

Theta Functions

Manuel Eberl

March 17, 2025

Abstract

This entry defines the Ramanujan theta function

$$f(a, b) = \sum_{n=-\infty}^{\infty} a^{\frac{n(n+1)}{2}} b^{\frac{n(n-1)}{2}}$$

and derives from it the more commonly known Jacobi theta function on the unit disc

$$\vartheta_{00}(w, q) = \sum_{n=-\infty}^{\infty} w^{2n} q^{n^2},$$

its version in the complex plane

$$\vartheta_{00}(z; \tau) = \sum_{n=-\infty}^{\infty} \exp(i\pi(2nz + n^2\tau))$$

as well as its half-period variants ϑ_{01} , ϑ_{10} , and ϑ_{11} .

The most notable single result in this work is the proof of Jacobi's triple product

$$\prod_{n=1}^{\infty} (1 - q^{2n})(1 + q^{2n-1}w^2)(1 + q^{2n-1}w^{-2}) = \sum_{k=-\infty}^{\infty} q^{k^2} w^{2k}$$

and its corollary, Euler's famous pentagonal number theorem:

$$\prod_{n=1}^{\infty} (1 - q^n) = \sum_{k=-\infty}^{\infty} (-1)^k q^{k(3k-1)/2}$$

Contents

| | |
|--|-----------|
| 1 Auxiliary material | 3 |
| 1.1 Limits | 3 |
| 1.2 Continuity and analyticity | 3 |
| 1.3 Formal power and Laurent series | 4 |
| 1.4 Infinite sums | 4 |
| 1.5 Miscellanea | 6 |
| 2 Conversion from the complex plane to the nome | 8 |
| 3 General theta functions | 11 |
| 3.1 The Ramanujan theta function | 11 |
| 3.2 The Jacobi theta function in terms of the nome | 14 |
| 3.3 The Jacobi theta function in the upper half of the complex plane | 16 |
| 3.4 The auxiliary theta functions in terms of the nome | 18 |
| 3.5 The auxiliary theta functions in the complex plane | 24 |
| 4 The Jacobi Triple Product | 28 |
| 4.1 Versions for Jacobi's theta function | 28 |
| 4.2 Version of Ramanujan's theta function | 29 |
| 4.3 The pentagonal number theorem | 29 |
| 4.4 (Non-)vanishing of theta functinos | 30 |
| 5 The theta nullwert functions | 31 |
| 5.1 The Maclaurin series of ϑ_3 and ϑ_4 | 34 |
| 5.2 Identities | 37 |
| 5.3 Properties of the nullwert functions on the real line | 38 |

1 Auxiliary material

```
theory Theta_Functions_Library
imports
  "HOL-Complex_Analysis.Complex_Analysis"
  "HOL-Computational_Algebra.Computational_Algebra"
begin
```

1.1 Limits

```
abbreviation "finite_sets_at_top ≡ finite_subsets_at_top UNIV"
```

```
lemma filterlim_atLeastAtMost_at_bot_at_top:
  fixes f g :: "'a ⇒ 'b :: linorder_topology"
  assumes "filterlim f at_bot F" "filterlim g at_top F"
  assumes [simp]: "¬(a b. finite {a..b::'b})"
  shows "filterlim (λx. {f x..g x}) finite_sets_at_top F"
  ⟨proof⟩
```

1.2 Continuity and analyticity

```
lemmas [continuous_intros del] = continuous_on_power_int
```

```
lemma continuous_on_power_int [continuous_intros]:
  fixes f :: "'a::topological_space ⇒ 'b::real_normed_div_algebra"
  assumes "continuous_on s f" and "n ≥ 0 ∨ (∀x∈s. f x ≠ 0)"
  shows "continuous_on s (λx. power_int (f x) n)"
  ⟨proof⟩
```

```
lemma analytic_on_powr [analytic_intros]:
  assumes "f analytic_on X" "g analytic_on X" "¬(x. x ∈ X ⇒ f x ∈ ℝ≤0)"
  shows "(λx. f x powr g x) analytic_on X"
  ⟨proof⟩
```

```
lemma holomorphic_on_powr [holomorphic_intros]:
  assumes "f holomorphic_on X" "g holomorphic_on X" "¬(x. x ∈ X ⇒ f x ∈ ℝ≤0)"
  shows "(λx. f x powr g x) holomorphic_on X"
  ⟨proof⟩
```

```
lemma continuous_powr_complex [continuous_intros]:
  assumes "continuous F f" "continuous F g"
  assumes "Re (f (netlimit F)) ≥ 0 ∨ Im (f (netlimit F)) ≠ 0"
  assumes "f (netlimit F) = 0 → Re (g (netlimit F)) > 0"
  shows "continuous F (λz. f z powr g z :: complex)"
  ⟨proof⟩
```

```
lemma continuous_powr_real [continuous_intros]:
  assumes "continuous F f" "continuous F g"
```

```

assumes "f (netlimit F) = 0 —> g (netlimit F) > 0 ∧ (∀F z in F. 0
≤ f z)"
shows   "continuous F (λz. f z powr g z :: real)"
⟨proof⟩

```

1.3 Formal power and Laurent series

```

lemma fps_nth_compose_linear [simp]:
  fixes f :: "'a :: comm_ring_1 fps"
  shows "fps_nth (fps_compose f (fps_const c * fps_X)) n = c ^ n * fps_nth
f n"
⟨proof⟩

lemma has_fps_expansionI:
  fixes f :: "'a :: {banach, real_normed_div_algebra} ⇒ 'a"
  assumes "eventually (λu. (λn. fps_nth F n * u ^ n) sums f u) (nhds
0)"
  shows   "f has_fps_expansion F"
⟨proof⟩

lemma fps_mult_numeral_left [simp]: "fps_nth (numeral c * f) n = numeral
c * fps_nth f n"
⟨proof⟩

lemma eval_fls_eq:
  assumes "N ≤ fls_subdegree F" "fls_subdegree F ≥ 0 ∨ z ≠ 0"
  assumes "(λn. fls_nth F (int n + N) * z powi (int n + N)) sums S"
  shows   "eval_fls F z = S"
⟨proof⟩

```

1.4 Infinite sums

```

no_notation Infinite_Set_Sum.abs_summable_on (infix "abs'_summable'_on"
50)

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

lemma summable_on_of_real:
  "f summable_on A ⇒ (λx. of_real (f x) :: 'a :: real_normed_algebra_1)
summable_on A"
⟨proof⟩

lemma has_sum_of_real_iff:
  "((λx. of_real (f x) :: 'a :: real_normed_div_algebra) has_sum (of_real
c)) A ←→
(f has_sum c) A"
⟨proof⟩

```

```

lemma has_sum_of_real:
  "(f has_sum S) A  $\implies$  (( $\lambda x.$  of_real (f x) :: 'a :: real_normed_algebra_1)
  has_sum of_real S) A"
  ⟨proof⟩

lemma has_sum_finite_iff:
  fixes S :: "'a :: {topological_comm_monoid_add, t2_space}"
  assumes "finite A"
  shows "(f has_sum S) A  $\longleftrightarrow$  S = ( $\sum_{x \in A}$  f x)"
  ⟨proof⟩

lemma has_sum_finite_neutralI:
  assumes "finite B" "B ⊆ A" " $\bigwedge x. x \in A - B \implies f x = 0$ " "c = ( $\sum_{x \in B}$  f x)"
  shows "(f has_sum c) A"
  ⟨proof⟩

lemma has_sum_strict_mono_neutral:
  fixes f :: "'a ⇒ 'b :: {ordered_ab_group_add, topological_ab_group_add,
  linorder_topology}"
  assumes <(f has_sum a) A> and "(g has_sum b) B"
  assumes < $\bigwedge x. x \in A \cap B \implies f x \leq g x$ >
  assumes < $\bigwedge x. x \in A - B \implies f x \leq 0$ >
  assumes < $\bigwedge x. x \in B - A \implies g x \geq 0$ >
  assumes < $x \in B$  <if  $x \in A$  then  $f x < g x$  else  $0 < g x$ >
  shows "a < b"
  ⟨proof⟩

lemma has_sum_strict_mono:
  fixes f :: "'a ⇒ 'b :: {ordered_ab_group_add, topological_ab_group_add,
  linorder_topology}"
  assumes <(f has_sum a) A> and "(g has_sum b) A"
  assumes < $\bigwedge x. x \in A \implies f x \leq g x$ >
  assumes < $x \in A$  < $f x < g x$ >
  shows "a < b"
  ⟨proof⟩

lemma has_sum_scaleR:
  fixes f :: "'a ⇒ 'b :: real_normed_vector"
  assumes "(f has_sum S) A"
  shows "(( $\lambda x.$  c *R f x) has_sum (c *R S)) A"
  ⟨proof⟩

lemma has_sum_scaleR_iff:
  fixes f :: "'a ⇒ 'b :: real_normed_vector"
  assumes "c ≠ 0"
  shows "(( $\lambda x.$  c *R f x) has_sum S) A  $\longleftrightarrow$  (f has_sum (S /R c)) A"
  ⟨proof⟩

```

```

lemma summable_on_reindex_bij_witness:
  assumes "Aa. a ∈ S ⟹ i (j a) = a"
  assumes "Aa. a ∈ S ⟹ j a ∈ T"
  assumes "Ab. b ∈ T ⟹ j (i b) = b"
  assumes "Ab. b ∈ T ⟹ i b ∈ S"
  assumes "Aa. a ∈ S ⟹ h (j a) = g a"
  shows   "g summable_on S ⟷ h summable_on T"
  ⟨proof⟩

lemma sums_nonneg_imp_has_sum_strong:
  assumes "f sums (S::real)" "eventually (λn. f n ≥ 0) sequentially"
  shows   "(f has_sum S) UNIV"
  ⟨proof⟩

lemma sums_nonneg_imp_has_sum:
  assumes "f sums (S::real)" and "A n. f n ≥ 0"
  shows   "(f has_sum S) UNIV"
  ⟨proof⟩

lemma summable_nonneg_imp_summable_on_strong:
  assumes "summable f" "eventually (λn. f n ≥ (0::real)) sequentially"
  shows   "f summable_on UNIV"
  ⟨proof⟩

lemma summable_nonneg_imp_summable_on:
  assumes "summable f" "A n. f n ≥ (0::real)"
  shows   "f summable_on UNIV"
  ⟨proof⟩

lemma Weierstrass_m_test_general':
  fixes f :: "'a ⇒ 'b ⇒ 'c :: banach"
  fixes M :: "'a ⇒ real"
  assumes norm_le: "A x y. x ∈ X ⟹ y ∈ Y ⟹ norm (f x y) ≤ M x"
  assumes has_sum: "A y. y ∈ Y ⟹ ((λx. f x y) has_sum S y) X"
  assumes summable: "M summable_on X"
  shows "uniform_limit Y (λX y. ∑x∈X. f x y) S (finite_subsets_at_top
X)"
  ⟨proof⟩

```

1.5 Miscellanea

```

lemma fraction_numeral_not_in_Ints [simp]:
  assumes "¬(numeral b :: int) dvd numeral a"
  shows   "numeral a / numeral b ∉ (Z :: 'a :: {division_ring, ring_char_0}
set)"
  ⟨proof⟩

lemma fraction_numeral_not_in_Ints' [simp]:
  assumes "b ≠ Num.One"

```

```

shows    "1 / numeral b ∈ (Z :: 'a :: {division_ring, ring_char_0} set)"
⟨proof⟩

lemmas [simp] = not_in_Ints_imp_not_in_nonpos_Ints minus_in_Ints_iff

lemma power_int_power: "(a ^ b :: 'a :: division_ring) powi c = a powi
(int b * c)"
⟨proof⟩

lemma power_int_power': "(a powi b :: 'a :: division_ring) ^ c = a powi
(b * int c)"
⟨proof⟩

lemma real_sqrt_abs': "sqrt (abs x) = abs (sqrt x)"
⟨proof⟩

lemma power_int_nonneg_exp: "n ≥ 0 ⇒ x powi n = x ^ nat n"
⟨proof⟩

lemma sin_npi_complex' [simp]: "sin (of_nat n * of_real pi) = 0"
⟨proof⟩

lemma cos_npi_complex' [simp]: "cos (of_nat n * of_real pi) = (-1) ^ n"
for n
⟨proof⟩

lemma cis_power_int: "cis x powi n = cis (of_int n * x)"
⟨proof⟩

lemma complex_cnj_power_int [simp]: "cnj (x powi n) = cnj x powi n"
⟨proof⟩

lemma uniform_limit_singleton: "uniform_limit {x} f g F ↔ ((λn. f
n x) ⟶ g x) F"
⟨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⟩

lemma is_square_mult_prime_left_iff:
assumes "prime p"

```

```

shows    "is_square (p * x)  $\longleftrightarrow$  p dvd x  $\wedge$  is_square (x div p)"
⟨proof⟩

lemma is_square_mult2_nat_iff:
  "is_square (2 * b :: nat)  $\longleftrightarrow$  even b  $\wedge$  is_square (b div 2)"
⟨proof⟩

lemma is_square_mult2_int_iff:
  "is_square (2 * b :: int)  $\longleftrightarrow$  even b  $\wedge$  is_square (b div 2)"
⟨proof⟩

lemma is_nth_power_mult_cancel_left:
  fixes a b :: "'a :: semiring_gcd"
  assumes "is_nth_power n a" "a  $\neq 0"
  shows    "is_nth_power n (a * b)  $\longleftrightarrow$  is_nth_power n b"
⟨proof⟩

lemma is_nth_power_mult_cancel_right:
  fixes a b :: "'a :: semiring_gcd"
  assumes "is_nth_power n b" "b  $\neq 0"
  shows    "is_nth_power n (a * b)  $\longleftrightarrow$  is_nth_power n a"
⟨proof⟩

end$$ 
```

2 Conversion from the complex plane to the nome

```

theory Nome
  imports "HOL-Complex_Analysis.Complex_Analysis" "HOL-Library.Going_To_Filter"
begin

definition to_nome :: "complex  $\Rightarrow$  complex"
  where "to_nome z = exp (i * of_real pi * z)"

lemma to_nome_nonzero [simp]: "to_nome z  $\neq 0"
⟨proof⟩

lemma norm_to_nome: "norm (to_nome z) = exp (-pi * Im z)"
⟨proof⟩

lemma to_nome_add: "to_nome (z + w) = to_nome z * to_nome w"
⟨proof⟩

lemma to_nome_diff: "to_nome (z - w) = to_nome z / to_nome w"
⟨proof⟩

lemma to_nome_minus: "to_nome (-z) = inverse (to_nome z)"
⟨proof⟩$ 
```

```

lemma to_nome_cnj: "to_nome (cnj z) = cnj (to_nome (-z))"
  ⟨proof⟩

lemma to_nome_power: "to_nome z ^ n = to_nome (of_nat n * z)"
  ⟨proof⟩

lemma to_nome_power_int: "to_nome z powi n = to_nome (of_int n * z)"
  ⟨proof⟩

lemma cis_conv_to_nome: "cis x = to_nome (of_real (x / pi))"
  ⟨proof⟩

lemma to_nome_powr:
  assumes "Re z ∈ {-1<..1}"
  shows   "to_nome z powr w = to_nome (z * w)"
  ⟨proof⟩

lemma to_nome_0 [simp]: "to_nome 0 = 1"
  ⟨proof⟩

lemma to_nome_1 [simp]: "to_nome 1 = -1"
  and to_nome_neg1 [simp]: "to_nome (-1) = -1"
  ⟨proof⟩

lemma to_nome_of_nat [simp]: "to_nome (of_nat n) = (-1) ^ n"
  ⟨proof⟩

lemma to_nome_of_int [simp]: "to_nome (of_int n) = (-1) powi n"
  ⟨proof⟩

lemma to_nome_one_half [simp]: "to_nome (1 / 2) = i"
  ⟨proof⟩

lemma to_nome_three_halves [simp]: "to_nome (3 / 2) = -i"
  ⟨proof⟩

lemma to_nome_eq_1_iff: "to_nome z = 1 ↔ (∃ n. even n ∧ z = of_int n)"
  ⟨proof⟩

lemma to_nome_eq_neg1_iff: "to_nome z = -1 ↔ (∃ n. odd n ∧ z = of_int n)"
  ⟨proof⟩

lemma to_nome_eq_1_iff': "to_nome z = 1 ↔ (z / 2) ∈ ℤ"
  ⟨proof⟩

lemma to_nome_eq_neg1_iff': "to_nome z = -1 ↔ ((z-1) / 2) ∈ ℤ"

```

```

⟨proof⟩

lemma to_nome_neg_one_half [simp]: "to_nome(-(1 / 2)) = -i"
⟨proof⟩

lemma to_nome_2 [simp]: "to_nome 2 = 1"
⟨proof⟩

lemma holomorphic_to_nome [holomorphic_intros]:
  "f holomorphic_on A ⟹ (λz. to_nome(f z)) holomorphic_on A"
⟨proof⟩

lemma analytic_to_nome [analytic_intros]:
  "f analytic_on A ⟹ (λz. to_nome(f z)) analytic_on A"
⟨proof⟩

lemma tendsto_to_nome [tendsto_intros]:
  assumes "(f ⟶ w) F"
  shows   "((λz. to_nome(f z)) ⟶ to_nome w) F"
⟨proof⟩

lemma continuous_on_to_nome [continuous_intros]:
  assumes "continuous_on A f"
  shows   "continuous_on A (λz. to_nome(f z))"
⟨proof⟩

lemma continuous_to_nome [continuous_intros]:
  assumes "continuous F f"
  shows   "continuous F (λz. to_nome(f z))"
⟨proof⟩

lemma tendsto_0_to_nome:
  assumes "filterlim (λx. Im(f x)) at_top F"
  shows   "filterlim (λx. to_nome(f x)) (nhds 0) F"
⟨proof⟩

lemma tendsto_0_to_nome': "(to_nome ⟶ 0) (Im going_to at_top)"
⟨proof⟩

lemma filterlim_at_0_to_nome:
  assumes "filterlim (λx. Im(f x)) at_top F"
  shows   "filterlim (λx. to_nome(f x)) (at 0) F"
⟨proof⟩

end

```

3 General theta functions

```
theory Theta_Functions
imports
  Nome
  "Combinatorial_Q_Analogues.Q_Binomial_Identities"
  Theta_Functions_Library
begin
```

3.1 The Ramanujan theta function

We define the other theta functions in terms of the Ramanujan theta function:

$$f(a, b) = \sum_{n=-\infty}^{\infty} a^{n(n+1)/2} b^{n(n-1)/2} \quad (\text{for } |ab| < 1)$$

This is, in some sense, more general than Jacobi's theta function: Jacobi's theta function can be expressed very easily in terms of Ramanujan's; the other direction is only straightforward in the real case. Due to the presence of square roots, the complex case becomes tedious due to branch cuts.

However, even in the complex case, results can be transferred from Jacobi's theta function to Ramanujan's by using the connection on the real line and then doing analytic continuation.

Some of the proofs below are loosely based on Ramanujan's lost notebook (as edited by Berndt [1]).

```
definition ramanujan_theta :: "'a :: {real_normed_field, banach} ⇒ 'a
⇒ 'a" where
  "ramanujan_theta a b =
  (if norm (a*b) < 1 then (∑ n. a powi (n*(n+1) div 2) * b powi (n*(n-1)
  div 2)) else 0)"

lemma ramanujan_theta_outside [simp]: "norm (a * b) ≥ 1 ⇒ ramanujan_theta
a b = 0"
  ⟨proof⟩

lemma uniform_limit_ramanujan_theta:
  fixes A :: "('a × 'a :: {real_normed_field, banach}) set"
  assumes "compact A" "¬ ∃ a b. (a, b) ∈ A ⇒ norm (a * b) < 1"
  shows "uniform_limit A (λX (a,b). ∑ n∈X. a powi (n*(n+1) div 2) *
  b powi (n*(n-1) div 2))
  (λ(a,b). ∑ n. a powi (n*(n+1) div 2) * b powi (n*(n-1)
  div 2))" (finite_subsets_at_top UNIV)"
  ⟨proof⟩

lemma has_sum_ramanujan_theta:
  assumes "norm (a*b) < 1"
```

```

shows    "((λn. a powi (n*(n+1) div 2) * b powi (n*(n-1) div 2)) has_sum
ramanujan_theta a b) UNIV"
⟨proof⟩

lemma ramanujan_theta_commute: "ramanujan_theta a b = ramanujan_theta
b a"
⟨proof⟩

lemma ramanujan_theta_0_left [simp]: "ramanujan_theta 0 b = 1 + b"
⟨proof⟩

lemma ramanujan_theta_0_right [simp]: "ramanujan_theta a 0 = 1 + a"
⟨proof⟩

lemma has_sum_ramanujan_theta1:
assumes "norm (a*b) < 1" and [simp]: "a ≠ 0"
shows    "((λn. a powi n * (a*b) powi (n*(n-1) div 2)) has_sum ramanujan_theta
a b) UNIV"
⟨proof⟩

lemma has_sum_ramanujan_theta2:
assumes "norm (a * b) < 1"
shows    "((λn. (a*b) powi (n*(n-1) div 2) * (a powi n + b powi n))
has_sum
(ramanujan_theta a b - 1)) {1..}"
⟨proof⟩

lemma ramanujan_theta_of_real:
"ramanujan_theta (of_real a) (of_real b) = of_real (ramanujan_theta
a b)"
⟨proof⟩

lemma ramanujan_theta_cnj:
"ramanujan_theta (cnj a) (cnj b) = cnj (ramanujan_theta a b)"
⟨proof⟩

lemma ramanujan_theta_holomorphic [holomorphic_intros]:
assumes "f holomorphic_on A" "g holomorphic_on A"
assumes "∀z. z ∈ A ⇒ norm (f z * g z) < 1" "open A"
shows    "(λz. ramanujan_theta (f z) (g z)) holomorphic_on A"
⟨proof⟩

lemma ramanujan_theta_analytic [analytic_intros]:
assumes "f analytic_on A" "g analytic_on A" "∀z. z ∈ A ⇒ norm (f
z * g z) < 1"
shows    "(λz. ramanujan_theta (f z) (g z)) analytic_on A"
⟨proof⟩

```

```

lemma tendsto_ramanujan_theta [tendsto_intros]:
  fixes f g :: "'a :: {real_normed_field, banach, heine_borel}"
  assumes "(f --> a) F" "(g --> b) F" "norm (a * b) < 1"
  shows   "((λz. ramanujan_theta (f z) (g z)) --> ramanujan_theta a
b) F"
⟨proof⟩

lemma continuous_on_ramanujan_theta [continuous_intros]:
  fixes f g :: "'a :: topological_space ⇒ 'b :: {real_normed_field, banach,
heine_borel}"
  assumes "continuous_on A f" "continuous_on A g" "λz. z ∈ A ⇒ norm
(f z * g z) < 1"
  shows   "continuous_on A (λz. ramanujan_theta (f z) (g z))"
⟨proof⟩

lemma continuous_ramanujan_theta [continuous_intros]:
  fixes f g :: "'a :: t2_space ⇒ 'b :: {real_normed_field, banach, heine_borel}"
  assumes "continuous F f" "continuous F g" "norm (f (netlimit F) * g
(netlimit F)) < 1"
  shows   "continuous F (λz. ramanujan_theta (f z) (g z))"
⟨proof⟩

lemma ramanujan_theta_1_left:
  "ramanujan_theta 1 a = 2 * ramanujan_theta a (a ^ 3)"
⟨proof⟩

lemma ramanujan_theta_1_right: "ramanujan_theta a 1 = 2 * ramanujan_theta
a (a ^ 3)"
⟨proof⟩

lemma ramanujan_theta_neg1_left [simp]: "ramanujan_theta (-1) a = 0"
⟨proof⟩

lemma ramanujan_theta_neg1_right [simp]: "ramanujan_theta a (-1) = 0"
⟨proof⟩

lemma ramanujan_theta_mult_power_int:
  assumes [simp]: "a ≠ 0" "b ≠ 0"
  shows   "ramanujan_theta a b =
    a powi (m*(m+1) div 2) * b powi (m*(m-1) div 2) *
    ramanujan_theta (a * (a*b) powi m) (b * (a*b) powi (-m))"
⟨proof⟩

lemma ramanujan_theta_mult:
  assumes [simp]: "a ≠ 0" "b ≠ 0"
  shows   "ramanujan_theta a b = a * ramanujan_theta (a^2 * b) (1 / a)"
⟨proof⟩

lemma ramanujan_theta_mult':

```

```

assumes [simp]: "a ≠ 0" "b ≠ 0"
shows   "ramanujan_theta a b = b * ramanujan_theta (1 / b) (a * b²)"
⟨proof⟩

```

3.2 The Jacobi theta function in terms of the nome

Based on Ramanujan's ϑ function, we introduce a version of Jacobi's ϑ function:

$$\vartheta(w, q) = \sum_{n=-\infty}^{\infty} w^n q^{n^2} \quad (\text{for } |q| < 1, w \neq 0)$$

Both parameters are still in terms of the nome rather than the complex plane. This has some advantages, and we can easily derive the other versions from it later.

```

definition jacobi_theta_nume :: "'a :: {real_normed_field, banach} ⇒ 'a
⇒ 'a" where
  "jacobi_theta_nume w q = (if w = 0 then 0 else ramanujan_theta (q*w)
  (q/w))"

lemma jacobi_theta_nume_0_left [simp]: "jacobi_theta_nume 0 q = 0"
⟨proof⟩

lemma jacobi_theta_nume_outside [simp]:
  assumes "norm q ≥ 1"
  shows   "jacobi_theta_nume w q = 0"
⟨proof⟩

lemma has_sum_jacobi_theta_nume:
  assumes "norm q < 1" and [simp]: "w ≠ 0"
  shows   "((λn. w powi n * q powi (n ^ 2)) has_sum jacobi_theta_nume
  w q) UNIV"
⟨proof⟩

lemma jacobi_theta_nume_same:
  "q ≠ 0 ⇒ jacobi_theta_nume q q = 2 * jacobi_theta_nume (1 / q ^ 2)
  (q ^ 4)"
⟨proof⟩

lemma jacobi_theta_nume_minus_same: "q ≠ 0 ⇒ jacobi_theta_nume (-q)
q = 0"
⟨proof⟩

lemma jacobi_theta_nume_minus_same': "q ≠ 0 ⇒ jacobi_theta_nume q
(-q) = 0"
⟨proof⟩

lemma jacobi_theta_nume_0_right [simp]: "w ≠ 0 ⇒ jacobi_theta_nume
w 0 = 1"

```

$\langle proof \rangle$

```
lemma jacobi_theta_nome_of_real:
  "jacobi_theta_nome (of_real w) (of_real q) = of_real (jacobi_theta_nome
w q)"
  ⟨proof⟩

lemma jacobi_theta_nome_cnj:
  "jacobi_theta_nome (cnj w) (cnj q) = cnj (jacobi_theta_nome w q)"
  ⟨proof⟩

lemma jacobi_theta_nome_minus_left:
  "jacobi_theta_nome (-w) q = jacobi_theta_nome w (-q)"
  ⟨proof⟩

lemma jacobi_theta_nome_quasiperiod':
  assumes [simp]: "w ≠ 0" "q ≠ 0"
  shows   "w * q * jacobi_theta_nome (q² * w) q = jacobi_theta_nome w
q"
  ⟨proof⟩

lemma jacobi_theta_nome_ii_left: "jacobi_theta_nome i q = jacobi_theta_nome
(-1) (q⁴)"
  ⟨proof⟩

lemma jacobi_theta_nome_quasiperiod:
  assumes [simp]: "w ≠ 0" "q ≠ 0"
  shows   "jacobi_theta_nome (q² * w) q = jacobi_theta_nome w q / (w
* q)"
  ⟨proof⟩

lemma jacobi_theta_nome_holomorphic [holomorphic_intros]:
  assumes "f holomorphic_on A" "g holomorphic_on A"
  assumes "¬(z. z ∈ A ⇒ norm (f z) ≠ 0)" "¬(z. z ∈ A ⇒ norm (g z)
< 1)" "open A"
  shows   "(λz. jacobi_theta_nome (f z) (g z)) holomorphic_on A"
  ⟨proof⟩

lemma jacobi_theta_nome_analytic [analytic_intros]:
  assumes "f analytic_on A" "g analytic_on A"
  assumes "¬(z. z ∈ A ⇒ f z ≠ 0)" "¬(z. z ∈ A ⇒ norm (g z) < 1)"
  shows   "(λz. jacobi_theta_nome (f z) (g z)) analytic_on A"
  ⟨proof⟩

lemma tendsto_jacobi_theta_nome [tendsto_intros]:
  fixes f g :: "'a ⇒ 'b :: {real_normed_field, banach, heine_borel}"
  assumes "(f ⟶ w) F" "(g ⟶ q) F" "w ≠ 0" "norm q < 1"
  shows   "((λz. jacobi_theta_nome (f z) (g z)) ⟶ jacobi_theta_nome
w q) F"
```

(proof)

```
lemma continuous_on_jacobi_theta_nome [continuous_intros]:
  fixes f g :: "'a :: topological_space ⇒ 'b :: {real_normed_field, banach,
heine_borel}"
  assumes "continuous_on A f" "continuous_on A g"
  assumes "∀z. z ∈ A ⇒ f z ≠ 0" "∀z. z ∈ A ⇒ norm (g z) < 1"
  shows   "continuous_on A (λz. jacobi_theta_nome (f z) (g z))"
(proof)

lemma continuous_jacobi_theta_nome [continuous_intros]:
  fixes f g :: "'a :: t2_space ⇒ 'b :: {real_normed_field, banach, heine_borel}"
  assumes "continuous F f" "continuous F g" "f (netlimit F) ≠ 0" "norm
(g (netlimit F)) < 1"
  shows   "continuous F (λz. jacobi_theta_nome (f z) (g z))"
(proof)
```

3.3 The Jacobi theta function in the upper half of the complex plane

We now define the more usual version of the Jacobi ϑ function, which takes two complex parameters z and t where z is arbitrary and t must lie in the upper half of the complex plane.

```
definition jacobi_theta_00 :: "complex ⇒ complex ⇒ complex" where
  "jacobi_theta_00 z t = jacobi_theta_nome (to_nome z ^ 2) (to_nome t)"

lemma jacobi_theta_00_outside: "Im t ≤ 0 ⇒ jacobi_theta_00 z t = 0"
(proof)

lemma has_sum_jacobi_theta_00:
  assumes "Im t > 0"
  shows   "((λn. to_nome (of_int n ^ 2 * t + 2 * of_int n * z)) has_sum
jacobi_theta_00 z t) UNIV"
(proof)

lemma sums_jacobi_theta_00:
  assumes "Im t > 0"
  shows   "((λn. if n = 0 then 1 else 2 * to_nome t ^ n ^ 2 *
cos (2 * of_nat n * of_real pi * z)) sums jacobi_theta_00
z t)"
(proof)

lemma jacobi_theta_00_holomorphic [holomorphic_intros]:
  assumes "f holomorphic_on A" "g holomorphic_on A" "∀z. z ∈ A ⇒ Im
(g z) > 0" "open A"
  shows   "(λz. jacobi_theta_00 (f z) (g z)) holomorphic_on A"
(proof)
```

```

lemma jacobi_theta_00_analytic [analytic_intros]:
  assumes "f analytic_on A" "g analytic_on A" " $\bigwedge z. z \in A \implies \text{Im } (g z) > 0"
  shows   " $(\lambda z. \text{jacobi\_theta\_00 } (f z) \ (g z))$  analytic_on A"
  ⟨proof⟩

lemma jacobi_theta_00_plus_half_left:
  "jacobi_theta_00 (z + 1 / 2) t = jacobi_theta_00 z (t + 1)"
  ⟨proof⟩

lemma jacobi_theta_00_plus_2_right: "jacobi_theta_00 z (t + 2) = jacobi_theta_00 z t"
  ⟨proof⟩

interpretation jacobi_theta_00_left: periodic_fun_simple' " $\lambda z. \text{jacobi\_theta\_00 } z t$ "
  ⟨proof⟩

interpretation jacobi_theta_00_right: periodic_fun_simple " $\lambda t. \text{jacobi\_theta\_00 } z t$ " 2
  ⟨proof⟩

lemma jacobi_theta_00_plus_quasiperiod:
  "jacobi_theta_00 (z + t) t = jacobi_theta_00 z t / to nome (t + 2 * z)"
  ⟨proof⟩

lemma jacobi_theta_00_quasiperiodic:
  "jacobi_theta_00 (z + of_int m + of_int n * t) t =
    jacobi_theta_00 z t / to nome (of_int (n^2) * t + 2 * of_int n * z)"
  ⟨proof⟩

lemma jacobi_theta_00_onequarter_left:
  "jacobi_theta_00 (1/4) t = jacobi_theta_00 (1/2) (4 * t)"
  ⟨proof⟩

lemma jacobi_theta_00_eq_0: "jacobi_theta_00 ((t + 1) / 2) t = 0"
  ⟨proof⟩

lemma jacobi_theta_00_eq_0': "jacobi_theta_00 ((of_int m + 1/2) + (of_int n + 1/2) * t) t = 0"
  ⟨proof⟩

lemma tendsto_jacobi_theta_00 [tendsto_intros]:
  assumes "(f —> w) F" "(g —> q) F" "Im q > 0"
  shows   " $((\lambda z. \text{jacobi\_theta\_00 } (f z) \ (g z)) —> \text{jacobi\_theta\_00 } w$  q) F"
  ⟨proof⟩$ 
```

```

lemma continuous_on_jacobi_theta_00 [continuous_intros]:
  assumes "continuous_on A f" "continuous_on A g"
  assumes "λz. z ∈ A ⇒ Im (g z) > 0"
  shows   "continuous_on A (λz. jacobi_theta_00 (f z) (g z))"
  ⟨proof⟩

lemma continuous_jacobi_theta_00 [continuous_intros]:
  assumes "continuous F f" "continuous F g" "Im (g (netlimit F)) > 0"
  shows   "continuous F (λz. jacobi_theta_00 (f z) (g z))"
  ⟨proof⟩

3.4 The auxiliary theta functions in terms of the nome

definition jacobi_theta_nome_00 :: "'a :: {real_normed_field, banach} ⇒
'a ⇒ 'a" where
  "jacobi_theta_nome_00 w q = jacobi_theta_nome (w^2) q"

definition jacobi_theta_nome_01 :: "'a :: {real_normed_field, banach} ⇒
'a ⇒ 'a" where
  "jacobi_theta_nome_01 w q = jacobi_theta_nome (-(w^2)) q"

definition jacobi_theta_nome_10 :: "'a :: {real_normed_field, banach, ln} ⇒
'a ⇒ 'a" where
  "jacobi_theta_nome_10 w q = w * q powr (1/4) * jacobi_theta_nome (w
^ 2 * q) q"

definition jacobi_theta_nome_11 :: "complex ⇒ complex ⇒ complex" where
  "jacobi_theta_nome_11 w q = i * w * q powr (1/4) * jacobi_theta_nome
(-(w ^ 2) * q) q"

lemmas jacobi_theta_nome_xx_defs =
  jacobi_theta_nome_00_def jacobi_theta_nome_01_def
  jacobi_theta_nome_10_def jacobi_theta_nome_11_def

lemma jacobi_theta_nome_00_outside [simp]: "norm q ≥ 1 ⇒ jacobi_theta_nome_00
w q = 0"
  and jacobi_theta_nome_01_outside [simp]: "norm q ≥ 1 ⇒ jacobi_theta_nome_01
w q = 0"
  and jacobi_theta_nome_10_outside [simp]: "norm q' ≥ 1 ⇒ jacobi_theta_nome_10
w' q' = 0"
  and jacobi_theta_nome_11_outside [simp]: "norm q'' ≥ 1 ⇒ jacobi_theta_nome_11
w'' q'' = 0"
  ⟨proof⟩

lemma jacobi_theta_nome_01_conv_00: "jacobi_theta_nome_01 w' q' = jacobi_theta_nome_00
w' (-q')"
  and jacobi_theta_nome_11_conv_10: "jacobi_theta_nome_11 w q = jacobi_theta_nome_10
(i * w) q"

```

```

⟨proof⟩

lemma jacobi_theta_nome_00_0_right [simp]: "w ≠ 0 ⟹ jacobi_theta_nome_00
w 0 = 1"
  and jacobi_theta_nome_01_0_right [simp]: "w ≠ 0 ⟹ jacobi_theta_nome_01
w 0 = 1"
  and jacobi_theta_nome_10_0_right [simp]: "jacobi_theta_nome_10 w' 0
= 0"
  and jacobi_theta_nome_11_0_right [simp]: "jacobi_theta_nome_11 w' 0
= 0"
⟨proof⟩

lemma jacobi_theta_nome_00_of_real:
  "jacobi_theta_nome_00 (of_real w :: 'a :: {banach, real_normed_field})
(of_real q) =
  of_real (jacobi_theta_nome_00 w q)"
and jacobi_theta_nome_01_of_real:
  "jacobi_theta_nome_01 (of_real w :: 'a) (of_real q) = of_real
(jacobi_theta_nome_01 w q)"
and jacobi_theta_nome_10_complex_of_real:
  "q ≥ 0 ⟹ jacobi_theta_nome_10 (complex_of_real w) (of_real
q) =
  of_real (jacobi_theta_nome_10 w q)"
⟨proof⟩

lemma jacobi_theta_nome_00_cnj:
  "jacobi_theta_nome_00 (cnj w) (cnj q) = cnj (jacobi_theta_nome_00
w q)"
and jacobi_theta_nome_01_cnj:
  "jacobi_theta_nome_01 (cnj w) (cnj q) = cnj (jacobi_theta_nome_01
w q)"
and jacobi_theta_nome_10_cnj:
  "(Im q = 0 ⟹ Re q ≥ 0) ⟹
  jacobi_theta_nome_10 (cnj w) (cnj q) = cnj (jacobi_theta_nome_10
w q)"
and jacobi_theta_nome_11_cnj:
  "(Im q = 0 ⟹ Re q ≥ 0) ⟹
  jacobi_theta_nome_11 (cnj w) (cnj q) = -cnj (jacobi_theta_nome_11
w q)"
⟨proof⟩

lemma has_sum_jacobi_theta_nome_00:
  assumes "norm q < 1" "w ≠ 0"
  shows "((λn. w powi (2*n) * q powi n2) has_sum jacobi_theta_nome_00
w q) UNIV"
⟨proof⟩

lemma has_sum_jacobi_theta_nome_01:

```

```

assumes "norm q < 1" "w ≠ 0"
shows   "((λn. (-1) powi n * w powi (2*n) * q powi n²) has_sum jacobi_theta_nome_01
w q) UNIV"
⟨proof⟩

lemma has_sum_jacobi_theta_nome_10':
assumes q: "norm q < 1" and [simp]: "w ≠ 0" "q ≠ 0"
shows   "((λn. w powi (2*n+1) * q powi (n*(n+1))) has_sum
(jacobi_theta_nome_10 w q / q powr (1/4))) UNIV"
⟨proof⟩

lemma has_sum_jacobi_theta_nome_10:
fixes q :: "'a :: {real_normed_field, banach, ln}"
assumes q: "norm q < 1" and [simp]: "w ≠ 0" "exp (ln q) = q"
shows   "((λn. w powi (2*n+1) * q powr (of_int n + 1 / 2) ^ 2) has_sum
(jacobi_theta_nome_10 w q)) UNIV"
⟨proof⟩

lemma has_sum_jacobi_theta_nome_11':
assumes q: "norm q < 1" and [simp]: "w ≠ 0" "q ≠ 0"
shows   "((λn. (-1) powi n * w powi (2*n+1) * q powi (n*(n+1))) has_sum
(jacobi_theta_nome_11 w q / (i * q powr (1/4)))) UNIV"
⟨proof⟩

lemma has_sum_jacobi_theta_nome_11:
assumes q: "norm q < 1" and [simp]: "w ≠ 0" "q ≠ 0"
shows   "((λn. i * (-1) powi n * w powi (2*n+1) * q powr (of_int n +
1/2) ^ 2) has_sum
(jacobi_theta_nome_11 w q)) UNIV"
⟨proof⟩

lemma jacobi_theta_nome_00_holomorphic [holomorphic_intros]:
assumes "f holomorphic_on A" "g holomorphic_on A"
assumes "¬(z. z ∈ A ⇒ norm (f z) ≠ 0)" "¬(z. z ∈ A ⇒ norm (g z)
< 1)" "open A"
shows   "(λz. jacobi_theta_nome_00 (f z) (g z)) holomorphic_on A"
⟨proof⟩

lemma jacobi_theta_nome_01_holomorphic [holomorphic_intros]:
assumes "f holomorphic_on A" "g holomorphic_on A"
assumes "¬(z. z ∈ A ⇒ norm (f z) ≠ 0)" "¬(z. z ∈ A ⇒ norm (g z)
< 1)" "open A"
shows   "(λz. jacobi_theta_nome_01 (f z) (g z)) holomorphic_on A"
⟨proof⟩

lemma jacobi_theta_nome_10_holomorphic [holomorphic_intros]:
assumes "f holomorphic_on A" "g holomorphic_on A"
assumes "¬(z. z ∈ A ⇒ norm (f z) ≠ 0)"

```

```

assumes " $\bigwedge z. z \in A \implies \text{norm}(g z) < 1 \wedge g z \notin \mathbb{R}_{\leq 0}$ " "open A"
shows "(\lambda z. jacobi_theta_nome_10 (f z) (g z)) holomorphic_on A"
⟨proof⟩

lemma jacobi_theta_nome_11_holomorphic [holomorphic_intros]:
assumes "f holomorphic_on A" "g holomorphic_on A"
assumes " $\bigwedge z. z \in A \implies \text{norm}(f z) \neq 0$ "
assumes " $\bigwedge z. z \in A \implies \text{norm}(g z) < 1 \wedge g z \notin \mathbb{R}_{\leq 0}$ " "open A"
shows "(\lambda z. jacobi_theta_nome_11 (f z) (g z)) holomorphic_on A"
⟨proof⟩

lemma jacobi_theta_nome_00_analytic [analytic_intros]:
assumes "f analytic_on A" "g analytic_on A"
assumes " $\bigwedge z. z \in A \implies \text{norm}(f z) \neq 0$ " " $\bigwedge z. z \in A \implies \text{norm}(g z) < 1$ "
shows "(\lambda z. jacobi_theta_nome_00 (f z) (g z)) analytic_on A"
⟨proof⟩

lemma jacobi_theta_nome_01_analytic [analytic_intros]:
assumes "f analytic_on A" "g analytic_on A"
assumes " $\bigwedge z. z \in A \implies \text{norm}(f z) \neq 0$ " " $\bigwedge z. z \in A \implies \text{norm}(g z) < 1$ "
shows "(\lambda z. jacobi_theta_nome_01 (f z) (g z)) analytic_on A"
⟨proof⟩

lemma jacobi_theta_nome_10_analytic [analytic_intros]:
assumes "f analytic_on A" "g analytic_on A"
assumes " $\bigwedge z. z \in A \implies \text{norm}(f z) \neq 0$ "
assumes " $\bigwedge z. z \in A \implies \text{norm}(g z) < 1 \wedge g z \notin \mathbb{R}_{\leq 0}$ "
shows "(\lambda z. jacobi_theta_nome_10 (f z) (g z)) analytic_on A"
⟨proof⟩

lemma jacobi_theta_nome_11_analytic [analytic_intros]:
assumes "f analytic_on A" "g analytic_on A"
assumes " $\bigwedge z. z \in A \implies \text{norm}(f z) \neq 0$ "
assumes " $\bigwedge z. z \in A \implies \text{norm}(g z) < 1 \wedge g z \notin \mathbb{R}_{\leq 0}$ "
shows "(\lambda z. jacobi_theta_nome_11 (f z) (g z)) analytic_on A"
⟨proof⟩

lemma tendsto_jacobi_theta_nome_00 [tendsto_intros]:
fixes f g :: "'a ⇒ 'b :: {real_normed_field, banach, heine_borel}"
assumes "(f ⟶ w)" "(g ⟶ q)" "w ≠ 0" "norm q < 1"
shows "((\lambda z. jacobi_theta_nome_00 (f z) (g z)) ⟶ jacobi_theta_nome_00
w q) F"
⟨proof⟩

lemma continuous_on_jacobi_theta_nome_00 [continuous_intros]:

```

```

fixes f g :: "'a :: topological_space ⇒ 'b :: {real_normed_field, banach,
heine_borel}"
assumes "continuous_on A f" "continuous_on A g"
assumes "¬(z. z ∈ A ⇒ f z ≠ 0)" "¬(z. z ∈ A ⇒ norm (g z) < 1)"
shows "continuous_on A (λz. jacobi_theta_nome_00 (f z) (g z))"
⟨proof⟩

lemma continuous_jacobi_theta_nome_00 [continuous_intros]:
fixes f g :: "'a :: t2_space ⇒ 'b :: {real_normed_field, banach, heine_borel}"
assumes "continuous F f" "continuous F g" "f (netlimit F) ≠ 0" "norm
(g (netlimit F)) < 1"
shows "continuous F (λz. jacobi_theta_nome_00 (f z) (g z))"
⟨proof⟩

lemma tendsto_jacobi_theta_nome_01 [tendsto_intros]:
fixes f g :: "'a ⇒ 'b :: {real_normed_field, banach, heine_borel}"
assumes "(f → w) F" "(g → q) F" "w ≠ 0" "norm q < 1"
shows "((λz. jacobi_theta_nome_01 (f z) (g z)) → jacobi_theta_nome_01
w q) F"
⟨proof⟩

lemma continuous_on_jacobi_theta_nome_01 [continuous_intros]:
fixes f g :: "'a :: topological_space ⇒ 'b :: {real_normed_field, banach,
heine_borel}"
assumes "continuous_on A f" "continuous_on A g"
assumes "¬(z. z ∈ A ⇒ f z ≠ 0)" "¬(z. z ∈ A ⇒ norm (g z) < 1)"
shows "continuous_on A (λz. jacobi_theta_nome_01 (f z) (g z))"
⟨proof⟩

lemma continuous_jacobi_theta_nome_01 [continuous_intros]:
fixes f g :: "'a :: t2_space ⇒ 'b :: {real_normed_field, banach, heine_borel}"
assumes "continuous F f" "continuous F g" "f (netlimit F) ≠ 0" "norm
(g (netlimit F)) < 1"
shows "continuous F (λz. jacobi_theta_nome_01 (f z) (g z))"
⟨proof⟩

lemma tendsto_jacobi_theta_nome_10_complex [tendsto_intros]:
fixes f g :: "complex ⇒ complex"
assumes "(f → w) F" "(g → q) F" "w ≠ 0" "norm q < 1" "q ∉
ℝ≤₀"
shows "((λz. jacobi_theta_nome_10 (f z) (g z)) → jacobi_theta_nome_10
w q) F"
⟨proof⟩

lemma continuous_on_jacobi_theta_nome_10_complex [continuous_intros]:
fixes f g :: "complex ⇒ complex"
assumes "continuous_on A f" "continuous_on A g"

```

```

assumes "¬(¬(Re (g z) > 0 ∨ Im (g z) ≠ 0))" "¬(¬(norm (g z) < 1 ∨
(Re (g z) = 0 ∧ Im (g z) = 0)))"
shows   "continuous_on A (λz. jacobi_theta_nome_10 (f z) (g z))"
⟨proof⟩

lemma continuous_jacobi_theta_nome_10_complex [continuous_intros]:
assumes "continuous F f" "continuous F g" "f (netlimit F) ≠ 0"
assumes "norm (g (netlimit F)) < 1" "Re (g (netlimit F)) > 0 ∨ Im (g
(netlimit F)) ≠ 0"
shows   "continuous F (λz. jacobi_theta_nome_10 (f z) (g z))"
⟨proof⟩

lemma tendsto_jacobi_theta_nome_10_real [tendsto_intros]:
fixes f g :: "real ⇒ real"
assumes "(f ----> w) F" "(g ----> q) F" "w ≠ 0" "norm q < 1" "q > 0"
shows   "((λz. jacobi_theta_nome_10 (f z) (g z)) ----> jacobi_theta_nome_10
w q) F"
⟨proof⟩

lemma continuous_on_jacobi_theta_nome_10_real [continuous_intros]:
fixes f g :: "real ⇒ real"
assumes "continuous_on A f" "continuous_on A g"
assumes "¬(¬(¬(λz. z ∈ A ⇒ f z ≠ 0) ∨ ¬(λz. z ∈ A ⇒ g z ∈ {0<..<1}))"
shows   "continuous_on A (λz. jacobi_theta_nome_10 (f z) (g z))"
⟨proof⟩

lemma continuous_jacobi_theta_nome_10_real [continuous_intros]:
fixes f g :: "real ⇒ real"
assumes "continuous F f" "continuous F g" "f (netlimit F) ≠ 0" "g (netlimit
F) ∈ {0<..<1}"
shows   "continuous F (λz. jacobi_theta_nome_10 (f z) (g z))"
⟨proof⟩

lemma tendsto_jacobi_theta_nome_11_complex [tendsto_intros]:
fixes f g :: "complex ⇒ complex"
assumes "(f ----> w) F" "(g ----> q) F" "w ≠ 0" "norm q < 1" "q ∉
ℝ≤₀"
shows   "((λz. jacobi_theta_nome_11 (f z) (g z)) ----> jacobi_theta_nome_11
w q) F"
⟨proof⟩

lemma continuous_on_jacobi_theta_nome_11_complex [continuous_intros]:
fixes f g :: "complex ⇒ complex"
assumes "continuous_on A f" "continuous_on A g"
assumes "¬(¬(¬(λz. z ∈ A ⇒ f z ≠ 0) ∨ ¬(λz. z ∈ A ⇒ norm (g z) < 1 ∨
(Re (g z) > 0 ∨ Im (g z) ≠ 0)))"
shows   "continuous_on A (λz. jacobi_theta_nome_11 (f z) (g z))"

```

```

⟨proof⟩

lemma continuous_jacobi_theta_nome_11_complex [continuous_intros]:
  assumes "continuous F f" "continuous F g" "f (netlimit F) ≠ 0"
  assumes "norm (g (netlimit F)) < 1" "Re (g (netlimit F)) > 0 ∨ Im (g
  (netlimit F)) ≠ 0"
  shows   "continuous F (λz. jacobi_theta_nome_11 (f z) (g z))"
  ⟨proof⟩

lemma tendsto_jacobi_theta_nome_11_real [tendsto_intros]:
  fixes f g :: "real ⇒ real"
  assumes "(f ⟶ w) F" "(g ⟶ q) F" "w ≠ 0" "norm q < 1" "q > 0"
  shows   "((λz. jacobi_theta_nome_11 (f z) (g z)) ⟶ jacobi_theta_nome_11
  w q) F"
  ⟨proof⟩

lemma continuous_on_jacobi_theta_nome_11_real [continuous_intros]:
  fixes f g :: "real ⇒ real"
  assumes "continuous_on A f" "continuous_on A g"
  assumes "¬(z ∈ A ⇒ f z ≠ 0)" "¬(z ∈ A ⇒ g z ∈ {0..<1})"
  shows   "continuous_on A (λz. jacobi_theta_nome_11 (f z) (g z))"
  ⟨proof⟩

lemma continuous_jacobi_theta_nome_11_real [continuous_intros]:
  fixes f g :: "real ⇒ real"
  assumes "continuous F f" "continuous F g" "f (netlimit F) ≠ 0" "g (netlimit
  F) ∈ {0..<1}"
  shows   "continuous F (λz. jacobi_theta_nome_11 (f z) (g z))"
  ⟨proof⟩

3.5 The auxiliary theta functions in the complex plane

definition jacobi_theta_01 :: "complex ⇒ complex ⇒ complex" where
  "jacobi_theta_01 z t = jacobi_theta_00 (z + 1/2) t"

definition jacobi_theta_10 :: "complex ⇒ complex ⇒ complex" where
  "jacobi_theta_10 z t = to_nome (z + t/4) * jacobi_theta_00 (z + t/2)
  t"

definition jacobi_theta_11 :: "complex ⇒ complex ⇒ complex" where
  "jacobi_theta_11 z t = to_nome (z + t/4 + 1/2) * jacobi_theta_00 (z
  + t/2 + 1/2) t"

lemma jacobi_theta_00_conv_nome:
  "jacobi_theta_00 z t = jacobi_theta_nome_00 (to_nome z) (to_nome t)"
  ⟨proof⟩

lemma jacobi_theta_01_conv_nome:
  "jacobi_theta_01 z t = jacobi_theta_nome_01 (to_nome z) (to_nome t)"

```

```

⟨proof⟩

lemma jacobi_theta_10_conv_nome:
  assumes "Re t ∈ {-1<....1}"
  shows   "jacobi_theta_10 z t = jacobi_theta_nome_10 (to_nome z) (to_nome
t)"
⟨proof⟩

lemma jacobi_theta_11_conv_nome:
  assumes "Re t ∈ {-1<....1}"
  shows   "jacobi_theta_11 z t = jacobi_theta_nome_11 (to_nome z) (to_nome
t)"
⟨proof⟩

lemma has_sum_jacobi_theta_01:
  assumes "Im t > 0"
  shows   "((λn. (-1) powi n * to_nome (of_int n ^ 2 * t + 2 * of_int
n * z)) has_sum jacobi_theta_01 z t) UNIV"
⟨proof⟩

lemma sums_jacobi_theta_01:
  assumes "Im t > 0"
  shows   "((λn. if n = 0 then 1 else 2 * (-1) ^ n * to_nome t ^ n^2
*
          cos (2 * of_nat n * of_real pi * z)) sums jacobi_theta_01
z t)"
⟨proof⟩

interpretation jacobi_theta_01_left: periodic_fun_simple' "λz. jacobi_theta_01
z t"
⟨proof⟩

interpretation jacobi_theta_01_right: periodic_fun_simple "λt. jacobi_theta_01
z t" 2
⟨proof⟩

lemma jacobi_theta_10_plus1_left: "jacobi_theta_10 (z + 1) t = -jacobi_theta_10
z t"
⟨proof⟩

lemma jacobi_theta_11_plus1_left: "jacobi_theta_11 (z + 1) t = -jacobi_theta_11
z t"
⟨proof⟩

lemma jacobi_theta_10_plus2_right: "jacobi_theta_10 z (t + 2) = i * jacobi_theta_10
z t"

```

```

⟨proof⟩

lemma jacobi_theta_11_plus2_right: "jacobi_theta_11 z (t + 2) = i * jacobi_theta_11
z t"
⟨proof⟩

lemma jacobi_theta_00_plus_half_left': "jacobi_theta_00 (z + 1/2) t =
jacobi_theta_01 z t"
⟨proof⟩

lemma jacobi_theta_01_plus_half_left: "jacobi_theta_01 (z + 1/2) t =
jacobi_theta_00 z t"
⟨proof⟩

lemma jacobi_theta_10_plus_half_left': "jacobi_theta_10 (z + 1/2) t =
jacobi_theta_11 z t"
⟨proof⟩

lemma jacobi_theta_11_plus_half_left': "jacobi_theta_11 (z + 1/2) t =
-jacobi_theta_10 z t"
⟨proof⟩

lemma tendsto_jacobi_theta_01 [tendsto_intros]:
assumes "(f ⟶ w) F" "(g ⟶ q) F" "Im q > 0"
shows   "((λz. jacobi_theta_01 (f z) (g z)) ⟶ jacobi_theta_01 w
q) F"
⟨proof⟩

lemma continuous_on_jacobi_theta_01 [continuous_intros]:
assumes "continuous_on A f" "continuous_on A g"
assumes "∀z. z ∈ A ⇒ Im (g z) > 0"
shows   "continuous_on A (λz. jacobi_theta_01 (f z) (g z))"
⟨proof⟩

lemma continuous_jacobi_theta_01 [continuous_intros]:
assumes "continuous F f" "continuous F g" "Im (g (netlimit F)) > 0"
shows   "continuous F (λz. jacobi_theta_01 (f z) (g z))"
⟨proof⟩

lemma holomorphic_jacobi_theta_01 [holomorphic_intros]:
assumes "f holomorphic_on A" "g holomorphic_on A" "∀z. z ∈ A ⇒ Im
(g z) > 0" "open A"
shows   "((λz. jacobi_theta_01 (f z) (g z)) holomorphic_on A"
⟨proof⟩

```

```

lemma analytic_jacobi_theta_01 [analytic_intros]:
  assumes "f analytic_on A" "g analytic_on A" " $\bigwedge z. z \in A \implies \operatorname{Im} (g z) > 0"
  shows   " $(\lambda z. \operatorname{jacobi\_theta\_01} (f z) (g z))$  analytic_on A"
  ⟨proof⟩

lemma tendsto_jacobi_theta_10 [tendsto_intros]:
  assumes "(f —> w) F" "(g —> q) F" "Im q > 0"
  shows   " $((\lambda z. \operatorname{jacobi\_theta\_10} (f z) (g z)) \xrightarrow{q} \operatorname{jacobi\_theta\_10} w)$ 
  ⟨proof⟩

lemma continuous_on_jacobi_theta_10 [continuous_intros]:
  assumes "continuous_on A f" "continuous_on A g"
  assumes " $\bigwedge z. z \in A \implies \operatorname{Im} (g z) > 0$ "
  shows   "continuous_on A ( $\lambda z. \operatorname{jacobi\_theta\_10} (f z) (g z))$ "
  ⟨proof⟩

lemma continuous_jacobi_theta_10 [continuous_intros]:
  assumes "continuous F f" "continuous F g" "Im (g (netlimit F)) > 0"
  shows   "continuous F ( $\lambda z. \operatorname{jacobi\_theta\_10} (f z) (g z))$ "
  ⟨proof⟩

lemma holomorphic_jacobi_theta_10 [holomorphic_intros]:
  assumes "f holomorphic_on A" "g holomorphic_on A" " $\bigwedge z. z \in A \implies \operatorname{Im} (g z) > 0$ " "open A"
  shows   " $(\lambda z. \operatorname{jacobi\_theta\_10} (f z) (g z))$  holomorphic_on A"
  ⟨proof⟩

lemma analytic_jacobi_theta_10 [analytic_intros]:
  assumes "f analytic_on A" "g analytic_on A" " $\bigwedge z. z \in A \implies \operatorname{Im} (g z) > 0$ "
  shows   " $(\lambda z. \operatorname{jacobi\_theta\_10} (f z) (g z))$  analytic_on A"
  ⟨proof⟩

lemma tendsto_jacobi_theta_11 [tendsto_intros]:
  assumes "(f —> w) F" "(g —> q) F" "Im q > 0"
  shows   " $((\lambda z. \operatorname{jacobi\_theta\_11} (f z) (g z)) \xrightarrow{q} \operatorname{jacobi\_theta\_11} w)$ 
  ⟨proof⟩

lemma continuous_on_jacobi_theta_11 [continuous_intros]:
  assumes "continuous_on A f" "continuous_on A g"
  assumes " $\bigwedge z. z \in A \implies \operatorname{Im} (g z) > 0$ "
  shows   "continuous_on A ( $\lambda z. \operatorname{jacobi\_theta\_11} (f z) (g z))$ "
  ⟨proof⟩$ 
```

```

lemma continuous_jacobi_theta_11 [continuous_intros]:
  assumes "continuous F f" "continuous F g" "Im (g (netlimit F)) > 0"
  shows   "continuous F ((\lambda z. jacobi_theta_11 (f z) (g z)))"
  ⟨proof⟩

lemma holomorphic_jacobi_theta_11 [holomorphic_intros]:
  assumes "f holomorphic_on A" "g holomorphic_on A" "\A z. z \in A \implies Im (g z) > 0" "open A"
  shows   "((\lambda z. jacobi_theta_11 (f z) (g z))) holomorphic_on A"
  ⟨proof⟩

lemma analytic_jacobi_theta_11 [analytic_intros]:
  assumes "f analytic_on A" "g analytic_on A" "\A z. z \in A \implies Im (g z) > 0"
  shows   "((\lambda z. jacobi_theta_11 (f z) (g z))) analytic_on A"
  ⟨proof⟩

end

```

4 The Jacobi Triple Product

```

theory Jacobi_Triple_Product
  imports Theta_Functions "Lambert_Series.Lambert_Series_Library"
begin

```

4.1 Versions for Jacobi's theta function

```
unbundle qpochhammer_inf_notation
```

The following follows the short proof given by Andrews [?], which makes use of two series expansions for $(a; b)_\infty$ and $1/(a; b)_\infty$ due to Euler.

We prove it for Jacobi's theta function and derive a version for Ramanujan's later on. One could possibly also adapt the prove to work for Ramanujan's version directly, which makes the transfer to Jacobi's function a bit easier. However, I chose to do it for Jacobi's version first in order to stay closer to the text by Andrews.

The proof is fairly straightforward; the only messy part is proving the absolute convergence of the double sum (which Andrews does not bother doing). This is necessary in order to allow the exchange of the order of summation.

```

theorem jacobi_theta_nome_triple_product_complex:
  fixes w q :: complex
  assumes "w \neq 0" "norm q < 1"
  shows   "jacobi_theta_nome w q = (q^2 ; q^2)_\infty * (-q*w ; q^2)_\infty * (-q/w ; q^2)_\infty"

```

(proof)

```
lemma jacobi_theta_nome_triple_product_real:
  fixes w q :: real
  assumes "w ≠ 0" "|q| < 1"
  shows   "jacobi_theta_nome w q = (q² ; q²)∞ * (-q*w ; q²)∞ * (-q/w
; q²)∞"
(proof)
```

4.2 Version of Ramanujan's theta function

The triple product for Ramanujan's theta function, which follows easily from the above one, has a particularly elegant form:

$$f(a, b) = (-a; ab)_∞ (-b; ab)_∞ (ab; ab)_∞$$

It follows directly from the version for Jacobi's theta function, although I again have to employ analytic continuation to avoid dealing with the branch cuts when converting Ramanujan's theta function to Jacobi's.

```
theorem ramanujan_theta_triple_product_complex:
  fixes a b :: complex
  assumes "norm (a * b) < 1"
  shows   "ramanujan_theta a b = (-a ; a*b)_∞ * (-b ; a*b)_∞ * (a*b ;
a*b)_∞"
(proof)

lemma ramanujan_theta_triple_product_real:
  fixes a b :: real
  assumes ab: "|a * b| < 1"
  shows   "ramanujan_theta a b = (-a ; a * b)_∞ * (-b ; a * b)_∞ * (a
* b ; a * b)_∞"
(proof)
```

4.3 The pentagonal number theorem

An easy corollary of this is the Pentagonal Number Theorem, which, in our notation, simply reads:

$$(q; q)_∞ = f(-q, -q²) = \theta(-\sqrt{q}, q\sqrt{q})$$

```
corollary pentagonal_number_theorem_complex:
  fixes q :: complex
  assumes q: "norm q < 1"
  shows   "(q; q)_∞ = ramanujan_theta (-q) (-(q²))"
(proof)

lemma pentagonal_number_theorem_real:
  fixes q :: real
```

```

assumes q: "|q| < 1"
shows   "(q; q)_∞ = ramanujan_theta (-q) (-(q²))"
⟨proof⟩

```

The following is the more explicit form of the Pentagonal Number Theorem usually found in textbooks:

$$\prod_{n=1}^{\infty} (1 - q^n) = \sum_{k=-\infty}^{\infty} (-1)^k q^{k(3k-1)/2}$$

The exponents $g_k = k(3k - 1)/2$ (for $k \in \mathbb{Z}$) are called the *generalised pentagonal numbers*.

```

corollary pentagonal_number_theorem_complex':
fixes q :: complex
assumes q: "norm q < 1"
shows "abs_convergent_prod (λn. 1 - q^(n+1))" (is ?th1)
and "(λk. (-1) powi k * q powi (k*(3*k-1) div 2)) abs_summable_on
UNIV" (is ?th2)
and "(∏n::nat. 1 - q^(n+1)) = (∑∞(k::int). (-1) powi k * q powi
(k * (3*k-1) div 2))" (is ?th3)
⟨proof⟩

```

4.4 (Non-)vanishing of theta functinos

A corollary of the Jacobi triple product: the Jacobi theta function has no zeros apart from the “obvious” ones, i.e. the ones at the center of the cells of the lattice generated by the period 1 and the quasiperiod t .

```

corollary jacobi_theta_00_eq_0_iff_complex:
fixes z t :: complex
assumes "Im t > 0"
shows   "jacobi_theta_00 z t = 0 ↔ (∃m n. z = (of_int m + 1/2) +
(of_int n + 1/2) * t)"
⟨proof⟩

```

```

lemma jacobi_theta_00_nonzero:
assumes z: "Im t > 0" and "Im z / Im t - 1 / 2 ∉ ℤ"
shows   "jacobi_theta_00 z t ≠ 0"
⟨proof⟩

```

```

lemma jacobi_theta_00_0_left_nonzero:
assumes "Im t > 0"
shows   "jacobi_theta_00 0 t ≠ 0"
⟨proof⟩

```

```

lemma jacobi_theta_nome_nonzero_complex:
fixes q w :: complex
assumes [simp]: "w ≠ 0" "norm q < 1"
assumes "q = 0 ∨ (ln (norm w) / ln (norm q) - 1) / 2 ∉ ℤ"

```

```

shows      "jacobi_theta_nome w q ≠ 0"
⟨proof⟩

lemma jacobi_theta_nome_q_q_nonzero_complex:
  assumes "norm (q::complex) < 1" "q ≠ 0"
  shows   "jacobi_theta_nome q q ≠ 0"
⟨proof⟩

lemma jacobi_theta_nome_nonzero_real:
  fixes q w :: real
  assumes [simp]: "w ≠ 0" "norm q < 1" and "(ln |w| / ln |q| - 1) / 2
  ≠ ZZ"
  shows   "jacobi_theta_nome w q ≠ 0"
⟨proof⟩

lemma jacobi_theta_nome_1_left_nonzero_complex:
  assumes "norm (q :: complex) < 1"
  shows   "jacobi_theta_nome 1 q ≠ 0"
⟨proof⟩

lemma jacobi_theta_nome_1_left_nonzero_real:
  assumes "|q::real| < 1"
  shows   "jacobi_theta_nome 1 q ≠ 0"
⟨proof⟩

unbundle no_qpochhammer_inf_notation

end

```

5 The theta nullwert functions

```

theory Theta_Nullwert
  imports "Sum_Of_Squares_Count.Sum_Of_Squares_Count" Jacobi_Triple_Product
begin

```

The theta nullwert function (nullwert being German for “zero value”) are the four functions $\vartheta_{xy}(z; \tau)$ with $z = 0$. However, they are very commonly denoted in terms of the nome instead, corresponding to $\vartheta_{xy}(w, q)$ with $w = 1$. It is easy to see that $\vartheta_{11}(0; \tau) = \vartheta_{11}(1, q)$ is identically zero and therefore uninteresting. The remaining three functions $\vartheta_{10}(0, q)$, $\vartheta_{00}(0, q)$, and $\vartheta_{01}(0, q)$ are denoted $\vartheta_2(q)$, $\vartheta_3(q)$, and $\vartheta_4(q)$.

It is also not hard to see that in fact $\vartheta_4(q) = \vartheta_3(-q)$, but we still introduce separate notation for ϑ_4 since it is very commonly used in the literature.

```

lemma jacobi_theta_nome_11_1_left [simp]: "jacobi_theta_nome_11 1 q =
0"
⟨proof⟩

```

```

abbreviation jacobi_theta_nw_10 :: "'a :: {real_normed_field, banach,
ln} ⇒ 'a" where
"jacobi_theta_nw_10 q ≡ jacobi_theta_nome_10 1 q"

abbreviation jacobi_theta_nw_00 :: "'a :: {real_normed_field, banach} ⇒ 'a" where
"jacobi_theta_nw_00 q ≡ jacobi_theta_nome_00 1 q"

abbreviation jacobi_theta_nw_01 :: "'a :: {real_normed_field, banach} ⇒ 'a" where
"jacobi_theta_nw_01 q ≡ jacobi_theta_nome_01 1 q"

bundle jacobi_theta_nw_notation
begin
notation jacobi_theta_nw_10 ("ϑ₂")
notation jacobi_theta_nw_00 ("ϑ₃")
notation jacobi_theta_nw_01 ("ϑ₄")
end

bundle no_jacobi_theta_nw_notation
begin
no_notation jacobi_theta_nw_10 ("ϑ₂")
no_notation jacobi_theta_nw_00 ("ϑ₃")
no_notation jacobi_theta_nw_01 ("ϑ₄")
end

unbundle jacobi_theta_nw_notation

lemma jacobi_theta_nw_10_0 [simp]: "ϑ₂ 0 = 0"
and jacobi_theta_nw_00_0 [simp]: "ϑ₃ 0 = 1"
and jacobi_theta_nw_01_0 [simp]: "ϑ₄ 0 = 1"
⟨proof⟩

lemma jacobi_theta_nw_01_conv_00: "ϑ₄ q = ϑ₃ (-q)"
⟨proof⟩

lemma jacobi_theta_nw_10_of_real:
"y ≥ 0 ⇒ ϑ₂ (complex_of_real y) = of_real (ϑ₂ y)"
and jacobi_theta_nw_00_of_real: "ϑ₃ (of_real x) = of_real (ϑ₃ x)"
and jacobi_theta_nw_01_of_real: "ϑ₄ (of_real x) = of_real (ϑ₄ x)"
⟨proof⟩

lemma jacobi_theta_nw_10_cnj:
"(Im q = 0 ⇒ Re q ≥ 0) ⇒ ϑ₂ (cnj q) = cnj (ϑ₂ q)"
and jacobi_theta_nw_00_cnj: "ϑ₃ (cnj q) = cnj (ϑ₃ q)"
and jacobi_theta_nw_01_cnj: "ϑ₄ (cnj q) = cnj (ϑ₄ q)"

```

$\langle proof \rangle$

The nullwerte have the following definitions as infinite sums:

$$\begin{aligned}\vartheta_2(q) &= \sum_{-\infty}^{\infty} q^{(n+\frac{1}{2})^2} \\ \vartheta_3(q) &= \sum_{-\infty}^{\infty} q^{n^2} \\ \vartheta_4(q) &= \sum_{-\infty}^{\infty} (-1)^n q^{n^2}\end{aligned}$$

```
lemma has_sum_jacobi_theta_nw_10_complex:
  assumes "norm (q :: complex) < 1"
  shows   "((\lambda n. q powr ((of_int n + 1 / 2) ^ 2)) has_sum \vartheta_2 q) UNIV"
⟨proof⟩

lemma has_sum_jacobi_theta_nw_10_real:
  assumes "q ∈ {0<..<1::real}"
  shows   "((\lambda n. q powr ((of_int n + 1 / 2) ^ 2)) has_sum \vartheta_2 q) UNIV"
⟨proof⟩

lemma has_sum_jacobi_theta_nw_00:
  assumes "norm q < 1"
  shows   "((\lambda n. q powi (n ^ 2)) has_sum \vartheta_3 q) UNIV"
⟨proof⟩

lemma has_sum_jacobi_theta_nw_01:
  assumes "norm q < 1"
  shows   "((\lambda n. (-1) powi n * q powi (n ^ 2)) has_sum \vartheta_4 q) UNIV"
⟨proof⟩
```

The theta nullwert functions do not vanish (except for $\vartheta_2(0) = 0$).

```
lemma jacobi_theta_00_nw_nonzero_complex: "norm (q::complex) < 1 ==>
\vartheta_3 q ≠ 0"
⟨proof⟩

lemma jacobi_theta_01_nw_nonzero_complex: "norm (q::complex) < 1 ==>
\vartheta_4 q ≠ 0"
⟨proof⟩

lemma jacobi_theta_10_nw_nonzero_complex:
  assumes "q ≠ 0" "norm (q::complex) < 1"
  shows   "\vartheta_2 q ≠ 0"
⟨proof⟩

lemma jacobi_theta_00_nw_nonzero_real: "|q::real| < 1 ==> \vartheta_3 q ≠ 0"
and jacobi_theta_01_nw_nonzero_real: "|q::real| < 1 ==> \vartheta_4 q ≠ 0"
```

```

and jacobi_theta_10_nw_nonzero_real: "q ∈ {0.. $\infty$ } ⇒ q ≠ 0 ⇒  $\vartheta_2$ 
q ≠ 0"
⟨proof⟩

```

5.1 The Maclaurin series of ϑ_3 and ϑ_4

It is easy to see from the above infinite sums that $\vartheta_3(q)$ and $\vartheta_4(q)$ have the Maclaurin series

$$1 + 2 \sum_{n=1}^{\infty} [\exists i. n = i^2] c^n q^n$$

for $c = 1$ and $c = -1$, respectively.

In other words, $\vartheta_3(q)$ is the generating function of the number $r_1(n)$ of ways to write an integer n as a square of an integer -1 for $n = 0, 2$ if n is a non-zero perfect square, and 0 otherwise.

Consequently, $\vartheta_3(q)^k$ is the generating function of the number $r_k(n)$ of ways to write an integer n as a square of k integers. The function $r_k(n)$ is written as `count_sos k n` in Isabelle.

```

definition fps_jacobi_theta_nw :: "'a :: field ⇒ 'a fps" where
  "fps_jacobi_theta_nw c = Abs_fps (λn. if n = 0 then 1 else if is_square
  n then 2 * c ^ n else 0)"

theorem fps_jacobi_theta_power_eq:
  "fps_jacobi_theta_nw c ^ k = Abs_fps (λn. of_nat (count_sos k n) * c
  ^ n)"
⟨proof⟩

corollary fps_jacobi_theta_altdef:
  "fps_jacobi_theta_nw c = Abs_fps (λn. of_nat (count_sos 1 n) * c ^ n)"
⟨proof⟩

lemma norm_summable_fps_jacobi_theta:
  fixes q :: "'a :: {real_normed_field, banach}"
  assumes "norm (c * q) < 1"
  shows   "summable (λn. norm (fps_nth (fps_jacobi_theta_nw c) n * q
  ^ n))"
⟨proof⟩

lemma summable_fps_jacobi_theta:
  fixes q :: "'a :: {real_normed_field, banach}"
  assumes "norm (c * q) < 1"
  shows   "summable (λn. fps_nth (fps_jacobi_theta_nw c) n * q ^ n)"
⟨proof⟩

lemma summable_ints_symmetric:
  fixes f :: "int ⇒ 'a :: {real_normed_vector, banach}"
  assumes "summable (λn. norm (f (int n)))"

```

```

assumes " $\bigwedge n. f(-n) = f n$ "
shows "f abs_summable_on UNIV" and "summable (( $\lambda n. \text{norm} ((\text{if } n = 0 \text{ then } 1 \text{ else } 2) *_R f (\text{int } n)))$ )"
⟨proof⟩

lemma has_sum_ints_symmetric_iff:
  fixes f :: "int ⇒ 'a :: {banach, real_normed_vector}"
  assumes " $\bigwedge n. f(-n) = f n$ "
  shows "(f has_sum S) UNIV ←→ (( $\lambda n. (\text{if } n = 0 \text{ then } 1 \text{ else } 2) *_R f (\text{int } n)$ ) has_sum S) UNIV"
⟨proof⟩

lemma sums_jacobi_theta_nw_00:
  assumes "norm q < 1"
  shows "(( $\lambda n. \text{fps_nth} (\text{fps_jacobi_theta_nw } 1) n * q ^ n$ ) sums  $\vartheta_3(q)$ )"
⟨proof⟩

lemma sums_jacobi_theta_nw_01:
  assumes "norm q < 1"
  shows "(( $\lambda n. \text{fps_nth} (\text{fps_jacobi_theta_nw } (-1)) n * q ^ n$ ) sums  $\vartheta_4(q)$ )"
⟨proof⟩

lemma has_fps_expansion_jacobi_theta_3 [fps_expansion_intros]:
  " $\vartheta_3$  has_fps_expansion fps_jacobi_theta_nw 1"
⟨proof⟩

lemma has_fps_expansion_jacobi_theta_4 [fps_expansion_intros]:
  " $\vartheta_4$  has_fps_expansion fps_jacobi_theta_nw (-1)"
⟨proof⟩

lemma fps_conv_radius_jacobi_theta_nw [simp]:
  fixes c :: "'a :: {banach, real_normed_field}"
  shows "fps_conv_radius (fps_jacobi_theta_nw c) = 1 / ereal (norm c)"
⟨proof⟩

```

Recall that $\vartheta_2(q) = q^{1/4}\vartheta(q, q)$. Since the factor $q^{1/4}$ has a branch cut, it is somewhat unpleasant to deal with and we will concentrate only on the factor $\vartheta(q, q)$ for now. This is a holomorphic function on the unit disc except for a removable singularity at $q = 0$.

For $q \neq 0$ and $|q| < 1$, $\vartheta(q, q)$ has following the power series expansion:

$$\vartheta(q, q) = \sum_{n=-\infty}^{\infty} q^{n(n+1)} = \sum_{n=0}^{\infty} 2q^{n(n+1)}$$

Note that $n(n+1)$ is twice the triangular number $n(n+1)/2$, so we can also see this as a series expansion in terms of powers of q^2 .

lemma has_sum_jacobi_theta_nw_10_aux:

```

assumes q: "norm q < 1" "q ≠ 0"
shows   "((λn. 2 * q ^ (n*(n+1))) has_sum jacobi_theta_nome q q) UNIV"
⟨proof⟩

lemma sums_jacobi_theta_nw_10_aux:
assumes q: "norm q < 1" "q ≠ 0"
shows   "(λn. if ∃i. n = i*(i+1) then 2 * q ^ n else 0) sums jacobi_theta_nome
q q"
⟨proof⟩

definition fps_jacobi_theta_nw_10 :: "'a :: field fps" where
"fps_jacobi_theta_nw_10 = Abs_fps (λn. if ∃i. n = i*(i+1) then 2 else
0)"

lemma fps_conv_radius_jacobi_theta_2 [simp]: "fps_conv_radius fps_jacobi_theta_nw_10
= 1"
⟨proof⟩

lemma has_laurent_expansion_jacobi_theta_2 [laurent_expansion_intros]:
"(λq. jacobi_theta_nome q q) has_laurent_expansion fps_to_fls fps_jacobi_theta_nw_10"
⟨proof⟩

```

For $\vartheta(q, q)^2$, we can find the following expansion into a double sum:

$$\vartheta(q, q)^2 = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} q^{i(i+1)+j(j+1)}$$

```

lemma has_sum_jacobi_theta_nw_10_aux_square:
fixes q :: complex
assumes q: "norm q < 1" "q ≠ 0"
shows "((λ(i,j). q powi (i*(i+1) + j*(j+1))) has_sum jacobi_theta_nome
q q ^ 2) UNIV"
⟨proof⟩

```

With some creative reindexing, we find the following power series expansion:

$$q\vartheta(q^2, q^2)^2 = \sum_{n=0}^{\infty} r_2(2n+1)q^{2n+1}$$

```

lemma sums_q_times_jacobi_theta_nw_10_aux_square_square:
fixes q :: complex
assumes q: "q ≠ 0" "norm q < 1"
shows "(λn. (if odd n then of_nat (count_sos 2 n) else 0) * q ^ n)
sums
(q * jacobi_theta_nome (q^2) (q^2) ^ 2)"
⟨proof⟩

```

```
lemma has_laurent_expansion_q_times_jacobi_theta_nw_10_aux_square_square:
```

```

defines "F ≡ Abs_fps (λn. if odd n then of_nat (count_sos 2 n) else
0)"
shows "(λq. q * jacobi_theta_nome (q2) (q2) ^ 2) has_laurent_expansion
fps_to_fls F"
⟨proof⟩

```

5.2 Identities

Lastly, we derive a variety of identities between the different theta functions.

```

theorem jacobi_theta_nw_00_plus_01_complex: "ϑ3 q + ϑ4 q = 2 * ϑ3 (q
^ 4 :: complex)"
⟨proof⟩

```

```

lemma jacobi_theta_nw_00_plus_01_real: "ϑ3 q + ϑ4 q = 2 * ϑ3 (q ^ 4 :: real)"
⟨proof⟩

```

```

theorem jacobi_theta_nw_00_plus_01_square_complex:
"ϑ3 q ^ 2 + ϑ4 q ^ 2 = 2 * ϑ3 (q ^ 2 :: complex) ^ 2"
⟨proof⟩

```

```

corollary midpoint_jacobi_theta_nw_00_01_square_complex:
"midpoint (ϑ3 q ^ 2) (ϑ4 q ^ 2) = ϑ3 (q ^ 2 :: complex) ^ 2"
⟨proof⟩

```

```

lemma jacobi_theta_nw_00_plus_01_square_real: "ϑ3 q ^ 2 + ϑ4 q ^ 2 =
2 * ϑ3 (q ^ 2 :: real) ^ 2"
⟨proof⟩

```

```

theorem jacobi_theta_nw_00_times_01_complex: "ϑ3 q * ϑ4 q = (ϑ4 (q
^ 2) ^ 2 :: complex)"
⟨proof⟩

```

```

lemma jacobi_theta_nw_00_times_01_real: "ϑ3 q * ϑ4 q = (ϑ4 (q ^ 2) ^
2 :: real)"
⟨proof⟩

```

```

lemma jacobi_theta_nw_00_plus_10_square_square_aux:
fixes q :: complex
shows "ϑ3 q ^ 2 - ϑ3 (q2) ^ 2 = q * jacobi_theta_nome (q2) (q2) ^ 2"
⟨proof⟩

```

```

theorem jacobi_theta_nw_00_plus_10_square_square_complex:
fixes q :: complex
assumes "Re q ≥ 0 ∧ (Re q = 0 ⟹ Im q ≥ 0)"
shows "ϑ3 (q2) ^ 2 + ϑ2 (q2) ^ 2 = ϑ3 q ^ 2"
⟨proof⟩

```

```

lemma jacobi_theta_nw_00_plus_10_square_square_real:
  assumes "q ≥ (0::real)"
  shows   "ϑ3 (q2) ^ 2 + ϑ2 (q2) ^ 2 = ϑ3 q ^ 2"
  ⟨proof⟩

theorem jacobi_theta_nw_00_minus_10_square_square_complex:
  assumes "0 ≤ Re q ∧ (Re q = 0 → 0 ≤ Im q)"
  shows   "ϑ3 (q2) ^ 2 - ϑ2 (q2) ^ 2 = ϑ4 (q :: complex) ^ 2"
  ⟨proof⟩

lemma jacobi_theta_nw_00_minus_10_square_square_real:
  assumes "q ≥ (0::real)"
  shows   "ϑ3 (q2) ^ 2 - ϑ2 (q2) ^ 2 = ϑ4 q ^ 2"
  ⟨proof⟩

```

The following shows that the theta nullwerte provide a parameterisation of the Fermat curve $X^4 + Y^4 = Z^4$:

```

theorem jacobi_theta_nw_pow4_complex: "ϑ2 q ^ 4 + ϑ4 q ^ 4 = (ϑ3 q ^ 4 :: complex)"
  ⟨proof⟩

```

```

lemma jacobi_theta_nw_pow4_real: "q ≥ 0 ⇒ ϑ2 q ^ 4 + ϑ4 q ^ 4 = (ϑ3 q ^ 4 :: real)"
  ⟨proof⟩

```

5.3 Properties of the nullwert functions on the real line

```

lemma has_field_derivative_jacobi_theta_nw_00:
  fixes q :: "'a :: {real_normed_field, banach}"
  assumes q: "norm q < 1"
  defines "a ≡ (λn. 2 * (of_nat n + 1)2 * q ^ (n * (n + 2)))"
  shows "summable a" "(ϑ3 has_field_derivative (∑ n. a n)) (at q)"
  ⟨proof⟩

lemma jacobi_theta_nw_10_le_00:
  assumes "q ≥ (0::real)"
  shows   "ϑ2 q ≤ ϑ3 q"
  ⟨proof⟩

lemma jacobi_theta_nw_00_pos:
  fixes q :: real
  assumes "q ∈ {-1..1}"
  shows   "ϑ3 q > 0"
  ⟨proof⟩

lemma jacobi_theta_nw_01_pos: "q ∈ {-1..1} ⇒ ϑ4 q > (0::real)"
  ⟨proof⟩

```

```

lemma jacobi_theta_nw_00_nonneg: " $\vartheta_3 q \geq (0::\text{real})$ "
   $\langle proof \rangle$ 

lemma jacobi_theta_nw_01_nonneg: " $\vartheta_4 q \geq (0::\text{real})$ "
   $\langle proof \rangle$ 

lemma strict_mono_jacobi_theta_nw_00: "strict_mono_on {-1<.. $<1::\text{real}$ }  $\vartheta_3$ "
   $\langle proof \rangle$ 
  include qpochhammer_inf_notation
   $\langle proof \rangle$ 

lemma strict_antimono_jacobi_theta_nw_01: "strict_antimono_on {-1<.. $<1::\text{real}$ }  $\vartheta_4$ "
   $\langle proof \rangle$ 

lemma jacobi_theta_nw_10_nonneg:
  assumes "x  $\geq 0$ "
  shows " $\vartheta_2 x \geq (0::\text{real})$ "
   $\langle proof \rangle$ 

lemma strict_mono_jacobi_theta_nw_10: "strict_mono_on {0::\text{real}.. $<1\}$   $\vartheta_2$ "
   $\langle proof \rangle$ 

lemma jacobi_theta_nw_10_pos:
  assumes "x  $\in \{0<..<1\}$ "
  shows " $\vartheta_2 x > (0::\text{real})$ "
   $\langle proof \rangle$ 

end

```

References

- [1] B. C. Berndt. *Ramanujans Notebooks*. Springer New York, 1991.