Holonomic gradient method — a symbolic-numeric method to evaluate integral with parameters

N.Takayama, T.Yaguchi, Y.Zhang

February 24, 2026

The holonomic gradient method[1] is a method for evaluating approximate values of parameterized multiple integrals through the following 3 steps:

1. Derive a holonomic system of linear partial differential equations (holonomic system) satisfied by the multiple integral with respect to its parameters, using a computational algebraic algorithm.

2. Numerically compute approximate values of the multiple integral at a few parameter values.

3. Convert the holonomic system obtained via computational algebra into an ordinary differential equation (ODE), and numerically compute approximate values of the multiple integral for all parameter values using numerical solvers for ODEs.

ODE $Lf = 0$ obtained via computational algebra, such as through the creative telescoping method or D-module integration, are often large in size, and standard methods like the Runge-Kutta method frequently fail to work well for Step 3. We introduce effective methods for performing the numerical analysis in Step 3 (specifically, the sparse interpolation/extrapolation (SIE) method) and their implementations, using various examples of multiple integrals that appear in mathematical statistics.

While the above provides an overview, we will explain the SIE method using a specific example. Let M_x be a “certain” random manifold parameterized by x. When x is large, the approximate expected value of the Euler characteristic of M_x is expressed by a 6-fold integral with respect to $s_1, s_2, \sigma, b, s, t$ of the integral kernel:

$$\frac{s_1 s_2(\sigma^2 - b^2)}{(1+s^2)(1+t^2)} \exp\left(-\frac{1}{2}R\right) Y(\sigma - x)$$

[1] For a survey, see N. Takayama, Hypergeometric systems, statistics, and algorithms, Combinatorial, Computational, and Applied Algebraic Geometry (Proceedings of Symposia in Pure Mathematics, Vol. 111), AMS, 2025, pp. 113–136.

Here, Y is the Heaviside function, and R is a certain rational expression of these variables. By finding the ODE satisfied by this integral with respect to x using the creative telescoping method, we obtain a 26 Kbyte, rank 11 ODE $Lf = 0$. Consider the integral:

$$\int_I |Lf|^2 dx$$

on a suitable interval I. Applying a numerical integration scheme with weights T_j and reference points x_j to this integral yields the approximation:

$$\sum_j |(Lf)(x_j)|^2 T_j.$$

We approximate f by a linear combination of basis functions e_k as $f = \sum_k f_k e_k$. Then, the approximate value of the integral is expressed as:

$$\sum_j \left| \sqrt{T_j} \sum_k f_k (Le_k)(x_j) \right|^2 . \tag{1}$$

The computation of $(Le_k)(x_j)$ is performed exactly using a computer algebra system (our implementation uses Risa/Asir). It is the puchline of the SIE method. After completing this preprocessing, we transition to double-precision computation. The integral values in Step 2 of the HGM is obtained using the Monte Carlo method. Using these values as a set of constraints, we minimize the above approximate integral value (1) with f_k as unknown variables to determine f. Executing Step 3 in this manner constitutes the SIE method. We used a Python package to solve the constrained optimization problem. Through this approach, we were able to rapidly and stably compute approximate expected values of the Euler characteristic over a wide domain.

The numerical analysis program in Python used to find the optimal values is automatically generated by the computer algebra system through computation with rational coefficients. This preprocessing can often suppress the catastrophic numerical error generation that occurs when a massive ODE is processed directly in double precision. For details on various methods and their comparisons across different examples including this one, see our preprint: `https://arxiv.org/abs/2111.10947`.

Although a test implementation has already been published, we plan to finalize it into a more user-friendly package by this summer.