

Numerical integration

Anne Kværnø

Oct 14, 2018

1 Introduction

Given the finite integral

$$I[f](a, b) = \int_a^b f(x) dx.$$

A *numerical quadrature* or a *quadrature rule* is a formula for approximating such integrals. Quadratures are usually of the form

$$Q[f](a, b) = \sum_{i=0}^n w_i f(x_i),$$

where x_i , w_i for $i = 0, 1, \dots, n$ are respectively the *nodes* and the *weights* of the quadrature rule. If the function f is given from the context, we will for simplicity denote the integral and the quadrature simply as $I(a, b)$ and $Q(a, b)$.

The [trapezoidal rule](#), the [midpoint rule](#) and [Simpson's rule](#) known from previous courses are all examples of numerical quadratures.

In this note we will see how quadrature rules can be constructed from integration of interpolation polynomials. We will demonstrate how to do error analysis and how to find error estimates. We will also demonstrate how the integration interval can be automatically partitioned in subintervals, so called *adaptive* integration.

In the sequel, we will use material from *Preliminaries*, section 3.2, 4 and 5.

2 Quadrature based on polynomial interpolation.

This section relies on the content of the note on polynomial interpolation, in particular the section on Lagrange polynomials.

Choose $n + 1$ distinct nodes x_i , $i = 0, \dots, n$ in the interval $[a, b]$, and let $p_n(x)$ be the interpolation polynomial satisfying the interpolation condition

$$p_n(x_i) = f(x_i), \quad i = 0, 1, \dots, n.$$

We will then use $\int_a^b p_n(x) dx$ as an approximation to $\int_a^b f(x) dx$. By using the Lagrange form of the polynomial

$$p_n(x) = \sum_{i=0}^n f(x_i) \ell_i(x)$$

with the cardinal functions $\ell_i(x)$ given by

$$\ell_i(x) = \prod_{j=0, j \neq i}^n \frac{x - x_j}{x_i - x_j},$$

the following quadrature formula is obtained

$$Q[f](a, b) = \int_a^b p_n(x) dx = \sum_{i=0}^n f(x_i) \int_a^b \ell_i(x) dx = \sum_{i=0}^n w_i f(x_i) = Q(a, b).$$

The weights in the quadrature is simply the integral of the cardinal functions over the interval.

Let us derive two schemes for integration over the interval $[0, 1]$, and apply them to the integral

$$I(0, 1) = \int_0^1 \cos\left(\frac{\pi}{2}x\right) dx = \frac{2}{\pi} = 0.636619 \dots$$

Example 1: Let $x_0 = 0$ and $x_1 = 1$. The cardinal functions and thus the weights are given by

$$\begin{aligned} \ell_0(x) &= 1 - x, & w_0 &= \int_0^1 (1 - x) dx = 1/2 \\ \ell_1(x) &= x, & w_1 &= \int_0^1 x dx = 1/2 \end{aligned}$$

and the corresponding quadrature rule, better known as the trapezoidal rule and usually denoted by T , is given by

$$T(0, 1) = \frac{1}{2} [f(0) + f(1)].$$

This formula applied to the function $f(x) = \cos(\pi x/2)$ gives

$$T(0, 1) = \frac{1}{2} \left[\cos(0) + \cos\left(\frac{\pi}{2}\right) \right] = \frac{1}{2},$$

and the error is

$$I(0, 1) - T(0, 1) = \frac{2}{\pi} - \frac{1}{2} = 0.138 \dots$$

Example 2: Let $x_0 = 1/2 + \sqrt{3}/6$ and $x_1 = 1/2 - \sqrt{3}/6$. Then

$$\begin{aligned} \ell_0(x) &= -\sqrt{3}x + \frac{1 + \sqrt{3}}{2}, & w_0 &= \int_0^1 \ell_0(x) dx = 1/2, \\ \ell_1(x) &= \sqrt{3}x + \frac{1 - \sqrt{3}}{2}, & w_1 &= \int_0^1 \ell_1(x) dx = 1/2. \end{aligned}$$

The quadrature rule is

$$Q(0, 1) = \frac{1}{2} \left[f\left(\frac{1}{2} - \frac{\sqrt{3}}{6}\right) + f\left(\frac{1}{2} + \frac{\sqrt{3}}{6}\right) \right].$$

And this quadrature applied to $f(x) = \cos(\pi x/2)$ is given by

$$Q(0, 1) = \frac{1}{2} \left[\cos\left(\frac{\pi}{2}x_0\right) + \cos\left(\frac{\pi}{2}x_1\right) \right] = 0.635647 \dots$$

with an error

$$I(0, 1) - Q(0, 1) = 9.72 \dots \cdot 10^{-4}.$$

So the choice of nodes clearly matters!

Before concluding this section, let us present simple indication on the quality of a method:

Definition: The degree of precision.

A numerical quadrature has degree of precision d if $Q[p](a, b) = I[p](a, b)$ for alle $p \in \mathbb{P}_d$.

Since both integrals and quadratures are linear in the integrand f , the degree of precision is d if

$$\begin{aligned} I[x^j](a, b) &= Q[x^j](a, b), & j = 0, 1, \dots, d, \\ I[x^{d+1}](a, b) &\neq Q[x^{d+1}](a, b) \end{aligned}$$

All quadratures constructed from Lagrange interpolation polynomials in $n + 1$ distinct nodes will automatically be of precision at least n .

It is left to the reader to show that the trapezoidal rule from Example 1 is of precision 1, the formula from Example 2 of precision 3.

3 Construction of numerical quadratures

In the following, you will learn the steps on how to construct realistic algorithms for numerical integration, similar to those used in software like Matlab or SciPy. The steps are:

Construction.

1. Choose $n + 1$ distinct nodes on a standard interval $[-1, 1]$.
2. Let $p_n(x)$ be the polynomial interpolating some general function f in the nodes, and let the $Q[f](-1, 1) = I[p_n](-1, 1)$.
3. Transfer the formula Q from $[-1, 1]$ to some interval $[a, b]$.
4. Find the composite formula, by dividing the interval $[a, b]$ into subintervals and applying the quadrature formula on each subinterval.
5. Find an expression for the error $E[f](a, b) = I[f](a, b) - Q[f](a, b)$.
6. Find an expression for an estimate of the error, and use this to create an adaptive algorithm.

3.1 Simpson's rule

We will go through the steps above for one method, Simpson's formula. The strategy is quite generic, so it is more important to understand and remember how results are derived, not exactly what they are. The different algorithms will be implemented and tested, and theoretical results will be verified by numerical experiments.

We will adopt the standard notation and denote this particular quadrature by $S[f](a, b)$.

The quadrature formula on the standard interval $[-1, 1]$. The quadrature rule is defined by the choice of nodes on a standard interval $[-1, 1]$. For Simpson's rule, choose the nodes $t_0 = -1$, $t_1 = 0$ and $t_2 = 1$. The corresponding cardinal functions are

$$\ell_0 = \frac{1}{2}(t^2 - t), \quad \ell_1(t) = 1 - t^2, \quad \ell_2(t) = \frac{1}{2}(t^2 + t).$$

which gives the weights

$$w_0 = \int_{-1}^1 \ell_0(t) dt = \frac{1}{3}, \quad w_1 = \int_{-1}^1 \ell_1(t) dt = \frac{4}{3}, \quad w_2 = \int_{-1}^1 \ell_2(t) dt = \frac{1}{3}$$

such that

$$\int_{-1}^1 f(t) dt \approx \int_{-1}^1 p_2(t) dt = \sum_{i=0}^2 w_i f(t_i) = \frac{1}{3} [f(-1) + 4f(0) + f(1)].$$

Simpson's rule has degree of precision 3 (check it yourself).

Example 3:

$$\int_{-1}^1 \cos\left(\frac{\pi t}{2}\right) dt = \frac{4}{\pi} \approx \frac{1}{3} [\cos(-\pi/2) + 4\cos(0) + \cos(\pi/2)] = \frac{4}{3}.$$

Transfer the integral and the quadrature to the interval $[a, b]$. The integral and the quadrature is transferred to some arbitrary interval $[a, b]$ by the transformation

$$x = \frac{b-a}{2}t + \frac{b+a}{2}, \quad \text{so} \quad dx = \frac{b-a}{2}dt.$$

Thus

$$\int_a^b f(x) dx = \frac{b-a}{2} \int_{-1}^1 f\left(\frac{b-a}{2}t + \frac{b+a}{2}\right) dt \approx \frac{b-a}{6} \left[f(a) + 4f\left(\frac{b+a}{2}\right) + f(b) \right].$$

Simpson's rule over the interval $[a, b]$ becomes

$$S(a, b) = \frac{b-a}{6} [f(a) + 4f(c) + f(b)], \quad c = \frac{b+a}{2}.$$

Composite Simpson's rule. Divide $[a, b]$ into $2m$ equal intervals of length $h = (b-a)/(2m)$. Let $x_j = a + jh$, $j = 0, \dots, 2m$, and apply Simpson's rule on each subinterval $[x_{2j}, x_{2j+2}]$. The result is:

$$\int_a^b f(x) dx = \sum_{j=0}^{m-1} \int_{x_{2j}}^{x_{2j+2}} f(x) dx \approx \sum_{j=0}^{m-1} S(x_{2j}, x_{2j+2}) \quad (1)$$

$$= \sum_{j=0}^{m-1} \frac{h}{3} [f(x_{2j}) + 4f(x_{2j+1}) + f(x_{2j+2})] \quad (2)$$

$$= \frac{h}{3} \left[f(x_0) + 4 \sum_{j=0}^{m-1} f(x_{2j+1}) + 2 \sum_{j=1}^{m-1} f(x_{2j}) + f(x_{2m}) \right] \quad (3)$$

We will use the notation $S_m(a, b)$ for the composite Simpson's rule on m subintervals.

Implementation and testing. It is now time to implement the composite Simpson's method, and see how well it works. See the function `simpson` in the file `quadrature.py`.

Test if the code is correct. We know that Simpson's rule has precision 3, thus all third degree polynomials can be integrated exactly. Choose one such polynomial, find the exact integral, and compare.

Numerical experiment 1: Apply the code on the integral, and compare with the exact result.

$$\int_{-1}^2 (4x^3 + x^2 + 2x - 1)dx = 18.$$

Numerical experiment 2: We will assume that the error decreases when the number of subintervals m increases. But how much?

Apply the composite method on the integral (again with a known solution):

$$\int_0^1 \cos\left(\frac{\pi x}{2}\right) dx = \frac{2}{\pi}.$$

Use the function 'simpson' with $m = 1, 2, 4, 8, 16$ and see how the error changes with m . Comment on the result.

From the experiment we observe that the error is reduced by a factor approximately $0.0625 = 1/16$ whenever the number of subintervals increases with a factor 2. In the following, we will prove that this is in fact what can be expected.

3.2 Error analysis

First we will find an expression for the error $E(a, b) = I(a, b) - S(a, b)$ over one interval (a, b) . This will then be used to find an expression for the composite formula.

Let $c = (a + b)/2$ be the midpoint of the interval, and $h = (b - a)/2$ be the distance between c and the endpoints a and b . Do a Taylor series expansion of the integrand f around the midpoint, and integrate each term in the series.

$$\int_a^b f(x)dx = \int_{-h}^h f(c + s)ds = \int_{-h}^h \left(f(c) + sf'(c) + \frac{1}{2}s^2 f''(c) + \frac{1}{6}s^3 f'''(c) + \frac{1}{24}s^4 f^{(4)}(c) + \dots \right) ds \quad (4)$$

$$= 2hf(c) + \frac{h^3}{3}f''(c) + \frac{h^5}{60}f^{(4)}(c) + \dots \quad (5)$$

Similarly, do a Taylor series expansion of the quadrature $S(a, b)$ around c :

$$S(a, b) = \frac{h}{3} (f(c - h) + 4f(c) + f(c + h)) \quad (6)$$

$$= \frac{h}{3} \left(f(c) - hf'(c) + \frac{1}{2}h^2 f''(c) - \frac{1}{6}h^3 f'''(c) + \frac{1}{24}h^4 f^{(4)}(c) + \dots \right. \quad (7)$$

$$\left. + 4f(c) \right. \quad (8)$$

$$\left. + f(c) + hf'(c) + \frac{1}{2}h^2 f''(c) + \frac{1}{6}h^3 f'''(c) + \frac{1}{24}h^4 f^{(4)}(c) + \dots \right) \quad (9)$$

$$= 2hf(c) + \frac{h^3}{3}f''(c) + \frac{h^5}{36}f^{(4)}(c) + \dots \quad (10)$$

The series expansion of the error becomes:

$$E(a, b) = \int_a^b f(x)dx - S(a, b) = -\frac{h^5}{90}f^{(4)}(c) + \dots = -\frac{(b-a)^5}{2^5 \cdot 90}f^{(4)}(c) + \dots,$$

using $h = (b - a)/2$.

NB! By choosing to do the Taylor-expansions around the midpoint, every second term disappear thanks to symmetry. Choosing another point $\hat{c}$ in the interval will give the same dominant error term (with c replaced by $\hat{c}$), but the calculations will be much more cumbersome.

Usually, we will assume h to be small, such that the first nonzero term in the series dominates the error, and the rest of the series can be ignored. It is however possible, but not trivial, to prove the following result:

Theorem: Error in Simpson's method.

Let $f(x) \in C^4[a, b]$. There exist a $\xi \in (a, b)$ such that

$$E(a, b) = \int_a^b f(x)dx - \frac{b-a}{6} \left[f(a) + 4f\left(\frac{b+a}{2}\right) + f(b) \right] = -\frac{(b-a)^5}{2880} f^{(4)}(\xi).$$

NB!: Since $p^{(4)}(x) = 0$ for all $p \in \mathbb{P}_3$ the degree of precision is 3.

Use the theorem to find an expression for the error in the composite Simpson's formula $S_m(a, b)$:

$$\int_a^b f(x)dx - S_m(a, b) = \sum_{j=0}^{m-1} \left(\int_{x_{2j}}^{x_{2j+2}} f(x)dx - \frac{h}{3} [f(x_{2j}) + 4f(x_{2j+1}) + f(x_{2j+2})] \right) \quad (11)$$

$$= \sum_{j=0}^{m-1} -\frac{(2h)^5}{2880} f^{(4)}(\xi_j) \quad (12)$$

where $\xi_j \in (x_{2j}, x_{2j+2})$. We can then use the generalized mean value theorem, see *Preliminaries*, section 5, result 2. According to this, there is a $\xi \in (a, b)$ such that

$$\sum_{j=0}^{m-1} f^{(4)}(\xi_j) = m f^{(4)}(\xi).$$

Use $2mh = b - a$, and the following theorem has been proved:

Theorem: Error in composite Simpson's method.

Let $f(x) \in C^4[a, b]$. There exist a $\xi \in (a, b)$ such that

$$\int_a^b f(x)dx - S_m(a, b) = -\frac{(b-a)h^4}{180} f^{(4)}(\xi).$$

Example 4: Find the upper bound for the error when the composite Simpson's rule is applied to the integral $\int_0^1 \cos(\pi x/2)dx$.

In this case $f^{(4)}(x) = (\pi^4/16) \cos(\pi x/2)$, so that $|f^{(4)}(x)| \leq (\pi/2)^4$. The error bound becomes

$$|I(a, b) - S_m(a, b)| \leq \frac{1}{180} \left(\frac{1}{2m} \right)^4 \left(\frac{\pi}{2} \right)^4 = \frac{\pi^4}{46080} \frac{1}{m^4}.$$

If m is increased by a factor 2, the error will be reduced by a factor of 1/16, as indicated by Numerical experiment 2.

Numerical exercise: Include the error bound in the output of Numerical experiment 2, and confirm that it really holds.

3.3 Error estimate

From a practical point of view, the error expression derived above has some limitations. First, the bounds are often much too high. Second, we do not always know (or want to find) $|f^{(4)}(x)|$, even less its upper bound. So the question arises: How can we find an estimate of the error, without any extra analytical calculations?

This is the idea: Let the interval (a, b) chosen small, such that $f^{(4)}(x)$ can be assumed to be almost constant over the interval. Let $H = b - a$ be the length of the interval. Let $S_1(a, b)$ and $S_2(a, b)$ be the results from Simpson's formula over one and two subintervals respectively. Further, let $C = -f^{(4)}(x)/2880$ for some $x \in [a, b]$, which x does not matter since $f^{(4)}$ is assumed almost constant anyway. The errors of the two approximations are then given by

$$\begin{aligned} I(a, b) - S_1(a, b) &\approx CH^5, \\ I(a, b) - S_2(a, b) &\approx 2C \left(\frac{H}{2}\right)^5. \end{aligned}$$

Subtract the two expressions to eliminate $I(a, b)$:

$$S_2(a, b) - S_1(a, b) \approx \frac{15}{16}CH^5 \quad \Rightarrow \quad CH^5 \approx \frac{16}{15}(S_2(a, b) - S_1(a, b)).$$

Insert this in the expression for the error:

$$E_1(a, b) = I(a, b) - S_1(a, b) \approx \frac{16}{15}(S_2(a, b) - S_1(a, b)) = \mathcal{E}_1(a, b), \quad (13)$$

$$E_2(a, b) = I(a, b) - S_2(a, b) \approx \frac{1}{15}(S_2(a, b) - S_1(a, b)) = \mathcal{E}_2(a, b). \quad (14)$$

This gives us a computable estimate for the error, both in S_1 and S_2 . As the error in $S_2(a, b)$ is about 1/16 of the error in $S_1(a, b)$, and we anyway need to compute both, we will use $S_2(a, b)$ as our approximation. An even better approximation to the integral is given for free by just adding the error estimate.

Example 5: Find an approximation to the integral $\int_0^1 \cos(x)dx = \sin(1)$ by Simpson's rule over one and two subintervals. Find the error estimates $\mathcal{E}_m$, $m = 1, 2$ and compare with the exact error.

Solution:

$$S_1(0, 1) = \frac{1}{6} [\cos(0.0) + 4\cos(0.5) + \cos(1.0)] = 0.8417720923 \quad (15)$$

$$S_2(0, 1) = \frac{1}{12} [\cos(0.0) + 4\cos(0.25) + 2\cos(0.5) + 4\cos(0.75) + \cos(1.0)] = 0.8414893826 \quad (16)$$

The exact error and the error estimate become:

$$E_1(0, 1) = \sin(1) - S_1(0, 1) = -3.011 \cdot 10^{-4}, \quad \mathcal{E}_1(0, 1) = \frac{16}{15}(S_2 - S_1) = -3.016 \cdot 10^{-4}, \quad (17)$$

$$E_2(0, 1) = \sin(1) - S_2(0, 1) = -1.840 \cdot 10^{-5}, \quad \mathcal{E}_2(0, 1) = \frac{1}{16}(S_2 - S_1) = -1.885 \cdot 10^{-5}. \quad (18)$$

In this case, it is a very good correspondence between the error estimate and the exact error. An even better approximation is obtained by adding the error estimate to S_2 :

$$Q = S_2(0, 1) + \mathcal{E}_2(0, 1) = 0.8414705353607151$$

with an error $\sin(1) - Q = 4.4945 \cdot 10^{-7}$. This gives a lot of additional accuracy without any extra work.

Implementation of Simpson's method with an error estimate. The function `simpson_basic` returns

$$S_2(a, b) \approx \int_a^b f(x) dx$$

including an error estimate.

Test: As a first check of the implementation, use the example above, and make sure that the results are the same:

Next, let us see how reliable the quadrature and the error estimates are for another example, which you have to do yourself:

Numerical experiment 3: Given the integral (with solution)

$$I(a, b) = \int_a^b \frac{1}{1 + 16x^2} dx = \frac{\arctan(4x)}{4} \Big|_a^b$$

1. Use `simsn_basic` to find an approximation to the integral over the interval $[0, 8]$. Print out $S_2(0, 8)$, the error estimate $\mathcal{E}_2(0, 8)$ and the real error $E_2(0, 8)$. How reliable are the error estimates?
2. Repeat the experiment over the intervals $[0, 1]$ and $[4, 8]$. Notice the difference between exact error of the two intervals.
3. Repeat the experiment over the interval $[0, 0.1]$.

This is what you should observe from the experiment:

1. Interval $[0, 8]$: The error is large, and the error estimate is significantly smaller than the real error (the error is *under-estimated*).
2. Interval $[0, 1]$: As for the interval $[0, 8]$.
3. Interval $[4, 8]$: Small error, and a reasonable error estimate.
4. Interval $[0, 0.1]$: Small error, reasonable error estimate.

Why is it so, and how can we deal with it? Obviously, we need small subintervals near $x = 0$, while large subintervals are acceptable in the last half of the interval.

Explanation: The error in Simpson's method is given by

$$E(a, b) = -\frac{(b-a)^5}{2880} f^{(4)}(\xi).$$

So let us take a look at $f^{(4)}(x)$:

$$f(x) = \frac{1}{1 + 16x^2} \quad \Rightarrow \quad f^{(4)}(x) = 6144 \frac{1280x^4 - 160x^2 + 1}{(1 + 16x^2)^5}$$

It is no surprise that the error is large and the error estimates fail (we have assumed $f^{(4)}$ almost constant for the estimates) over the interval $[0, 1]$. The part of the interval where $f^{(4)}(x)$ is large has to be partitioned in significantly smaller subintervals to get an acceptable result. But how, as $f^{(4)}$ is in general not known? This is the topic of the next section.

3.4 Adaptive integration

Given a basic function, for example `simpson_basic`, returning an approximation $Q(a, b)$ to the integral, as well as an error estimate $\mathcal{E}(a, b)$. Based on this, we want to find a partitioning of the interval:

$$a = X_0 < X_1 < \dots < X_m = b$$

such that

$$|\mathcal{E}(X_j, X_{j+1})| \approx \frac{X_{k+1} - X_k}{b - a} \cdot Tol$$

where Tol is a tolerance given by the user. In this case

$$\text{Accumulated error over } (a, b) \approx \sum_{j=0}^{m-1} \mathcal{E}(X_j, X_{j+1}) \leq Tol.$$

Such a partitioning can be done by an recursive algorithm:

Algorithm: Adaptive quadrature.

Given f , a , b and a user defined tolerance Tol .

- Calculate $Q(a, b)$ and $\mathcal{E}(a, b)$.
- **if** $|\mathcal{E}(a, b)| \leq Tol$:
 Accept the result, return $Q(a, b) + \mathcal{E}(a, b)$ as an approximation to $I(a, b)$.
- **else**:
 Let $c = (a + b)/2$, and repeat the process on each of the subintervals $[a, c]$ and $[c, b]$, with tolerance $Tol/2$.
- Sum up the accepted results from each subinterval.

Implementation. The adaptive algorithm is implemented below with `simpson_basic` as the basic quadrature routine. The function `simpson_adaptive` is a recursive function, that is a function that calls itself. To avoid it to do so infinitely many times, an extra variable `level` is introduced, this will increase by one for each time the function calls itself. If `level` is over some maximum value, the result is returned, and a warning printed.

Numerical experiment 4: Use adaptive Simpson to find an approximation to the integral $\int_0^5 1/(1 + 16x^2)dx$ using the tolerances $Tol=10^{-3}, 10^{-5}, 10^{-7}$. Compare the numerical result with the exact one.

3.5 Other quadrature formulas

Simpson's rule is only one example of quadrature rules derived from polynomial interpolations. There are many others, and the whole process of deriving the methods, do error analysis, develop error estimates and adaptive algorithms can be repeated.

Let us just conclude with a few other popular classes of methods:

Newton-Cotes formulas. These are based on equidistributed nodes. Both the Trapezoidal rule and Simpson's rule are examples of closed Newton-Cotes methods, the midpoint rule is an example of an open one.

Gauss-Legendre quadrature: For the standard interval $[-1, 1]$ choose the nodes as the zeros of the polynomial of degree m :

$$L_m(t) = \frac{d^m}{dt^m}(t^2 - 1)^m.$$

These are of precision $d = 2m - 1$, which is the best result possible with m nodes. The midpoint rule is an example of a Gauss-Legendre quadrature with $m = 1$.