Perron's Formula

Manuel Eberl

July 14, 2026

Abstract

This entry provides a proof of *Perron's Formula*, which states that for a Dirichlet series $f(s) = \sum_{n=1}^{\infty} a_n n^{-s}$ with abscissa of convergence σ_c we have for any c, z, x with $c > 0$, $x > 0$, $\text{Re}(z) > \sigma_c - c$:

$$\sum_{n \leq x}{}' a_n n^{-z} = \frac{1}{2\pi i} \lim_{T \to \infty} \int_{c-iT}^{c+iT} f(z+w)\, x^s\, w \,\frac{\mathrm{d}w}{w}$$

Additionally, various explicit bounds for the remainder (i.e. when using some finite value of T instead of the limit) are shown.

As an interesting nontrivial auxiliary result, asymptotic bounds for a Dirichlet series near $\pm i\infty$ are also included, namely that if $a \in [0, 1)$ then $f(\sigma + it) \in o(t^{1-a})$ as $t \to \pm\infty$, uniformly for all $\sigma \geq \sigma_c + a + \varepsilon$.

The proofs mainly follow Tenenbaum's *Introduction to Analytic and Probabilistic Number Theory* [1] and Titchmarsh's *Theory of Functions* [2].

Contents

1 Auxiliary material

```
theory Perron_Prerequisites
  imports "Dirichlet_Series.Dirichlet_Series_Analysis"
begin
```

1.1 General analysis

```
lemma at_infinity_conv_filtercomap_norm_at_top: "at_infinity = filtercomap
norm at_top"
  ⟨proof⟩

lemma at_infinity_conv_filtercomap_abs_at_top:
  "at_infinity = filtercomap (abs :: real ⇒ real) at_top"
  ⟨proof⟩

lemma tendsto_dist_sandwich:
  assumes "eventually (λn. dist (f n) c ≤ g n) F"
  assumes "(g ⟶ 0) F"
  shows   "(f ⟶ c) F"
⟨proof⟩

lemma summation_by_parts:
  fixes f g :: "nat ⇒ 'a :: comm_ring_1"
  assumes "m ≤ n"
  shows "(∑k=m..n. f k * (g (Suc k) - g k)) =
           f (Suc n) * g (Suc n) - f m * g m - (∑k=m..n. g (Suc k) *
(f (Suc k) - f k))"
⟨proof⟩

lemma bounded_normE:
  assumes "bounded A"
  obtains C where "C > c" "⋀x. x ∈ A ⟹ norm x < C"
⟨proof⟩

lemma norm_suminf_le':
  fixes f :: "nat ⇒ 'a :: banach"
  assumes "⋀n. norm (f n) ≤ g n" "g sums A"
  shows   "norm (suminf f) ≤ A"
⟨proof⟩

lemma uniform_limit_compose':
  assumes "uniform_limit A f g F" and "h ` B ⊆ A"
  shows   "uniform_limit B (λn x. f n (h x)) (λx. g (h x)) F"
  ⟨proof⟩
```

1.2 Complex analysis

```
lemma analytic_imp_contour_integrable:
```

```
  "f analytic_on path_image g ⟹ valid_path g ⟹ f contour_integrable_on
g"
  ⟨proof⟩

lemma contour_integral_rectpath:
  assumes "f analytic_on path_image (rectpath a b)"
  shows   "contour_integral (rectpath a b) f =
             contour_integral (linepath a (Complex (Re b) (Im a))) f +
             contour_integral (linepath (Complex (Re b) (Im a)) b) f +
             contour_integral (linepath b (Complex (Re a) (Im b))) f +
             contour_integral (linepath (Complex (Re a) (Im b)) a) f"
⟨proof⟩

lemma contour_integral_bound_linepath':
    "⟦f contour_integrable_on (linepath a b);
      0 ≤ B; ⋀x. x ∈ closed_segment a b ⟹ norm(f x) ≤ B; c = norm
(b - a)⟧
     ⟹ norm(contour_integral (linepath a b) f) ≤ B * c"
  ⟨proof⟩

lemma contour_integral_linepath_same_Im:
  assumes "Im z = T" "Im z' = T" "Re z = a" "Re z' = b" "a < b"
  shows   "contour_integral (linepath z z') f =
           integral {a..b} (λx. f (Complex x T))"
⟨proof⟩

lemma contour_integral_linepath_same_Re:
  assumes "Re z = c" "Re z' = c" "Im z = a" "Im z' = b" "a < b"
  shows   "contour_integral (linepath z z') f =
           i * integral {a..b} (λx. f (Complex c x))"
⟨proof⟩

lemma continuous_on_Complex [continuous_intros]:
  assumes "continuous_on A f" "continuous_on A g"
  shows   "continuous_on A (λx. Complex (f x) (g x))"
  ⟨proof⟩

lemma contour_integral_primitive':
  assumes "⋀x. x ∈ S ⟹ (f has_field_derivative f' x) (at x within
S)"
      and "valid_path g" "path_image g ⊆ S" "pathfinish g = b" "pathstart
g = a"
    shows "(f' has_contour_integral (f b - f a)) g"
  ⟨proof⟩

lemma fds_converges_0_imp_summable_fds_nth:
  assumes "fds_converges f 0"
  shows   "summable (fds_nth f)"
⟨proof⟩
```

lemma *contour_integral_rmul: "contour_integral g (λx. f x * c) = contour_integral g f * c"*
⟨*proof*⟩

lemma *contour_integral_lmul: "contour_integral g (λx. c * f x) = c * contour_integral g f"*
⟨*proof*⟩

lemma *contour_integral_divide: "contour_integral g (λx. f x / c) = contour_integral g f / c"*
⟨*proof*⟩

lemma *uniform_limit_contour_integral_linepath:*
assumes *u: "uniform_limit (path_image (linepath a b)) f g F"*
assumes *c: "⋀n. continuous_on (path_image (linepath a b)) (f n)"*
assumes *[simp]: "F ≠ bot"*
obtains *I J* **where**
"⋀n. (f n has_contour_integral I n) (linepath a b)"
"(g has_contour_integral J) (linepath a b)"
"(I ⟶ J) F"
⟨*proof*⟩

lemma *contour_integral_sums_linepath:*
assumes *u: "uniform_limit (closed_segment a b) (λN w. ∑n<N. f n w) g sequentially"*
assumes *c: "⋀n. continuous_on (closed_segment a b) (f n)"*
obtains *J* **where**
"(g has_contour_integral J) (linepath a b)"
"(λn. contour_integral (linepath a b) (f n)) sums J"
⟨*proof*⟩

1.3 Dirichlet series

lemma *uniform_limit_eval_fds:*
fixes *f :: "'a :: dirichlet_series fds"*
assumes *"compact B" "⋀z. z ∈ B ⟹ conv_abscissa f < ereal (z · 1)"*
shows *"uniform_limit B (λN z. ∑n≤N. fds_nth f n / nat_power n z) (eval_fds f) sequentially"*
⟨*proof*⟩

lemma *uniform_limit_eval_fds':*
fixes *f :: "'a :: dirichlet_series fds"*
assumes *"compact B" "⋀z. z ∈ B ⟹ conv_abscissa f < ereal (z · 1)"*
shows *"uniform_limit B (λN z. ∑n<N. fds_nth f n / nat_power n z) (eval_fds f) sequentially"*
⟨*proof*⟩

lemma *conv_abscissa_shift [simp]:*
  "conv_abscissa (fds_shift c f) = conv_abscissa (f :: 'a :: dirichlet_series fds) + c · 1"
⟨*proof*⟩

lemma *abs_conv_abscissa_fds_zeta [simp]:*
  "abs_conv_abscissa (fds_zeta :: 'a :: dirichlet_series fds) = 1"
⟨*proof*⟩

end

2 Bounding Dirichlet series at $\pm i\infty$

theory *Dirichlet_Series_At_I_Infinity_Bound*
  imports *"Dirichlet_Series.Dirichlet_Series_Analysis" Perron_Prerequisites*
begin

This lemma corresponds to 9.11 (2) in Titchmarsh's Theory of Functions. It bounds the difference of two successive terms in the Dirichlet series expansion of the Riemann ζ function.

lemma *dist_consec_nat_powr_complex_le:*
  assumes *"n > 0"* **and** *"Re s ≠ 0"*
  shows *"norm (of_nat n powr (-s) - of_nat (Suc n) powr (-s)) ≤
      norm s / Re s * (real n powr (-Re s) - real (Suc n) powr (-Re s))"*
⟨*proof*⟩

For any $c > 0$, the real-valued function $f(x) = \frac{c^x - 1}{x}$ is differentiable everywhere, with a removable singularity at $x = 0$. If $c = 1$ it is the constant zero function; otherwise it is strictly increasing.

definition *fds_at_ii_infinity_bound :: "real ⇒ real ⇒ real"* **where**
  "fds_at_ii_infinity_bound c x = (if x = 0 then ln c else ((c powr x - 1) / x))"

lemma *fds_at_ii_infinity_bound_1_left [simp]: "fds_at_ii_infinity_bound 1 = (λx. 0)"*
  ⟨*proof*⟩

definition *fds_at_ii_infinity_bound_deriv :: "real ⇒ real ⇒ real"* **where**
  *"fds_at_ii_infinity_bound_deriv c x =
    (if x = 0 then ln c ^ 2 / 2 else (1 + c powr x * (x * ln c - 1)) / x ^ 2)"*

lemma *fds_at_ii_infinity_bound_deriv_pos:*
  assumes *"c > 0" "c ≠ 1"*
  shows *"fds_at_ii_infinity_bound_deriv c x > 0"*

⟨proof⟩

```
lemma has_field_derivative_fds_at_ii_infinity_bound:
  assumes "c > 0"
  defines "f ≡ fds_at_ii_infinity_bound c"
  defines "f' ≡ fds_at_ii_infinity_bound_deriv c"
  shows   "(f has_field_derivative f' x) (at x within A)"
⟨proof⟩

lemma strict_mono_fds_at_ii_infinity_bound:
  assumes "c > 0" "c ≠ 1"
  defines "f ≡ fds_at_ii_infinity_bound c"
  defines "f' ≡ fds_at_ii_infinity_bound_deriv c"
  shows   "strict_mono f"
⟨proof⟩

lemma mono_fds_at_ii_infinity_bound:
  assumes "c > 0"
  shows   "mono (fds_at_ii_infinity_bound c)"
⟨proof⟩
```

In the rest of this section, we will derive Theorem 9.33 in Titchmarsh's Theory of Functions, which bounds the value of a Dirichlet series as the imaginary part of its argument tends to $\pm\infty$.

```
lemma eval_fds_bigo_Im_going_to_infinity_aux1:
  fixes f :: "complex fds"
  assumes "fds_converges f 0" and c: "0 < c" "c < 1"
  defines "fltr ≡ Im going_to at_infinity within {s. Re s ≥ c}"
  shows   "eval_fds f ∈ O[fltr](λs. of_real (|Im s| powr (1 - c)))"
⟨proof⟩

lemma eval_fds_bigo_Im_going_to_infinity_aux2:
  fixes f :: "complex fds"
  assumes "fds_converges f s" and c: "Re s < c" "c < 1 + Re s"
  defines "fltr ≡ Im going_to at_infinity within {s. Re s ≥ c}"
  shows   "eval_fds f ∈ O[fltr](λw. of_real (|Im w| powr (1 - c + Re s)))"
⟨proof⟩
```

Now, the final theorem in its full generality: Let f be a Dirichlet series with abscissa of convergence σ_c. Then for any reals a, c with $a \in [0, 1)$ and $c - a > \sigma_c$ we have $f(\sigma + it) \in o(t^{1-a})$ as $t \to \pm\infty$ uniformly for $\sigma \geq c$.

```
theorem eval_fds_smallo_Im_going_to_infinity:
  fixes f :: "complex fds" and c :: real
  assumes c: "ereal (c - a) > conv_abscissa f" and a: "a ∈ {0..<1}"
  defines "fltr ≡ Im going_to at_infinity within {s. Re s ≥ c}"
  shows   "eval_fds f ∈ o[fltr](λw. of_real (|Im w| powr (1 - a)))"
⟨proof⟩

end
```

3 Perron's formula

theory *Perrons_Formula*
imports
  "Dirichlet_Series.Dirichlet_Series_Analysis"
  Perron_Prerequisites
  Dirichlet_Series_At_I_Infinity_Bound
begin

lemma (in *comm_monoid_set)* *union_disjoint':*
  assumes *"A ∪ B = C" "A ∩ B = {}" "finite C"*
  shows *"F g C = f (F g A) (F g B)"*
  ⟨*proof*⟩

lemma *infsum_Un_disjoint':*
  fixes *f :: "'a ⇒ 'b::{topological_comm_monoid_add, t2_space}"*
  assumes *"f summable_on A" "f summable_on B" "A ∩ B = {}" "C = A ∪ B"*
  shows *‹infsum f C = infsum f A + infsum f B›*
  ⟨*proof*⟩

3.1 Definitions

definition *sum_upto' :: "(nat ⇒ 'a :: real_vector) ⇒ real ⇒ 'a"* **where**
  "sum_upto' f x = sum_upto (λi. (if real i = x then (1/2) else 1) $*_R$ f i) x"

definition *perron_indicator :: "real ⇒ 'a :: field"* **where**
  "perron_indicator a = (if a > 1 then 1 else if a = 1 then 1/2 else 0)"

lemma *perron_indicator_simps [simp]:*
  "a > 1 ⟹ perron_indicator a = 1"
  "perron_indicator 1 = 1 / 2"
  "a < 1 ⟹ perron_indicator a = 0"
  ⟨*proof*⟩

3.2 The integral $\int_{c-i\infty}^{c+i\infty} a^z/z\,\mathbf{d}z$

The following integral is a key important in Perron's formula: If $a, c > 0$ then:

$$\int_{c-i\infty}^{c+i\infty} \frac{a^s}{s}\,\mathrm{d}s = \begin{cases} 0 & \text{if } a \in (0,1) \\ i\pi & \text{if } a = 1 \\ 2i\pi & \text{if } a > 1 \end{cases}$$

Note that this integral is *not* absolutely convergent (i.e. it does not exist in the sense of a Lebesgue integral) but only in the sense of a Cauchy principal value around the singularity at ∞.

context

fixes *L :: "real ⇒ real ⇒ real ⇒ complex"*
defines *"L ≡ (λc T. linepath (Complex c (-T)) (Complex c T))"*
begin

definition *perron_aux_integral :: "real ⇒ real ⇒ real ⇒ complex"* **where**
"perron_aux_integral a c T = contour_integral (L c T) (λz. of_real a powr z / z)"

definition *perron_remainder :: "real ⇒ real ⇒ real ⇒ real"* **where**
*"perron_remainder a c T = a powr c / (pi * T) * (if a = 1 then c else 1 / |ln a|)"*

lemma *perron_remainder_1: "perron_remainder 1 c T = c / (pi * T)"*
⟨*proof*⟩

lemma *perron_remainder_not_1: "a ≠ 1 ⟹ perron_remainder a c T = a powr c / (pi * T * |ln a|)"*
⟨*proof*⟩

lemma *perron_aux_integral_1:*
assumes *c: "c > 0"*
shows *"perron_aux_integral 1 c T = 2 * of_real (arctan (T / c)) * i"*
⟨*proof*⟩

lemma *perron_aux_integral_bound:*
fixes *a c T :: real*
assumes *a: "a > 0"* **and** *c: "c > 0"* **and** *T: "T > 0"*
defines *"I ≡ 1 / (2 * pi * i) * perron_aux_integral a c T"*
shows *"dist I (perron_indicator a) ≤ perron_remainder a c T"*
⟨*proof*⟩

lemma *perron_aux_integral_bound':*
fixes *a c T :: real*
assumes *a: "a > 0"* **and** *c: "c > 0"* **and** *T: "T > 0"*
shows *"dist (perron_aux_integral a c T) (2 * pi * i * perron_indicator a) ≤
2 * pi * perron_remainder a c T"*
⟨*proof*⟩

lemma *perron_aux_integral_bound2:*
fixes *a c T :: real*
assumes *a: "a > 0"* **and** *c: "c > 0"* **and** *T: "T > 0"*
shows *"dist (perron_aux_integral a c T) (2 * pi * i * perron_indicator a) ≤
2 * (arctan (T / c) + |ln a| * T + pi) * a powr c"*
⟨*proof*⟩

lemma *perron_aux_integral:*

fixes a c :: real
  assumes "a > 0" "c > 0"
  shows "(perron_aux_integral a c ⟶ 2 * pi * i * perron_indicator a) at_top"
⟨proof⟩

3.3 The textbook version

The textbook version of Perron's formula says the following: Consider a Dirichlet series $f(s) = \sum_{n=1}^{\infty} a_n n^{-s}$ whose abscissa of convergence is σ_c. Then, for any c, z, x with $c > 0$, $x > 0$, $\mathrm{Re}(z) > \sigma_c - c$ we have:

$$\sum_{n\leq x}{}' a_n n^{-z} = \frac{1}{2\pi i} \int_{c-i\infty}^{c+i\infty} f(z+s)\, x^s \,\frac{\mathrm{d}s}{s}$$

Note that the integral on the right-hand side must be interpretet as a Cauchy principal value around the singularity at ∞ and the $\sum'$ on the left-hand side means that if x is an positive integer then the last summand $a(x)$ must be multiplified with $\frac{1}{2}$.

lemma *perron_asymp_aux1:*
  fixes b x :: real
  assumes c: "c > 0" "ereal c > abs_conv_abscissa f"
  assumes x: "x > 0"
  shows "((λT. contour_integral (L c T) (λs. eval_fds f s * of_real x powr s / s))
        ⟶ 2 * pi * i * sum_upto' (fds_nth f) x) at_top"
⟨proof⟩

lemma *perron_asymp_aux2:*
  fixes c x :: real
  assumes c: "c > 0" "ereal c > conv_abscissa f"
  assumes x: "x > 0"
  shows "((λT. contour_integral (L c T) (λs. eval_fds f s * of_real x powr s / s))
        ⟶ 2 * pi * i * sum_upto' (fds_nth f) x) at_top"
⟨proof⟩

theorem *perron_asymp:*
  fixes fds :: "complex fds" **and** z :: complex **and** c x T :: real
  assumes c: "c > 0" "ereal (Re z + c) > conv_abscissa f" **and** x: "x > 0"
  shows "((λT. contour_integral (L c T) (λs. eval_fds f (z + s) * of_real x powr s / s))
        ⟶ 2 * pi * i * sum_upto' (λn. fds_nth f n / of_nat n powr z) x) at_top"
⟨proof⟩

3.4 The first effective version

As a more quantitative version of Perron's formula, we get for any $T, x > 0$ and $c > \max(0, \sigma_a)$:

$$\left| 2\pi i \sum_{n \le x}{}' a_n n^{-z} - = \int_{c-iT}^{c+iT} f(z+s)\, x^s \, \frac{\mathrm{d}s}{s} \right| \le C\, x^c \sum_{n=1}^{\infty} \frac{|a_n| n^{-c}}{1 + T |\log(x/n)|}$$

where $C = 4(\arctan(T/c) + \pi + 1)$.

Note that although C is not a constant we have $C \le 6\pi + 4$ for all T, c.

definition *perron_bound :: "complex fds ⇒ real ⇒ real ⇒ real ⇒ real"*
where
  *"perron_bound f c x T = (∑n. norm (fds_nth f n) / (n powr c * (1 + T * |ln (x / n)|)))"*

lemma *perron_bound_fds_shift [simp]:*
  "perron_bound (fds_shift s f) c x T = perron_bound f (c - Re s) x T"
  ⟨*proof*⟩

lemma *sums_perron_bound:*
  fixes *fds :: "complex fds"* **and** *c x T :: real*
  assumes *c: "ereal c > abs_conv_abscissa f"* **and** *T: "T > 0"* **and** *x: "x > 0"*
  shows *"(λn. norm (fds_nth f n) / (n powr c * (1 + T * |ln (x / n)|))) sums perron_bound f c x T"*
⟨*proof*⟩

lemma *perron_bound_nonneg:*
  fixes *fds :: "complex fds"* **and** *c x T :: real*
  assumes *c: "ereal c > abs_conv_abscissa f"* **and** *T: "T > 0"* **and** *x: "x > 0"*
  shows *"perron_bound f c x T ≥ 0"*
⟨*proof*⟩

lemma *perron_effective_strong_aux:*
  fixes *fds :: "complex fds"* **and** *c x T :: real*
  assumes *c: "c > 0" "ereal c > abs_conv_abscissa f"* **and** *T: "T > 0"* **and** *x: "x > 0"*
  defines *"A ≡ contour_integral (L c T) (λs. eval_fds f s * of_real x powr s / s)"*
  defines *"B ≡ 2 * pi * i * sum_upto' (fds_nth f) x"*
    shows *"dist A B ≤ 4 * (arctan (T / c) + pi + 1) * x powr c * perron_bound f c x T"*
⟨*proof*⟩

theorem *perron_effective_strong:*
  fixes *fds :: "complex fds"* **and** *z :: complex* **and** *c x T :: real*

assumes *c: "c > 0" "ereal c + ereal (Re z) > abs_conv_abscissa f"* **and** *T: "T > 0"* **and** *x: "x > 0"*
defines *"A ≡ contour_integral (L c T) (λs. eval_fds f (z + s) * of_real x powr s / s)"*
defines *"B ≡ 2 * pi * i * sum_upto' (λn. fds_nth f n / of_nat n powr z) x"*
shows *"dist A B ≤ 4 * (arctan (T / c) + pi + 1) * x powr c * perron_bound f (c + Re z) x T"*
⟨*proof*⟩

This corresponds to Theorem 2 in § II.2.1 of Tenenbaum's book.

corollary *perron_effective:*
fixes *fds :: "complex fds"* **and** *z :: complex* **and** *c x T :: real*
assumes *c: "c > 0" "ereal c + ereal (Re z) > abs_conv_abscissa f"* **and** *T: "T > 0"* **and** *x: "x > 0"*
defines *"A ≡ contour_integral (L c T) (λs. eval_fds f (z + s) * of_real x powr s / s)"*
defines *"B ≡ 2 * pi * i * sum_upto' (λn. fds_nth f n / of_nat n powr z) x"*
defines *"C ≡ 6 * pi + 4"*
shows *"dist A B ≤ C * x powr c * perron_bound f (c + Re z) x T"*
⟨*proof*⟩

3.5 The second effective version

Lastly, we derive Tenenbaum's Corollary 2.1, which he calles the *second* effective Perron formula. We first need a small auxiliary theorem that estimates $|ln(x/n)|$ in terms of $|x-n|/x$. This is easily derived using the Mean Value Theorem.

lemma *MVT2':*
assumes *"a ≠ b" "⋀x. x ∈ closed_segment a (b :: real) ⟹ (f has_field_derivative f' x) (at x)"*
shows *"∃z∈open_segment a b. f b - f a = (b - a) * f' z"*
⟨*proof*⟩

lemma *perron_effective'_aux:*
fixes *x b :: real* **and** *n :: nat*
assumes *b: "b ≥ 1"* **and** *x: "x > 0"* **and** *n: "n > 0" "n ≤ b * x"*
shows *"|ln (x / real n)| ≥ |x - real n| / (b * x)"*
⟨*proof*⟩

Now, the second effective Perron formula works in the following setting. Consider a Dirichlet series $f(s) := \sum_{n\geq 1} a_n n^{-s}$ with a finite abscissa of absolute convergence σ_a. Let $x, T \geq 1$ be real numbers and s a complex number with real part $\sigma \leq \sigma_a$. Let c be a real number with $c > \sigma_a - \sigma$.

Let α be a real number such that

$$\sum_{n\geq 1} |a_n| n^{-c-\sigma} \leq C(c+\sigma-\sigma_a)^{-\alpha}$$

and $B : [1, \infty] \to \mathbb{R}$ a non-decreasing function such that $|a_n| \leq B(n)$ for all $n \geq 1$.

Then we have:

$$\left| {\sum_{n \leq x}}' a_n n^s - \frac{1}{2\pi i} \int_{c-iT}^{c+iT} \frac{f(s+w)x^w}{w} \,\mathrm{d}w \right| \leq$$
$$C_1 C_3^{c+\sigma} B(2x) x^{-\sigma} \left(1 + \frac{x}{T}(4 \ln T + 9)\right) +$$
$$C_2 \frac{x^c}{T(c + \sigma - \sigma_a)^\alpha}$$

for $C_1 = 4(3\pi + 2)$ and $C_2 = 2(3\pi + 2)C/\ln(2)$, and $C_3 = 2$ if $c + \sigma \geq 0$ and $C_3 = \frac{1}{2}$ otherwise.

Note that Tenenbaum's version looks somewhat different since it hides the constants behind "Big-O" notation and also specialises to $c := \sigma_a - \sigma + 1/\ln x$.

```
theorem perron_effective':
  fixes c x T :: real and s :: complex
  assumes "abs_conv_abscissa f = ereal σa"
  assumes C: "(∑n. norm (fds_nth f n) / real n powr (c + Re s)) ≤ C
* (c + Re s - σa) powr -α"
  assumes B: "mono_on {1..} B" "⋀n. norm (fds_nth f n) ≤ B n"
  assumes x: "x ≥ 1" and T: "T ≥ 1" and s: "Re s ≤ σa"
  assumes c: "c > σa - Re s"
  defines "A ≡ (λx. 2 * pi * i * sum_upto' (λn. fds_nth f n / n powr s)
x)"
  defines "I ≡ contour_integral (L c T) (λs'. eval_fds f (s + s') * of_real
x powr s' / s')"
  defines "C1 ≡ 4 * (3 * pi + 2)"
  defines "C2 ≡ 2 * (3 * pi + 2) / ln 2 * C"
  defines "C3 ≡ (if c + Re s ≥ 0 then 2 else 1/2 :: real)"
  shows "dist (A x) I ≤
           C1 * C3 powr (c + Re s) * B (2 * x) / x powr Re s * (1 + x
/ T * (4 * ln T + 9)) +
           C2 * x powr c * (c + Re s - σa) powr (-α) / T"
⟨proof⟩

end

end
```

References

[1] G. Tenenbaum. *Introduction to Analytic and Probabilistic Number Theory*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 1995.

[2] E. C. Titchmarsh. *The Theory of Functions.* Oxford University Press, 2nd edition, 1939.