



WIKIPEDIA
The Free Encyclopedia

[Main page](#)
[Contents](#)
[Featured content](#)
[Current events](#)
[Random article](#)
[Donate to Wikipedia](#)
[Wikipedia store](#)

[Interaction](#)
[Help](#)
[About Wikipedia](#)
[Community portal](#)
[Recent changes](#)
[Contact page](#)

[Tools](#)
[What links here](#)
[Related changes](#)
[Upload file](#)
[Special pages](#)
[Permanent link](#)
[Page information](#)
[Wikidata item](#)
[Cite this page](#)

[Print/export](#)
[Create a book](#)
[Download as PDF](#)
[Printable version](#)

[Languages](#)
[Português](#)
[Edit links](#)

[Create account](#) [Log in](#)

[Article](#) [Talk](#)

[Read](#) [Edit](#) [View history](#)

Trapezoidal rule (differential equations)

From Wikipedia, the free encyclopedia

In [numerical analysis](#) and [scientific computing](#), the **trapezoidal rule** is a [numerical method to solve ordinary differential equations](#) derived from the [trapezoidal rule](#) for computing integrals. The trapezoidal rule is an [implicit](#) second-order method, which can be considered as both a [Runge–Kutta method](#) and a [linear multistep method](#).

Contents

- [1 Method](#)
- [2 Motivation](#)
- [3 Error analysis](#)
- [4 Stability](#)
- [5 Notes](#)
- [6 References](#)
- [7 See also](#)

Method

Suppose that we want to solve the differential equation

$$y' = f(t, y).$$

The trapezoidal rule is given by the formula

$$y_{n+1} = y_n + \frac{1}{2}h \left(f(t_n, y_n) + f(t_{n+1}, y_{n+1}) \right),$$

where $h = t_{n+1} - t_n$ is the step size.^[1]

This is an implicit method: the value y_{n+1} appears on both sides of the equation, and to actually calculate it, we have to solve an equation which will usually be nonlinear. One possible method for solving this equation is [Newton's method](#). We can use the [Euler method](#) to get a fairly good estimate for the solution, which can be used as the initial guess of Newton's method.^[2]

Motivation

Integrating the differential equation from t_n to t_{n+1} , we find that

$$y(t_{n+1}) - y(t_n) = \int_{t_n}^{t_{n+1}} f(t, y(t)) \, dt.$$

The [trapezoidal rule](#) states that the integral on the right-hand side can be approximated as

$$\int_{t_n}^{t_{n+1}} f(t, y(t)) \, dt \approx \frac{1}{2}h \left(f(t_n, y(t_n)) + f(t_{n+1}, y(t_{n+1})) \right).$$

Now combine both formulas and use that $y_n \approx y(t_n)$ and $y_{n+1} \approx y(t_{n+1})$ to get the trapezoidal rule for solving ordinary differential equations.^[3]

Error analysis

It follows from the error analysis of the trapezoidal rule for quadrature that the [local truncation error](#) τ_n of the trapezoidal rule for solving differential equations can be bounded as:

$$|\tau_n| \leq \frac{1}{12}h^3 \max_t |y'''(t)|.$$

Thus, the trapezoidal rule is a second-order method. This result can be used to show that the global error is $O(h^2)$ as the step size h tends to zero (see [big O notation](#) for the meaning of this).^[4]

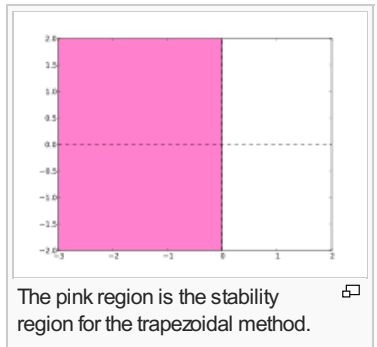
Stability

The [region of absolute stability](#) for the trapezoidal rule is

$$\{z \in \mathbb{C} \mid \operatorname{Re}(z) < 0\}.$$

This includes the left-half plane, so the trapezoidal rule is A-stable. The second Dahlquist barrier states that the trapezoidal rule is the most accurate amongst the A-stable linear multistep methods. More precisely, a linear multistep method that is A-stable has at most order two, and the error constant of a second-order A-stable linear multistep method cannot be better than the error constant of the trapezoidal rule.^[5]

In fact, the region of absolute stability for the trapezoidal rule is precisely the left-half plane. This means that if the trapezoidal rule is applied to the linear test equation $y' = \lambda y$, the numerical solution decays to zero if and only if the exact solution does.

Notes [\[edit\]](#)

1. [^] [Iserles 1996](#), p. 8; [Süli & Mayers 2003](#), p. 324
2. [^] [Süli & Mayers 2003](#), p. 324
3. [^] [Iserles 1996](#), p. 8; [Süli & Mayers 2003](#), p. 324
4. [^] [Iserles 1996](#), p. 9; [Süli & Mayers 2003](#), p. 325
5. [^] [Süli & Mayers 2003](#), p. 324

References [\[edit\]](#)

- Iserles, Ariele (1996), *A First Course in the Numerical Analysis of Differential Equations*, Cambridge University Press, ISBN 978-0-521-55655-2.
- Süli, Endre; Mayers, David (2003), *An Introduction to Numerical Analysis*, Cambridge University Press, ISBN 0521007941.

See also [\[edit\]](#)

- Crank–Nicolson method

| | | | |
|--|---|---|--------|
| v · t · e | Numerical integration methods by order | | [hide] |
| First-order | Second-order | Higher-order | |
| Euler (backward · semi-implicit · exponential) | Verlet (velocity) · Trapezoidal · Beeman · Mdpoint · Heun · Newmark-beta · Leapfrog | Exponential integrators · General linear (Runge–Kutta (list) · multistep) | |

Categories: Runge–Kutta methods

This page was last modified on 15 November 2013, at 15:01.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#) [Contact Wikipedia](#) [Developers](#) [Mobile view](#)

