flatness criterion. The modification factor (f) is decreased when all energy levels are visited by the random walker. Second, taking $F = \log(f)$, when $F < 1/t$, the modification factor is decreased by $F \rightarrow 1/t$, instead of $F \rightarrow F/2$. Unlike the original method, the error obtained by this method does not saturate in discrete models. Both Wang–Landau and $1/t$ algorithms have been applied to multidimensional integrals (continuous models) in order to analyze their convergence [1,6].

To calculate $\int_a^b y(x)dx$, the lower bound (y_{min}) and the upper bound (y_{max}) are determined by the self-adaptive range method [8]. Then the y-space is divided into a number of equal intervals $[y_i, y_i + dy]$, for $i = 1, \ldots, n$, where dy denotes a bin width of y and n is the number of subintervals in y-space. A distribution depending on y, namely $g(y_i)$, i.e. the measure $\{x | x \in [a, b], y_i \leq y(x) \leq y_i + dy\}$, can be generated. This distribution $g(y)$ corresponds to the density of states $g(E)$ in a physical system. Since $y_1 = y_{min}$ and $y_n + dy = y_{max}$, the integral can then be approximated using the following definition:

$$I = \int_a^b y(x)dx \simeq \sum_{y_{min}}^{y_{max}} g(y).y = \sum_{i=1}^{n} g(y_i).y_i, \tag{3}$$

where y_i is the midpoint of the subinterval $[y_i, y_i + dy]$.

Belardinelli et al. [1] evaluated seven integrals up to six dimensions (I_π, I_{nD} where $n = 1, \ldots, 6$). The estimated results indicate that the $1/t$ algorithm is more accurate than the Wang–Landau algorithm in all cases. They also analyzed the source of errors in both algorithms by using the dynamic behavior of the error. However, the error saturation in the $1/t$ algorithm arises from the grid discretization of the y-space (the bin width effect). The estimated integrals obtained by the Wang–Landau and $1/t$ algorithms are calculated by the midpoint approximation. Therefore, the bin width dy impacts on the accuracy of the estimated integral. Accuracy can be improved by reducing the bin width, but at the cost of increasing computation time. In traditional methods, there is no way of determining the optimal bin width.

Therefore, the main focus of this work will be a strategy to avoid the bin width effect by introducing a new definition of the approximation. In addition, the theoretical proof for the convergence of the method will be analyzed to demonstrate that the estimated integral always converges to the exact value without saturation of the error. Moreover, to compare the efficiencies of the proposed method and the conventional method, we apply the proposed method to the multidimensional integrals (as in Refs. [1,6]) up to six dimensions.

The remainder of the paper is as follows. In Section 2, the proposed method and theoretical analysis are described. In Section 3, the results and errors of the new method are compared with those from the original algorithm. Finally, the conclusions are outlined in Section 4.

2. Method

2.1. Description of the method

The source of error saturation in the $1/t$ algorithm is exclusively due to the approximation method. It uses a fixed value of y to represent the subinterval $[y_i, y_i + dy]$ used to calculate the integral in the approximation (3). In particular, the definition in Refs. [1,6] uses a midpoint, i.e. $\frac{2y_i+dy}{2}$, to represent the subinterval $[y_i, y_i + dy]$. Fixing the value of y_i results in error saturation because it does not depend on the number of Monte Carlo trials (N).

The proposed strategy is to introduce a new approximation method which obviates the bin width effect of the $1/t$ algorithm. We use the average of y values in the subinterval $[y_i, y_i + dy]$, which varies as N changes. The definite integral can then be approximated by

$$I = \int_a^b y(x)dx \simeq \sum_{y_{min}}^{y_{max}} g(y).\bar{y} = \sum_{i=1}^{n} g(y_i).\bar{y}_i, \tag{4}$$