

NACRE-INTEGRATORS: INTEGRAL CALCULATION PROGRAMS

ABSTRACT. The NACRE set of programs demonstrates mathematical algorithms (in educational use) and solves practical problems of numeric analysis. NACRE-integrators contain quadrature algorithms and applied utilities. Lastest version is 0.8.9d modified 2006-04-28.

NACRE is free and Open Source (GNU GPL). You can help project by donating and code developing.

1. chebyshev

Chebyshev quadrature:

$$(1) \quad \int_{-1}^1 f(x)dx \cong \frac{2}{n} \sum_{i=1}^n f(x_i), n = 2, 3, 4, 5, 6, 7, 9.$$

Values of x_i are the roots of polynomials:

$n = 2$	$x^2 - \frac{1}{3}$,
$n = 3$	$x^3 - \frac{1}{5}x$,
$n = 4$	$x^4 - \frac{2}{5}x^2 + \frac{1}{45}$,
$n = 5$	$x^5 - \frac{3}{6}x^3 + \frac{4}{72}x$,
$n = 6$	$x^6 - x^4 + \frac{1}{5}x^2 - \frac{1}{105}$,
$n = 7$	$x^7 - \frac{7}{6}x^5 + \frac{119}{360}x^3 - \frac{149}{6480}x$,
$n = 9$	$x^9 - \frac{9}{2}x^7 + \frac{27}{40}x^5 - \frac{57}{560}x^3 + \frac{53}{22400}x$.

See `chebyshev_calculate()` at `integrators.hpp`.

2. composite

Newton-cotes quadratures:

$$(2) \quad \int_a^b f(x)dx \cong \frac{nh}{P_n} \sum_{i=0}^n f(a + ih)p_{in}, P_n = \sum_{i=0}^n p_{in}, h = \frac{b-a}{n}.$$

Values of p_{in} :

$n = 1$	1, 1
$n = 2$	1, 4, 1;
$n = 3$	1, 3, 3, 1;
$n = 4$	7, 32, 12, 32, 7;
$n = 5$	19, 75, 50, 50, 75, 19;
$n = 6$	41, 216, 27, 272, 27, 216, 41;
$n = 7$	751, 3577, 1323, 2989, 2989, 1323, 3577, 751.

Composite quadratures:

$$(3) \quad \int_a^b f(x)dx = \sum_{i=1}^s \int_{a_{i-1}}^{a_i} f(x)dx, a_i = a + ih, h = \frac{b-a}{s}.$$

See `composite_calculate()`, `newton_cotes_calculate()` at `integrators.hpp`.

3. gauss-chebyshev

Gauss-Chebyshev quadrature:

$$(4) \quad \int_{-1}^1 \frac{1}{\sqrt{1-x^2}} f(x) dx \cong \sum_{i=1}^{n-1} w_i f(x_i), x_i = \cos \frac{\pi(i - \frac{1}{2})}{n}, w_i = \frac{\pi}{n}.$$

See `gauss_chebyshev_calculate()` at `integrators.hpp`.

4. ppa

Preferred Precision Algorithm uses **indirect** parameter of calculations precision. We increasing s (see the *composite* section above) until difference between calculations before and after increasing s becomes less than preferred `eps`.

See `PPA_calculate()` at `integrators.hpp`.

5. fourier-series

Fourier-series¹:

$$(5) \quad f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos \frac{\pi n x}{l} + b_n \sin \frac{\pi n x}{l}), x \in (-l, l);$$

$$(6) \quad a_k = \frac{1}{l} \int_{-l}^l f(x) \cos \frac{\pi k x}{l} dx, b_k = \frac{1}{l} \int_{-l}^l f(x) \sin \frac{\pi k x}{l} dx.$$

See `fourier-series.cpp`.

6. interpolate

Lagrange form of interpolation polynomial:

$$(7) \quad L_n(x) = \sum_{i=0}^n f(x_i) \frac{\prod_{j=0, j \neq i}^n (x - x_j)}{\prod_{j=0, j \neq i}^n (x_i - x_j)}.$$

See `interpolate()` at `integrators.hpp`.

7. gravity

Gravity center for functionally defined figure²:

$$(8) \quad \left(\frac{\int_a^b x f(x) dx}{\int_a^b f(x) dx}, \frac{\frac{1}{2} \int_a^b f^2(x) dx}{\int_a^b f(x) dx} \right).$$

See `gravity.cpp`.

¹In general, $\lim_{n \rightarrow \infty} [\frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx)] = \frac{f(x+0) + f(x-0)}{2}$, where x is point of discontinuity for $f(x)$, but at this time integrators processing only functions without points of discontinuity (it will be done at NACRE-integrators v. 1.0).

²We plan to implement calculations for more complex figures than functionally defined, but **at this time** we focused on definite integrals of 1th dimension.