

# Lambert Series

Manuel Eberl

April 18, 2024

## Abstract

This entry provides a formalisation of *Lambert series*, i.e. series of the form  $L(a_n, q) = \sum_{n=1}^{\infty} a_n q^n / (1 - q^n)$  where  $a_n$  is a sequence of real or complex numbers. Proofs for all the basic properties are provided, such as

- the precise region in which  $L(a_n, q)$  converges
- the functional equation  $L(a_n, \frac{1}{q}) = -(\sum_{n=1}^{\infty} a_n) - L(a_n, q)$
- the power series expansion of  $L(a_n, q)$  at  $q = 0$
- the connection  $L(a_n, q) = \sum_{k=1}^{\infty} f(q^k)$  for  $f(z) = \sum_{n=1}^{\infty} a_n z^n$  that links a Lambert series to its “corresponding” power series
- connections to various number-theoretic functions, e.g. the divisor  $\sigma$  function via  $\sum_{n=1}^{\infty} \sigma_{\alpha}(n) q^n = L(n^{\alpha}, q)$

The formalisation mainly follows the chapter on Lambert series in Konrad Knopp’s classic textbook *Theory and Application of Infinite Series* [1] and includes all results presented therein.

## Contents

<b>1</b>	<b>Missing Library Material</b>	<b>3</b>
1.1	Miscellaneous . . . . .	3
1.2	Infinite sums . . . . .	3
1.3	Convergence radius . . . . .	5
1.4	Limits . . . . .	5
<b>2</b>	<b>Some Facts About Number-Theoretic Functions</b>	<b>7</b>
<b>3</b>	<b>Some Abel-Style Summation Tests</b>	<b>8</b>
<b>4</b>	<b>Lambert Series</b>	<b>9</b>
4.1	Definition . . . . .	10
4.2	Uniform convergence, continuity, holomorphicity . . . . .	11
4.3	Power series expansion . . . . .	14
4.3.1	Divisor $\sigma$ function . . . . .	15
4.3.2	Möbius $\mu$ function . . . . .	15
4.3.3	Euler's totient function $\varphi$ . . . . .	16
4.3.4	Mangoldt's $\Lambda$ function . . . . .	16
4.3.5	Liouville's $\lambda$ function . . . . .	17
4.4	Expressing a Lambert series in terms of a power series . . . . .	17
4.5	Connection to Euler's function . . . . .	19
4.6	Application: Fibonacci numbers . . . . .	21

# 1 Missing Library Material

```
theory Lambert_Series_Library
imports
  "HOL-Complex_Analysis.Complex_Analysis"
  "HOL-Library.Landau_Symbols"
  "HOL-Real_Asymp.Real_Asymp"
begin
```

## 1.1 Miscellaneous

```
lemma power_less_1_iff: "x ≥ 0 ⇒ (x :: real) ^ n < 1 ↔ x < 1 ∧
n > 0"
  ⟨proof⟩
```

```
lemma fls_nth_sum: "fls_nth (∑ x∈A. f x) n = (∑ x∈A. fls_nth (f x)
n)"
  ⟨proof⟩
```

```
lemma two_times_choose_two: "2 * (n choose 2) = n * (n - 1)"
  ⟨proof⟩
```

```
lemma Nats_not_empty [simp]: "ℕ ≠ {}"
  ⟨proof⟩
```

## 1.2 Infinite sums

```
lemma has_sum_iff: "(f has_sum S) A ↔ f summable_on A ∧ infsum f A
= S"
  ⟨proof⟩
```

```
lemma summable_on_reindex_bij_witness:
  assumes "∧ a. a ∈ S ⇒ i (j a) = a"
  assumes "∧ a. a ∈ S ⇒ j a ∈ T"
  assumes "∧ b. b ∈ T ⇒ j (i b) = b"
  assumes "∧ b. b ∈ T ⇒ i b ∈ S"
  assumes "∧ a. a ∈ S ⇒ h (j a) = g a"
  shows "g summable_on S ↔ h summable_on T"
  ⟨proof⟩
```

```
lemma has_sum_diff:
  fixes f g :: "'a ⇒ 'b::{topological_ab_group_add}"
  assumes <(f has_sum a) A>
  assumes <(g has_sum b) A>
  shows <((λx. f x - g x) has_sum (a - b)) A>
  ⟨proof⟩
```

```
lemma summable_on_diff:
  fixes f g :: "'a ⇒ 'b::{topological_ab_group_add}"
  assumes <f summable_on A>
```

```

    assumes <g summable_on A>
    shows <( $\lambda x. f x - g x$ ) summable_on A>
    <proof>

lemma infsum_diff:
  fixes f g :: "'a  $\Rightarrow$  'b::{topological_ab_group_add, t2_space}"
  assumes <f summable_on A>
  assumes <g summable_on A>
  shows <infsum ( $\lambda x. f x - g x$ ) A = infsum f A - infsum g A>
  <proof>

lemma summable_norm_add:
  assumes "summable ( $\lambda n. \text{norm } (f n)$ )" "summable ( $\lambda n. \text{norm } (g n)$ )"
  shows "summable ( $\lambda n. \text{norm } (f n + g n)$ )"
  <proof>

lemma summable_norm_diff:
  assumes "summable ( $\lambda n. \text{norm } (f n)$ )" "summable ( $\lambda n. \text{norm } (g n)$ )"
  shows "summable ( $\lambda n. \text{norm } (f n - g n)$ )"
  <proof>

lemma sums_imp_has_prod_exp:
  fixes f :: "'_  $\Rightarrow$  'a::{real_normed_field,banach}"
  assumes "f sums F"
  shows "( $\lambda n. \text{exp } (f n)$ ) has_prod exp F"
  <proof>

lemma telescope_summable_iff:
  fixes f :: "nat  $\Rightarrow$  'a::{real_normed_vector}"
  shows "summable ( $\lambda n. f (\text{Suc } n) - f n$ )  $\longleftrightarrow$  convergent f"
  <proof>

lemma telescope_summable_iff':
  fixes f :: "nat  $\Rightarrow$  'a::{real_normed_vector}"
  shows "summable ( $\lambda n. f n - f (\text{Suc } n)$ )  $\longleftrightarrow$  convergent f"
  <proof>

lemma norm_summable_mult_bounded:
  assumes "summable ( $\lambda n. \text{norm } (f n)$ )"
  assumes "g  $\in O(\lambda_. 1)$ "
  shows "summable ( $\lambda n. \text{norm } (f n * g n)$ )"
  <proof>

lemma summable_powser_comparison_test_bigo:
  fixes f g :: "nat  $\Rightarrow$  'a :: {real_normed_field, banach}"
  assumes "summable f" "g  $\in O(\lambda n. f n * c ^ n)$ " "norm c < 1"
  shows "summable ( $\lambda n. \text{norm } (g n)$ )"
  <proof>

```

```

lemma geometric_sums_gen:
  assumes "norm (x :: 'a :: real_normed_field) < 1"
  shows "(λn. x ^ (n + k)) sums (x ^ k / (1 - x))"
⟨proof⟩

lemma has_sum_geometric:
  fixes x :: "'a :: {real_normed_field, banach}"
  assumes "norm x < 1"
  shows "(λn. x ^ n) has_sum (x ^ m / (1 - x)) {m..}"
⟨proof⟩

lemma n_powser_sums:
  fixes q :: "'a :: {real_normed_field, banach}"
  assumes q: "norm q < 1"
  shows "(λn. of_nat n * q ^ n) sums (q / (1 - q) ^ 2)"
⟨proof⟩

```

### 1.3 Convergence radius

```

lemma tendsto_imp_conv_radius_eq:
  assumes "(λn. ereal (norm (f n) powr (1 / real n))) ⟶ c'" "c =
inverse c'"
  shows "conv_radius f = c"
⟨proof⟩

```

```

lemma conv_radius_powr_real: "conv_radius (λn. real n powr a) = 1"
⟨proof⟩

```

```

lemma conv_radius_one_over: "conv_radius (λn. 1 / of_nat n :: 'a :: {real_normed_field,
banach}) = 1"
⟨proof⟩

```

```

lemma conv_radius_mono:
  assumes "eventually (λn. norm (f n) ≥ norm (g n)) sequentially"
  shows "conv_radius f ≤ conv_radius g"
⟨proof⟩

```

```

lemma conv_radius_const [simp]:
  assumes "c ≠ 0"
  shows "conv_radius (λ_. c) = 1"
⟨proof⟩

```

```

lemma conv_radius_bigo_polynomial:
  assumes "f ∈ O(λn. of_nat n ^ k)"
  shows "conv_radius f ≥ 1"
⟨proof⟩

```

### 1.4 Limits

```

lemma oscillation_imp_not_tendsto:

```

```

    assumes "eventually ( $\lambda n. f (g n) \in A$ ) sequentially" "filterlim g F sequentially"
    assumes "eventually ( $\lambda n. f (h n) \in B$ ) sequentially" "filterlim h F sequentially"
    assumes "closed A" "closed B" "A  $\cap$  B = {}"
    shows " $\neg$ filterlim f (nhds c) F"
<proof>

```

```

lemma oscillation_imp_not_convergent:
  assumes "frequently ( $\lambda n. f n \in A$ ) sequentially"
  assumes "frequently ( $\lambda n. f n \in B$ ) sequentially"
  assumes "closed A" "closed B" "A  $\cap$  B = {}"
  shows " $\neg$ convergent f"
<proof>

```

```

lemma seq_bigo_1_iff:
  " $g \in O(\lambda_. 1) \iff$  bounded (range g)"
<proof>

```

```

lemma incseq_convergent':
  assumes "incseq (g :: nat  $\Rightarrow$  real)" " $g \in O(\lambda_. 1)$ "
  shows "convergent g"
<proof>

```

```

lemma decseq_convergent':
  assumes "decseq (g :: nat  $\Rightarrow$  real)" " $g \in O(\lambda_. 1)$ "
  shows "convergent g"
<proof>

```

```

lemma filterlim_of_int_iff:
  fixes c :: "'a :: real_normed_algebra_1"
  assumes "F  $\neq$  bot"
  shows "filterlim ( $\lambda x. \text{of\_int } (f x)$ ) (nhds c) F  $\iff$ 
    ( $\exists c'. c = \text{of\_int } c' \wedge$  eventually ( $\lambda x. f x = c'$ ) F)"
<proof>

```

```

lemma filterlim_of_nat_iff:
  fixes c :: "'a :: real_normed_algebra_1"
  assumes "F  $\neq$  bot"
  shows "filterlim ( $\lambda x. \text{of\_nat } (f x)$ ) (nhds c) F  $\iff$ 
    ( $\exists c'. c = \text{of\_nat } c' \wedge$  eventually ( $\lambda x. f x = c'$ ) F)"
<proof>

```

```

lemma uniform_limit_compose:
  assumes "uniform_limit B ( $\lambda x y. f x y$ ) ( $\lambda y. f' y$ ) F" " $\bigwedge y. y \in A \implies$ 
g y  $\in$  B"
  shows "uniform_limit A ( $\lambda x y. f x (g y)$ ) ( $\lambda y. f' (g y)$ ) F"
<proof>

```

```

lemma uniform_limit_const':
  assumes "filterlim f (nhds c) F"
  shows "uniform_limit A (λx y. f x) (λy. c) F"
⟨proof⟩

lemma uniform_limit_singleton_iff [simp]:
  "uniform_limit {x} f g F ↔ filterlim (λy. f y x) (nhds (g x)) F"
⟨proof⟩

end

```

## 2 Some Facts About Number-Theoretic Functions

```

theory Number_Theoretic_Functions_Extras
imports
  "Dirichlet_Series.Dirichlet_Series_Analysis"
  "Dirichlet_Series.Divisor_Count"
  Lambert_Series_Library
begin

lemma (in nat_power_field) nat_power_minus:
  "a ≠ 0 ∨ n ≠ 0 ⇒ nat_power n (-a) = inverse (nat_power n a)"
⟨proof⟩

lemma divisor_sigma_minus:
  fixes a :: "'a :: {nat_power_field, field_char_0}"
  shows "divisor_sigma (-a) n = divisor_sigma a n / nat_power n a"
⟨proof⟩

lemma norm_moebius_mu:
  "norm (moebius_mu n :: 'a :: {real_normed_algebra_1, comm_ring_1}) =
ind squarefree n"
⟨proof⟩

lemma conv_radius_nat_power: "conv_radius (λn. nat_power n a :: 'a ::
{nat_power_normed_field, banach}) = 1"
⟨proof⟩

lemma not_convergent_liouville_lambda:
  "¬convergent (liouville_lambda :: nat ⇒ 'a :: {real_normed_algebra,
comm_ring_1, semiring_char_0})"
⟨proof⟩

lemma conv_radius_liouville_lambda:
  "conv_radius (liouville_lambda :: nat ⇒ 'a :: {real_normed_field, banach})
= 1"
⟨proof⟩

```

```
lemma not_convergent_mangoldt: "¬convergent (mangoldt :: nat ⇒ 'a ::
{real_normed_algebra_1})"
⟨proof⟩
```

```
lemma conv_radius_mangoldt:
  "conv_radius (mangoldt :: nat ⇒ 'a :: {real_normed_field, banach})
= 1"
⟨proof⟩
```

```
lemma not_convergent_moebius_mu: "¬convergent (moebius_mu :: nat ⇒ 'a
:: real_normed_field)"
⟨proof⟩
```

```
lemma conv_radius_moebius_mu:
  "conv_radius (moebius_mu :: nat ⇒ 'a :: {real_normed_field, banach})
= 1"
⟨proof⟩
```

```
lemma not_convergent_totient:
  "¬convergent (λn. of_nat (totient n) :: 'a :: {real_normed_field, banach})"
⟨proof⟩
```

```
lemma conv_radius_totient:
  "conv_radius (λn. of_nat (totient n) :: 'a :: {real_normed_field, banach})
= 1"
⟨proof⟩
```

end

### 3 Some Abel-Style Summation Tests

```
theory Summation_Tests_More
  imports "HOL-Analysis.Analysis" "HOL-Library.Landau_Symbols" Lambert_Series_Library
begin
```

The following five summation tests are taken from Chapter 10 of Knopp's textbook [?]. He introduces a strong variant of Abel's summation test and then deduces from it four summation tests named after Abel, Dirichlet, du Bois-Reymond, and Dedekind.

```
lemma abel_partial_summation:
  fixes f g :: "nat ⇒ 'a :: comm_ring_1"
  defines "F ≡ (λn. ∑ k≤n. f k)"
  shows "(∑ r=n+1..n+k. f r * g r) =
          (∑ r=n+1..n+k. F r * (g r - g (Suc r))) -
          F n * g (Suc n) + F (n + k) * g (n + k + 1)"
  ⟨proof⟩
```

```

theorem abel_summation_test_strong:
  fixes f g :: "nat  $\Rightarrow$  'a :: {real_normed_field, banach}"
  defines "F  $\equiv$  ( $\lambda$ n.  $\sum_{k \leq n} f k$ )"
  assumes "summable ( $\lambda$ r. F r * (g r - g (Suc r)))"
  assumes "convergent ( $\lambda$ r. F r * g (Suc r))"
  shows "summable ( $\lambda$ r. f r * g r)"
  <proof>

corollary abel_summation_test:
  fixes f g :: "nat  $\Rightarrow$  real"
  assumes "summable f"
  assumes "incseq g" "g  $\in O(\lambda_. 1)$ "
  shows "summable ( $\lambda$ r. f r * g r)"
  <proof>

corollary dirichlet_summation_test:
  fixes f g :: "nat  $\Rightarrow$  real"
  assumes " $(\lambda$ n.  $\sum_{r \leq n} f r) \in O(\lambda_. 1)$ "
  assumes "decseq g" "g  $\in o(\lambda_. 1)$ "
  shows "summable ( $\lambda$ r. f r * g r)"
  <proof>

corollary dubois_reymond_summation_test:
  fixes f g :: "nat  $\Rightarrow$  'a :: {real_normed_field, banach}"
  assumes "summable f"
  assumes "summable ( $\lambda$ r. norm (g r - g (Suc r)))"
  shows "summable ( $\lambda$ r. f r * g r)"
  <proof>

corollary dedekind_summation_test:
  fixes f g :: "nat  $\Rightarrow$  'a :: {real_normed_field, banach}"
  assumes " $(\lambda$ n.  $\sum_{k \leq n} f k) \in O(\lambda_. 1)$ "
  assumes "summable ( $\lambda$ r. norm (g r - g (Suc r)))"
  assumes "g  $\in o(\lambda_. 1)$ "
  shows "summable ( $\lambda$ r. f r * g r)"
  <proof>

end

```

## 4 Lambert Series

```

theory Lambert_Series
  imports
    "HOL-Complex_Analysis.Complex_Analysis"
    "HOL-Real_Asymp.Real_Asymp"
    "Dirichlet_Series.Dirichlet_Series_Analysis"
    "Dirichlet_Series.Divisor_Count"
  Polylog.Polylog
  Lambert_Series_Library

```

begin

#### 4.1 Definition

Given any sequence  $a(n)$  for  $n \geq 1$ , the corresponding *Lambert series* is defined as

$$L(a, q) = \sum_{n=1}^{\infty} a(n) \frac{q^n}{1 - q^n} .$$

**definition** `lambert` :: "(nat  $\Rightarrow$  'a :: {real\_normed\_field, banach})  $\Rightarrow$  'a  $\Rightarrow$  'a" where

`"lambert a q =`  
`(let f = ( $\lambda$ n. a (Suc n) * q ^ (Suc n) / (1 - q ^ (Suc n))) in`  
`if summable f then  $\sum$ n. f n else 0)"`

**lemma** `lambert_eqI`:

`assumes "( $\lambda$ n. a (Suc n) * q ^ (Suc n) / (1 - q ^ (Suc n))) sums x"`  
`shows "lambert a q = x"`  
`<proof>`

**lemma** `lambert_cong [cong]`:

`"( $\bigwedge$ n. n > 0  $\implies$  a n = a' n)  $\implies$  q = q'  $\implies$  lambert a q = lambert a' q'"`  
`<proof>`

**lemma** `lambert_0 [simp]`: "lambert a 0 = 0"

`<proof>`

**lemma** `lambert_0' [simp]`: "lambert ( $\lambda$ \_. 0) q = 0"

`<proof>`

**lemma** `lambert_cmult`: "lambert ( $\lambda$ n. c \* a n) q = c \* lambert a q"

`<proof>`

**lemma** `lambert_cmult'`: "lambert ( $\lambda$ n. a n \* c) q = lambert a q \* c"

`<proof>`

**lemma** `lambert_uminus`: "lambert ( $\lambda$ n. -a n) q = -lambert a q"

`<proof>`

We will later see that if  $\sum_{n=1}^{\infty} a(n)$  exists then the Lambert series converges everywhere except on the unit circle; otherwise it has the same convergence radius as  $a$  (and that radius then has to be  $< 1$ ).

**definition** `lambert_conv_radius` :: "(nat  $\Rightarrow$  'a :: {banach, real\_normed\_field})  $\Rightarrow$  ereal"

`where "lambert_conv_radius a = (if summable a then  $\infty$  else conv_radius a)"`

```

lemma lambert_conv_radius_gt_1_iff: "lambert_conv_radius a > 1  $\longleftrightarrow$  summable
a"
<proof>

```

## 4.2 Uniform convergence, continuity, holomorphicity

We will now show some (uniform) convergence results for  $L(a, q)$ , which will then give us the holomorphicity and continuity of  $L(a, q)$ . We will also show some absolute summability results.

```

context
  fixes a :: "nat  $\Rightarrow$  'a :: {real_normed_field, banach}"
  fixes f :: "nat  $\Rightarrow$  'a  $\Rightarrow$  'a" and A :: "'a"
  defines "f  $\equiv$   $\lambda$ k q. a k * q ^ k / (1 - q ^ k)"
  defines "A  $\equiv$  ( $\sum$  n. a (Suc n))"
begin

```

Let  $a(n)$  have convergence radius  $r$ . In discs of radius  $\min(1, r)$ , the Lambert series for  $a(n)$  converges uniformly. This is a simple application of Weierstraß's  $M$  test.

```

lemma uniform_limit_lambert1_aux:
  fixes r :: real
  assumes "0 < r" "r < min 1 (conv_radius a)"
  shows "uniform_limit (ball 0 r) ( $\lambda$ n q. ( $\sum$  k<n. f (Suc k) q)) ( $\lambda$ q.
 $\sum$  k. f (Suc k) q) sequentially"
<proof>

```

```

lemma uniform_limit_lambert1:
  fixes r :: real
  assumes "0 < r" "r < min 1 (conv_radius a)"
  shows "uniform_limit (ball 0 r) ( $\lambda$ n q. ( $\sum$  k<n. f (Suc k) q)) (lambert
a) sequentially"
<proof>

```

Since  $a_n \frac{q^n}{1-q^n} = -a_n - a_n \frac{(\frac{1}{q})^n}{1-(\frac{1}{q})^n}$ , we can substitute  $q \mapsto \frac{1}{q}$  in the above uniform convergence result to deduce that uniform convergence also holds on any annulus  $r \leq |q| \leq R$  with  $1 < r < R$ .

```

lemma uniform_limit_lambert2:
  fixes r R :: real
  assumes r: "1 < r" "r < R"
  assumes "summable a"
  defines "D  $\equiv$  cball 0 R - ball 0 r"
  shows "uniform_limit D ( $\lambda$ n q. ( $\sum$  k<n. f (Suc k) q)) ( $\lambda$ q. -A - lambert
a (1 / q)) sequentially"
<proof>

```

With some more book-keeping, we show that the series converges uniformly

in all compact sets that do not touch the unit circle and, if  $\sum_{n=1}^{\infty} a(n)$  does not exist, lie fully within the convergence radius of  $a(n)$ . This is mentioned in Knopp's Theorem 259.

**theorem uniform\_limit\_lambert:**

assumes "compact X" "X  $\subseteq$  eball 0 (lambert\_conv\_radius a) - sphere 0 1"

shows "uniform\_limit X ( $\lambda n$  q. ( $\sum k < n$ . f (Suc k) q)) (lambert a) sequentially"  
(proof)

**lemma sums\_lambert:**

assumes "norm q < lambert\_conv\_radius a" "norm q  $\neq$  1"

shows "( $\lambda k$ . f (Suc k) q) sums lambert a q"  
(proof)

A side effect of this: the functional equation

$$L(a, \frac{1}{q}) = -(\sum_{n=1}^{\infty} a(n)) - L(a, q),$$

which is valid for all  $q$  with  $q \neq 0$  and  $|q| \neq 1$  if  $\sum_{n=1}^{\infty} a(n)$  exists.

**theorem lambert\_reciprocal:**

assumes "summable a" and "q  $\neq$  0" and "norm q  $\neq$  1"

shows "lambert a (1 / q) = -A - lambert a q"  
(proof)

**lemma summable\_lambert:**

assumes "norm q < lambert\_conv\_radius a" "norm q  $\neq$  1"

shows "summable ( $\lambda k$ . f k q)"  
(proof)

We have shown that the Lambert series for  $a(n)$  converges everywhere except on the unit circle if  $\sum_{n=1}^{\infty} a(n)$  exists, and it converges within the convergence radius of  $R$  of  $a(n)$  otherwise.

We will now show that within  $\min(1, R)$ , this convergence is absolute.

**lemma norm\_summable\_lambert:**

assumes "norm q < min 1 (conv\_radius a)"

shows "summable ( $\lambda k$ . norm (f k q))"  
(proof)

If additionally  $\sum_{k=1}^{\infty} a(k)$  converges absolutely, the absolute convergence of the Lambert series also holds everywhere.

**lemma norm\_summable\_lambert':**

assumes "summable ( $\lambda k$ . norm (a k))" and "norm q  $\neq$  1"

shows "summable ( $\lambda k$ . norm (f k q))"  
(proof)

**lemma abs\_summable\_on\_lambert:**

assumes "norm q < min 1 (conv\_radius a)"

shows "( $\lambda k. f k q$ ) abs\_summable\_on {1..}"  
 <proof>

lemma abs\_summable\_on\_lambert':  
 assumes "summable ( $\lambda k. norm (a k)$ )" and "norm q  $\neq$  1"  
 shows "( $\lambda k. f k q$ ) abs\_summable\_on {1..}"  
 <proof>

lemma summable\_on\_lambert:  
 assumes "norm q < min 1 (conv\_radius a)"  
 shows "( $\lambda k. f k q$ ) summable\_on {1..}"  
 <proof>

lemma has\_sum\_lambert:  
 assumes "norm q < min 1 (conv\_radius a)"  
 shows "(( $\lambda k. f k q$ ) has\_sum lambert a q) {1..}"  
 <proof>

We can also show a more precise convergence result that essentially fully reduces the question of convergence of a Lambert series to that of its “corresponding” power series:  $\sum_{k=1}^{\infty} a(k) \frac{q^k}{1-q^k}$  converges if and only if the “corresponding” power series  $\sum_{k=1}^{\infty} a(k)q^k$  converges or if  $\sum_{k=1}^{\infty} a(k)$  converges. This is Theorem 259 in Knopp’s book. A key ingredient, aside from the results we have amassed so far, is the du-Bois Reymond summation test.

theorem summable\_lambert\_iff:  
 assumes "norm q  $\neq$  1"  
 shows "summable ( $\lambda k. f k q$ )  $\longleftrightarrow$  summable a  $\vee$  summable ( $\lambda k. a k * q \wedge k$ )"  
 <proof>

end

lemma holomorphic\_lambert [holomorphic\_intros]:  
 assumes "X  $\subseteq$  eball 0 (lambert\_conv\_radius a) - sphere 0 1"  
 shows "lambert a holomorphic\_on X"  
 <proof>

lemma holomorphic\_lambert' [holomorphic\_intros]:  
 assumes "f holomorphic\_on A" " $\wedge z. z \in A \implies f z \in$  eball 0 (lambert\_conv\_radius a) - sphere 0 1"  
 shows "( $\lambda z. lambert a (f z)$ ) holomorphic\_on A"  
 <proof>

lemma analytic\_lambert [analytic\_intros]:  
 fixes a :: "nat  $\Rightarrow$  complex"  
 assumes "A  $\subseteq$  eball 0 (lambert\_conv\_radius a) - sphere 0 1"  
 shows "lambert a analytic\_on A"

*<proof>*

```
lemma analytic_lambert' [analytic_intros]:  
  assumes "f analytic_on A" "\z. z \in A \implies f z \in eball 0 (lambert_conv_radius  
a) - sphere 0 1"  
  shows "(λz. lambert a (f z)) analytic_on A"  
<proof>
```

```
lemma continuous_on_lambert [continuous_intros]:  
  fixes a :: "nat \Rightarrow 'a :: {real_normed_field, banach, heine_borel}"  
  assumes "A \subseteq eball 0 (lambert_conv_radius a) - sphere 0 1"  
  shows "continuous_on A (lambert a)"  
<proof>
```

```
lemma continuous_on_lambert' [continuous_intros]:  
  fixes a :: "nat \Rightarrow 'a :: {real_normed_field, banach, heine_borel}"  
  assumes "continuous_on A f" "\z. z \in A \implies f z \in eball 0 (lambert_conv_radius  
a) - sphere 0 1"  
  shows "continuous_on A (λz. lambert a (f z))"  
<proof>
```

```
lemma tendsto_lambert [tendsto_intros]:  
  fixes a :: "nat \Rightarrow 'a :: {real_normed_field, banach, heine_borel}"  
  assumes "(f \longrightarrow c) F" "c \in eball 0 (lambert_conv_radius a) - sphere  
0 1"  
  shows "((λx. lambert a (f x)) \longrightarrow lambert a c) F"  
<proof>
```

If  $\sum_{n=1}^{\infty} a(n)$  exists, the Lambert series of  $a(n)$  tends to it for  $q \rightarrow \infty$ .

```
lemma tendsto_lambert_at_infinity:  
  assumes "summable (a :: nat \Rightarrow 'a :: {real_normed_field, banach, heine_borel})"  
  shows "(lambert a \longrightarrow -(\sum n. a (Suc n))) at_infinity"  
<proof>
```

### 4.3 Power series expansion

By exchanging the order of summation, we can prove the power series expansion of  $L(a, q)$  as

$$L(a, q) = \sum_{n=1}^{\infty} (a * 1)(n) q^n$$

where  $*$  denotes the Dirichlet product, i.e.  $(a * 1)(n) = \sum_{d|n} a(d)$ .

This gives particularly nice results when  $a(n)$  is a number-theoretic function.

```
theorem has_sum_lambert_powser:  
  assumes "norm q < min 1 (conv_radius a)"  
  assumes "dirichlet_prod a (λ_. 1) = b"  
  shows "((λn. b n * q ^ n) has_sum lambert a q) {1..}"  
<proof>
```

```

lemma sums_lambert_powser:
  assumes "norm q < min 1 (conv_radius a)"
  assumes "dirichlet_prod a (λ_. 1) = b"
  shows "(λn. b n * q ^ n) sums lambert a q"
⟨proof⟩

```

```

lemma conv_radius_dirichlet_prod_1_ge:
  fixes a b :: "nat ⇒ 'a :: {real_normed_field, banach}"
  defines "b ≡ dirichlet_prod a (λ_. 1)"
  shows "conv_radius b ≥ min 1 (conv_radius a)"
⟨proof⟩

```

```

lemma sums_lambert_powser':
  assumes "norm q < min 1 (conv_radius a)"
  assumes "fds b = fds a * fds_zeta" "b 0 = 0"
  shows "(λn. b n * q ^ n) sums lambert a q"
⟨proof⟩

```

#### 4.3.1 Divisor $\sigma$ function

For any  $q$  with  $|q| < 1$  and any  $\alpha \in \mathbb{C}$ , we have

$$\sum_{n=1}^{\infty} \sigma_{\alpha}(n) q^n = \sum_{n=1}^{\infty} n^{\alpha} \frac{q^n}{1 - q^n}$$

where  $\sigma_{\alpha}(n)$  is the divisor  $\sigma$  function, i.e.  $\sigma_{\alpha}(n) = \sum_{d|n} d^{\alpha}$ .

```

lemma divisor_sigma_powser_conv_lambert:
  fixes α q :: "'a :: {nat_power_normed_field, banach}"
  assumes q: "norm q < 1"
  shows "(λn. divisor_sigma α n * q ^ n) sums lambert (λn. nat_power
n α) q"
⟨proof⟩

```

```

lemma divisor_count_powser_conv_lambert:
  fixes q :: "'a :: {nat_power_normed_field, banach}"
  assumes q: "norm q < 1"
  shows "(λn. of_nat (divisor_count n) * q ^ n) sums lambert (λ_. 1)
q"
⟨proof⟩

```

#### 4.3.2 Möbius $\mu$ function

For any  $q$  with  $|q| < 1$ , we have

$$\sum_{n=1}^{\infty} \mu(n) \frac{q^n}{1 - q^n} = q$$

where  $\mu(n)$  is Möbus'  $\mu$  function, which is 0 if  $n$  is not squarefree (i.e. contains the same prime factor more than once) and otherwise equal to  $(-1)^k$ , where  $k$  is the number of prime factors of  $n$ .

```
lemma lambert_moebius_mu:
  fixes q :: "'a :: {real_normed_field, banach}"
  assumes q: "norm q < 1"
  shows "lambert moebius_mu q = q"
  <proof>
```

```
lemma lambert_conv_radius_moebius_mu:
  "lambert_conv_radius (moebius_mu :: nat  $\Rightarrow$  'a :: {real_normed_field,
banach}) = 1"
  <proof>
```

### 4.3.3 Euler's totient function $\varphi$

For any  $q$  with  $|q| < 1$ , we have

$$\frac{q}{(1-q)^2} = \sum_{n=1}^{\infty} nq^n = \sum_{n=1}^{\infty} \varphi(n) \frac{q^n}{1-q^n}$$

where  $\varphi(n)$  is Euler's totient function, i.e. the number of positive integers not greater than  $n$  that are coprime to  $n$ .

```
lemma lambert_totient:
  fixes q :: "'a :: {real_normed_field, banach}"
  assumes q: "norm q < 1"
  shows "lambert ( $\lambda n$ . of_nat (totient n) :: 'a) q = q / (1 - q) ^ 2"
  <proof>
```

```
lemma lambert_conv_radius_totient:
  "lambert_conv_radius ( $\lambda n$ . of_nat (totient n) :: 'a :: {real_normed_field,
banach}) = 1"
  <proof>
```

### 4.3.4 Mangoldt's $\Lambda$ function

For any  $q$  with  $|q| < 1$ , we have

$$\sum_{n=1}^{\infty} \ln n q^n = \sum_{n=1}^{\infty} \Lambda(n) \frac{q^n}{1-q^n}$$

where  $\Lambda(n)$  is Mangoldt's function, which is defined to be equal to  $\log n$  if  $n$  is prime and 0 otherwise.

```
lemma lambert_mangoldt:
  fixes q :: "'a :: {real_normed_field, banach}"
  assumes q: "norm q < 1"
```

shows "(λn. of\_real (ln (Suc n)) \* q ^ (Suc n)) sums lambert mangoldt q"  
 ⟨proof⟩

lemma lambert\_conv\_radius\_mangoldt:  
 "lambert\_conv\_radius (mangoldt :: nat ⇒ 'a :: {real\_normed\_field, banach})  
 = 1"  
 ⟨proof⟩

#### 4.3.5 Liouville's λ function

For any  $q$  with  $|q| < 1$ , we have

$$\sum_{n=1}^{\infty} q^{n^2} = \sum_{n=1}^{\infty} \lambda(n) \frac{q^n}{1 - q^n}$$

where  $\lambda(n)$  is Liouville's function, which is defined as the number of prime factors of  $n$  (taking multiplicity into account).

lemma lambert\_liouville\_lambda:  
 fixes q :: "'a :: {real\_normed\_field, banach}"  
 assumes q: "norm q < 1"  
 shows "(λn. ind is\_square n \* q ^ n) sums lambert liouville\_lambda q"  
 ⟨proof⟩

lemma lambert\_liouville\_lambda':  
 fixes q :: "'a :: {real\_normed\_field, banach}"  
 assumes q: "norm q < 1"  
 shows "(λn. q ^ ((n+1) ^ 2)) sums lambert liouville\_lambda q"  
 ⟨proof⟩

lemma lambert\_conv\_radius\_liouville\_lambda:  
 "lambert\_conv\_radius (liouville\_lambda :: nat ⇒ 'a :: {real\_normed\_field, banach}) = 1"  
 ⟨proof⟩

#### 4.4 Expressing a Lambert series in terms of a power series

Let  $a(n)$  be a sequence of numbers. Then we can express the value of the Lambert series as an infinite sum in terms of the “normal” power series  $f(q) = \sum_{k=1}^{\infty} a(k)q^k$ :

$$L(a, q) = \sum_{n=1}^{\infty} f(q^n)$$

The proof is quite obvious, by expanding  $f(q^n)$  into its power series and then switching the order of summation.

This gives us a number of interesting relationships, including a connection between  $L(n^a, q)$  and the polylogarithm function  $\text{Li}_{-a}$ .

**theorem** `lambert_conv_powser_has_sum`:

assumes `q: "norm q < min 1 (conv_radius a)"` and `[simp]: "a 0 = 0"`

defines `"f ≡ (λq. ∑n. a n * q ^ n)"`

shows `"((λn. f (q ^ n)) has_sum lambert a q) {1..}"`

*<proof>*

**lemma** `lambert_conv_powser_has_sum'`:

assumes `"norm q < r" and "r ≤ 1"`

assumes `"∧q. norm q < r ⇒ (λn. a (Suc n) * q ^ Suc n) sums f q"`

shows `"((λn. f (q ^ n)) has_sum lambert a q) {1..}"`

*<proof>*

**lemma** `lambert_conv_powser_sums`:

assumes `q: "norm q < min 1 (conv_radius a)"` and `[simp]: "a 0 = 0"`

defines `"f ≡ (λq. ∑n. a n * q ^ n)"`

shows `"(λn. f (q ^ Suc n)) sums lambert a q"`

*<proof>*

**lemma** `lambert_conv_powser_sums'`:

assumes `"norm q < r" and "r ≤ 1"`

assumes `"∧q. norm q < r ⇒ (λn. a (Suc n) * q ^ Suc n) sums f q"`

shows `"(λn. f (q ^ Suc n)) sums lambert a q"`

*<proof>*

**lemma** `lambert_mult_exp_conv_powser_has_sum`:

assumes `"norm q < r" and "r ≤ 1" and c: "norm c ≤ 1"`

assumes `"∧q. norm q < r ⇒ (λn. a (Suc n) * q ^ Suc n) sums f q"`

shows `"((λn. f (c * q ^ n)) has_sum lambert (λn. c ^ n * a n) q) {1..}"`

*<proof>*

**lemma** `lambert_mult_exp_conv_powser_sums`:

assumes `"norm q < r" and "r ≤ 1" and c: "norm c ≤ 1"`

assumes `"∧q. norm q < r ⇒ (λn. a (Suc n) * q ^ Suc n) sums f q"`

shows `"((λn. f (c * q ^ Suc n)) sums lambert (λn. c ^ n * a n) q)"`

*<proof>*

**lemma** `lambert_power_int_has_sum_polylog_gen`:

fixes `q :: complex`

assumes `q: "norm q < 1" and c: "norm c ≤ 1"`

shows `"((λn. polylog (-a) (c * q ^ n)) has_sum lambert (λn. c ^ n * of_nat n powi a) q) {1..}"`

*<proof>*

**lemma** `has_sum_lambert_recip_complex_gen`:

fixes `q :: complex`

assumes `q: "norm q < 1" and c: "norm c ≤ 1"`

```

  shows "(λk. -ln (1 - c * q ^ k)) has_sum lambert (λn. c ^ n / of_nat
n) q) {1..}"
⟨proof⟩

```

```

lemma has_sum_lambert_recip_complex:
  fixes q :: complex
  assumes q: "norm q < 1"
  shows "(λk. -ln (1 - q ^ k)) has_sum lambert (λn. 1 / of_nat n)
q) {1..}"
⟨proof⟩

```

```

lemma has_sum_lambert_recip_complex':
  fixes q :: complex
  assumes q: "norm q < 1"
  shows "(λk. -ln (1 + q ^ k)) has_sum lambert (λn. (-1) ^ n / of_nat
n) q) {1..}"
⟨proof⟩

```

```

lemma has_sum_lambert_poly_complex:
  fixes q :: complex and a :: nat
  assumes q: "norm q < 1" and a: "a > 0"
  defines "E ≡ poly (eulerian_poly a)"
  shows "(λn. E (q ^ n) * q ^ n / (1 - q ^ n) ^ (a + 1)) has_sum
lambert (λn. complex_of_nat n ^ a) q) {1..}"
⟨proof⟩

```

```

lemma lambert_minus1_power_has_sum:
  assumes q: "norm q < 1"
  shows "(λn. q ^ n / (1 + q ^ n)) has_sum lambert (λn. (-1) ^ Suc
n) q) {1..}"
⟨proof⟩

```

```

lemma lambert_exp_has_sum:
  fixes q :: "'a :: {real_normed_field, banach}"
  assumes q: "norm q < 1" and a: "norm a ≤ 1"
  shows "(λn. a * q ^ n / (1 - a * q ^ n)) has_sum lambert (λn. a
^ n) q) {1..}"
⟨proof⟩

```

## 4.5 Connection to Euler's function

In this section, we show a connection between Lambert series and Euler's function:

$$\varphi(q) = \prod_{k=1}^{\infty} (1 - q^k)$$

(not to be confused with Euler's totient function, commonly denoted with  $\varphi(n)$ )

For this, we apply the results from the previous section to  $a(n) = \frac{1}{n}$  to obtain:

$$\sum_{k=1}^{\infty} \ln(1 - q^k) = -L\left(\frac{1}{n}, q\right)$$

```
lemma sums_lambert_recip_complex:
  fixes q :: complex
  assumes q: "norm q < 1"
  shows "(λk. -ln (1 - q ^ Suc k)) sums_lambert (λn. 1 / of_nat n)
q)"
  <proof>
```

```
lemma sums_lambert_recip_complex':
  fixes q :: complex
  assumes q: "norm q < 1"
  shows "(λk. -ln (1 + q ^ Suc k)) sums_lambert (λn. (-1)^n / of_nat
n) q)"
  <proof>
```

By exponentiating this, we get:

$$\varphi(q) \stackrel{\text{def}}{=} \prod_{n=1}^{\infty} (1 - q^n) = \exp\left(-\sum_{n=1}^{\infty} \frac{1}{n} \frac{q^n}{1 - q^n}\right)$$

In other words, the Lambert sum  $\sum \frac{1}{n} \frac{q^n}{1 - q^n}$  is a logarithm of Euler's function  $\varphi(q)$ .

Note that this does not show that this is *the* logarithm of  $\varphi(q)$ , but merely that it is *one* of the branches of the multi-valued logarithm of  $\varphi(q)$ . Nevertheless, we will – just like is typically in textbooks – ignore this in our informal explanations and write  $\ln \varphi(q)$ .

```
theorem euler_phi_conv_lambert:
  fixes q :: complex
  assumes q: "norm q < 1"
  shows "(λn. 1 - q ^ Suc n) has_prod exp (-lambert (λn. 1 / of_nat n)
q)"
  <proof>
```

With our general results on Lambert series, we also know that  $\ln \varphi(q)$  has the power series expansion

$$\ln \varphi(q) = -\sum_{n=1}^{\infty} \sigma_{-1}(n) q^n = -\sum_{n=1}^{\infty} \frac{\sigma_1(n)}{n} q^n .$$

```
lemma ln_euler_phi_powser:
  fixes q :: complex
  assumes q: "norm q < 1"
```

```

  shows "(λn. divisor_sigma (-1) n * q ^ n) sums lambert (λn. 1 / of_nat
n) q"
  ⟨proof⟩

```

```

lemma ln_euler_phi_powser':
  fixes q :: complex
  assumes q: "norm q < 1"
  shows "(λn. divisor_sum n / n * q ^ n) sums lambert (λn. 1 / of_nat
n) q"
  ⟨proof⟩

```

We also show the following variant of the above, also mentioned by Knopp:

```

theorem euler_phi_variant_conv_lambert:
  fixes q :: complex
  assumes q: "norm q < 1"
  shows "(λn. 1 + q ^ Suc n) has_prod exp (-lambert (λn. (-1) ^ n / of_nat
n) q)"
  ⟨proof⟩

```

## 4.6 Application: Fibonacci numbers

Lastly, we show a connection between the Fibonacci numbers and Lambert series, namely that:

$$\sum_{n=1}^{\infty} \frac{1}{F_n} = \sqrt{5} \left[ L\left(1, \frac{1}{2}(3 - \sqrt{5})\right) - L\left(1, \frac{1}{2}(7 - 3\sqrt{5})\right) \right]$$

```

lemma fib_closed_form_alt:
  defines "φ ≡ (1 + sqrt 5) / 2"
  shows "real (fib n) = (φ ^ n - (-1 / φ) ^ n) / sqrt 5"
  ⟨proof⟩

```

```

theorem sum_inv_even_fib_conv_lambert:
  defines "L ≡ lambert (λ_. 1)"
  shows "(λn. 1 / real (fib (2*n))) has_sum
  (sqrt 5 * (L ((3 - sqrt 5) / 2) - L ((7 - 3 * sqrt 5) / 2)))"
  {1..}"
  ⟨proof⟩

```

end

## References

- [1] K. Knopp. *Theorie und Anwendung der Unendlichen Reihen*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 2013.