The Transcendence of e

Manuel Eberl

May 21, 2019

Abstract

This work contains a formalisation of the proof that Euler's number e is transcendental. The proof follows the standard approach of assuming that e is algebraic and then using a specific integer polynomial to derive two inconsistent bounds, leading to a contradiction.

This approach can be found in many different sources; this formalisation mostly follows a PlanetMath article [1] by Roger Lipsett.

Contents

1 Proof of the Transcendence of e

theory *E-Transcendental*
imports
HOL−Analysis.Analysis
HOL−Number-Theory.Number-Theory
HOL−Computational-Algebra.Polynomial
begin

1.1 Various auxiliary facts

lemma *fact-dvd-pochhammer*:
assumes $m \leq n + 1$
shows *fact m dvd pochhammer* $(int\ n - int\ m + 1)\ m$
⟨*proof*⟩

lemma *of-nat-eq-1-iff* [*simp*]: *of-nat* $x = (1 :: 'a :: semiring\text{-}char\text{-}0) \longleftrightarrow x = 1$
⟨*proof*⟩

lemma *prime-elem-int-not-dvd-neg1-power*:

prime-elem (*p* :: *int*) $\Longrightarrow$ ¬*p* *dvd* (−*1*) ˆ *n*
⟨*proof*⟩

lemma *nat-fact* [*simp*]: *nat* (*fact n*) = *fact n*
⟨*proof*⟩

lemma *prime-dvd-fact-iff-int*:
p dvd fact n ⟷ *p* ≤ *int n* **if** *prime p*
⟨*proof*⟩

lemma *filterlim-minus-nat-at-top*:
filterlim (λ*n*. *n* − *k* :: *nat*) *at-top at-top*
⟨*proof*⟩

lemma *power-over-fact-tendsto-0*:
(λ*n*. (*x* :: *real*) ˆ *n* / *fact n*) ⟶ *0*
⟨*proof*⟩

lemma *power-over-fact-tendsto-0'*:
(λ*n*. *c* ∗ (*x* :: *real*) ˆ *n* / *fact n*) ⟶ *0*
⟨*proof*⟩

1.2 Lifting integer polynomials

lift-definition *of-int-poly* :: *int poly* ⇒ *'a* :: *comm-ring-1 poly* **is** λ*g x*. *of-int* (*g x*)
⟨*proof*⟩

lemma *coeff-of-int-poly* [*simp*]: *coeff* (*of-int-poly p*) *n* = *of-int* (*coeff p n*)
⟨*proof*⟩

lemma *of-int-poly-0* [*simp*]: *of-int-poly 0* = *0*
⟨*proof*⟩

lemma *of-int-poly-pCons* [*simp*]: *of-int-poly* (*pCons c p*) = *pCons* (*of-int c*) (*of-int-poly p*)
⟨*proof*⟩

lemma *of-int-poly-smult* [*simp*]: *of-int-poly* (*smult c p*) = *smult* (*of-int c*) (*of-int-poly p*)
⟨*proof*⟩

lemma *of-int-poly-1* [*simp*]: *of-int-poly 1* = *1*
⟨*proof*⟩

lemma *of-int-poly-add* [*simp*]: *of-int-poly* (*p* + *q*) = *of-int-poly p* + *of-int-poly q*
⟨*proof*⟩

lemma *of-int-poly-mult* [*simp*]: *of-int-poly* (*p* ∗ *q*) = (*of-int-poly p* ∗ *of-int-poly q*)

⟨*proof*⟩

lemma *of-int-poly-sum* [*simp*]: *of-int-poly* (*sum f A*) = *sum* (λ*x*. *of-int-poly* (*f x*)) *A*
⟨*proof*⟩

lemma *of-int-poly-prod* [*simp*]: *of-int-poly* (*prod f A*) = *prod* (λ*x*. *of-int-poly* (*f x*)) *A*
⟨*proof*⟩

lemma *of-int-poly-power* [*simp*]: *of-int-poly* (*p* ^ *n*) = *of-int-poly p* ^ *n*
⟨*proof*⟩

lemma *of-int-poly-monom* [*simp*]: *of-int-poly* (*monom c n*) = *monom* (*of-int c*) *n*
⟨*proof*⟩

lemma *poly-of-int-poly* [*simp*]: *poly* (*of-int-poly p*) (*of-int x*) = *of-int* (*poly p x*)
⟨*proof*⟩

lemma *poly-of-int-poly-of-nat* [*simp*]: *poly* (*of-int-poly p*) (*of-nat x*) = *of-int* (*poly p* (*int x*))
⟨*proof*⟩

lemma *poly-of-int-poly-0* [*simp*]: *poly* (*of-int-poly p*) *0* = *of-int* (*poly p 0*)
⟨*proof*⟩

lemma *poly-of-int-poly-1* [*simp*]: *poly* (*of-int-poly p*) *1* = *of-int* (*poly p 1*)
⟨*proof*⟩

lemma *poly-of-int-poly-of-real* [*simp*]:
poly (*of-int-poly p*) (*of-real x*) = *of-real* (*poly* (*of-int-poly p*) *x*)
⟨*proof*⟩

lemma *of-int-poly-eq-iff* [*simp*]:
of-int-poly p = (*of-int-poly q* :: *'a* :: {*comm-ring-1*, *ring-char-0*} *poly*) ⟷ *p* = *q*
⟨*proof*⟩

lemma *of-int-poly-eq-0-iff* [*simp*]:
of-int-poly p = (*0* :: *'a* :: {*comm-ring-1*, *ring-char-0*} *poly*) ⟷ *p* = *0*
⟨*proof*⟩

lemma *degree-of-int-poly* [*simp*]:
degree (*of-int-poly p* :: *'a* :: {*comm-ring-1*, *ring-char-0*} *poly*) = *degree p*
⟨*proof*⟩

lemma *pderiv-of-int-poly* [*simp*]: *pderiv* (*of-int-poly p*) = *of-int-poly* (*pderiv p*)
⟨*proof*⟩

lemma *higher-pderiv-of-int-poly* [*simp*]:
(*pderiv* ^^ *n*) (*of-int-poly p*) = *of-int-poly* ((*pderiv* ^^ *n*) *p*)
⟨*proof*⟩

lemma *int-polyE*:
assumes $\bigwedge n$. *coeff* (*p* :: *'a* :: {*comm-ring-1*, *ring-char-0*} *poly*) *n* ∈ ℤ
obtains *p'* **where** *p* = *of-int-poly p'*
⟨*proof*⟩

1.3 General facts about polynomials

lemma *pderiv-power*:
pderiv (*p* ^ *n*) = *smult* (*of-nat n*) (*p* ^ (*n* − *1*) ∗ *pderiv p*)
⟨*proof*⟩

lemma *degree-prod-sum-eq*:
($\bigwedge x$. *x* ∈ *A* ⟹ *f x* ≠ *0*) ⟹
degree (*prod f A* :: *'a* :: *idom poly*) = ($\sum$ *x*∈*A*. *degree* (*f x*))
⟨*proof*⟩

lemma *pderiv-monom*:
pderiv (*monom c n*) = *monom* (*of-nat n* ∗ *c*) (*n* − *1*)
⟨*proof*⟩

lemma *power-poly-const* [*simp*]: [:*c*:] ^ *n* = [:*c* ^ *n*:]
⟨*proof*⟩

lemma *monom-power*: *monom c n* ^ *k* = *monom* (*c* ^ *k*) (*n* ∗ *k*)
⟨*proof*⟩

lemma *coeff-higher-pderiv*:
coeff ((*pderiv* ^^ *m*) *f*) *n* = *pochhammer* (*of-nat* (*Suc n*)) *m* ∗ *coeff f* (*n* + *m*)
⟨*proof*⟩

lemma *higher-pderiv-add*: (*pderiv* ^^ *n*) (*p* + *q*) = (*pderiv* ^^ *n*) *p* + (*pderiv* ^^ *n*) *q*
⟨*proof*⟩

lemma *higher-pderiv-smult*: (*pderiv* ^^ *n*) (*smult c p*) = *smult c* ((*pderiv* ^^ *n*) *p*)
⟨*proof*⟩

lemma *higher-pderiv-0* [*simp*]: (*pderiv* ^^ *n*) *0* = *0*
⟨*proof*⟩

lemma *higher-pderiv-monom*:
m ≤ *n* + *1* ⟹ (*pderiv* ^^ *m*) (*monom c n*) = *monom* (*pochhammer* (*int n* − *int m* + *1*) *m* ∗ *c*) (*n* − *m*)
⟨*proof*⟩

lemma *higher-pderiv-monom-eq-zero*:
$m > n + 1 \Longrightarrow (pderiv\ \hat{}\hat{}\ m)\ (monom\ c\ n) = 0$
⟨*proof*⟩

lemma *higher-pderiv-sum*: $(pderiv\ \hat{}\hat{}\ n)\ (sum\ f\ A) = (\sum x \in A.\ (pderiv\ \hat{}\hat{}\ n)\ (f\ x))$
⟨*proof*⟩

lemma *fact-dvd-higher-pderiv*:
[:*fact n* :: *int*:] *dvd* (*pderiv* ^^ *n*) *p*
⟨*proof*⟩

lemma *fact-dvd-poly-higher-pderiv-aux*:
(*fact n* :: *int*) *dvd poly* ((*pderiv* ^^ *n*) *p*) *x*
⟨*proof*⟩

lemma *fact-dvd-poly-higher-pderiv-aux'*:
$m \le n \Longrightarrow$ (*fact m* :: *int*) *dvd poly* ((*pderiv* ^^ *n*) *p*) *x*
⟨*proof*⟩

lemma *algebraicE'*:
assumes *algebraic* (*x* :: *'a* :: *field-char-0*)
obtains *p* **where** $p \ne 0$ *poly* (*of-int-poly p*) $x = 0$
⟨*proof*⟩

lemma *algebraicE'-nonzero*:
assumes *algebraic* (*x* :: *'a* :: *field-char-0*) $x \ne 0$
obtains *p* **where** $p \ne 0$ *coeff p* $0 \ne 0$ *poly* (*of-int-poly p*) $x = 0$
⟨*proof*⟩

lemma *algebraic-of-real-iff* [*simp*]:
algebraic (*of-real x* :: *'a* :: {*real-algebra-1*,*field-char-0*}) ⟷ *algebraic x*
⟨*proof*⟩

1.4 Main proof

lemma *lindemann-weierstrass-integral*:
fixes *u* :: *complex* **and** *f* :: *complex poly*
defines $df \equiv \lambda n.\ (pderiv\ \hat{}\hat{}\ n)\ f$
defines $m \equiv degree\ f$
defines $I \equiv \lambda f\ u.\ exp\ u * (\sum j \le degree\ f.\ poly\ ((pderiv\ \hat{}\hat{}\ j)\ f)\ 0) - (\sum j \le degree\ f.\ poly\ ((pderiv\ \hat{}\hat{}\ j)\ f)\ u)$
shows $((\lambda t.\ exp\ (u - t) * poly\ f\ t)$ *has-contour-integral* $I\ f\ u)$ (*linepath* $0\ u$)
⟨*proof*⟩

locale *lindemann-weierstrass-aux* =
fixes *f* :: *complex poly*
begin

definition I :: *complex* ⇒ *complex* **where**
$I\ u = exp\ u * (\sum j{\le}degree\ f.\ poly\ ((pderiv\ \hat{}\hat{}\ j)\ f)\ 0) -$
$(\sum j{\le}degree\ f.\ poly\ ((pderiv\ \hat{}\hat{}\ j)\ f)\ u)$

lemma *lindemann-weierstrass-integral-bound*:
fixes u :: *complex*
assumes $C \ge 0$ $\bigwedge t.\ t \in closed\text{-}segment\ 0\ u \Longrightarrow norm\ (poly\ f\ t) \le C$
shows $norm\ (I\ u) \le norm\ u * exp\ (norm\ u) * C$
⟨*proof*⟩

end

lemma *poly-higher-pderiv-aux1*:
fixes c :: *'a* :: *idom*
assumes $k < n$
shows $poly\ ((pderiv\ \hat{}\hat{}\ k)\ ([:-c,\ 1:]\ \hat{}\ n * p))\ c = 0$
⟨*proof*⟩

lemma *poly-higher-pderiv-aux1'*:
fixes c :: *'a* :: *idom*
assumes $k < n$ $[:-c,\ 1:]\ \hat{}\ n\ dvd\ p$
shows $poly\ ((pderiv\ \hat{}\hat{}\ k)\ p)\ c = 0$
⟨*proof*⟩

lemma *poly-higher-pderiv-aux2*:
fixes c :: *'a* :: {*idom*, *semiring-char-0*}
shows $poly\ ((pderiv\ \hat{}\hat{}\ n)\ ([:-c,\ 1:]\ \hat{}\ n * p))\ c = fact\ n * poly\ p\ c$
⟨*proof*⟩

lemma *poly-higher-pderiv-aux3*:
fixes c :: *'a* :: {*idom*,*semiring-char-0*}
assumes $k \ge n$
shows $\exists q.\ poly\ ((pderiv\ \hat{}\hat{}\ k)\ ([:-c,\ 1:]\ \hat{}\ n * p))\ c = fact\ n * poly\ q\ c$
⟨*proof*⟩

lemma *poly-higher-pderiv-aux3'*:
fixes c :: *'a* :: {*idom*, *semiring-char-0*}
assumes $k \ge n$ $[:-c,\ 1:]\ \hat{}\ n\ dvd\ p$
shows $fact\ n\ dvd\ poly\ ((pderiv\ \hat{}\hat{}\ k)\ p)\ c$
⟨*proof*⟩

lemma *e-transcendental-aux-bound*:
obtains C **where** $C \ge 0$
$\bigwedge x.\ x \in closed\text{-}segment\ 0\ (of\text{-}nat\ n) \Longrightarrow$
$norm\ (\prod k{\in}\{1..n\}.\ (x - of\text{-}nat\ k :: complex)) \le C$
⟨*proof*⟩

theorem *e-transcendental-complex*: $\neg\ algebraic\ (exp\ 1 :: complex)$

⟨*proof*⟩

corollary *e-transcendental-real*: ¬ *algebraic* (*exp 1* :: *real*)
⟨*proof*⟩

end

References

[1] R. Lipsett. Planetmath. http://planetmath.org/proofoflindemannweierstrasstheoremandthateandpiaretranscendental, 2007.