

The Irrationality of $\zeta(3)$

Manuel Eberl

December 14, 2021

Abstract

This article provides a formalisation of Beukers’s straightforward analytic proof [2] that $\zeta(3)$ is irrational. This was first proven by Apéry [1] (which is why this result is also often called ‘Apéry’s Theorem’) using a more algebraic approach. This formalisation follows Filaseta’s presentation of Beukers’s proof [5].

Contents

1	The Irrationality of $\zeta(3)$	2
1.1	Auxiliary facts about polynomials	2
1.2	Auxiliary facts about integrals	4
1.3	Shifted Legendre polynomials	10
1.4	Auxiliary facts about the ζ function	13
1.5	Divisor of a sum of rationals	14
1.6	The first double integral	16
1.7	The second double integral	26
1.8	The triple integral	29
1.9	Connecting the double and triple integral	33
1.10	The main result	44

1 The Irrationality of $\zeta(3)$

theory *Zeta-3-Irrational*

imports

E-Transcendental.E-Transcendental

Prime-Number-Theorem.Prime-Number-Theorem

Prime-Distribution-Elementary.PNT-Consequences

begin

Apéry's original proof of the irrationality of $\zeta(3)$ contained several tricky identities of sums involving binomial coefficients that are difficult to prove. I instead follow Beukers's proof, which consists of several fairly straightforward manipulations of integrals with absolutely no caveats.

Both Apéry and Beukers make use of an asymptotic upper bound on $\text{lcm}\{1 \dots n\}$ – namely $\text{lcm}\{1 \dots n\} \in o(c^n)$ for any $c > e$, which is a consequence of the Prime Number Theorem (which, fortunately, is available in the *Archive of Formal Proofs* [4, 3]).

I follow the concise presentation of Beukers's proof in Filaseta's lecture notes [5], which turned out to be very amenable to formalisation.

There is another earlier formalisation of the irrationality of $\zeta(3)$ by Mahboubi *et al.* [6], who followed Apéry's original proof, but were ultimately forced to find a more elementary way to prove the asymptotics of $\text{lcm}\{1 \dots n\}$ than the Prime Number Theorem.

First, we will require some auxiliary material before we get started with the actual proof.

1.1 Auxiliary facts about polynomials

lemma *degree-higher-pderiv*: $\text{degree } ((\text{pderiv } \sim n) p) = \text{degree } p - n$

for $p :: 'a :: \{\text{comm-semiring-1, semiring-no-zero-divisors, semiring-char-0}\}$ *poly*

by (*induction n arbitrary: p (auto simp: degree-pderiv)*)

lemma *pcompose-power-left*: $\text{pcompose } (p \wedge n) q = \text{pcompose } p q \wedge n$

by (*induction n (auto simp: pcompose-mult pcompose-1)*)

lemma *pderiv-sum*: $\text{pderiv } (\sum_{x \in A} f x) = (\sum_{x \in A} \text{pderiv } (f x))$

by (*induction A rule: infinite-finite-induct (auto simp: pderiv-add)*)

lemma *higher-pderiv-minus*: $(\text{pderiv } \sim n) (-p :: 'a :: \text{idom poly}) = -(\text{pderiv } \sim n) p$

by (*induction n (auto simp: pderiv-minus)*)

lemma *pderiv-power*: $\text{pderiv } (p \wedge n) = \text{smult } (\text{of-nat } n) (p \wedge (n - 1)) * \text{pderiv } p$

by (*cases n (simp-all add: pderiv-power-Suc del: power-Suc)*)

lemma *pderiv-monom*: $\text{pderiv } (\text{monom } c n) = \text{monom } (\text{of-nat } n * c) (n - 1)$

by (*simp add: monom-altdef pderiv-smult pderiv-power pderiv-pCons mult-ac*)

lemma *higher-pderiv-monom*:

$k \leq n \implies (\text{pderiv } \overset{\sim}{\sim} k) (\text{monom } c \ n) = \text{monom } (\text{of-nat } (\text{pochhammer } (n - k + 1) \ k) * c) (n - k)$

by (*induction k*) (*auto simp: pderiv-monom pochhammer-rec Suc-diff-le Suc-diff-Suc mult-ac*)

lemma *higher-pderiv-mult*:

$(\text{pderiv } \overset{\sim}{\sim} n) (p * q) = (\sum_{k \leq n}. \text{Polynomial.smult } (\text{of-nat } (n \ \text{choose } k)) ((\text{pderiv } \overset{\sim}{\sim} k) \ p * (\text{pderiv } \overset{\sim}{\sim} (n - k)) \ q))$

proof (*induction n*)

case (*Suc n*)

have *eq*: $(\text{Suc } n \ \text{choose } k) = (n \ \text{choose } k) + (n \ \text{choose } (k-1))$ **if** $k > 0$ **for** k

using *that by (cases k) auto*

have $(\text{pderiv } \overset{\sim}{\sim} \text{Suc } n) (p * q) =$

$(\sum_{k \leq n}. \text{smult } (\text{of-nat } (n \ \text{choose } k)) ((\text{pderiv } \overset{\sim}{\sim} k) \ p * (\text{pderiv } \overset{\sim}{\sim} (\text{Suc } n - k)) \ q)) +$

$(\sum_{k \leq n}. \text{smult } (\text{of-nat } (n \ \text{choose } k)) ((\text{pderiv } \overset{\sim}{\sim} \text{Suc } k) \ p * (\text{pderiv } \overset{\sim}{\sim} (n - k)) \ q))$

by (*simp add: Suc pderiv-sum pderiv-smult pderiv-mult sum.distrib smult-add-right algebra-simps Suc-diff-le*)

also have $(\sum_{k \leq n}. \text{smult } (\text{of-nat } (n \ \text{choose } k)) ((\text{pderiv } \overset{\sim}{\sim} k) \ p * (\text{pderiv } \overset{\sim}{\sim} (\text{Suc } n - k)) \ q)) =$

$(\sum_{k \in \text{insert } 0 \ \{1..n\}}. \text{smult } (\text{of-nat } (n \ \text{choose } k)) ((\text{pderiv } \overset{\sim}{\sim} k) \ p * (\text{pderiv } \overset{\sim}{\sim} (\text{Suc } n - k)) \ q))$

by (*intro sum.cong auto*)

also have $\dots = (\sum_{k=1..n}. \text{smult } (\text{of-nat } (n \ \text{choose } k)) ((\text{pderiv } \overset{\sim}{\sim} k) \ p * (\text{pderiv } \overset{\sim}{\sim} (\text{Suc } n - k)) \ q)) + p * (\text{pderiv } \overset{\sim}{\sim} \text{Suc } n) \ q$

by (*subst sum.insert (auto simp: add-ac)*)

also have $(\sum_{k \leq n}. \text{smult } (\text{of-nat } (n \ \text{choose } k)) ((\text{pderiv } \overset{\sim}{\sim} \text{Suc } k) \ p * (\text{pderiv } \overset{\sim}{\sim} (n - k)) \ q)) =$

$(\sum_{k=1..n+1}. \text{smult } (\text{of-nat } (n \ \text{choose } (k-1))) ((\text{pderiv } \overset{\sim}{\sim} k) \ p * (\text{pderiv } \overset{\sim}{\sim} (\text{Suc } n - k)) \ q))$

by (*intro sum.reindex-bij-witness[of - $\lambda k. k - 1$ Suc] auto*)

also have $\dots = (\sum_{k \in \text{insert } (n+1) \ \{1..n\}}. \text{smult } (\text{of-nat } (n \ \text{choose } (k-1))) ((\text{pderiv } \overset{\sim}{\sim} k) \ p * (\text{pderiv } \overset{\sim}{\sim} (\text{Suc } n - k)) \ q))$

by (*intro sum.cong auto*)

also have $\dots = (\sum_{k=1..n}. \text{smult } (\text{of-nat } (n \ \text{choose } (k-1))) ((\text{pderiv } \overset{\sim}{\sim} k) \ p * (\text{pderiv } \overset{\sim}{\sim} (\text{Suc } n - k)) \ q)) + (\text{pderiv } \overset{\sim}{\sim} \text{Suc } n) \ p * q$

by (*subst sum.insert (auto simp: add-ac)*)

also have $(\sum_{k=1..n}. \text{smult } (\text{of-nat } (n \ \text{choose } k)) ((\text{pderiv } \overset{\sim}{\sim} k) \ p * (\text{pderiv } \overset{\sim}{\sim} (\text{Suc } n - k)) \ q)) +$

$p * (\text{pderiv } \overset{\sim}{\sim} \text{Suc } n) \ q + \dots = (\sum_{k=1..n}. \text{smult } (\text{of-nat } (\text{Suc } n \ \text{choose } k)) ((\text{pderiv } \overset{\sim}{\sim} k) \ p * (\text{pderiv } \overset{\sim}{\sim} (\text{Suc } n - k)) \ q)) +$

$p * (\text{pderiv } \overset{\sim}{\sim} \text{Suc } n) \ q + (\text{pderiv } \overset{\sim}{\sim} \text{Suc } n) \ p * q$

by (*simp add: sum.distrib algebra-simps smult-add-right eq smult-add-left*)

also have $\dots = (\sum_{k \in \{1..n\} \cup \{0, \text{Suc } n\}}. \text{smult } (\text{of-nat } (\text{Suc } n \ \text{choose } k)))$

((*pderiv* $\widehat{\sim}$ *k*) *p* * (*pderiv* $\widehat{\sim}$ (*Suc n* - *k*)) *q*)
 by (*subst sum.union-disjoint*) (*auto simp: algebra-simps*)
 also have $\{1..n\} \cup \{0, \text{Suc } n\} = \{..\text{Suc } n\}$ by *auto*
 finally show ?*case* .
 qed *auto*

1.2 Auxiliary facts about integrals

theorem (*in pair-sigma-finite*) *Fubini-set-integrable*:
 fixes *f* :: - \Rightarrow - :: {*banach*, *second-countable-topology*}
 assumes *f*[*measurable*]: *set-borel-measurable* (*M1* \otimes_M *M2*) (*A* \times *B*) *f*
 and *integ1*: *set-integrable* *M1* *A* ($\lambda x. \int y \in B. \text{norm } (f \ (x, y)) \ \partial M2$)
 and *integ2*: $\text{AE } x \in A \text{ in } M1. \text{ set-integrable } M2 \ B \ (\lambda y. f \ (x, y))$
 shows *set-integrable* (*M1* \otimes_M *M2*) (*A* \times *B*) *f*
 unfolding *set-integrable-def*
proof (*rule Fubini-integrable*)
 note *integ1*
 also have *set-integrable* *M1* *A* ($\lambda x. \int y \in B. \text{norm } (f \ (x, y)) \ \partial M2$) \longleftrightarrow
integrable *M1* ($\lambda x. \text{LINT } y | M2. \text{norm } (\text{indicat-real } (A \times B) \ (x, y)) *_{\mathbb{R}} f \ (x, y)$)
 unfolding *set-integrable-def*
 by (*intro integrable-cong*) (*auto simp: indicat-def set-lebesgue-integral-def*)
 finally show
next
 from *integ2* show $\text{AE } x \text{ in } M1. \text{ integrable } M2 \ (\lambda y. \text{indicat-real } (A \times B) \ (x, y)) *_{\mathbb{R}} f \ (x, y)$
proof *eventually-elim*
 case (*elim x*)
 show *integrable* *M2* ($\lambda y. \text{indicat-real } (A \times B) \ (x, y)) *_{\mathbb{R}} f \ (x, y)$
proof (*cases x* \in *A*)
 case *True*
 with *elim* have *set-integrable* *M2* *B* ($\lambda y. f \ (x, y)$) by *simp*
 also have ?*this* \longleftrightarrow ?*thesis*
 unfolding *set-integrable-def* using *True*
 by (*intro integrable-cong*) (*auto simp: indicat-def*)
 finally show ?*thesis* .
 qed *auto*
 qed

lemma (*in pair-sigma-finite*) *set-integral-fst'*:
 fixes *f* :: - \Rightarrow '*c* :: {*second-countable-topology*, *banach*}
 assumes *set-integrable* (*M1* \otimes_M *M2*) (*A* \times *B*) *f*
 shows *set-lebesgue-integral* (*M1* \otimes_M *M2*) (*A* \times *B*) *f* =
 $(\int x \in A. (\int y \in B. f \ (x, y)) \ \partial M2) \ \partial M1$
proof -
 have *set-lebesgue-integral* (*M1* \otimes_M *M2*) (*A* \times *B*) *f* =
 $(\int z. \text{indicator } (A \times B) \ z *_{\mathbb{R}} f \ z \ \partial (M1 \ \otimes_M \ M2))$
 by (*simp add: set-lebesgue-integral-def*)

also have $\dots = (\int x. \int y. \text{indicator } (A \times B) (x,y) *_R f (x,y) \partial M2 \partial M1)$
using *assms* **by** (*subst integral-fst'* [*symmetric*]) (*auto simp: set-integrable-def*)
also have $\dots = (\int x \in A. (\int y \in B. f (x,y) \partial M2) \partial M1)$
unfolding *set-lebesgue-integral-def*
by (*intro Bochner-Integration.integral-cong refl*) (*auto simp: indicator-def*)
finally show *?thesis* .
qed

lemma (*in pair-sigma-finite*) *set-integral-snd*:

fixes $f :: - \Rightarrow 'c :: \{\text{second-countable-topology, banach}\}$
assumes *set-integrable* $(M1 \otimes_M M2) (A \times B) f$
shows *set-lebesgue-integral* $(M1 \otimes_M M2) (A \times B) f =$
 $(\int y \in B. (\int x \in A. f (x, y) \partial M1) \partial M2)$

proof –

have *set-lebesgue-integral* $(M1 \otimes_M M2) (A \times B) f =$
 $(\int z. \text{indicator } (A \times B) z *_R f z \partial(M1 \otimes_M M2))$
by (*simp add: set-lebesgue-integral-def*)
also have $\dots = (\int y. \int x. \text{indicator } (A \times B) (x,y) *_R f (x,y) \partial M1 \partial M2)$
using *assms* **by** (*subst integral-snd*) (*auto simp: set-integrable-def case-prod-unfold*)
also have $\dots = (\int y \in B. (\int x \in A. f (x,y) \partial M1) \partial M2)$
unfolding *set-lebesgue-integral-def*
by (*intro Bochner-Integration.integral-cong refl*) (*auto simp: indicator-def*)
finally show *?thesis* .
qed

proposition (*in pair-sigma-finite*) *Fubini-set-integral*:

fixes $f :: - \Rightarrow - \Rightarrow - :: \{\text{banach, second-countable-topology}\}$
assumes f : *set-integrable* $(M1 \otimes_M M2) (A \times B) (\text{case-prod } f)$
shows $(\int y \in B. (\int x \in A. f x y \partial M1) \partial M2) = (\int x \in A. (\int y \in B. f x y \partial M2) \partial M1)$

proof –

have $(\int y \in B. (\int x \in A. f x y \partial M1) \partial M2) = (\int y. (\int x. \text{indicator } (A \times B) (x, y)$
 $*_R f x y \partial M1) \partial M2)$
unfolding *set-lebesgue-integral-def*
by (*intro Bochner-Integration.integral-cong*) (*auto simp: indicator-def*)
also have $\dots = (\int x. (\int y. \text{indicator } (A \times B) (x, y) *_R f x y \partial M2) \partial M1)$
using *assms* **by** (*intro Fubini-integral*) (*auto simp: set-integrable-def case-prod-unfold*)
also have $\dots = (\int x \in A. (\int y \in B. f x y \partial M2) \partial M1)$
unfolding *set-lebesgue-integral-def*
by (*intro Bochner-Integration.integral-cong*) (*auto simp: indicator-def*)
finally show *?thesis* .
qed

lemma (*in pair-sigma-finite*) *nn-integral-swap*:

assumes [*measurable*]: $f \in \text{borel-measurable } (M1 \otimes_M M2)$
shows $(\int^+ x. f x \partial(M1 \otimes_M M2)) = (\int^+ (y,x). f (x,y) \partial(M2 \otimes_M M1))$
by (*subst distr-pair-swap, subst nn-integral-distr*) (*auto simp: case-prod-unfold*)

lemma *set-integrable-bound*:

fixes $f :: 'a \Rightarrow 'b :: \{\text{banach, second-countable-topology}\}$

```

  and g :: 'a ⇒ 'c::{banach, second-countable-topology}
  shows set-integrable M A f ⇒ set-borel-measurable M A g ⇒
    (AE x in M. x ∈ A → norm (g x) ≤ norm (f x)) ⇒ set-integrable M
A g
  unfolding set-integrable-def
  apply (erule Bochner-Integration.integrable-bound)
  apply (simp add: set-borel-measurable-def)
  apply (erule eventually-mono)
  apply (auto simp: indicator-def)
  done

```

lemma set-integrableI-nonneg:

```

  fixes f :: 'a ⇒ real
  assumes set-borel-measurable M A f
  assumes AE x in M. x ∈ A → 0 ≤ f x (∫+x∈A. f x ∂M) < ∞
  shows set-integrable M A f
  unfolding set-integrable-def
  proof (rule integrableI-nonneg)
    from assms show (λx. indicator A x *R f x) ∈ borel-measurable M
      by (simp add: set-borel-measurable-def)
    from assms(2) show AE x in M. 0 ≤ indicat-real A x *R f x
      by eventually-elim (auto simp: indicator-def)
    have (∫+x. ennreal (indicator A x *R f x) ∂M) = (∫+x∈A. f x ∂M)
      by (intro nn-integral-cong) (auto simp: indicator-def)
    also have ... < ∞ by fact
    finally show (∫+x. ennreal (indicator A x *R f x) ∂M) < ∞ .
  qed

```

lemma set-integral-eq-nn-integral:

```

  assumes set-borel-measurable M A f
  assumes set-nn-integral M A f = ennreal x x ≥ 0
  assumes AE x in M. x ∈ A → f x ≥ 0
  shows set-integrable M A f
    and set-lebesgue-integral M A f = x
  proof -
    have eq: (λx. (if x ∈ A then 1 else 0) * f x) = (λx. if x ∈ A then f x else 0)
      (λx. if x ∈ A then ennreal (f x) else 0) =
      (λx. ennreal (f x) * (if x ∈ A then 1 else 0))
      (λx. ennreal (f x * (if x ∈ A then 1 else 0))) =
      (λx. ennreal (f x) * (if x ∈ A then 1 else 0))
    by auto
    from assms show *: set-integrable M A f
      by (intro set-integrableI-nonneg) auto
    have set-lebesgue-integral M A f = enn2real (set-nn-integral M A f)
      unfolding set-lebesgue-integral-def using assms(1,4) * eq
      by (subst integral-eq-nn-integral)
      (auto intro!: nn-integral-cong simp: indicator-def of-bool-def set-integrable-def
      mult-ac)
  qed

```

also have $\dots = x$ using *assms* by *simp*
 finally show *set-lebesgue-integral* $M A f = x$.
 qed

lemma *set-integral-0* [*simp*, *intro*]: *set-integrable* $M A (\lambda y. 0)$
 by (*simp add: set-integrable-def*)

lemma *set-integrable-sum*:
 fixes $f :: - \Rightarrow - \Rightarrow - :: \{\text{banach, second-countable-topology}\}$
 assumes *finite* B
 assumes $\bigwedge x. x \in B \implies \text{set-integrable } M A (f x)$
 shows *set-integrable* $M A (\lambda y. \sum_{x \in B}. f x y)$
 using *assms* by (*induction rule: finite-induct*) (*auto intro!: set-integral-add*)

lemma *set-integral-sum*:
 fixes $f :: - \Rightarrow - \Rightarrow - :: \{\text{banach, second-countable-topology}\}$
 assumes *finite* B
 assumes $\bigwedge x. x \in B \implies \text{set-integrable } M A (f x)$
 shows *set-lebesgue-integral* $M A (\lambda y. \sum_{x \in B}. f x y) = (\sum_{x \in B}. \text{set-lebesgue-integral } M A (f x))$
 using *assms*
 apply (*induction rule: finite-induct*)
 apply *simp*
 apply *simp*
 apply (*subst set-integral-add*)
 apply (*auto intro!: set-integrable-sum*)
 done

lemma *set-nn-integral-cong*:
 assumes $M = M' A = B \bigwedge x. x \in \text{space } M \cap A \implies f x = g x$
 shows *set-nn-integral* $M A f = \text{set-nn-integral } M' B g$
 using *assms* **unfolding** *assms(1,2)* by (*intro nn-integral-cong*) (*auto simp: indicator-def*)

lemma *set-nn-integral-mono*:
 assumes $\bigwedge x. x \in \text{space } M \cap A \implies f x \leq g x$
 shows *set-nn-integral* $M A f \leq \text{set-nn-integral } M A g$
 using *assms* by (*intro nn-integral-mono*) (*auto simp: indicator-def*)

lemma
 fixes $f :: \text{real} \Rightarrow \text{real}$
 assumes $a \leq b$
 assumes *deriv*: $\bigwedge x. a \leq x \implies x \leq b \implies (F \text{ has-field-derivative } f x)$ (*at x within* $\{a..b\}$)
 assumes *cont*: *continuous-on* $\{a..b\}$ f
 shows *has-bochner-integral-FTC-Icc-real*:
 has-bochner-integral l borel $(\lambda x. f x * \text{indicator } \{a .. b\} x) (F b - F a)$ (**is** *?has*)
 and *integral-FTC-Icc-real*: $(\int x. f x * \text{indicator } \{a .. b\} x \partial \text{l borel}) = F b - F a$

a (is ?eq)
proof –
have $1: \bigwedge x. a \leq x \implies x \leq b \implies (F \text{ has-vector-derivative } f x) \text{ (at } x \text{ within } \{a .. b\})$
unfolding *has-field-derivative-iff-has-vector-derivative[symmetric]*
using *deriv by auto*
show ?has ?eq
using *has-bochner-integral-FTC-Icc[OF <a ≤ b> 1 cont] integral-FTC-Icc[OF <a ≤ b> 1 cont]*
by (*auto simp: mult.commute*)
qed

lemma *integral-by-parts-integrable*:
fixes $f g F G::\text{real} \Rightarrow \text{real}$
assumes $a \leq b$
assumes *cont-f[intro]: continuous-on {a..b} f*
assumes *cont-g[intro]: continuous-on {a..b} g*
assumes *[intro]: $\bigwedge x. x \in \{a..b\} \implies (F \text{ has-field-derivative } f x) \text{ (at } x \text{ within } \{a..b\})$*
assumes *[intro]: $\bigwedge x. x \in \{a..b\} \implies (G \text{ has-field-derivative } g x) \text{ (at } x \text{ within } \{a..b\})$*
shows *integrable lborel ($\lambda x. (F x) * (g x) + (f x) * (G x) * \text{indicator } \{a .. b\} x$)*
proof –
have *integrable lborel ($\lambda x. \text{indicator } \{a..b\} x *_{\mathbb{R}} ((F x) * (g x) + (f x) * (G x))$)*
by (*intro borel-integrable-compact continuous-intros assms*)
(auto intro!: DERIV-continuous-on assms)
thus ?thesis **by** (*simp add: mult-ac*)
qed

lemma *integral-by-parts*:
fixes $f g F G::\text{real} \Rightarrow \text{real}$
assumes *[arith]: $a \leq b$*
assumes *cont-f[intro]: continuous-on {a..b} f*
assumes *cont-g[intro]: continuous-on {a..b} g*
assumes *[intro]: $\bigwedge x. x \in \{a..b\} \implies (F \text{ has-field-derivative } f x) \text{ (at } x \text{ within } \{a..b\})$*
assumes *[intro]: $\bigwedge x. x \in \{a..b\} \implies (G \text{ has-field-derivative } g x) \text{ (at } x \text{ within } \{a..b\})$*
shows ($\int x. (F x * g x) * \text{indicator } \{a .. b\} x \partial \text{lborel}$)
 $= F b * G b - F a * G a - \int x. (f x * G x) * \text{indicator } \{a .. b\} x \partial \text{lborel}$
proof –
have $0: (\int x. (F x * g x + f x * G x) * \text{indicator } \{a .. b\} x \partial \text{lborel}) = F b * G b - F a * G a$
by (*rule integral-FTC-Icc-real, auto intro!: derivative-eq-intros continuous-intros*)
(auto intro!: assms DERIV-continuous-on)
have *[continuous-intros]: continuous-on {a..b} F*
by (*rule DERIV-continuous-on assms*)
have *[continuous-intros]: continuous-on {a..b} G*

by (rule DERIV-continuous-on assms)+
 have $(\int x. \text{indicator } \{a..b\} x *_R (F x * g x + f x * G x) \partial \text{lborel}) =$
 $(\int x. \text{indicator } \{a..b\} x *_R (F x * g x) \partial \text{lborel}) + \int x. \text{indicator } \{a..b\} x *_R (f$
 $x * G x) \partial \text{lborel}$
 apply (subst Bochner-Integration.integral-add[symmetric])
 apply (rule borel-integrable-compact; force intro!: continuous-intros assms)
 apply (rule borel-integrable-compact; force intro!: continuous-intros assms)
 apply (simp add: algebra-simps)
 done
 thus ?thesis using 0 by (simp add: algebra-simps)
 qed

lemma interval-lebesgue-integral-by-parts:
 assumes $a < b$
 assumes $\text{cont-}f[\text{intro}]: \text{continuous-on } \{a..b\} f$
 assumes $\text{cont-}g[\text{intro}]: \text{continuous-on } \{a..b\} g$
 assumes $[\text{intro}]: \bigwedge x. x \in \{a..b\} \implies (F \text{ has-field-derivative } f x) \text{ (at } x \text{ within } \{a..b\})$
 assumes $[\text{intro}]: \bigwedge x. x \in \{a..b\} \implies (G \text{ has-field-derivative } g x) \text{ (at } x \text{ within } \{a..b\})$
 shows $(\text{LBINT } x=a..b. F x * g x) = F b * G b - F a * G a - (\text{LBINT } x=a..b. f x * G x)$
 using interval-by-parts[of a b f g F G] assms
 by (simp add: interval-integral-Icc set-lebesgue-integral-def mult-ac)

lemma interval-lebesgue-integral-by-parts-01:
 assumes $\text{cont-}f[\text{intro}]: \text{continuous-on } \{0..1\} f$
 assumes $\text{cont-}g[\text{intro}]: \text{continuous-on } \{0..1\} g$
 assumes $[\text{intro}]: \bigwedge x. x \in \{0..1\} \implies (F \text{ has-field-derivative } f x) \text{ (at } x \text{ within } \{0..1\})$
 assumes $[\text{intro}]: \bigwedge x. x \in \{0..1\} \implies (G \text{ has-field-derivative } g x) \text{ (at } x \text{ within } \{0..1\})$
 shows $(\text{LBINT } x=0..1. F x * g x) = F 1 * G 1 - F 0 * G 0 - (\text{LBINT } x=0..1. f x * G x)$
 using interval-lebesgue-integral-by-parts[of 0 1 f g F G] assms
 by (simp add: zero-ereal-def one-ereal-def)

lemma continuous-on-imp-set-integrable-cbox:
 fixes $h :: 'a :: \text{euclidean-space} \implies \text{real}$
 assumes $\text{continuous-on } (\text{cbox } a b) h$
 shows $\text{set-integrable lborel } (\text{cbox } a b) h$
proof –
 from assms have $h \text{ absolutely-integrable-on } \text{cbox } a b$
 by (rule absolutely-integrable-continuous)
 moreover have $(\lambda x. \text{indicat-real } (\text{cbox } a b) x *_R h x) \in \text{borel-measurable borel}$
 by (rule borel-measurable-continuous-on-indicator) (use assms in auto)
 ultimately show ?thesis

unfolding *set-integrable-def* **using** *assms* **by** (*subst (asm) integrable-completion*)
auto
qed

1.3 Shifted Legendre polynomials

The first ingredient we need to show Apéry's theorem is the *shifted Legendre polynomials*

$$P_n(X) = \frac{1}{n!} \frac{\partial^n}{\partial X^n} (X^n (1 - X)^n)$$

and the auxiliary polynomials

$$Q_{n,k}(X) = \frac{\partial^k}{\partial X^k} (X^n (1 - X)^n) .$$

Note that P_n is in fact an *integer* polynomial.

Only some very basic properties of these will be proven, since that is all we will need.

context

fixes $n :: nat$

begin

definition *gen-shleg-poly* :: $nat \Rightarrow int\ poly$ **where**

gen-shleg-poly $k = (pderiv \hat{\sim} k) ([:0, 1, -1:] \hat{\sim} n)$

definition *shleg-poly* **where** *shleg-poly* = *gen-shleg-poly* $n\ div\ [:fact\ n:]$

We can easily prove the following more explicit formula for $Q_{n,k}$:

$$Q_{n,k}(X) = \sum_{i=0}^k (-1)^{k-1} \binom{k}{i} n^i n^{k-i} X^{n-i} (1 - X)^{n-k+i}$$

lemma *gen-shleg-poly-altdef*:

assumes $k \leq n$

shows *gen-shleg-poly* $k =$

$$\left(\sum_{i \leq k} smult ((-1) \hat{\sim} (k-i)) * of-nat (k\ choose\ i) * \right. \\
\left. pochhammer (n-i+1)\ i * pochhammer (n-k+i+1)\ (k-i) \right) \\
([:0, 1:] \hat{\sim} (n-i)) * [:1, -1:] \hat{\sim} (n-k+i))$$

proof –

have *: $(pderiv \hat{\sim} i) (x \circ_p [:1, -1:]) =$

$smult ((-1) \hat{\sim} i) ((pderiv \hat{\sim} i) x \circ_p [:1, -1:])$ **for** i **and** $x :: int\ poly$

by (*induction* i *arbitrary*: x)

(*auto simp*: *pderiv-smult pderiv-pcompose funpow-Suc-right pderiv-pCons*
higher-pderiv-minus simp del: *funpow.simps(2)*)

have *gen-shleg-poly* $k = (pderiv \hat{\sim} k) ([:0, 1, -1:] \hat{\sim} n)$

by (*simp add*: *gen-shleg-poly-def*)

also have $[:0, 1, -1::int:] = [:0, 1:] * [:1, -1:]$
by *simp*
also have $\dots \wedge n = [:0, 1:] \wedge n * [:1, -1:] \wedge n$
by (*simp flip: power-mult-distrib*)
also have $(pderiv \wedge k) \dots =$
 $(\sum_{i \leq k}. smult (of-nat (k \text{ choose } i)) ((pderiv \wedge i)$
 $([:0, 1:] \wedge n) * (pderiv \wedge (k - i)) ([:1, -1:] \wedge n)))$
by (*simp add: higher-pderiv-mult*)
also have $\dots = (\sum_{i \leq k}. smult (of-nat (k \text{ choose } i))$
 $((pderiv \wedge i) (monom 1 n) * (pderiv \wedge (k - i)) (monom 1 n) \circ_p$
 $[:1, -1:])))$
by (*simp add: monom-altdef pcompose-pCons pcompose-power-left*)
also have $\dots = (\sum_{i \leq k}. smult ((-1) \wedge (k - i) * of-nat (k \text{ choose } i))$
 $((pderiv \wedge i) (monom 1 n) * ((pderiv \wedge (k - i)) (monom 1 n) \circ_p$
 $[:1, -1:])))$
by (*simp add: * mult-ac*)
also have $\dots = (\sum_{i \leq k}. smult ((-1) \wedge (k - i) * of-nat (k \text{ choose } i))$
 $(monom (pochhammer (n - i + 1) i) (n - i) *$
 $monom (pochhammer (n - k + i + 1) (k - i)) (n - k + i) \circ_p$
 $[:1, -1:])))$
using *assms* **by** (*simp add: higher-pderiv-monom*)
also have $\dots = (\sum_{i \leq k}. smult ((-1) \wedge (k - i) * of-nat (k \text{ choose } i) * pochham-$
 $mer (n - i + 1) i * pochhammer (n - k + i + 1) (k - i))$
 $([:0, 1:] \wedge (n - i) * [:1, -1:] \wedge (n - k + i)))$
by (*simp add: monom-altdef algebra-simps pcompose-smult pcompose-power-left*
pcompose-pCons)
finally show *?thesis* .
qed

lemma *degree-gen-shleg-poly [simp]: degree (gen-shleg-poly k) = 2 * n - k*
by (*simp add: gen-shleg-poly-def degree-higher-pderiv degree-power-eq*)

lemma *gen-shleg-poly-n: gen-shleg-poly n = smult (fact n) shleg-poly*
proof –

obtain *r* **where** *r: gen-shleg-poly n = [:fact n:] * r*
unfolding *gen-shleg-poly-def* **using** *fact-dvd-higher-pderiv* [*of n [:0,1,-1:] \wedge n*]
by *blast*
have $smult (fact n) shleg-poly = smult (fact n) (gen-shleg-poly n \text{ div } [:fact n:])$
by (*simp add: shleg-poly-def*)
also note *r*
also have $[:fact n:] * r \text{ div } [:fact n:] = r$
by (*rule nonzero-mult-div-cancel-left*) *auto*
finally show *?thesis*
by (*simp add: r*)
qed

lemma *degree-shleg-poly [simp]: degree shleg-poly = n*
using *degree-gen-shleg-poly* [*of n*] **by** (*simp add: gen-shleg-poly-n*)

lemma *pderiv-gen-shleg-poly* [*simp*]: $pderiv (gen-shleg-poly k) = gen-shleg-poly (Suc k)$

by (*simp add: gen-shleg-poly-def*)

The following functions are the interpretation of the shifted Legendre polynomials and the auxiliary polynomials as a function from reals to reals.

definition *Gen-Shleg* :: $nat \Rightarrow real \Rightarrow real$

where $Gen-Shleg k x = poly (of-int-poly (gen-shleg-poly k)) x$

definition *Shleg* :: $real \Rightarrow real$ **where** $Shleg = poly (of-int-poly shleg-poly)$

lemma *Gen-Shleg-altdef*:

assumes $k \leq n$

shows $Gen-Shleg k x = (\sum_{i \leq k}. (-1)^{\wedge(k-i)} * of-nat (k \text{ choose } i) * of-int (pochhammer (n-i+1) i * pochhammer (n-k+i+1) (k-i)) * x^{\wedge(n-i)} * (1-x)^{\wedge(n-k+i)})$

using *assms* **by** (*simp add: Gen-Shleg-def gen-shleg-poly-altdef poly-sum mult-ac*)

lemma *Gen-Shleg-0* [*simp*]: $k < n \implies Gen-Shleg k 0 = 0$

by (*simp add: Gen-Shleg-altdef zero-power*)

lemma *Gen-Shleg-1* [*simp*]: $k < n \implies Gen-Shleg k 1 = 0$

by (*simp add: Gen-Shleg-altdef zero-power*)

lemma *Gen-Shleg-n-0* [*simp*]: $Gen-Shleg n 0 = fact n$

proof –

have $Gen-Shleg n 0 = (\sum_{i \leq n}. (-1)^{\wedge(n-i)} * real (n \text{ choose } i) * (real (pochhammer (Suc (n-i)) i) * real (pochhammer (Suc i) (n-i))) * 0^{\wedge(n-i)})$

by (*simp add: Gen-Shleg-altdef*)

also have $\dots = (\sum_{i \in \{n\}}. (-1)^{\wedge(n-i)} * real (n \text{ choose } i) * (real (pochhammer (Suc (n-i)) i) * real (pochhammer (Suc i) (n-i))) * 0^{\wedge(n-i)})$

by (*intro sum.mono-neutral-right*) *auto*

also have $\dots = fact n$

by (*simp add: pochhammer-fact flip: pochhammer-of-nat*)

finally show *?thesis* .

qed

lemma *Gen-Shleg-n-1* [*simp*]: $Gen-Shleg n 1 = (-1)^{\wedge n} * fact n$

proof –

have $Gen-Shleg n 1 = (\sum_{i \leq n}. (-1)^{\wedge(n-i)} * real (n \text{ choose } i) * (real (pochhammer (Suc (n-i)) i) * real (pochhammer (Suc i) (n-i))) * 0^{\wedge i})$

by (*simp add: Gen-Shleg-altdef*)

also have $\dots = (\sum_{i \in \{0\}}. (-1)^{\wedge(n-i)} * real (n \text{ choose } i) * (real (pochhammer (Suc (n-i)) i) * real (pochhammer (Suc i) (n-i))) * 0^{\wedge i})$

by (*intro sum.mono-neutral-right*) auto
 also have $\dots = (-1)^{\wedge n} * \text{fact } n$
 by (*simp add: pochhammer-fact flip: pochhammer-of-nat*)
 finally show ?thesis .
 qed

lemma *Shleg-altdef*: $\text{Shleg } x = \text{Gen-Shleg } n \ x / \text{fact } n$
 by (*simp add: Shleg-def Gen-Shleg-def gen-shleg-poly-n*)

lemma *Shleg-0* [*simp*]: $\text{Shleg } 0 = 1$ and *Shleg-1* [*simp*]: $\text{Shleg } 1 = (-1)^{\wedge n}$
 by (*simp-all add: Shleg-altdef*)

lemma *Gen-Shleg-0-left*: $\text{Gen-Shleg } 0 \ x = x^{\wedge n} * (1 - x)^{\wedge n}$
 by (*simp add: Gen-Shleg-def gen-shleg-poly-def power-mult-distrib*)

lemma *has-field-derivative-Gen-Shleg*:
 (*Gen-Shleg k has-field-derivative Gen-Shleg (Suc k) x*) (at *x*)
proof –
 note [*derivative-intros*] = *poly-DERIV*
 show ?thesis **unfolding** *Gen-Shleg-def*
 by (*rule derivative-eq-intros*) auto
 qed

lemma *continuous-on-Gen-Shleg*: *continuous-on A (Gen-Shleg k)*
 by (*auto simp: Gen-Shleg-def intro!: continuous-intros*)

lemma *continuous-on-Gen-Shleg'* [*continuous-intros*]:
continuous-on A f \implies continuous-on A ($\lambda x. \text{Gen-Shleg } k \ (f \ x)$)
 by (*rule continuous-on-compose2[OF continuous-on-Gen-Shleg[of UNIV]]*) auto

lemma *continuous-on-Shleg*: *continuous-on A Shleg*
 by (*auto simp: Shleg-def intro!: continuous-intros*)

lemma *continuous-on-Shleg'* [*continuous-intros*]:
continuous-on A f \implies continuous-on A ($\lambda x. \text{Shleg } (f \ x)$)
 by (*rule continuous-on-compose2[OF continuous-on-Shleg[of UNIV]]*) auto

lemma *measurable-Gen-Shleg* [*measurable*]: $\text{Gen-Shleg } n \in \text{borel-measurable borel}$
 by (*intro borel-measurable-continuous-onI continuous-on-Gen-Shleg*)

lemma *measurable-Shleg* [*measurable*]: $\text{Shleg} \in \text{borel-measurable borel}$
 by (*intro borel-measurable-continuous-onI continuous-on-Shleg*)

end

1.4 Auxiliary facts about the ζ function

lemma *Re-zeta-ge-1*:
 assumes $x > 1$

```

shows  $Re (zeta (of-real x)) \geq 1$ 
proof -
  have *:  $(\lambda n. real (Suc n) powr -x) sums Re (zeta (complex-of-real x))$ 
    using  $sums-Re[OF sums-zeta[of of-real x]]$  assms by  $(simp add: powr-Reals-eq)$ 
  show  $Re (zeta (of-real x)) \geq 1$ 
  proof  $(rule sums-le[OF - - *])$ 
    show  $(\lambda n. if n = 0 then 1 else 0) sums 1$ 
    by  $(rule sums-single)$ 
  qed auto
qed

```

```

lemma sums-zeta-of-nat-offset:
  fixes  $r :: nat$ 
  assumes  $n: n > 1$ 
  shows  $(\lambda k. 1 / (r + k + 1) ^ n) sums (zeta (of-nat n) - (\sum k=1..r. 1 / k ^ n))$ 
proof -
  have  $(\lambda k. 1 / (k + 1) ^ n) sums zeta (of-nat n)$ 
    using  $sums-zeta[of of-nat n] n$ 
    by  $(simp add: powr-minus field-simps flip: of-nat-Suc)$ 
  from  $sums-split-initial-segment[OF this, of r]$ 
  have  $(\lambda k. 1 / (r + k + 1) ^ n) sums (zeta (of-nat n) - (\sum k < r. 1 / Suc k ^ n))$ 
    by  $(simp add: algebra-simps)$ 
  also have  $(\sum k < r. 1 / Suc k ^ n) = (\sum k=1..r. 1 / k ^ n)$ 
    by  $(intro sum.reindex-bij-witness[of - \lambda k. k - 1 Suc]) auto$ 
  finally show ?thesis .
qed

```

```

lemma sums-Re-zeta-of-nat-offset:
  fixes  $r :: nat$ 
  assumes  $n: n > 1$ 
  shows  $(\lambda k. 1 / (r + k + 1) ^ n) sums (Re (zeta (of-nat n)) - (\sum k=1..r. 1 / k ^ n))$ 
proof -
  have  $(\lambda k. Re (1 / (r + k + 1) ^ n)) sums (Re (zeta (of-nat n)) - (\sum k=1..r. 1 / k ^ n))$ 
    by  $(intro sums-Re sums-zeta-of-nat-offset assms)$ 
  thus ?thesis by simp
qed

```

1.5 Divisor of a sum of rationals

A finite sum of rationals of the form $\frac{a_1}{b_1} + \dots + \frac{a_n}{b_n}$ can be brought into the form $\frac{c}{d}$, where d is the LCM of the b_i (or some integer multiple thereof).

```

lemma sum-rationals-common-divisor:
  fixes  $f g :: 'a \Rightarrow int$ 
  assumes finite A
  assumes  $\bigwedge x. x \in A \implies g x \neq 0$ 

```

```

shows  $\exists c. (\sum x \in A. f x / g x) = \text{real-of-int } c / (\text{LCM } x \in A. g x)$ 
using assms
proof (induction rule: finite-induct)
  case empty
  thus ?case by auto
next
  case (insert x A)
  define d where  $d = (\text{LCM } x \in A. g x)$ 
  from insert have [simp]:  $d \neq 0$ 
    by (auto simp: d-def Lcm-0-iff)
  from insert have [simp]:  $g x \neq 0$  by auto
  from insert obtain c where  $c: (\sum x \in A. f x / g x) = \text{real-of-int } c / \text{real-of-int } d$ 
    by (auto simp: d-def)
  define e1 where  $e1 = \text{lcm } d (g x) \text{ div } d$ 
  define e2 where  $e2 = \text{lcm } d (g x) \text{ div } g x$ 
  have  $(\sum y \in \text{insert } x A. f y / g y) = c / d + f x / g x$ 
    using insert c by simp
  also have  $c / d = (c * e1) / \text{lcm } d (g x)$ 
    by (simp add: e1-def real-of-int-div)
  also have  $f x / g x = (f x * e2) / \text{lcm } d (g x)$ 
    by (simp add: e2-def real-of-int-div)
  also have  $(c * e1) / \text{lcm } d (g x) + \dots = (c * e1 + f x * e2) / (\text{LCM } x \in \text{insert } x A. g x)$ 
    using insert by (simp add: add-divide-distrib lcm commute d-def)
  finally show ?case ..
qed

```

```

lemma sum-rationals-common-divisor':
  fixes  $f g :: 'a \Rightarrow \text{int}$ 
  assumes finite A
  assumes  $\bigwedge x. x \in A \implies g x \neq 0$  and  $(\bigwedge x. x \in A \implies g x \text{ dvd } d)$  and  $d \neq 0$ 
  shows  $\exists c. (\sum x \in A. f x / g x) = \text{real-of-int } c / \text{real-of-int } d$ 
proof -
  define d' where  $d' = (\text{LCM } x \in A. g x)$ 
  have  $d' \text{ dvd } d$ 
    unfolding d'-def using assms(3) by (auto simp: Lcm-dvd-iff)
  then obtain e where  $e: d = d' * e$  by blast
  have  $\exists c. (\sum x \in A. f x / g x) = \text{real-of-int } c / (\text{LCM } x \in A. g x)$ 
    by (rule sum-rationals-common-divisor) fact+
  then obtain c where  $c: (\sum x \in A. f x / g x) = \text{real-of-int } c / \text{real-of-int } d'$ 
    unfolding d'-def ..
  also have  $\dots = \text{real-of-int } (c * e) / \text{real-of-int } d$ 
    using  $\langle d \neq 0 \rangle$  by (simp add: e)
  finally show ?thesis ..
qed

```

1.6 The first double integral

We shall now investigate the double integral

$$I_1 := \int_0^1 \int_0^1 -\frac{\ln(xy)}{1-xy} x^r y^s dx dy .$$

Since everything is non-negative for now, we can work over the extended non-negative real numbers and the issues of integrability or summability do not arise at all.

definition *beukers-nn-integral1* :: *nat* \Rightarrow *nat* \Rightarrow *ennreal* **where**

beukers-nn-integral1 *r s* =
 $(\int^+(x,y) \in \{0 <..< 1\} \times \{0 <..< 1\}. \text{ennreal } (-\ln (x*y) / (1 - x*y) * x^{\widehat{r}} * y^{\widehat{s}})$
∂lborel)

definition *beukers-integral1* :: *nat* \Rightarrow *nat* \Rightarrow *real* **where**

beukers-integral1 *r s* = $(\int (x,y) \in \{0 <..< 1\} \times \{0 <..< 1\}. (-\ln (x*y) / (1 - x*y) * x^{\widehat{r}} * y^{\widehat{s}}) \text{∂lborel})$

lemma

fixes *x y z* :: *real*

assumes *xyz*: $x \in \{0 <..< 1\} \ y \in \{0 <..< 1\} \ z \in \{0..1\}$

shows *beukers-denom-ineq*: $(1 - x * y) * z < 1$ **and** *beukers-denom-neg*: $(1 - x * y) * z \neq 1$

proof –

from *xyz* **have** *: $x * y < 1 * 1$

by (*intro mult-strict-mono*) *auto*

from * **have** $(1 - x * y) * z \leq (1 - x * y) * 1$

using *xyz* **by** (*intro mult-left-mono*) *auto*

also have ... $< 1 * 1$

using *xyz* **by** (*intro mult-strict-right-mono*) *auto*

finally show $(1 - x * y) * z < 1 \ (1 - x * y) * z \neq 1$ **by** *simp-all*

qed

We first evaluate the improper integral

$$\int_0^1 -\ln x \cdot x^e dx = \frac{1}{(e+1)^2} .$$

for any $e > -1$.

lemma *integral-0-1-ln-times-powr*:

assumes $e > -1$

shows (*LBINT* $x=0..1. -\ln x * x^{\text{powr } e}$) = $1 / (e + 1)^2$

and *interval-lebesgue-integrable lborel* $0 \ 1 \ (\lambda x. -\ln x * x^{\text{powr } e})$

proof –

define *f* **where** $f = (\lambda x. -\ln x * x^{\text{powr } e})$

define *F* **where** $F = (\lambda x. x^{\text{powr } (e+1)} * (1 - (e+1) * \ln x) / (e+1)^2)$

have 0: *isCont f x* **if** $x \in \{0 < .. < 1\}$ **for** x
using *that* **by** (*auto intro!*: *continuous-intros simp: f-def*)
have 1: (*F has-real-derivative f x*) (*at x*) **if** $x \in \{0 < .. < 1\}$ **for** x
proof –
show (*F has-real-derivative f x*) (*at x*)
unfolding *F-def f-def* **using** *that* *assms*
apply (*insert that assms*)
apply (*rule derivative-eq-intros refl | simp*) +
apply (*simp add: divide-simps*)
apply (*simp add: power2-eq-square algebra-simps powr-add power-numeral-reduce*)
done
qed
have 2: *AE x in lborel. ereal 0 < ereal x \longrightarrow ereal x < ereal 1 \longrightarrow 0 \leq f x*
by (*intro AE-I2*) (*auto simp: f-def mult-nonpos-nonneg*)
have 3: ((*F \circ real-of-ereal*) \longrightarrow 0) (*at-right (ereal 0)*)
unfolding *ereal-tendsto-simps F-def* **using** *assms* **by** *real-asymp*
have 4: ((*F \circ real-of-ereal*) \longrightarrow *F 1*) (*at-left (ereal 1)*)
unfolding *ereal-tendsto-simps F-def*
using *assms* **by** *real-asymp (simp add: field-simps)*

have (*LBINT x=ereal 0..ereal 1. f x*) = *F 1 - 0*
by (*rule interval-integral-FTC-nonneg*[**where** *F = F*])
(*use 0 1 2 3 4 in auto*)
thus (*LBINT x=0..1. -ln x * x powr e*) = *1 / (e + 1)²*
by (*simp add: F-def zero-ereal-def one-ereal-def f-def*)
have *set-integrable lborel (einterval (ereal 0) (ereal 1)) f*
by (*rule interval-integral-FTC-nonneg*)
(*use 0 1 2 3 4 in auto*)
thus *interval-lebesgue-integrable lborel 0 1 f*
by (*simp add: interval-lebesgue-integrable-def einterval-def*)
qed

lemma *interval-lebesgue-integral-lborel-01-cong*:
assumes $\bigwedge x. x \in \{0 < .. < 1\} \implies f x = g x$
shows *interval-lebesgue-integral lborel 0 1 f* =
interval-lebesgue-integral lborel 0 1 g
using *assms*
by (*subst (1 2) interval-integral-Ioo*)
(*auto intro!: set-lebesgue-integral-cong assms*)

lemma *nn-integral-0-1-ln-times-powr*:
assumes $e > -1$
shows ($\int^{+} y \in \{0 < .. < 1\}. \text{ennreal } (-\ln y * y \text{ powr } e) \partial \text{lborel}$) = *ennreal (1 / (e + 1)²)*
proof –
have *: (*LBINT x=0..1. -ln x * x powr e*) = *1 / (e + 1)²*
interval-lebesgue-integrable lborel 0 1 ($\lambda x. -\ln x * x \text{ powr } e$)
using *integral-0-1-ln-times-powr*[*OF assms*] **by** *simp-all*
have *eq: ($\lambda y. (\text{if } 0 < y \wedge y < 1 \text{ then } 1 \text{ else } 0) * \ln y * y \text{ powr } e$) =*

$(\lambda y. \text{if } 0 < y \wedge y < 1 \text{ then } \ln y * y \text{ powr } e \text{ else } 0)$
 by *auto*
have $(\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln y * y \text{ powr } e) \partial \text{lborel}) =$
 $(\int^{+y}. \text{ennreal } (-\ln y * y \text{ powr } e * \text{indicator } \{0 < .. < 1\} y) \partial \text{lborel})$
 by *(intro nn-integral-cong) (auto simp: indicator-def)*
also have $\dots = \text{ennreal } (1 / (e + 1)^2)$
 using ** eq*
 by *(subst nn-integral-eq-integral)*
(auto intro!: AE-I2 simp: indicator-def interval-lebesgue-integrable-def
set-integrable-def one-ereal-def zero-ereal-def interval-integral-Ioo
mult-ac mult-nonpos-nonneg set-lebesgue-integral-def)
finally show *?thesis* .
qed

lemma *nn-integral-0-1-ln-times-power:*
 $(\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln y * y ^ n) \partial \text{lborel}) = \text{ennreal } (1 / (n + 1)^2)$
proof –
have $(\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln y * y ^ n) \partial \text{lborel}) =$
 $(\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln y * y \text{ powr } \text{real } n) \partial \text{lborel})$
 by *(intro set-nn-integral-cong) (auto simp: powr-realpow)*
also have $\dots = \text{ennreal } (1 / (n + 1)^2)$
 by *(subst nn-integral-0-1-ln-times-power) auto*
finally show *?thesis by simp*
qed

Next, we also evaluate the more trivial integral

$$\int_0^1 x^n dx .$$

lemma *nn-integral-0-1-power:*
 $(\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (y ^ n) \partial \text{lborel}) = \text{ennreal } (1 / (n + 1))$
proof –
have **: (($\lambda a. a ^ (n + 1) / \text{real } (n + 1)$) has-real-derivative $x ^ n$) (at x)* **for** x
 by *(rule derivative-eq-intros refl | simp)+*
have $(\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (y ^ n) \partial \text{lborel}) = (\int^{+y \in \{0..1\}}. \text{ennreal } (y ^ n) \partial \text{lborel})$
 by *(intro nn-integral-cong-AE AE-I[of - - {0,1}])*
(auto simp: indicator-def emeasure-lborel-countable)
also have $\dots = \text{ennreal } (1 ^ (n + 1) / (n + 1) - 0 ^ (n + 1) / (n + 1))$
 using ** by (intro nn-integral-FTC-Icc) auto*
also have $\dots = \text{ennreal } (1 / (n + 1))$
 by *simp*
finally show *?thesis by simp*
qed

I_1 can alternatively be written as the triple integral

$$\int_0^1 \int_0^1 \int_0^1 \frac{x^r y^s}{1 - (1 - xy)w} dx dy dw .$$

lemma *beukers-nn-integral1-altdef*:

beukers-nn-integral1 $r s =$
 $(\int^+(w,x,y) \in \{0 < .. < 1\} \times \{0 < .. < 1\} \times \{0 < .. < 1\}).$
 $\text{ennreal } (1 / (1 - (1 - x * y) * w) * x^{\wedge} r * y^{\wedge} s) \partial \text{lborel})$

proof –

have $(\int^+(w,x,y) \in \{0 < .. < 1\} \times \{0 < .. < 1\} \times \{0 < .. < 1\}).$
 $\text{ennreal } (1 / (1 - (1 - x * y) * w) * x^{\wedge} r * y^{\wedge} s) \partial \text{lborel}) =$
 $(\int^+(x,y) \in \{0 < .. < 1\} \times \{0 < .. < 1\}). (\int^+ w \in \{0 < .. < 1\}).$
 $\text{ennreal } (1 / (1 - (1 - x * y) * w) * x^{\wedge} r * y^{\wedge} s) \partial \text{lborel}) \partial \text{lborel})$

by (*subst lborel-prod [symmetric]*, *subst lborel-pair.nn-integral-snd [symmetric]*)
(auto simp: case-prod-unfold indicator-def simp flip: lborel-prod intro!: nn-integral-cong)

also have $\dots = (\int^+(x,y) \in \{0 < .. < 1\} \times \{0 < .. < 1\}). \text{ennreal } (-\ln (x * y) / (1 - x * y))$
 $* x^{\wedge} r * y^{\wedge} s) \partial \text{lborel})$

proof (*intro nn-integral-cong, clarify*)

fix $x y :: \text{real}$

have $(\int^+ w \in \{0 < .. < 1\}). \text{ennreal } (1 / (1 - (1 - x * y) * w) * x^{\wedge} r * y^{\wedge} s) \partial \text{lborel}) =$
 $\text{ennreal } (-\ln (x * y) * x^{\wedge} r * y^{\wedge} s / (1 - x * y))$

if $xy: (x, y) \in \{0 < .. < 1\} \times \{0 < .. < 1\}$

proof –

from xy **have** $x * y < 1$

using *mult-strict-mono[of x 1 y 1]* **by** *simp*

have *deriv: (($\lambda w. -\ln (1 - (1 - x * y) * w) / (1 - x * y)$) has-real-derivative*
 $1 / (1 - (1 - x * y) * w))$ *(at w) if w: w $\in \{0..1\}$ for w*

by (*insert xy w $\langle x * y < 1 \rangle$ beukers-denom-ineq[of x y w]*)
(rule derivative-eq-intros refl | simp add: divide-simps)+

have $(\int^+ w \in \{0 < .. < 1\}). \text{ennreal } (1 / (1 - (1 - x * y) * w) * x^{\wedge} r * y^{\wedge} s) \partial \text{lborel}) =$
 $\text{ennreal } (x^{\wedge} r * y^{\wedge} s) * (\int^+ w \in \{0 < .. < 1\}). \text{ennreal } (1 / (1 - (1 - x * y) * w))$
 $\partial \text{lborel})$

using xy **by** (*subst nn-integral-cmult [symmetric]*)
(auto intro!: nn-integral-cong simp: indicator-def simp flip: ennreal-mult')

also have $(\int^+ w \in \{0 < .. < 1\}). \text{ennreal } (1 / (1 - (1 - x * y) * w)) \partial \text{lborel}) =$
 $(\int^+ w \in \{0..1\}). \text{ennreal } (1 / (1 - (1 - x * y) * w)) \partial \text{lborel})$

by (*intro nn-integral-cong-AE AE-I[of - - $\{0, 1\}$]*)
(auto simp: emeasure-lborel-countable indicator-def)

also have $(\int^+ w \in \{0..1\}). \text{ennreal } (1 / (1 - (1 - x * y) * w)) \partial \text{lborel}) =$
 $\text{ennreal } (-\ln (1 - (1 - x * y) * 1) / (1 - x * y) - (-\ln (1 - (1 - x * y) * 0) / (1 - x * y)))$

using xy *deriv less-imp-le[OF beukers-denom-ineq[of x y]]*

by (*intro nn-integral-FTC-Icc*) *auto*

finally show *?thesis using xy*

by (*simp flip: ennreal-mult' ennreal-mult'' add: mult-ac*)

qed

thus $(\int^+ w \in \{0 < .. < 1\}). \text{ennreal } (1 / (1 - (1 - x * y) * w) * x^{\wedge} r * y^{\wedge} s) \partial \text{lborel}) * \text{indicator}$
 $(\{0 < .. < 1\} \times \{0 < .. < 1\}) (x, y) =$
 $\text{ennreal } (-\ln (x * y) / (1 - x * y) * x^{\wedge} r * y^{\wedge} s) * \text{indicator } (\{0 < .. < 1\} \times \{0 < .. < 1\})$
 (x, y)

by (*auto simp: indicator-def*)

qed

also have $\dots = \text{beukers-nn-integral1 } r s$

by (simp add: beukers-nn-integral1-def)
 finally show ?thesis ..
 qed

context

fixes $r s :: \text{nat}$ and $I1 I2' :: \text{real}$ and $I2 :: \text{ennreal}$ and $D :: (\text{real} \times \text{real} \times \text{real})$
 set

assumes $rs: s \leq r$

defines $D \equiv \{0 < .. < 1\} \times \{0 < .. < 1\} \times \{0 < .. < 1\}$

begin

By unfolding the geometric series, pulling the summation out and evaluating the integrals, we find that

$$I_1 = \sum_{k=0}^{\infty} \frac{1}{(k+r+1)^2(k+s+1)} + \frac{1}{(k+r+1)(k+s+1)^2}.$$

lemma beukers-nn-integral1-series:

beukers-nn-integral1 $r s = (\sum k. \text{ennreal} (1 / ((k+r+1)^2 * (k+s+1)) + 1 / ((k+r+1) * (k+s+1)^2)))$

proof –

have beukers-nn-integral1 $r s =$

$(\int^{+}(x,y) \in \{0 < .. < 1\} \times \{0 < .. < 1\}. (\sum k. \text{ennreal} (-\ln (x*y) * x^{k+r} * y^{k+s})) \partial \text{lborel})$

unfolding beukers-nn-integral1-def

proof (intro set-nn-integral-cong refl, clarify)

fix $x y :: \text{real}$ assume $xy: x \in \{0 < .. < 1\} y \in \{0 < .. < 1\}$

from xy have $x * y < 1$ using mult-strict-mono[of x 1 y 1] by simp

have $(\sum k. \text{ennreal} (-\ln (x*y) * x^{k+r} * y^{k+s})) =$

$\text{ennreal} (-\ln (x*y) * x^r * y^s) * (\sum k. \text{ennreal} ((x*y)^k))$

using xy by (subst ennreal-suminf-cmult [symmetric], subst ennreal-mult'' [symmetric])

(auto simp: power-add mult-ac power-mult-distrib)

also have $(\sum k. \text{ennreal} ((x*y)^k)) = \text{ennreal} (1 / (1 - x*y))$

using geometric-sums[of $x*y$] $\langle x * y < 1 \rangle xy$ by (intro suminf-ennreal-eq)

auto

also have $\text{ennreal} (-\ln (x*y) * x^r * y^s) * \dots =$

$\text{ennreal} (-\ln (x*y) / (1 - x*y) * x^r * y^s)$

using $\langle x * y < 1 \rangle$ by (subst ennreal-mult'' [symmetric]) auto

finally show $\text{ennreal} (-\ln (x*y) / (1 - x*y) * x^r * y^s) =$

$(\sum k. \text{ennreal} (-\ln (x*y) * x^{k+r} * y^{k+s})) ..$

qed

also have $\dots = (\sum k. (\int^{+}(x,y) \in \{0 < .. < 1\} \times \{0 < .. < 1\}. (\text{ennreal} (-\ln (x*y) * x^{k+r} * y^{k+s})) \partial \text{lborel}))$

unfolding case-prod-unfold by (subst nn-integral-suminf [symmetric]) (auto simp flip: borel-prod)

also have $\dots = (\sum k. \text{ennreal} (1 / ((k+r+1)^2 * (k+s+1)) + 1 / ((k+r+1) * (k+s+1)^2)))$

proof (rule suminf-cong)

fix $k :: \text{nat}$

define F where $F = (\lambda x y :: \text{real}. x + y)$

have $(\int^{+x,y \in \{0 < .. < 1\}} \times \{0 < .. < 1\}. \text{ennreal } (-\ln(x*y) * x^{(k+r)} * y^{(k+s)})) \partial\text{lborel} =$
 $(\int^{+x \in \{0 < .. < 1\}}. (\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln(x*y) * x^{(k+r)} * y^{(k+s)})) \partial\text{lborel}) \partial\text{lborel}$
unfolding *case-prod-unfold lborel-prod [symmetric]*
by (*subst lborel.nn-integral-fst [symmetric]*) (*auto intro!: nn-integral-cong simp: indicator-def*)
also have $\dots = (\int^{+x \in \{0 < .. < 1\}}. \text{ennreal } (-\ln x * x^{(k+r)} / (k+s+1) + x^{(k+r)} / (k+s+1)^2) \partial\text{lborel})$
proof (*intro set-nn-integral-cong refl, clarify*)
fix $x :: \text{real}$ **assume** $x \in \{0 < .. < 1\}$
have $(\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln(x*y) * x^{(k+r)} * y^{(k+s)})) \partial\text{lborel} =$
 $(\int^{+y \in \{0 < .. < 1\}}. (\text{ennreal } (-\ln x * x^{(k+r)} * y^{(k+s)}) + \text{ennreal } (-\ln y * x^{(k+r)} * y^{(k+s)})) \partial\text{lborel})$
by (*intro set-nn-integral-cong*)
(use x in <auto simp: ln-mult ring-distrib mult-nonpos-nonneg simp flip: ennreal-plus>)
also have $\dots = (\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln x * x^{(k+r)} * y^{(k+s)})) \partial\text{lborel} +$
 $(\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln y * x^{(k+r)} * y^{(k+s)})) \partial\text{lborel}$
by (*subst nn-integral-add [symmetric]*) (*auto intro!: nn-integral-cong simp: indicator-def*)
also have $(\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln x * x^{(k+r)} * y^{(k+s)})) \partial\text{lborel} =$
 $\text{ennreal } (-\ln x * x^{(k+r)}) * (\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (y^{(k+s)})) \partial\text{lborel}$
by (*subst nn-integral-cmult [symmetric]*)
(auto intro!: nn-integral-cong simp: indicator-def simp flip: ennreal-mult'')
also have $(\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (y^{(k+s)})) \partial\text{lborel} = \text{ennreal } (1 / (k+s+1))$
by (*subst nn-integral-0-1-power*) *simp*
also have $\text{ennreal } (-\ln x * x^{(k+r)}) * \dots = \text{ennreal } (-\ln x * x^{(k+r)} / (k+s+1))$
by (*subst ennreal-mult'' [symmetric]*) *auto*
also have $(\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln y * x^{(k+r)} * y^{(k+s)})) \partial\text{lborel} =$
 $\text{ennreal } (x^{(k+r)}) * (\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln y * y^{(k+s)})) \partial\text{lborel}$
by (*subst nn-integral-cmult [symmetric]*)
(use x in <auto intro!: nn-integral-cong simp: indicator-def mult-ac simp flip: ennreal-mult'>)
also have $(\int^{+y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln y * y^{(k+s)})) \partial\text{lborel} = \text{ennreal } (1 / (k+s+1)^2)$
by (*subst nn-integral-0-1-ln-times-power*) *simp*
also have $\text{ennreal } (x^{(k+r)}) * \dots = \text{ennreal } (x^{(k+r)} / (k+s+1)^2)$
by (*subst ennreal-mult'' [symmetric]*) *auto*
also have $\text{ennreal } (-\ln x * x^{(k+r)} / (k+s+1)) + \dots =$
 $\text{ennreal } (-\ln x * x^{(k+r)} / (k+s+1) + x^{(k+r)} / (k+s+1)^2)$
using x **by** (*subst ennreal-plus*) (*auto simp: mult-nonpos-nonneg divide-nonpos-nonneg*)
finally show $(\int^{+x,y \in \{0 < .. < 1\}}. \text{ennreal } (-\ln(x*y) * x^{(k+r)} * y^{(k+s)})) \partial\text{lborel} =$

$ennreal (-\ln x * x^{(k+r)} / (k+s+1) + x^{(k+r)}/(k+s+1)^2) .$

qed

also have ... = $(\int^{+x \in \{0 < .. < 1\}}. (ennreal (-\ln x * x^{(k+r)} / (k+s+1)) + ennreal (x^{(k+r)}/(k+s+1)^2)) \partial lborel)$

by (*intro set-nn-integral-cong refl, subst ennreal-plus*)
(auto simp: mult-nonpos-nonneg divide-nonpos-nonneg)

also have ... = $(\int^{+x \in \{0 < .. < 1\}}. ennreal (-\ln x * x^{(k+r)} / (k+s+1)) \partial lborel) +$
 $(\int^{+x \in \{0 < .. < 1\}}. ennreal (x^{(k+r)}/(k+s+1)^2) \partial lborel)$

by (*subst nn-integral-add [symmetric]*) *(auto intro!: nn-integral-cong simp: indicator-def)*

also have $(\int^{+x \in \{0 < .. < 1\}}. ennreal (-\ln x * x^{(k+r)} / (k+s+1)) \partial lborel) =$
 $ennreal (1 / (k+s+1)) * (\int^{+x \in \{0 < .. < 1\}}. ennreal (-\ln x * x^{(k+r)}) \partial lborel)$

by (*subst nn-integral-cmult [symmetric]*)
(auto intro!: nn-integral-cong simp: indicator-def simp flip: ennreal-mult')

also have ... = $ennreal (1 / ((k+s+1) * (k+r+1)^2))$

by (*subst nn-integral-0-1-ln-times-power, subst ennreal-mult [symmetric]*) *(auto simp: algebra-simps)*

also have $(\int^{+x \in \{0 < .. < 1\}}. ennreal (x^{(k+r)}/(k+s+1)^2) \partial lborel) =$
 $ennreal (1/(k+s+1)^2) * (\int^{+x \in \{0 < .. < 1\}}. ennreal (x^{(k+r)}) \partial lborel)$

by (*subst nn-integral-cmult [symmetric]*)
(auto intro!: nn-integral-cong simp: indicator-def simp flip: ennreal-mult')

also have ... = $ennreal (1/((k+r+1)*(k+s+1)^2))$

by (*subst nn-integral-0-1-power, subst ennreal-mult [symmetric]*) *(auto simp: algebra-simps)*

also have $ennreal (1 / ((k+s+1) * (k+r+1)^2)) + ... =$
 $ennreal (1/((k+r+1)^2*(k+s+1)) + 1/((k+r+1)*(k+s+1)^2))$

by (*subst ennreal-plus [symmetric]*) *(auto simp: algebra-simps)*

finally show $(\int^{+(x,y) \in \{0 < .. < 1\} \times \{0 < .. < 1\}}. ennreal (-\ln (x*y) * x^{(k+r)} * y^{(k+s)}) \partial lborel) =$

qed

finally show *?thesis .*

qed

Remembering that $\zeta(3) = \sum k^{-3}$, it is easy to see that if $r = s$, this sum is simply

$$2 \left(\zeta(3) - \sum_{k=1}^r \frac{1}{k^3} \right) .$$

lemma *beukers-nn-integral1-same:*

assumes $r = s$

shows $beukers-nn-integral1\ r\ s = ennreal (2 * (Re (zeta 3) - (\sum_{k=1..r}. 1 / k^3)))$

and $2 * (Re (zeta 3) - (\sum_{k=1..r}. 1 / k^3)) \geq 0$

proof –

from *assms* **have** [*simp*]: $s = r$ **by** *simp*

have *: $Suc\ 2 = 3$ **by** *simp*

have *beukers-nn-integral1* $r\ s = (\sum k. \text{ennreal } (2 / (r + k + 1) ^ 3))$
unfolding *beukers-nn-integral1-series*
by (*simp only: assms power-Suc [symmetric] mult.commute[of $x ^ 2$ for x] * times-divide-eq-right mult-1-right add-ac flip: mult-2*)
also have **: $(\lambda k. 2 / (r + k + 1) ^ 3)$ *sums*
 $(2 * (\text{Re } (\text{zeta } 3) - (\sum k=1..r. 1 / k ^ 3)))$
using *sums-mult[OF sums-Re-zeta-of-nat-offset[of 3], of 2]* **by** *simp*
hence $(\sum k. \text{ennreal } (2 / (r + k + 1) ^ 3)) = \text{ennreal } \dots$
by (*intro suminf-ennreal-eq*) *auto*
finally show *beukers-nn-integral1* $r\ s = \text{ennreal } (2 * (\text{Re } (\text{zeta } 3) - (\sum k=1..r. 1 / k ^ 3)))$.
show $2 * (\text{Re } (\text{zeta } 3) - (\sum k=1..r. 1 / k ^ 3)) \geq 0$
by (*rule sums-le[OF - sums-zero **]*) *auto*
qed

lemma *beukers-integral1-same*:

assumes $r = s$
shows *beukers-integral1* $r\ s = 2 * (\text{Re } (\text{zeta } 3) - (\sum k=1..r. 1 / k ^ 3))$
proof –
have $\ln (a * b) * a ^ r * b ^ s / (1 - a * b) \leq 0$ **if** $a \in \{0 <.. < 1\}$ $b \in \{0 <.. < 1\}$
for $a\ b :: \text{real}$
using *that mult-strict-mono[of a 1 b 1]* **by** (*intro mult-nonpos-nonneg divide-nonpos-nonneg*) *auto*
thus *?thesis*
using *beukers-nn-integral1-same[OF assms]*
unfolding *beukers-nn-integral1-def beukers-integral1-def*
by (*intro set-integral-eq-nn-integral AE-I2*)
(auto simp flip: lborel-prod simp: case-prod-unfold set-borel-measurable-def intro: divide-nonpos-nonneg mult-nonpos-nonneg)
qed

In contrast, for $r > s$, we find that

$$I_1 = \frac{1}{r - s} \sum_{k=s+1}^r \frac{1}{k^2}.$$

lemma *beukers-nn-integral1-different*:

assumes $r > s$
shows *beukers-nn-integral1* $r\ s = \text{ennreal } ((\sum k \in \{s <.. r\}. 1 / k ^ 2) / (r - s))$
proof –
have $(\lambda k. 1 / (r - s) * (1 / (s + k + 1) ^ 2 - 1 / (r + k + 1) ^ 2))$
 $\text{sums } (1 / (r - s) * ((\text{Re } (\text{zeta } (\text{of-nat } 2)) - (\sum k=1..s. 1 / k ^ 2)) - (\text{Re } (\text{zeta } (\text{of-nat } 2)) - (\sum k=1..r. 1 / k ^ 2))))$
(is - sums ?S) **by** (*intro sums-mult sums-diff sums-Re-zeta-of-nat-offset*) *auto*
also have $?S = ((\sum k=1..r. 1 / k ^ 2) - (\sum k=1..s. 1 / k ^ 2)) / (r - s)$
by (*simp add: algebra-simps diff-divide-distrib*)
also have $(\sum k=1..r. 1 / k ^ 2) - (\sum k=1..s. 1 / k ^ 2) = (\sum k \in \{1..r\} - \{1..s\}. 1 / k ^ 2)$
using *assms* **by** (*subst Groups-Big.sum-diff*) *auto*

also have $\{1..r\} - \{1..s\} = \{s<..r\}$ by *auto*

also have $(\lambda k. 1 / (r - s) * (1 / (s + k + 1) ^ 2 - 1 / (r + k + 1) ^ 2)) =$
 $(\lambda k. 1 / ((k+r+1) * (k+s+1) ^ 2) + 1 / ((k+r+1) ^ 2 * (k+s+1)))$

proof (*intro ext, goal-cases*)
case (1 k)
define x **where** x = *real* (k + r + 1)
define y **where** y = *real* (k + s + 1)
have [*simp*]: x ≠ 0 y ≠ 0 **by** (*auto simp: x-def y-def*)
have (x² * y + x * y²) * (*real* r - *real* s) = x * y * (x² - y²)
by (*simp add: algebra-simps power2-eq-square x-def y-def*)
hence 1 / (x*y²) + 1 / (x²*y) = 1 / (r - s) * (1 / y² - 1 / x²)
using *assms* **by** (*simp add: divide-simps of-nat-diff*)
thus ?*case* **by** (*simp add: x-def y-def algebra-simps*)
qed
finally show ?*thesis*
unfolding *beukers-nn-integral1-series* **by** (*intro suminf-ennreal-eq*) (*auto simp:*
add-ac)
qed

lemma *beukers-integral1-different*:
assumes r > s
shows *beukers-integral1* r s = $(\sum k \in \{s<..r\}. 1 / k ^ 2) / (r - s)$
proof -
have $\ln (a * b) * a ^ r * b ^ s / (1 - a * b) \leq 0$ **if** a ∈ {0<.. <1 } b ∈ {0<.. <1 }
for a b :: *real*
using *that mult-strict-mono*[of a 1 b 1] **by** (*intro mult-nonpos-nonneg di-*
vide-nonpos-nonneg) *auto*
thus ?*thesis*
using *beukers-nn-integral1-different*[OF *assms*]
unfolding *beukers-nn-integral1-def beukers-integral1-def*
by (*intro set-integral-eq-nn-integral AE-I2*)
(auto simp flip: lborel-prod simp: case-prod-unfold set-borel-measurable-def
intro: divide-nonpos-nonneg mult-nonpos-nonneg intro!: sum-nonneg
divide-nonneg-nonneg)
qed

end

It is also easy to see that if we exchange r and s, nothing changes.

lemma *beukers-nn-integral1-swap*:
beukers-nn-integral1 r s = *beukers-nn-integral1* s r
unfolding *beukers-nn-integral1-def lborel-prod* [*symmetric*]
by (*subst lborel-pair.nn-integral-swap, simp*)
(intro nn-integral-cong, auto simp: indicator-def algebra-simps split: if-splits)

lemma *beukers-nn-integral1-finite*: *beukers-nn-integral1* r s < ∞
using *beukers-nn-integral1-different*[of r s] *beukers-nn-integral1-different*[of s r]
by (*cases r s rule: linorder-cases*)

(simp-all add: beukers-nn-integral1-same beukers-nn-integral1-swap)

lemma *beukers-integral1-integrable*:

set-integrable lborel ($\{0 < .. < 1\} \times \{0 < .. < 1\}$)
 $(\lambda(x,y). (-\ln(x*y) / (1 - x*y) * x^r * y^s :: \text{real}))$

proof (intro set-integrableI-nonneg AE-I2; clarify?)

fix $x y :: \text{real}$ **assume** $xy: x \in \{0 < .. < 1\} y \in \{0 < .. < 1\}$

have $0 \geq \ln(x * y) / (1 - x * y) * x^r * y^s$

using mult-strict-mono[of x 1 y 1]

by (intro mult-nonpos-nonneg divide-nonpos-nonneg) (use xy in auto)

thus $0 \leq -\ln(x * y) / (1 - x * y) * x^r * y^s$ **by** simp

next

show $\int^+ x \in \{0 < .. < 1\} \times \{0 < .. < 1\}. \text{ennreal}(\text{case } x \text{ of } (x, y) \Rightarrow$
 $-\ln(x * y) / (1 - x * y) * x^r * y^s) \partial \text{lborel} < \infty$

using beukers-nn-integral1-finite **by** (simp add: beukers-nn-integral1-def case-prod-unfold)

qed (simp-all flip: lborel-prod add: set-borel-measurable-def)

lemma *beukers-integral1-integrable'*:

set-integrable lborel ($\{0 < .. < 1\} \times \{0 < .. < 1\} \times \{0 < .. < 1\}$)
 $(\lambda(z,x,y). (x^r * y^s / (1 - (1 - x*y) * z) :: \text{real}))$

proof (intro set-integrableI-nonneg AE-I2; clarify?)

fix $x y z :: \text{real}$ **assume** $xyz: x \in \{0 < .. < 1\} y \in \{0 < .. < 1\} z \in \{0 < .. < 1\}$

show $0 \leq x^r * y^s / (1 - (1 - x*y) * z)$

using mult-strict-mono[of x 1 y 1] xyz beukers-denom-ineq[of x y z]

by (intro mult-nonneg-nonneg divide-nonneg-nonneg) auto

next

show $\int^+ x \in \{0 < .. < 1\} \times \{0 < .. < 1\} \times \{0 < .. < 1\}. \text{ennreal}(\text{case } x \text{ of } (z,x,y) \Rightarrow$
 $x^r * y^s / (1 - (1 - x*y) * z)) \partial \text{lborel} < \infty$

using beukers-nn-integral1-finite

by (simp add: beukers-nn-integral1-altdef case-prod-unfold)

qed (simp-all flip: lborel-prod add: set-borel-measurable-def)

lemma *beukers-integral1-conv-nn-integral*:

beukers-integral1 r s = enn2real (beukers-nn-integral1 r s)

proof -

have $\ln(a * b) * a^r * b^s / (1 - a * b) \leq 0$ **if** $a \in \{0 < .. < 1\} b \in \{0 < .. < 1\}$

for $a b :: \text{real}$

using mult-strict-mono[of a 1 b 1] **that** **by** (intro divide-nonpos-nonneg mult-nonpos-nonneg)

auto

thus ?thesis **unfolding** beukers-integral1-def **using** beukers-nn-integral1-finite[of r s]

by (intro set-integral-eq-nn-integral)

(auto simp: case-prod-unfold beukers-nn-integral1-def

set-borel-measurable-def simp flip: borel-prod

intro!: AE-I2 intro: divide-nonpos-nonneg mult-nonpos-nonneg)

qed

lemma *beukers-integral1-swap*: beukers-integral1 r s = beukers-integral1 s r

by (simp add: beukers-integral1-conv-nn-integral beukers-nn-integral1-swap)

1.7 The second double integral

context

fixes $n :: \text{nat}$
fixes $D :: (\text{real} \times \text{real}) \text{ set}$ **and** $D' :: (\text{real} \times \text{real} \times \text{real}) \text{ set}$
fixes $P :: \text{real} \Rightarrow \text{real}$ **and** $Q :: \text{nat} \Rightarrow \text{real} \Rightarrow \text{real}$
defines $D \equiv \{0 < .. < 1\} \times \{0 < .. < 1\}$ **and** $D' \equiv \{0 < .. < 1\} \times \{0 < .. < 1\} \times \{0 < .. < 1\}$
defines $Q \equiv \text{Gen-Shleg } n$ **and** $P \equiv \text{Shleg } n$
begin

The next integral to consider is the following variant of I_1 :

$$I_2 := \int_0^1 \int_0^1 -\frac{\ln(xy)}{1-xy} P_n(x) P_n(y) dx dy .$$

definition *beukers-integral2* $:: \text{real where}$

$$\text{beukers-integral2} = (\int (x,y) \in D. (-\ln(x*y) / (1-x*y) * P x * P y) \partial \text{lborel})$$

I_2 is simply a sum of integrals of type I_1 , so using our results for I_1 , we can write I_2 in the form $A\zeta(3) + \frac{B}{\text{lcm}\{1..n\}^3}$ where A and B are integers and $A > 0$:

lemma *beukers-integral2-conv-int-combination*:

obtains $A B :: \text{int where } A > 0$ **and**

$$\text{beukers-integral2} = \text{of-int } A * \text{Re } (\text{zeta } 3) + \text{of-int } B / \text{of-nat } (\text{Lcm } \{1..n\} \wedge 3)$$

proof –

let $?I1 = (\lambda i. (i, i)) \text{ ' } \{..n\}$
let $?I2 = \text{Set.filter } (\lambda(i,j). i \neq j) (\{..n\} \times \{..n\})$
let $?I3 = \text{Set.filter } (\lambda(i,j). i < j) (\{..n\} \times \{..n\})$
let $?I4 = \text{Set.filter } (\lambda(i,j). i > j) (\{..n\} \times \{..n\})$
define p **where** $p = \text{shleg-poly } n$
define I **where** $I = (\text{SIGMA } i:\{..n\}. \{1..i\})$
define J **where** $J = (\text{SIGMA } (i,j):?I4. \{j<..i\})$
define h **where** $h = \text{beukers-integral1}$
define $A :: \text{int where } A = (\sum i \leq n. 2 * \text{poly.coeff } p i \wedge 2)$
define $B1$ **where** $B1 = (\sum (i,k) \in I. \text{real-of-int } (-2 * \text{coeff } p i \wedge 2) / \text{real-of-int } (k \wedge 3))$
define $B2$ **where** $B2 = (\sum ((i,j),k) \in J. \text{real-of-int } (2 * \text{coeff } p i * \text{coeff } p j) / \text{real-of-int } (k \wedge 2 * (i-j)))$
define d **where** $d = \text{Lcm } \{1..n\} \wedge 3$

have $[\text{simp}]$: $h i j = h j i$ **for** $i j$

by $(\text{simp add: } h\text{-def beukers-integral1-swap})$

have *beukers-integral2* =

$$(\int (x,y) \in D. (\sum (i,j) \in \{..n\} \times \{..n\}. \text{coeff } p i * \text{coeff } p j * -\ln(x*y) / (1-x*y) * x \wedge i * y \wedge j) \partial \text{lborel})$$

unfolding *beukers-integral2-def*

by (*subst sum.cartesian-product [symmetric]*)
(simp add: poly-altdef P-def Shleg-def mult-ac case-prod-unfold p-def sum-distrib-left sum-distrib-right sum-negf sum-divide-distrib)
also have $\dots = (\sum_{(i,j) \in \{..n\} \times \{..n\}} \text{coeff } p \ i * \text{coeff } p \ j * h \ i \ j)$
unfolding *case-prod-unfold*
proof (*subst set-integral-sum*)
fix $ij :: \text{nat} \times \text{nat}$
have *set-integrable lborel D*
 $(\lambda(x,y). \text{real-of-int } (\text{coeff } p \ (\text{fst } ij) * \text{coeff } p \ (\text{snd } ij)) * (-\ln (x*y) / (1-x*y) * x^{\text{fst } ij} * y^{\text{snd } ij}))$
unfolding *case-prod-unfold using beukers-integral1-integrable[of fst ij snd ij]*
by (*intro set-integrable-mult-right*) (*auto simp: D-def case-prod-unfold*)
thus *set-integrable lborel D*
 $(\lambda pa. \text{real-of-int } (\text{coeff } p \ (\text{fst } ij) * \text{coeff } p \ (\text{snd } ij)) * -\ln (\text{fst } pa * \text{snd } pa) / (1 - \text{fst } pa * \text{snd } pa) * \text{fst } pa^{\text{fst } ij} * \text{snd } pa^{\text{snd } ij})$
by (*simp add: mult-ac case-prod-unfold*)
qed (*auto simp: beukers-integral1-def h-def case-prod-unfold mult.assoc D-def simp flip: set-integral-mult-right*)
also have $\dots = (\sum_{(i,j) \in ?I1 \cup ?I2} \text{coeff } p \ i * \text{coeff } p \ j * h \ i \ j)$
by (*intro sum.cong*) *auto*
also have $\dots = (\sum_{(i,j) \in ?I1} \text{coeff } p \ i * \text{coeff } p \ j * h \ i \ j) + (\sum_{(i,j) \in ?I2} \text{coeff } p \ i * \text{coeff } p \ j * h \ i \ j)$
by (*intro sum.union-disjoint*) *auto*
also have $(\sum_{(i,j) \in ?I1} \text{coeff } p \ i * \text{coeff } p \ j * h \ i \ j) = (\sum_{i \leq n} \text{coeff } p \ i^2 * h \ i \ i)$
by (*subst sum.reindex*) (*auto intro: inj-onI simp: case-prod-unfold power2-eq-square*)
also have $\dots = (\sum_{i \leq n} \text{coeff } p \ i^2 * 2 * (\text{Re } (\text{zeta } 3) - (\sum_{k=1..i} 1 / k^3)))$
unfolding *h-def D-def*
by (*intro sum.cong refl, subst beukers-integral1-same*) *auto*
also have $\dots = \text{of-int } A * \text{Re } (\text{zeta } 3) - (\sum_{i \leq n} 2 * \text{coeff } p \ i^2 * (\sum_{k=1..i} 1 / k^3))$
by (*simp add: sum-subtractf sum-distrib-left sum-distrib-right algebra-simps A-def*)
also have $\dots = \text{of-int } A * \text{Re } (\text{zeta } 3) + B1$
unfolding *I-def B1-def* **by** (*subst sum.Sigma [symmetric]*) (*auto simp: sum-distrib-left sum-negf*)
also have $(\sum_{(i,j) \in ?I2} \text{coeff } p \ i * \text{coeff } p \ j * h \ i \ j) = (\sum_{(i,j) \in ?I3 \cup ?I4} \text{coeff } p \ i * \text{coeff } p \ j * h \ i \ j)$
by (*intro sum.cong*) *auto*
also have $\dots = (\sum_{(i,j) \in ?I3} \text{coeff } p \ i * \text{coeff } p \ j * h \ i \ j) + (\sum_{(i,j) \in ?I4} \text{coeff } p \ i * \text{coeff } p \ j * h \ i \ j)$
by (*intro sum.union-disjoint*) *auto*
also have $(\sum_{(i,j) \in ?I3} \text{coeff } p \ i * \text{coeff } p \ j * h \ i \ j) = (\sum_{(i,j) \in ?I4} \text{coeff } p \ i * \text{coeff } p \ j * h \ i \ j)$
by (*intro sum.reindex-bij-witness[of - \lambda(i,j). (j,i) \lambda(i,j). (j,i)]*) *auto*
also have $\dots + \dots = 2 * \dots$ **by** *simp*
also have $\dots = (\sum_{(i,j) \in ?I4} \sum_{k \in \{j <..i\}} 2 * \text{coeff } p \ i * \text{coeff } p \ j / k^2 /$

$(i - j)$
unfolding *sum-distrib-left*
by (*intro sum.cong refl*)
(auto simp: h-def beukers-integral1-different sum-divide-distrib sum-distrib-left mult-ac)
also have $\dots = B2$
unfolding *J-def B2-def* **by** (*subst sum.Sigma [symmetric]*) (*auto simp: case-prod-unfold*)

also have $\exists B1'. B1 = \text{real-of-int } B1' / \text{real-of-int } d$
unfolding *B1-def case-prod-unfold*
by (*rule sum-rationals-common-divisor'*) (*auto simp: d-def I-def*)
then obtain $B1'$ **where** $B1 = \text{real-of-int } B1' / \text{real-of-int } d \dots$

also have $\exists B2'. B2 = \text{real-of-int } B2' / \text{real-of-int } d$
unfolding *B2-def case-prod-unfold J-def*
proof (*rule sum-rationals-common-divisor'; clarsimp?*)
fix $i j k :: \text{nat}$ **assume** $ijk: i \leq n \ j < k \ k \leq i$
have $\text{int } (k^2 * (i - j)) \ \text{dvd} \ \text{int } (\text{Lcm } \{1..n\}^2 * \text{Lcm } \{1..n\})$
unfolding *int-dvd-int-iff* **using** ijk
by (*intro mult-dvd-mono dvd-power-same dvd-Lcm*) *auto*
also have $\dots = d$
by (*simp add: d-def power-numeral-reduce*)
finally show $\text{int } k^2 * \text{int } (i - j) \ \text{dvd} \ \text{int } d$ **by** *simp*
qed(*auto simp: d-def J-def intro!: Nat.gr0I*)
then obtain $B2'$ **where** $B2 = \text{real-of-int } B2' / \text{real-of-int } d \dots$

finally have $\text{beukers-integral2} =$
 $\text{of-int } A * \text{Re } (\text{zeta } 3) + \text{of-int } (B1' + B2') / \text{of-nat } (\text{Lcm } \{1..n\})$
 3
by (*simp add: add-divide-distrib d-def*)

moreover have $\text{coeff } p \ 0 = P \ 0$
unfolding *P-def p-def Shleg-def* **by** (*simp add: poly-0-coeff-0*)
hence $\text{coeff } p \ 0 = 1$
by (*simp add: P-def*)
hence $A > 0$
unfolding *A-def* **by** (*intro sum-pos2[of - 0]*) *auto*

ultimately show *?thesis*
by (*intro that[of A B1' + B2']*) *auto*
qed

lemma *beukers-integral2-integrable*:
 $\text{set-integrable l borel } D \ (\lambda(x,y). -\ln(x*y) / (1 - x*y) * P \ x * P \ y)$
proof –
have $\text{bounded } (P \ ' \ \{0..1\})$
unfolding *P-def Shleg-def*
by (*intro compact-imp-bounded compact-continuous-image continuous-intros*)
auto

then obtain C **where** $C: \bigwedge x. x \in \{0..1\} \implies \text{norm } (P x) \leq C$
unfolding *bounded-iff* **by** *fast*
have [*measurable*]: $P \in \text{borel-measurable borel}$ **by** (*simp add: P-def*)
from $C[\text{of } 0]$ **have** $C \geq 0$ **by** *simp*
show *?thesis*
proof (*rule set-integrable-bound[OF - - AE-I2]; clarify?*)
show *set-integrable lborel D* ($\lambda(x,y). C^2 * (-\ln(x*y) / (1 - x*y))$)
using *beukers-integral1-integrable[of 0 0]* **unfolding** *case-prod-unfold*
by (*intro set-integrable-mult-right*) (*auto simp: D-def*)
next
fix $x y :: \text{real}$
assume $xy: (x, y) \in D$
from xy **have** $x * y < 1$
using *mult-strict-mono[of x 1 y 1]* **by** (*simp add: D-def*)
have $\text{norm } (-\ln(x*y) / (1 - x*y) * P x * P y) = (-\ln(x*y)) / (1 - x*y)$
 $* \text{norm } (P x) * \text{norm } (P y)$
using $xy \langle x * y < 1 \rangle$ **by** (*simp add: abs-mult abs-divide D-def*)
also have $\dots \leq (-\ln(x*y)) / (1 - x*y) * C * C$
using $xy C[\text{of } x] C[\text{of } y] \langle x * y < 1 \rangle \langle C \geq 0 \rangle$
by (*intro mult-mono divide-left-mono*)
(auto simp: D-def divide-nonpos-nonneg mult-nonpos-nonneg)
also have $\dots = \text{norm } ((-\ln(x*y)) / (1 - x*y) * C * C)$
using $xy \langle x * y < 1 \rangle \langle C \geq 0 \rangle$ **by** (*simp add: abs-divide abs-mult D-def*)
finally show $\text{norm } (-\ln(x*y) / (1 - x*y) * P x * P y)$
 $\leq \text{norm } (\text{case } (x, y) \text{ of } (x, y) \Rightarrow C^2 * (-\ln(x*y) / (1 - x*y)))$
by (*auto simp: algebra-simps power2-eq-square abs-mult abs-divide*)
qed (*auto simp: D-def set-borel-measurable-def case-prod-unfold simp flip: lborel-prod*)
qed

1.8 The triple integral

Lastly, we turn to the triple integral

$$I_3 := \int_0^1 \int_0^1 \int_0^1 \frac{(x(1-x)y(1-y)w(1-w))^n}{(1 - (1-xy)w)^{n+1}} dx dy dw .$$

definition *beukers-nn-integral3* :: *ennreal* **where**

beukers-nn-integral3 =
 $(\int^{+(w,x,y) \in D'. ((x*(1-x)*y*(1-y)*w*(1-w))^n / (1 - (1-xy)*w)^{(n+1)})}$
 $\partial \text{lborel})$

definition *beukers-integral3* :: *real* **where**

beukers-integral3 =
 $(\int^{+(w,x,y) \in D'. ((x*(1-x)*y*(1-y)*w*(1-w))^n / (1 - (1-xy)*w)^{(n+1)})}$
 $\partial \text{lborel})$

We first prove the following bound (which is a consequence of the arithmetic-geometric mean inequality) that will help us to bound the triple integral.

lemma *beukers-integral3-integrand-bound*:

```

fixes  $x y z :: \text{real}$ 
assumes  $xyz: x \in \{0 < .. < 1\} y \in \{0 < .. < 1\} z \in \{0 < .. < 1\}$ 
shows  $(x*(1-x)*y*(1-y)*z*(1-z)) / (1-(1-x*y)*z) \leq 1 / 27$  (is ?lhs  $\leq$  -)
proof -
  have  $ineq1: x * (1 - x) \leq 1 / 4$  if  $x: x \in \{0..1\}$  for  $x :: \text{real}$ 
  proof -
    have  $x * (1 - x) - 1 / 4 = -((x - 1 / 2) ^ 2)$ 
    by (simp add: algebra-simps power2-eq-square)
    also have  $\dots \leq 0$ 
    by simp
    finally show ?thesis by simp
  qed

  have  $ineq2: x * (1 - x) ^ 2 \leq 4 / 27$  if  $x: x \in \{0..1\}$  for  $x :: \text{real}$ 
  proof -
    have  $x * (1 - x) ^ 2 - 4 / 27 = (x - 4 / 3) * (x - 1 / 3) ^ 2$ 
    by (simp add: algebra-simps power2-eq-square)
    also have  $\dots \leq 0$ 
    by (rule mult-nonpos-nonneg) (use x in auto)
    finally show ?thesis by simp
  qed

  have  $1 - (1-x*y)*z = (1 - z) + x * y * z$ 
  by (simp add: algebra-simps)
  also have  $\dots \geq 2 * \text{sqrt} (1 - z) * \text{sqrt} x * \text{sqrt} y * \text{sqrt} z$ 
  using arith-geo-mean-sqrt[of  $1 - z x * y * z$ ] xyz
  by (auto simp: real-sqrt-mult)

  finally have  $*$ : ?lhs  $\leq (x*(1-x)*y*(1-y)*z*(1-z)) / (2 * \text{sqrt} (1 - z) * \text{sqrt} x * \text{sqrt} y * \text{sqrt} z)$ 
  using xyz beukers-denom-ineq[of  $x y z$ ]
  by (intro divide-left-mono mult-nonneg-nonneg mult-pos-pos) auto

  have  $(x*(1-x)*y*(1-y)*z*(1-z)) = (\text{sqrt} x * \text{sqrt} x * (1-x) * \text{sqrt} y * \text{sqrt} y$ 
  *
   $(1-y) * \text{sqrt} z * \text{sqrt} z * \text{sqrt} (1-z) * \text{sqrt} (1-z))$ 
  using xyz by simp
  also have  $\dots / (2 * \text{sqrt} (1 - z) * \text{sqrt} x * \text{sqrt} y * \text{sqrt} z) =$ 
   $\text{sqrt} (x * (1 - x) ^ 2) * \text{sqrt} (y * (1 - y) ^ 2) * \text{sqrt} (z * (1 - z)) / 2$ 
  using xyz by (simp add: divide-simps real-sqrt-mult del: real-sqrt-mult-self)
  also have  $\dots \leq \text{sqrt} (4 / 27) * \text{sqrt} (4 / 27) * \text{sqrt} (1 / 4) / 2$ 
  using xyz by (intro divide-right-mono mult-mono real-sqrt-le-mono ineq1 ineq2)
  auto
  also have  $\dots = 1 / 27$ 
  by (simp add: real-sqrt-divide)
  finally show ?thesis using  $*$  by argo
qed

```

Connecting the above bound with our results of I_1 , it is easy to see that

$$I_3 \leq 2 \cdot 27^{-n} \cdot \zeta(3):$$

lemma *beukers-nn-integral3-le*:

$$\text{beukers-nn-integral3} \leq \text{ennreal } (2 * (1 / 27) ^ n * \text{Re } (\text{zeta } 3))$$

proof –

have D' [measurable]: $D' \in \text{sets } (\text{borel } \otimes_M \text{borel } \otimes_M \text{borel})$

unfolding D' -def **by** (*simp flip: borel-prod*)

have *beukers-nn-integral3* =

$$\left(\int^{+(w,x,y) \in D'} ((x*(1-x)*y*(1-y)*w*(1-w)) ^ n / (1-(1-x*y)*w) ^ (n+1)) \right) \partial \text{lborel}$$

by (*simp add: beukers-nn-integral3-def*)

also have $\dots \leq \left(\int^{+(w,x,y) \in D'} ((1 / 27) ^ n / (1-(1-x*y)*w)) \right) \partial \text{lborel}$

proof (*intro set-nn-integral-mono ennreal-leI, clarify, goal-cases*)

case ($1 \ w \ x \ y$)

have $(x*(1-x)*y*(1-y)*w*(1-w)) ^ n / (1-(1-x*y)*w) ^ (n+1) =$

$$\left((x*(1-x)*y*(1-y)*w*(1-w)) / (1-(1-x*y)*w) \right) ^ n / (1-(1-x*y)*w)$$

by (*simp add: divide-simps*)

also have $\dots \leq (1 / 27) ^ n / (1 - (1 - x * y) * w)$

using *beukers-denom-ineq*[*of x y w*] 1

by (*intro divide-right-mono power-mono beukers-integral3-integrand-bound*)

(*auto simp: D'-def*)

finally show ?case .

qed

$$\text{also have } \dots = \text{ennreal } ((1 / 27) ^ n) * \left(\int^{+(w,x,y) \in D'} (1 / (1-(1-x*y)*w)) \right) \partial \text{lborel}$$

unfolding *lborel-prod* [*symmetric*] *case-prod-unfold*

by (*subst nn-integral-cmult* [*symmetric*])

(*auto intro!: nn-integral-cong simp: indicator-def simp flip: ennreal-mult'*)

also have $\left(\int^{+(w,x,y) \in D'} (1 / (1-(1-x*y)*w)) \right) \partial \text{lborel} =$

$$\left(\int^{+(x,y) \in \{0 <..<1\} \times \{0 <..<1\}} \text{ennreal } (- (\ln (x * y)) / (1 - x * y)) \right) \partial \text{lborel}$$

using *beukers-nn-integral1-altdef*[*of 0 0*]

by (*simp add: beukers-nn-integral1-def D'-def case-prod-unfold*)

also have $\dots = \text{ennreal } (2 * \text{Re } (\text{zeta } 3))$

using *beukers-nn-integral1-same*[*of 0 0*] **by** (*simp add: D-def beukers-nn-integral1-def*)

$$\text{also have } \text{ennreal } ((1 / 27) ^ n) * \dots = \text{ennreal } (2 * (1 / 27) ^ n * \text{Re } (\text{zeta } 3))$$

by (*subst ennreal-mult'* [*symmetric*]) (*simp-all add: mult-ac*)

finally show ?thesis .

qed

lemma *beukers-nn-integral3-finite*: *beukers-nn-integral3* $< \infty$

by (*rule le-less-trans, rule beukers-nn-integral3-le*) *simp-all*

lemma *beukers-integral3-integrable*:

set-integrable lborel D' ($\lambda(w,x,y). (x*(1-x)*y*(1-y)*w*(1-w)) ^ n / (1-(1-x*y)*w) ^ (n+1)$)

unfolding *case-prod-unfold* **using** *less-imp-le*[*OF beukers-denom-ineq*] *beukers-nn-integral3-finite*

by (*intro set-integrableI-nonneg AE-I2 impI*)

(*auto simp: D'-def set-borel-measurable-def beukers-nn-integral3-def case-prod-unfold*)

simp flip: lborel-prod intro!: divide-nonneg-nonneg mult-nonneg-nonneg)

lemma *beukers-integral3-conv-nn-integral*:
beukers-integral3 = enn2real beukers-nn-integral3
unfolding *beukers-integral3-def* **using** *beukers-nn-integral3-finite less-imp-le[OF beukers-denom-ineq]*
by (*intro set-integral-eq-nn-integral AE-I2 impI*)
(*auto simp: D'-def set-borel-measurable-def beukers-nn-integral3-def case-prod-unfold simp flip: lborel-prod*)

lemma *beukers-integral3-le*: *beukers-integral3 ≤ 2 * (1 / 2^γ) ^ n * Re (zeta 3)*
proof –
have *beukers-integral3 = enn2real beukers-nn-integral3*
by (*rule beukers-integral3-conv-nn-integral*)
also have *... ≤ enn2real (ennreal (2 * (1 / 2^γ) ^ n * Re (zeta 3)))*
by (*intro enn2real-mono beukers-nn-integral3-le*) *auto*
also have *... = 2 * (1 / 2^γ) ^ n * Re (zeta 3)*
using *Re-zeta-ge-1[of 3]* **by** (*intro enn2real-ennreal mult-nonneg-nonneg*) *auto*
finally show *?thesis* .
qed

It is also easy to see that $I_3 > 0$.

lemma *beukers-nn-integral3-pos*: *beukers-nn-integral3 > 0*
proof –
have *D' [measurable]: D' ∈ sets (borel ⊗_M borel ⊗_M borel)*
unfolding *D'-def* **by** (*simp flip: borel-prod*)

have **: ¬(AE (w,x,y) in lborel. ennreal ((x*(1-x)*y*(1-y)*w*(1-w)) ^ n / (1-(1-x*y)*w) ^ (n+1)) * indicator D' (w,x,y) ≤ 0)*
(*is ¬(AE z in lborel. ?P z)*)

proof –
{
fix *w x y :: real* **assume** *xyw: (w,x,y) ∈ D'*
hence *(x*(1-x)*y*(1-y)*w*(1-w)) ^ n / (1-(1-x*y)*w) ^ (n+1) > 0*
using *beukers-denom-ineq[of x y w]*
by (*intro divide-pos-pos mult-pos-pos zero-less-power*) (*auto simp: D'-def*)
with *xyw* **have** *¬?P (w,x,y)*
by (*auto simp: indicator-def D'-def*)
}

hence **: ¬?P z if z ∈ D' for z* **using** *that* **by** *blast*

hence *{z ∈ space lborel. ¬?P z} = D'* **by** *auto*

moreover **have** *emeasure lborel D' = 1*

proof –

have *D' = box (0,0,0) (1,1,1)*

by (*auto simp: D'-def box-def Basis-prod-def*)

also have *emeasure lborel ... = 1*

by (*subst emeasure-lborel-box*) (*auto simp: Basis-prod-def*)

finally show *?thesis* **by** *simp*

qed

ultimately show *?thesis*

by (subst AE-iff-measurable[of D']) (simp-all flip: borel-prod)
qed

hence nn-integral lborel ($\lambda::\text{real}\times\text{real}\times\text{real}.$ 0) < beukers-nn-integral3
unfolding beukers-nn-integral3-def
 by (intro nn-integral-less) (simp-all add: case-prod-unfold flip: lborel-prod)
thus ?thesis **by** simp
qed

lemma beukers-integral3-pos: beukers-integral3 > 0
proof –
 have 0 < enn2real beukers-nn-integral3
using beukers-nn-integral3-pos beukers-nn-integral3-finite
 by (subst enn2real-positive-iff) auto
also have ... = beukers-integral3
 by (rule beukers-integral3-conv-nn-integral [symmetric])
finally show ?thesis .
qed

1.9 Connecting the double and triple integral

In this section, we will prove the most technically involved part, namely that $I_2 = I_3$. I will not go into detail about how this works – the reader is advised to simply look at Filaseta’s presentation of the proof.

The basic idea is to integrate by parts n times with respect to y to eliminate the factor $P(y)$, then change variables $z = \frac{1-w}{1-(1-xy)w}$, and then apply the same integration by parts n times to x to eliminate $P(x)$.

The first expand

$$-\frac{\ln(xy)}{1-xy} = \int_0^1 \frac{1}{1-(1-xy)z} dz .$$

lemma beukers-aux-ln-conv-integral:
fixes $x y :: \text{real}$
assumes $xy: x \in \{0 < .. < 1\}$ $y \in \{0 < .. < 1\}$
shows $-\ln(x*y) / (1-x*y) = (LBINT z=0..1. 1 / (1-(1-x*y)*z))$
proof –
 have $x * y < 1$
using mult-strict-mono[of x 1 y 1] xy **by** simp
have less: $(1 - x * y) * u < 1$ **if** $u: u \in \{0..1\}$ **for** u
proof –
from $u \langle x * y < 1 \rangle$ **have** $(1 - x * y) * u \leq (1 - x * y) * 1$
by (intro mult-left-mono) auto
also have ... < 1 * 1
using xy **by** (intro mult-strict-right-mono) auto
finally show $(1 - x * y) * u < 1$ **by** simp
qed

```

have neq:  $(1 - x * y) * u \neq 1$  if  $u \in \{0..1\}$  for  $u$ 
  using less[of  $u$ ] that by simp

let  $?F = \lambda z. \ln (1 - (1 - x * y) * z) / (x * y - 1)$ 
have  $(LBINT\ z = \text{ereal } 0 .. \text{ereal } 1. 1 / (1 - (1 - x * y) * z)) = ?F\ 1 - ?F\ 0$ 
proof (rule interval-integral-FTC-finite, goal-cases cont deriv)
  case cont
  show ?case
    using neq by (intro continuous-intros) auto
next
  case (deriv z)
  show ?case
    unfolding has-field-derivative-iff-has-vector-derivative [symmetric]
    by (insert less[of  $z$ ]  $xy \langle x * y < 1 \rangle$  deriv)
      (rule derivative-eq-intros refl | simp)+
qed
also have  $\dots = -\ln (x * y) / (1 - x * y)$ 
  using  $\langle x * y < 1 \rangle$  by (simp add: field-simps)
finally show ?thesis
  by (simp add: zero-ereal-def one-ereal-def)
qed

```

The first part we shall show is the integration by parts.

```

lemma beukers-aux-by-parts-aux:
  assumes  $xz: x \in \{0 < .. < 1\}$   $z \in \{0 < .. < 1\}$  and  $k \leq n$ 
  shows  $(LBINT\ y = 0 .. 1. Q\ n\ y * (1 / (1 - (1 - x * y) * z))) =$ 
     $(LBINT\ y = 0 .. 1. Q\ (n - k)\ y * (fact\ k * (x * z) ^ k / (1 - (1 - x * y) * z) ^ (k + 1)))$ 
  using assms(3)
proof (induction k)
  case (Suc k)
  note [derivative-intros] = DERIV-chain2[OF has-field-derivative-Gen-Shleg]
  define  $G$  where  $G = (\lambda y. -fact\ k * (x * z) ^ k / (1 - (1 - x * y) * z) ^ (k + 1))$ 
  define  $g$  where  $g = (\lambda y. fact\ (Suc\ k) * (x * z) ^ {Suc\ k} / (1 - (1 - x * y) * z) ^ (k + 2))$ 

  have less:  $(1 - x * y) * z < 1$  and neq:  $(1 - x * y) * z \neq 1$ 
    if  $y \in \{0..1\}$  for  $y$ 
  proof  $-$ 
    from  $yz$  have  $x * y \leq x * 1$ 
      by (intro mult-left-mono) auto
    also have  $\dots < 1$ 
      using  $xz$  by simp
    finally have  $(1 - x * y) * z \leq 1 * z$ 
      using  $xz\ y$  by (intro mult-right-mono) auto
    also have  $\dots < 1$ 
      using  $xz$  by simp
    finally show  $(1 - x * y) * z < 1$  by simp
    thus  $(1 - x * y) * z \neq 1$  by simp
  qed

```

```

have cont: continuous-on {0..1} g
using neq by (auto simp: g-def intro!: continuous-intros)
have deriv: (G has-real-derivative g y) (at y within {0..1}) if y: y ∈ {0..1} for
y
unfolding G-def
by (insert neq xz y, (rule derivative-eq-intros refl power-not-zero)+)
(auto simp: divide-simps g-def)
have deriv2: (Q (n - Suc k) has-real-derivative Q (n - k) y) (at y within {0..1})
for y
using Suc.prems by (auto intro!: derivative-eq-intros simp: Suc-diff-Suc Q-def)

have (LBINT y=0..1. Q (n - Suc k) y * (fact (Suc k) * (x*z)Suc k / (1 - (1 - x*y)*z)k+2)) =
(LBINT y=0..1. Q (n - Suc k) y * g y)
by (simp add: g-def)
also have (LBINT y=0..1. Q (n - Suc k) y * g y) = -(LBINT y=0..1. Q (n - k)
y * G y)
using Suc.prems deriv deriv2 cont
by (subst interval-lebesgue-integral-by-parts-01 [where f = Q (n - k) and G =
G])
(auto intro!: continuous-intros simp: Q-def)
also have ... = (LBINT y=0..1. Q (n - k) y * (fact k * (x*z)k / (1 - (1 - x*y)*z)k+1))
by (simp add: G-def flip: interval-lebesgue-integral-uminus)
finally show ?case using Suc by simp
qed auto

```

```

lemma beukers-aux-by-parts:
assumes xz: x ∈ {0<..1} z ∈ {0<..1}
shows (LBINT y=0..1. P y / (1 - (1 - x*y)*z)) =
(LBINT y=0..1. (x*y*z)n * (1 - y)n / (1 - (1 - x*y)*z)n+1)
proof -
have (LBINT y=0..1. P y * (1 / (1 - (1 - x*y)*z))) =
1 / fact n * (LBINT y=0..1. Q n y * (1 / (1 - (1 - x*y)*z)))
unfolding interval-lebesgue-integral-mult-right [symmetric]
by (simp add: P-def Q-def Shleg-altdef)
also have ... = (LBINT y=0..1. (x*y*z)n * (1 - y)n / (1 - (1 - x*y)*z)n+1)
by (subst beukers-aux-by-parts-aux [OF assms, of n], simp,
subst interval-lebesgue-integral-mult-right [symmetric])
(simp add: Q-def mult-ac Gen-Shleg-0-left power-mult-distrib)
finally show ?thesis by simp
qed

```

We then get the following by applying the integration by parts to y :

```

lemma beukers-aux-integral-transform1:
fixes z :: real
assumes z: z ∈ {0<..1}
shows (LBINT (x,y):D. P x * P y / (1 - (1 - x*y)*z)) =

```

$(LBINT (x,y):D. P x * (x*y*z)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*z)^{\wedge}(n+1))$

proof –

have $cbox$: $cbox (0, 0) (1, 1) = (\{0..1\} \times \{0..1\}) :: (real \times real) \text{ set}$
by (*auto simp: cbox-def Basis-prod-def inner-prod-def*)

have box : $box (0, 0) (1, 1) = (\{0<..
by (*auto simp: box-def Basis-prod-def inner-prod-def*)$

have *set-integrable lborel* ($cbox (0,0) (1,1)$)
 $(\lambda(x, y). P x * P y / (1 - (1 - x * y) * z))$
unfolding *lborel-prod case-prod-unfold P-def*

proof (*intro continuous-on-imp-set-integrable-cbox continuous-intros ballI*)
fix $p :: real \times real$ **assume** $p: p \in cbox (0, 0) (1, 1)$
have $(1 - fst p * snd p) * z \leq 1 * z$
using *mult-mono[of fst p 1 snd p 1] p z cbox* **by** (*intro mult-right-mono*) *auto*
also have $\dots < 1$ **using** z **by** *simp*
finally show $1 - (1 - fst p * snd p) * z \neq 0$ **by** *simp*

qed

hence *integrable: set-integrable lborel* ($box (0,0) (1,1)$)
 $(\lambda(x, y). P x * P y / (1 - (1 - x * y) * z))$
by (*rule set-integrable-subset*) (*auto simp: box simp flip: borel-prod*)

have *set-integrable lborel* ($cbox (0,0) (1,1)$)
 $(\lambda(x, y). P x * (x*y*z)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*z)^{\wedge}(n+1))$
unfolding *lborel-prod case-prod-unfold P-def*

proof (*intro continuous-on-imp-set-integrable-cbox continuous-intros ballI*)
fix $p :: real \times real$ **assume** $p: p \in cbox (0, 0) (1, 1)$
have $(1 - fst p * snd p) * z \leq 1 * z$
using *mult-mono[of fst p 1 snd p 1] p z cbox* **by** (*intro mult-right-mono*) *auto*
also have $\dots < 1$ **using** z **by** *simp*
finally show $(1 - (1 - fst p * snd p) * z)^{\wedge}(n + 1) \neq 0$ **by** *simp*

qed

hence *integrable': set-integrable lborel* D
 $(\lambda(x, y). P x * (x*y*z)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*z)^{\wedge}(n+1))$
by (*rule set-integrable-subset*) (*auto simp: box D-def simp flip: borel-prod*)

have $(LBINT (x,y):D. P x * P y / (1-(1-x*y)*z)) =$
 $(LBINT x=0..1. (LBINT y=0..1. P x * P y / (1-(1-x*y)*z)))$
unfolding D -def *lborel-prod [symmetric]* **using** *box integrable*
by (*subst lborel-pair.set-integral-fst'*) (*simp-all add: interval-integral-Ioo lborel-prod*)

also have $\dots = (LBINT x=0..1. P x * (LBINT y=0..1. P y / (1-(1-x*y)*z)))$
by (*subst interval-lebesgue-integral-mult-right [symmetric]*) (*simp add: mult-ac*)

also have $\dots = (LBINT x=0..1. P x * (LBINT y=0..1. (x*y*z)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*z)^{\wedge}(n+1)))$
using z **by** (*intro interval-lebesgue-integral-lborel-01-cong, subst beukers-aux-by-parts*)

auto

also have $\dots = (LBINT x=0..1. (LBINT y=0..1. P x * (x*y*z)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*z)^{\wedge}(n+1)))$
by (*subst interval-lebesgue-integral-mult-right [symmetric]*) (*simp add: mult-ac*)

also have $\dots = (LBINT (x,y):D. P x * (x*y*z)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*z)^{\wedge}(n+1))$
unfolding D -def *lborel-prod [symmetric]* **using** *box integrable'*

by (*subst lborel-pair.set-integral-fst'*)
 (*simp-all add: D-def interval-integral-Ioo lborel-prod*)
finally show ($LBINT (x,y):D. P x * P y / (1-(1-x*y)*z) = \dots$) .
qed

We then change variables for z :

lemma *beukers-aux-integral-transform2*:

assumes $xy: x \in \{0 < .. < 1\} \ y \in \{0 < .. < 1\}$

shows ($LBINT z=0..1. (x*y*z)^n * (1-y)^n / (1-(1-x*y)*z)^{(n+1)} =$
 $(LBINT w=0..1. (1-w)^n * (1-y)^n / (1-(1-x*y)*w))$)

proof –

define g **where** $g = (\lambda z. (1 - z) / (1-(1-x*y)*z))$

define g' **where** $g' = (\lambda z. -x*y / (1-(1-x*y)*z)^2)$

have $x * y < 1$

using *mult-strict-mono[of x 1 y 1]* xy **by** *simp*

have less: $(1 - (x * y)) * w < 1$ **and** *neg:* $(1 - (x * y)) * w \neq 1$

if $w: w \in \{0..1\}$ **for** w

proof –

have $(1 - (x * y)) * w \leq (1 - (x * y)) * 1$

using $w \langle x * y < 1 \rangle$ **by** (*intro mult-left-mono*) *auto*

also have $\dots < 1$

using xy **by** *simp*

finally show $(1 - (x * y)) * w < 1$.

thus $(1 - (x * y)) * w \neq 1$ **by** *simp*

qed

have *deriv:* (g has-real-derivative $g' w$) (at w within $\{0..1\}$) **if** $w \in \{0..1\}$ **for** w

unfolding g -def g' -def

apply (*insert that xy neg*)

apply (*rule derivative-eq-intros refl*)+

apply (*simp-all add: divide-simps power2-eq-square*)

apply (*auto simp: algebra-simps*)

done

have *continuous-on* $\{0..1\}$ $(\lambda xa. (1 - xa)^n / (1 - (1 - x * y) * xa))$

using *neg* **by** (*auto intro!: continuous-intros*)

moreover have $g \text{ ' } \{0..1\} \subseteq \{0..1\}$

proof *clarify*

fix $w :: \text{real}$ **assume** $w: w \in \{0..1\}$

have $(1 - x * y) * w \leq 1 * w$

using $\langle x * y < 1 \rangle xy w$ **by** (*intro mult-right-mono*) *auto*

thus $g w \in \{0..1\}$

unfolding g -def **using** *less*[of w] w **by** (*auto simp: divide-simps*)

qed

ultimately have *cont:* *continuous-on* $(g \text{ ' } \{0..1\}) (\lambda xa. (1 - xa)^n / (1 - (1 - x * y) * xa))$

by (*rule continuous-on-subset*)

have *cont'*: *continuous-on* $\{0..1\}$ g'

using *neg* **by** (*auto simp: g'-def intro!: continuous-intros*)

have $(LBINT\ w=0..1.\ (1-w)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*w)) =$
 $(1-y)^{\wedge}n * (LBINT\ w=0..1.\ (1-w)^{\wedge}n / (1-(1-x*y)*w))$
unfolding *interval-lebesgue-integral-mult-right [symmetric]*
by *(simp add: algebra-simps power-mult-distrib)*
also have $(LBINT\ w=0..1.\ (1-w)^{\wedge}n / (1-(1-x*y)*w)) =$
 $-(LBINT\ w=g\ 0..g\ 1.\ (1-w)^{\wedge}n / (1-(1-x*y)*w))$
by *(subst interval-integral-endpoints-reverse)(simp add: g-def zero-ereal-def one-ereal-def)*
also have $(LBINT\ w=g\ 0..g\ 1.\ (1-w)^{\wedge}n / (1-(1-x*y)*w)) =$
 $(LBINT\ w=0..1.\ g' * w * ((1-g*w)^{\wedge}n / (1-(1-x*y)*g*w)))$
using *deriv cont cont'*
by *(subst interval-integral-substitution-finite [symmetric, where g = g and g' = g'])*
(simp-all add: zero-ereal-def one-ereal-def)
also have $-\dots = (LBINT\ z=0..1.\ ((x*y)^{\wedge}n * z^{\wedge}n / (1-(1-x*y)*z)^{\wedge}(n+1)))$
unfolding *interval-lebesgue-integral-uminus [symmetric]* **using** *xy*
apply *(intro interval-lebesgue-integral-lborel-01-cong)*
apply *(simp add: divide-simps g-def g'-def)*
apply *(auto simp: algebra-simps power-mult-distrib power2-eq-square)*
done
also have $(1-y)^{\wedge}n * \dots = (LBINT\ z=0..1.\ (x*y*z)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*z)^{\wedge}(n+1))$
unfolding *interval-lebesgue-integral-mult-right [symmetric]*
by *(simp add: algebra-simps power-mult-distrib)*
finally show $\dots = (LBINT\ w=0..1.\ (1-w)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*w))$
..
qed

Lastly, we apply the same integration by parts to x :

lemma *beukers-aux-integral-transform3:*

assumes $w: w \in \{0 < .. < 1\}$

shows $(LBINT\ (x,y):D.\ P\ x * (1-y)^{\wedge}n / (1-(1-x*y)*w)) =$

$(LBINT\ (x,y):D.\ (1-y)^{\wedge}n * (x*y*w)^{\wedge}n * (1-x)^{\wedge}n / (1-(1-x*y)*w)^{\wedge}(n+1))$

proof $-$

have *cbox: cbox (0, 0) (1, 1) = ({0..1} × {0..1}) :: (real × real) set*

by *(auto simp: cbox-def Basis-prod-def inner-prod-def)*

have *box: box (0, 0) (1, 1) = ({0 < .. < 1} × {0 < .. < 1}) :: (real × real) set*

by *(auto simp: box-def Basis-prod-def inner-prod-def)*

have *set-integrable lborel*

*(cbox (0,0) (1,1)) (λ(x,y). P x * (1-y)^n / (1-(1-x*y)*w))*

unfolding *lborel-prod case-prod-unfold P-def*

proof *(intro continuous-on-imp-set-integrable-cbox continuous-intros ballI)*

fix $p :: \text{real} \times \text{real}$ **assume** $p: p \in \text{cbox } (0,0) (1,1)$

have $(1 - \text{fst } p * \text{snd } p) * w \leq 1 * w$

using p *cbox w* **by** *(intro mult-right-mono) auto*

also have $\dots < 1$ **using** w **by** *simp*

finally have $(1 - \text{fst } p * \text{snd } p) * w < 1$ **by** *simp*

thus $1 - (1 - \text{fst } p * \text{snd } p) * w \neq 0$ **by** *simp*

qed

hence *integrable*: *set-integrable lborel* D $(\lambda(x,y). P x * (1-y)^{\wedge}n / (1-(1-x*y)*w))$
by (*rule set-integrable-subset*) (*auto simp: D-def simp flip: borel-prod*)

have *set-integrable lborel* (*cbox* $(0,0)$ $(1,1)$)
 $(\lambda(x,y). (1-y)^{\wedge}n * (x*y*w)^{\wedge}n * (1-x)^{\wedge}n / (1-(1-x*y)*w)^{\wedge}(n+1))$
unfolding *lborel-prod case-prod-unfold P-def*

proof (*intro continuous-on-imp-set-integrable-cbox continuous-intros ballI*)
fix $p :: \text{real} \times \text{real}$ **assume** $p: p \in \text{cbox } (0,0) (1,1)$
have $(1 - \text{fst } p * \text{snd } p) * w \leq 1 * w$
using p *cbox w* **by** (*intro mult-right-mono*) *auto*
also have $\dots < 1$ **using** w **by** *simp*
finally have $(1 - \text{fst } p * \text{snd } p) * w < 1$ **by** *simp*
thus $(1 - (1 - \text{fst } p * \text{snd } p) * w)^{\wedge}(n+1) \neq 0$ **by** *simp*

qed

hence *integrable'*: *set-integrable lborel* D
 $(\lambda(x,y). (1-y)^{\wedge}n * (x*y*w)^{\wedge}n * (1-x)^{\wedge}n / (1-(1-x*y)*w)^{\wedge}(n+1))$
by (*rule set-integrable-subset*) (*auto simp: D-def simp flip: borel-prod*)

have $(\text{LBINT } (x,y):D. P x * (1-y)^{\wedge}n / (1-(1-x*y)*w)) =$
 $(\text{LBINT } y=0..1. (\text{LBINT } x=0..1. P x * (1-y)^{\wedge}n / (1-(1-x*y)*w)))$
using *integrable unfolding case-prod-unfold D-def lborel-prod [symmetric]*
by (*subst lborel-pair.set-integral-snd*) (*auto simp: interval-integral-Ioo*)

also have $\dots = (\text{LBINT } y=0..1. (1-y)^{\wedge}n * (\text{LBINT } x=0..1. P x / (1-(1-y*x)*w)))$
by (*subst interval-lebesgue-integral-mult-right [symmetric]*) (*auto simp: mult-ac*)

also have $\dots = (\text{LBINT } y=0..1. (1-y)^{\wedge}n * (\text{LBINT } x=0..1. (x*y*w)^{\wedge}n * (1-x)^{\wedge}n / (1-(1-x*y)*w)^{\wedge}(n+1)))$
using w **by** (*intro interval-lebesgue-integral-lborel-01-cong, subst beukers-aux-by-parts*)
(auto simp: mult-ac)

also have $\dots = (\text{LBINT } y=0..1. (\text{LBINT } x=0..1. (1-y)^{\wedge}n * (x*y*w)^{\wedge}n * (1-x)^{\wedge}n / (1-(1-x*y)*w)^{\wedge}(n+1)))$
by (*subst interval-lebesgue-integral-mult-right [symmetric]*) (*auto simp: mult-ac*)

also have $\dots = (\text{LBINT } (x,y):D. (1-y)^{\wedge}n * (x*y*w)^{\wedge}n * (1-x)^{\wedge}n / (1-(1-x*y)*w)^{\wedge}(n+1))$
using *integrable' unfolding case-prod-unfold D-def lborel-prod [symmetric]*
by (*subst lborel-pair.set-integral-snd*) (*auto simp: interval-integral-Ioo*)

finally show *?thesis* .

qed

We need to show the existence of some of these triple integrals.

lemma *beukers-aux-integrable1*:

set-integrable lborel $((\{0 <..< 1\} \times \{0 <..< 1\}) \times \{0 <..< 1\})$
 $(\lambda((x,y),z). P x * P y / (1-(1-x*y)*z))$

proof –

have D [*measurable*]: $D \in \text{sets } (\text{borel} \otimes_M \text{borel})$
unfolding D -*def* **by** (*simp flip: borel-prod*)

have *bounded* $(P \text{ ‘ } \{0..1\})$
unfolding P -*def* **by** (*intro compact-imp-bounded compact-continuous-image continuous-intros*) *auto*

then obtain C **where** $C: \bigwedge x. x \in \{0..1\} \implies \text{norm } (P x) \leq C$
unfolding *bounded-iff* **by** *fast*

```

show ?thesis unfolding D'-def case-prod-unfold
proof (subst lborel-prod [symmetric],
  intro lborel-pair.Fubini-set-integrable AE-I2 impI; clarsimp?)
  fix x y :: real
  assume xy: x > 0 x < 1 y > 0 y < 1
  have x * y < 1 using xy mult-strict-mono[of x 1 y 1] by simp
  show set-integrable lborel {0<..by (rule set-integrable-subset[of - {0..1}], rule borel-integrable-atLeastAtMost')
      (use ⟨x*y<1⟩ beukers-denom-neq[of x y] xy in ⟨auto intro!: continuous-intros
simp: P-def⟩)
  next
  have set-integrable lborel D
    (λ(x,y). (∫ z∈{0<..proof (rule set-integrable-bound[OF - - AE-I2]; clarify?)
    show set-integrable lborel D (λ(x,y). C2 * (-ln (x*y) / (1 - x*y)))
      using beukers-integral1-integrable[of 0 0]
      unfolding case-prod-unfold by (intro set-integrable-mult-right) (auto simp:
D-def)
  next
  fix x y assume xy: (x, y) ∈ D
  have norm (LBINT z:{0<..proof (intro arg-cong[where f = norm] set-lebesgue-integral-cong allI impI,
goal-cases)
    case (2 z)
    with beukers-denom-ineq[of x y z] xy show ?case
      by (auto simp: abs-mult D-def)
    qed (auto simp: abs-mult D-def)
  also have ... = norm (|P x| * |P y| * (LBINT z=0..1. (1 / (1-(1-x*y)*z))))
    by (subst set-integral-mult-right) (auto simp: interval-integral-Ioo)
  also have ... = norm (norm (P x) * norm (P y) * (- ln (x * y) / (1 - x
* y)))
    using beukers-aux-ln-conv-integral[of x y] xy by (simp add: D-def)
  also have ... = norm (P x) * norm (P y) * (- ln (x * y) / (1 - x * y))
    using xy mult-strict-mono[of x 1 y 1]
    by (auto simp: D-def divide-nonpos-nonneg abs-mult)
  also have norm (P x) * norm (P y) * (- ln (x * y) / (1 - x * y)) ≤
    norm (C * C * (- ln (x * y) / (1 - x * y)))
    using xy C[of x] C[of y] mult-strict-mono[of x 1 y 1] unfolding norm-mult
norm-divide
    by (intro mult-mono C) (auto simp: D-def divide-nonpos-nonneg)
  finally show norm (LBINT z:{0<..2 * (- ln (x * y) / (1 - x * y)))
    by (simp add: power2-eq-square mult-ac)
  next
  show set-borel-measurable lborel D (λ(x, y).
    LBINT z:{0<..unfolding lborel-prod [symmetric] set-borel-measurable-def
    case-prod-unfold set-lebesgue-integral-def P-def

```



```

    by measurable
  qed
  thus set-integrable lborel ( $\{0 < .. < 1\} \times \{0 < .. < 1\}$ )
    ( $\lambda x. \text{LBINT } y: \{0 < .. < 1\}. |P (fst x) * P (snd x)| / |1 - (1 - fst x * snd
x) * y|$ )
    by (simp add: case-prod-unfold D-def)
  qed (auto simp: case-prod-unfold lborel-prod [symmetric] set-borel-measurable-def
P-def)
qed

lemma beukers-aux-integrable2:
  set-integrable lborel  $D' (\lambda(z,x,y). P x * (x*y*z)^{\hat{n}} * (1-y)^{\hat{n}} / (1-(1-x*y)*z)^{\hat{n}})$ 
  ^  $(n+1)$ )
proof -
  have [measurable]:  $P \in \text{borel-measurable borel unfolding } P\text{-def}$ 
    by (intro borel-measurable-continuous-onI continuous-intros)
  have bounded ( $P ' \{0..1\}$ )
    unfolding P-def by (intro compact-imp-bounded compact-continuous-image
continuous-intros) auto
  then obtain C where  $C: \bigwedge x. x \in \{0..1\} \implies \text{norm } (P x) \leq C$ 
    unfolding bounded-iff by fast
  show ?thesis unfolding D'-def
  proof (rule set-integrable-bound[OF - - AE-I2]; clarify?)
    show set-integrable lborel ( $\{0 < .. < 1\} \times \{0 < .. < 1\} \times \{0 < .. < 1\}$ )
      ( $\lambda(z,x,y). C * (1 / (1-(1-x*y)*z))$ )
      unfolding case-prod-unfold
      using beukers-integral1-integrable'[of 0 0]
      by (intro set-integrable-mult-right) (auto simp: lborel-prod case-prod-unfold)
    next
    fix x y z :: real assume xyz:  $x \in \{0 < .. < 1\} y \in \{0 < .. < 1\} z \in \{0 < .. < 1\}$ 
    have norm ( $P x * (x*y*z)^{\hat{n}} * (1-y)^{\hat{n}} / (1-(1-x*y)*z)^{\hat{n}}$ ) =
      norm ( $P x * (1-y)^{\hat{n}} * ((x*y*z) / (1-(1-x*y)*z))^{\hat{n}} / (1-(1-x*y)*z)^{\hat{n}}$ )
      using xyz beukers-denom-ineq[of x y z] by (simp add: abs-mult power-divide
mult-ac)
    also have  $(x*y*z) / (1-(1-x*y)*z) = 1/((1-z)/(z*x*y)+1)$ 
      using xyz by (simp add: field-simps)
    also have norm ( $P x * (1-y)^{\hat{n}} * \dots^{\hat{n}} / (1-(1-x*y)*z) \leq$ 
       $C * 1^{\hat{n}} * 1^{\hat{n}} / (1-(1-x*y)*z)$ )
      using xyz C[of x] beukers-denom-ineq[of x y z]
    by (intro mult-mono divide-right-mono power-mono zero-le-power mult-nonneg-nonneg
divide-nonneg-nonneg)
      (auto simp: field-simps)
    also have  $\dots \leq |C * 1^{\hat{n}} * 1^{\hat{n}} / (1-(1-x*y)*z)|$ 
      by linarith
    finally show norm ( $P x * (x*y*z)^{\hat{n}} * (1-y)^{\hat{n}} / (1-(1-x*y)*z)^{\hat{n}}$ )  $\leq$ 
      norm (case (z,x,y) of (z,x,y)  $\implies C * (1 / (1-(1-x*y)*z))$ )
      by (simp add: case-prod-unfold)
  qed (simp-all add: case-prod-unfold set-borel-measurable-def flip: borel-prod)
qed

```

lemma *beukers-aux-integrable3*:
set-integrable lborel $D' (\lambda(w,x,y). P x * (1-w)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*w))$
proof –
have [*measurable*]: $P \in \text{borel-measurable borel}$ **unfolding** $P\text{-def}$
by (*intro borel-measurable-continuous-onI continuous-intros*)
have *bounded* ($P \text{ ' } \{0..1\}$)
unfolding $P\text{-def}$ **by** (*intro compact-imp-bounded compact-continuous-image continuous-intros*) *auto*
then obtain C **where** $C: \bigwedge x. x \in \{0..1\} \implies \text{norm } (P x) \leq C$
unfolding *bounded-iff* **by** *fast*
show *?thesis* **unfolding** $D'\text{-def}$
proof (*rule set-integrable-bound[OF - - AE-I2]; clarify?*)
show *set-integrable lborel* ($\{0<..)
 $(\lambda(z,x,y). C * (1 / (1-(1-x*y)*z)))$)
unfolding *case-prod-unfold*
using *beukers-integral1-integrable'[of 0 0]*
by (*intro set-integrable-mult-right*) (*auto simp: lborel-prod case-prod-unfold*)
next
fix $x y w :: \text{real}$ **assume** $xyw: x \in \{0<..
have $\text{norm } (P x * (1-w)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*w)) =$
 $\text{norm } (P x) * (1-w)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*w)$
using xyw *beukers-denom-ineq[of x y w]* **by** (*simp add: abs-mult power-divide mult-ac*)
also have $\dots \leq C * 1^{\wedge}n * 1^{\wedge}n / (1-(1-x*y)*w)$
using xyw C [*of x*] *beukers-denom-ineq[of x y w]*
by (*intro mult-mono divide-right-mono power-mono zero-le-power mult-nonneg-nonneg divide-nonneg-nonneg*)
(auto simp: field-simps)
also have $\dots \leq |C * 1^{\wedge}n * 1^{\wedge}n / (1-(1-x*y)*w)|$
by *linarith*
finally show $\text{norm } (P x * (1-w)^{\wedge}n * (1-y)^{\wedge}n / (1-(1-x*y)*w)) \leq$
 $\text{norm } (\text{case } (w,x,y) \text{ of } (z,x,y) \Rightarrow C * (1 / (1-(1-x*y)*z)))$
by (*simp add: case-prod-unfold*)
qed (*simp-all add: case-prod-unfold set-borel-measurable-def flip: borel-prod*)
qed$$

Now we only need to put all of these results together:

lemma *beukers-integral2-conv-3*: *beukers-integral2 = beukers-integral3*

proof –

have *cont-P*: *continuous-on* $\{0..1\}$ P
by (*auto simp: P-def intro: continuous-intros*)
have D [*measurable*]: $D \in \text{sets borel } D \in \text{sets } (\text{borel } \otimes_M \text{ borel})$
unfolding $D\text{-def}$ **by** (*simp-all flip: borel-prod*)
have [*measurable*]: $P \in \text{borel-measurable borel}$ **unfolding** $P\text{-def}$
by (*intro borel-measurable-continuous-onI continuous-intros*)

have *beukers-integral2 = (LBINT (x,y):D. P x * P y * (LBINT z=0..1. 1 / (1-(1-x*y)*z)))*

unfolding *beukers-integral2-def case-prod-unfold*
by (*intro set-lebesgue-integral-cong allI impI, measurable*)
(subst beukers-aux-ln-conv-integral, auto simp: D-def)
also have ... = $(\text{LBINT } (x,y):D. (\text{LBINT } z=0..1. P x * P y / (1-(1-x*y)*z)))$
by (*subst interval-lebesgue-integral-mult-right [symmetric] auto*)
also have ... = $(\text{LBINT } (x,y):D. (\text{LBINT } z:\{0<..
by (*simp add: interval-integral-Ioo*)
also have ... = $(\text{LBINT } z:\{0<..
proof (*subst lborel-pair.Fubini-set-integral [symmetric]*)
have *set-integrable lborel* ($(\{0<..)
 $(\lambda((x, y), z). P x * P y / (1 - (1 - x * y) * z))$)
using *beukers-aux-integrable1 by simp*
also have *?this* \longleftrightarrow *set-integrable* (*lborel* \otimes_M *lborel*) ($\{0<..)
 $(\lambda(z,x,y). P x * P y / (1 - (1 - x * y) * z))$)
unfolding *set-integrable-def*
by (*subst lborel-pair.integrable-product-swap-iff [symmetric], intro integrable-cong*)
(auto simp: indicator-def case-prod-unfold lborel-prod D-def)
finally show
qed (*auto simp: case-prod-unfold*)
also have ... = $(\text{LBINT } z:\{0<..
by (*rule set-lebesgue-integral-cong*) (*use beukers-aux-integral-transform1 in simp-all*)
also have ... = $(\text{LBINT } (x,y):D. (\text{LBINT } z:\{0<..
using *beukers-aux-integrable2*
by (*subst lborel-pair.Fubini-set-integral [symmetric]*)
(auto simp: case-prod-unfold lborel-prod D-def D'-def)
also have ... = $(\text{LBINT } (x,y):D. (\text{LBINT } w:\{0<..
proof (*intro set-lebesgue-integral-cong allI impI; clarify?*)
fix $x y :: \text{real}$
assume $xy: (x, y) \in D$
have $(\text{LBINT } z:\{0<..
 $=$
 $P x * (\text{LBINT } z=0..1. (x*y*z)^n * (1-y)^n / (1-(1-x*y)*z)^{(n+1)})$
by (*subst interval-lebesgue-integral-mult-right [symmetric]*)
(simp add: mult-ac interval-integral-Ioo)
also have ... = $P x * (\text{LBINT } w=0..1. (1-w)^n * (1-y)^n / (1-(1-x*y)*w))$
using xy **by** (*subst beukers-aux-integral-transform2*) (*auto simp: D-def*)
also have ... = $(\text{LBINT } w:\{0<..
by (*subst interval-lebesgue-integral-mult-right [symmetric]*)
(simp add: mult-ac interval-integral-Ioo)
finally show $(\text{LBINT } z:\{0<..
 $=$
 $(\text{LBINT } w:\{0<..
.$$$$$$$$$$$

qed (*auto simp: D-def simp flip: borel-prod*)
also have ... = $(\text{LBINT } w:\{0<..$

```

(1-y) ^ n / (1-(1-x*y)*w))
  using beukers-aux-integrable3
  by (subst lborel-pair.Fubini-set-integral [symmetric])
    (auto simp: case-prod-unfold lborel-prod D-def D'-def)
  also have ... = (LBINT w=0..1. (1-w) ^ n * (LBINT (x,y):D. P x * (1-y) ^ n
/ (1-(1-x*y)*w)))
  unfolding case-prod-unfold
  by (subst set-integral-mult-right [symmetric]) (simp add: mult-ac interval-integral-Ioo)
  also have ... = (LBINT w=0..1. (1-w) ^ n * (LBINT (x,y):D. (x*y*w*(1-x)*(1-y)) ^ n
/ (1-(1-x*y)*w) ^ (n+1)))
  by (intro interval-lebesgue-integral-lborel-01-cong, subst beukers-aux-integral-transform3)
    (auto simp: mult-ac power-mult-distrib)
  also have ... = (LBINT w=0..1. (LBINT (x,y):D. (x*y*w*(1-x)*(1-y)*(1-w)) ^ n
/ (1-(1-x*y)*w) ^ (n+1)))
  by (subst set-integral-mult-right [symmetric])
    (auto simp: case-prod-unfold mult-ac power-mult-distrib)
  also have ... = beukers-integral3
  using beukers-integral3-integrable unfolding D'-def D-def beukers-integral3-def
  by (subst (2) lborel-prod [symmetric], subst lborel-pair.set-integral-fst')
    (auto simp: case-prod-unfold interval-integral-Ioo lborel-prod algebra-simps)
  finally show ?thesis .
qed

```

1.10 The main result

Combining all of the results so far, we can derive the key inequalities

$$0 < A\zeta(3) + B < 2\zeta(3) \cdot 27^{-n} \cdot \text{lcm}\{1 \dots n\}^3$$

for integers A, B with $A > 0$.

lemma *zeta-3-linear-combination-bounds*:

obtains $A B :: \text{int}$

where $A > 0$

$$A * \text{Re} (\text{zeta } 3) + B \in \{0 <.. 2 * \text{Re} (\text{zeta } 3) * \text{Lcm} \{1..n\} ^ 3 / 27 ^ n\}$$

proof –

define I **where** $I = \text{beukers-integral2}$

define d **where** $d = \text{Lcm} \{1..n\} ^ 3$

have $d > 0$ **by** (auto simp: d-def intro!: Nat.gr0I)

from *beukers-integral2-conv-int-combination* **obtain** $A' B :: \text{int}$

where $*$: $A' > 0$ $I = A' * \text{Re} (\text{zeta } 3) + B / d$ **unfolding** I -def d -def .

define A **where** $A = A' * d$

from $*$ **have** A : $A > 0$ $I = (A * \text{Re} (\text{zeta } 3) + B) / d$

using $\langle d > 0 \rangle$ **by** (simp-all add: A -def field-simps)

have $0 < I$

using *beukers-integral3-pos* **by** (simp add: I -def *beukers-integral2-conv-3*)

with $\langle d > 0 \rangle$ **have** $A * \text{Re} (\text{zeta } 3) + B > 0$

by (simp add: field-simps A)

moreover have $I \leq 2 * (1 / 27) ^ n * Re (zeta 3)$
using *beukers-integral2-conv-3 beukers-integral3-le* **by** (*simp add: I-def*)
hence $A * Re (zeta 3) + B \leq 2 * Re (zeta 3) * d / 27 ^ n$
using $\langle d > 0 \rangle$ **by** (*simp add: A field-simps*)

ultimately show *?thesis*
using *A* **by** (*intro that[of A B]*) (*auto simp: d-def*)

qed

If $\zeta(3) = \frac{a}{b}$ for some integers a and b , we can thus derive the inequality $2b\zeta(3) \cdot 27^{-n} \cdot \text{lcm}\{1 \dots n\}^3 \geq 1$ for any natural number n .

lemma *beukers-key-inequality*:

fixes $a :: int$ **and** $b :: nat$
assumes $b > 0$ **and** $ab: Re (zeta 3) = a / b$
shows $2 * b * Re (zeta 3) * Lcm \{1..n\} ^ 3 / 27 ^ n \geq 1$

proof –

from *zeta-3-linear-combination-bounds* **obtain** $A B :: int$

where $AB: A > 0$

$A * Re (zeta 3) + B \in \{0 <.. 2 * Re (zeta 3) * Lcm \{1..n\} ^ 3 / 27 ^ n\}$.

from AB **have** $0 < (A * Re (zeta 3) + B) * b$

using $\langle b > 0 \rangle$ **by** (*intro mult-pos-pos*) *auto*

also have $\dots = A * (Re (zeta 3) * b) + B * b$

using $\langle b > 0 \rangle$ **by** (*simp add: algebra-simps*)

also have $Re (zeta 3) * b = a$

using $\langle b > 0 \rangle$ **by** (*simp add: ab*)

also have $of-int A * of-int a + of-int (B * b) = of-int (A * a + B * b)$

by *simp*

finally have $1 \leq A * a + B * b$

by *linarith*

hence $1 \leq real-of-int (A * a + B * b)$

by *linarith*

also have $\dots = (A * (a / b) + B) * b$

using $\langle b > 0 \rangle$ **by** (*simp add: ring-distrib*)

also have $a / b = Re (zeta 3)$

by (*simp add: ab*)

also have $A * Re (zeta 3) + B \leq 2 * Re (zeta 3) * Lcm \{1..n\} ^ 3 / 27 ^ n$

using AB **by** *simp*

finally show $2 * b * Re (zeta 3) * Lcm \{1..n\} ^ 3 / 27 ^ n \geq 1$

using $\langle b > 0 \rangle$ **by** (*simp add: mult-ac*)

qed

end

lemma *smallo-power*: $n > 0 \implies f \in o[F](g) \implies (\lambda x. f x ^ n) \in o[F](\lambda x. g x ^ n)$

by (*induction n rule: nat-induct-non-zero*) (*auto intro: landau-o.small.mult*)

This is now a contradiction, since $\text{lcm}\{1 \dots n\} \in o(3^n)$ by the Prime Number Theorem – hence the main result.

theorem *zeta-3-irrational*: $\text{zeta } 3 \notin \mathbb{Q}$

proof

assume $\text{zeta } 3 \in \mathbb{Q}$

obtain $a :: \text{int}$ **and** $b :: \text{nat}$ **where** $b > 0$ **and** ab' : $\text{zeta } 3 = a / b$

proof –

from $\langle \text{zeta } 3 \in \mathbb{Q} \rangle$ **obtain** r **where** r : $\text{zeta } 3 = \text{of-rat } r$

by *(elim Rats-cases)*

also have $r = \text{rat-of-int } (\text{fst } (\text{quotient-of } r)) / \text{rat-of-int } (\text{snd } (\text{quotient-of } r))$

by *(intro quotient-of-div) auto*

also have $\text{of-rat } \dots = (\text{of-int } (\text{fst } (\text{quotient-of } r)) / \text{of-int } (\text{snd } (\text{quotient-of } r))) :: \text{complex}$

by *(simp add: of-rat-divide)*

also have $\text{of-int } (\text{snd } (\text{quotient-of } r)) = \text{of-nat } (\text{nat } (\text{snd } (\text{quotient-of } r)))$

using *quotient-of-denom-pos'[of r]* **by** *auto*

finally have $\text{zeta } 3 = \text{of-int } (\text{fst } (\text{quotient-of } r)) / \text{of-nat } (\text{nat } (\text{snd } (\text{quotient-of } r)))$.

thus *?thesis*

using *quotient-of-denom-pos'[of r]*

by *(intro that[of nat (snd (quotient-of r)) fst (quotient-of r)]) auto*

qed

hence ab : $\text{Re } (\text{zeta } 3) = a / b$ **by** *simp*

interpret *prime-number-theorem*

by *standard (rule prime-number-theorem)*

have *Lcm-upto-smallo*: $(\lambda n. \text{real } (\text{Lcm } \{1..n\})) \in o(\lambda n. c \wedge n)$ **if** c : $c > \exp 1$ **for** c

proof –

have $0 < \exp (1 :: \text{real})$ **by** *simp*

also note c

finally have $c > 0$.

have $(\lambda n. \text{real } (\text{Lcm } \{1..n\})) = (\lambda n. \text{real } (\text{Lcm } \{1.. \text{nat } \lfloor \text{real } n \rfloor\}))$

by *simp*

also have $\dots \in o(\lambda n. c \text{ powr } \text{real } n)$

using *Lcm-upto.smallo'*

by *(rule landau-o.small.compose) (simp-all add: c filterlim-real-sequentially)*

also have $(\lambda n. c \text{ powr } \text{real } n) = (\lambda n. c \wedge n)$

using $c \langle c > 0 \rangle$ **by** *(subst powr-realpow) auto*

finally show *?thesis* .

qed

have $(\lambda n. 2 * b * \text{Re } (\text{zeta } 3) * \text{real } (\text{Lcm } \{1..n\}) \wedge 3 / 27 \wedge n) \in$

$O(\lambda n. \text{real } (\text{Lcm } \{1..n\}) \wedge 3 / 27 \wedge n)$

using $\langle b > 0 \rangle$ *Re-zeta-ge-1[of 3]* **by** *simp*

also have $\exp 1 < (3 :: \text{real})$

using *e-approx-32* **by** *(simp add: abs-if split: if-splits)*

hence $(\lambda n. \text{real } (\text{Lcm } \{1..n\}) \wedge 3 / 27 \wedge n) \in o(\lambda n. (3 \wedge n) \wedge 3 / 27 \wedge n)$

unfolding *of-nat-power*
by (*intro landau-o.small.divide-right smallo-power Lcm-upto-smallo*) *auto*
also have $(\lambda n. (3 \wedge n) \wedge 3 / 27 \wedge n :: \text{real}) = (\lambda -. 1)$
by (*simp add: power-mult [of 3, symmetric] mult.commute [of - 3] power-mult [of - 3]*)
finally have $*$: $(\lambda n. 2 * b * \text{Re} (\text{zeta } 3) * \text{real} (\text{Lcm} \{1..n\}) \wedge 3 / 27 \wedge n) \in o(\lambda -. 1)$.
have *lim*: $(\lambda n. 2 * b * \text{Re} (\text{zeta } 3) * \text{real} (\text{Lcm} \{1..n\}) \wedge 3 / 27 \wedge n) \longrightarrow 0$
using *smalloD-tendsto [OF *]* **by** *simp*

moreover have $1 \leq \text{real} (2 * b) * \text{Re} (\text{zeta } 3) * \text{real} (\text{Lcm} \{1..n\}) \wedge 3 / 27 \wedge n$
for n
using *beukers-key-inequality [of b a] ab < b > 0* **by** *simp*

ultimately have $1 \leq (0 :: \text{real})$
by (*intro tendsto-lowerbound [OF lim] always-eventually allI*) *auto*
thus *False* **by** *simp*
qed

end

References

- [1] R. Apéry. Irrationalité de $\zeta(2)$ et $\zeta(3)$. In *Journées Arithmétiques de Luminy*, number 61 in *Astérisque*, pages 11–13. Société mathématique de France, 1979.
- [2] F. Beukers. A note on the irrationality of $\zeta(2)$ and $\zeta(3)$. *Bulletin of the London Mathematical Society*, 11(3):268–272, 1979.
- [3] M. Eberl. Elementary facts about the distribution of primes. *Archive of Formal Proofs*, Feb. 2019. http://isa-afp.org/entries/Prime_Distribution_Elementary.html, Formal proof development.
- [4] M. Eberl and L. C. Paulson. The prime number theorem. *Archive of Formal Proofs*, Sept. 2018. http://isa-afp.org/entries/Prime_Number_Theorem.html, Formal proof development.
- [5] M. Filaseta. Math 785: Transcendental number theory (lecture notes, part 4), 2011.
- [6] A. Mahboubi and T. Sibut-Pinote. A formal proof of the irrationality of $\zeta(3)$, 2019.