The Euler–MacLaurin summation formula

Manuel Eberl

March 17, 2025

Abstract

The Euler–MacLaurin formula relates the value of a discrete sum $\sum_{i=a}^{b} f(i)$ to that of the integral $\int_a^b f(x)\,\mathrm{d}x$ in terms of the derivatives of f at a and b and a remainder term. Since the remainder term is often very small as b grows, this can be used to compute asymptotic expansions for sums.

This entry contains a proof of this formula for functions from the reals to an arbitrary Banach space. Two variants of the formula are given: the standard textbook version and a variant outlined in *Concrete Mathematics* [3] that is more useful for deriving asymptotic estimates.

As example applications, we use that formula to derive the full asymptotic expansion of the harmonic numbers and the sum of inverse squares.

Contents

1 The Euler–MacLaurin summation formula **2**
- 1.1 Auxiliary facts . . . 2
- 1.2 The remainder terms . . . 3
- 1.3 The conventional version of the Euler–MacLaurin formula . . 8
- 1.4 The “Concrete Mathematics” version of the Euler–MacLaurin formula . . . 9
- 1.5 Bounds on the remainder term . . . 12
- 1.6 Application to harmonic numbers . . . 13
- 1.7 Application to sums of inverse squares . . . 14

2 Connection of Euler–MacLaurin summation to Landau symbols **14**
- 2.1 O-bound for the remainder term . . . 15
- 2.2 Asymptotic expansion of the harmonic numbers . . . 15
- 2.3 Asymptotic expansion of the sum of inverse squares . . . 16

1 The Euler–MacLaurin summation formula

theory *Euler-MacLaurin*
imports
HOL−Complex-Analysis.Complex-Analysis
Bernoulli.Periodic-Bernpoly
Bernoulli.Bernoulli-FPS
begin

1.1 Auxiliary facts

lemma *pbernpoly-of-int* [*simp*]: *pbernpoly n* (*of-int a*) = *bernoulli n*
⟨*proof*⟩

lemma *continuous-on-bernpoly′* [*continuous-intros*]:
assumes *continuous-on A f*
shows *continuous-on A* (λ*x*. *bernpoly n* (*f x*) :: ′*a* :: *real-normed-algebra-1*)
⟨*proof*⟩

lemma *sum-atLeastAtMost-int-last*:
assumes *a* < (*b* :: *int*)
shows *sum f* {*a*..*b*} = *sum f* {*a*..<*b*} + *f b*
⟨*proof*⟩

lemma *sum-atLeastAtMost-int-head*:
assumes *a* < (*b* :: *int*)
shows *sum f* {*a*..*b*} = *f a* + *sum f* {*a*<..*b*}
⟨*proof*⟩

lemma *not-in-nonpos-Reals-imp-add-nonzero*:
assumes $z \notin \mathbb{R}_{\leq 0}$ $x \geq 0$
shows *z* + *of-real x* ≠ *0*
⟨*proof*⟩

lemma *negligible-atLeastAtMostI*: *b* ≤ *a* ⟹ *negligible* {*a*..(*b*::*real*)}
⟨*proof*⟩

lemma *integrable-on-negligible*:
negligible A ⟹ (*f* :: ′*n* :: *euclidean-space* ⇒ ′*a* :: *banach*) *integrable-on A*
⟨*proof*⟩

lemma *Union-atLeastAtMost-real-of-int*:
assumes *a* < *b*
shows (⋃*n*∈{*a*..<*b*}. {*real-of-int n*..*real-of-int* (*n* + *1*)}) = {*real-of-int a*..*real-of-int b*}
⟨*proof*⟩

1.2 The remainder terms

The following describes the remainder term in the classical version of the Euler–MacLaurin formula.

definition *EM-remainder'* :: *nat* ⇒ (*real* ⇒ *'a* :: *banach*) ⇒ *real* ⇒ *real* ⇒ *'a* **where**
EM-remainder' n f a b = ((−*1*) ^ *Suc n* / *fact n*) $*_R$ *integral* {*a..b*} (λ*t*. *pbernpoly n t* $*_R$ *f t*)

Next, we define the remainder term that occurs when one lets the right bound of summation in the Euler–MacLaurin formula tend to infinity.

definition *EM-remainder-converges* :: *nat* ⇒ (*real* ⇒ *'a* :: *banach*) ⇒ *int* ⇒ *bool* **where**
EM-remainder-converges n f a ⟷ (∃ *L*. ((λ*x*. *EM-remainder' n f a* (*of-int x*)) ⟶ *L*) *at-top*)

definition *EM-remainder* :: *nat* ⇒ (*real* ⇒ *'a* :: *banach*) ⇒ *int* ⇒ *'a* **where**
EM-remainder n f a =
(*if EM-remainder-converges n f a then*
Lim at-top (λ*x*. *EM-remainder' n f a* (*of-int x*)) *else 0*)

The following lemmas are fairly obvious – but tedious to prove – properties of the remainder terms.

lemma *EM-remainder-eqI*:
fixes *L*
assumes ((λ*x*. *EM-remainder' n f b* (*of-int x*)) ⟶ *L*) *at-top*
shows *EM-remainder n f b* = *L*
⟨*proof*⟩

lemma *integrable-EM-remainder'-int*:
fixes *a b* :: *int* **and** *f* :: *real* ⇒ *'a* :: *banach*
assumes *continuous-on* {*of-int a..of-int b*} *f*
shows (λ*t*. *pbernpoly n t* $*_R$ *f t*) *integrable-on* {*a..b*}
⟨*proof*⟩

lemma *integrable-EM-remainder'*:
fixes *a b* :: *real* **and** *f* :: *real* ⇒ *'a* :: *banach*
assumes *continuous-on* {*a..b*} *f*
shows (λ*t*. *pbernpoly n t* $*_R$ *f t*) *integrable-on* {*a..b*}
⟨*proof*⟩

lemma *EM-remainder'-bounded-linear*:
assumes *bounded-linear h*
assumes *continuous-on* {*a..b*} *f*
shows *EM-remainder' n* (λ*x*. *h* (*f x*)) *a b* = *h* (*EM-remainder' n f a b*)
⟨*proof*⟩

lemma *EM-remainder-converges-of-real*:

assumes *EM-remainder-converges n f a continuous-on* {*of-int a..*} *f*
shows *EM-remainder-converges n* (*λx. of-real* (*f x*)) *a*
⟨*proof*⟩

lemma *EM-remainder-converges-of-real-iff*:
fixes *f* :: *real* ⇒ *real*
assumes *continuous-on* {*of-int a..*} *f*
shows *EM-remainder-converges n* (*λx. of-real* (*f x*) ::
′*a* :: {*banach*, *real-normed-algebra-1*, *real-inner*}) *a* ⟷ *EM-remainder-converges n f a*
⟨*proof*⟩

lemma *EM-remainder-of-real*:
assumes *continuous-on* {*a..*} *f*
shows *EM-remainder n* (*λx. of-real* (*f x*) ::
′*a* :: {*banach*, *real-normed-algebra-1*, *real-inner*}) *a* =
of-real (*EM-remainder n f a*)
⟨*proof*⟩

lemma *EM-remainder′-cong*:
assumes ⋀*x. x* ∈ {*a..b*} ⟹ *f x* = *g x n* = *n′ a* = *a′ b* = *b′*
shows *EM-remainder′ n f a b* = *EM-remainder′ n′ g a′ b′*
⟨*proof*⟩

lemma *EM-remainder-converges-cong*:
assumes ⋀*x. x* ≥ *of-int a* ⟹ *f x* = *g x n* = *n′ a* = *a′*
shows *EM-remainder-converges n f a* = *EM-remainder-converges n′ g a′*
⟨*proof*⟩

lemma *EM-remainder-cong*:
assumes ⋀*x. x* ≥ *of-int a* ⟹ *f x* = *g x n* = *n′ a* = *a′*
shows *EM-remainder n f a* = *EM-remainder n′ g a′*
⟨*proof*⟩

lemma *EM-remainder-converges-cnj*:
assumes *continuous-on* {*a..*} *f* **and** *EM-remainder-converges n f a*
shows *EM-remainder-converges n* (*λx. cnj* (*f x*)) *a*
⟨*proof*⟩

lemma *EM-remainder-converges-cnj-iff*:
assumes *continuous-on* {*of-int a..*} *f*
shows *EM-remainder-converges n* (*λx. cnj* (*f x*)) *a* ⟷ *EM-remainder-converges n f a*
⟨*proof*⟩

lemma *EM-remainder-cnj*:
assumes *continuous-on* {*a..*} *f*
shows *EM-remainder n* (*λx. cnj* (*f x*)) *a* = *cnj* (*EM-remainder n f a*)
⟨*proof*⟩

lemma *EM-remainder′-combine*:
  fixes f :: *real* ⇒ ′*a* :: *banach*
  assumes [*continuous-intros*]: *continuous-on* $\{a..c\}$ f
  assumes $a \leq b$ $b \leq c$
  shows *EM-remainder′* n f a b + *EM-remainder′* n f b c = *EM-remainder′* n f a c
⟨*proof*⟩

lemma *uniformly-convergent-EM-remainder′*:
  fixes f :: ′*a* ⇒ *real* ⇒ ′*b* :: {*banach*,*real-normed-algebra*}
  assumes *deriv*: $\bigwedge y.\ a \leq y \Longrightarrow$ (*G has-real-derivative g y*) (*at y within* $\{a..\}$)
  assumes *integrable*: $\bigwedge a'\ b\ y.\ y \in A \Longrightarrow\ a \leq a' \Longrightarrow a' \leq b \Longrightarrow$
      ($\lambda t.$ *pbernpoly n t* $*_R$ *f y t*) *integrable-on* $\{a'..b\}$
  assumes *conv*: *convergent* ($\lambda y.$ *G* (*real y*))
  assumes *bound*: *eventually* ($\lambda x.\ \forall y{\in}A.$ *norm* (*f y x*) $\leq$ *g x*) *at-top*
  shows *uniformly-convergent-on A* ($\lambda b\ s.$ *EM-remainder′ n* (*f s*) *a b*)
⟨*proof*⟩

lemma *uniform-limit-EM-remainder*:
  fixes f :: ′*a* ⇒ *real* ⇒ ′*b* :: {*banach*,*real-normed-algebra*}
  assumes *deriv*: $\bigwedge y.\ a \leq y \Longrightarrow$ (*G has-real-derivative g y*) (*at y within* $\{a..\}$)
  assumes *integrable*: $\bigwedge a'\ b\ y.\ y \in A \Longrightarrow\ a \leq a' \Longrightarrow a' \leq b \Longrightarrow$
      ($\lambda t.$ *pbernpoly n t* $*_R$ *f y t*) *integrable-on* $\{a'..b\}$
  assumes *conv*: *convergent* ($\lambda y.$ *G* (*real y*))
  assumes *bound*: *eventually* ($\lambda x.\ \forall y{\in}A.$ *norm* (*f y x*) $\leq$ *g x*) *at-top*
  shows *uniform-limit A* ($\lambda b\ s.$ *EM-remainder′ n* (*f s*) *a b*)
      ($\lambda s.$ *EM-remainder n* (*f s*) *a*) *sequentially*
⟨*proof*⟩

lemma *tendsto-EM-remainder*:
  fixes f :: *real* ⇒ ′*b* :: {*banach*,*real-normed-algebra*}
  assumes *deriv*: $\bigwedge y.\ a \leq y \Longrightarrow$ (*G has-real-derivative g y*) (*at y within* $\{a..\}$)
  assumes *integrable*: $\bigwedge a'\ b\ .\ a \leq a' \Longrightarrow a' \leq b \Longrightarrow$
      ($\lambda t.$ *pbernpoly n t* $*_R$ *f t*) *integrable-on* $\{a'..b\}$
  assumes *conv*: *convergent* ($\lambda y.$ *G* (*real y*))
  assumes *bound*: *eventually* ($\lambda x.$ *norm* (*f x*) $\leq$ *g x*) *at-top*
  shows *filterlim* ($\lambda b.$ *EM-remainder′ n f a b*)
      (*nhds* (*EM-remainder n f a*)) *sequentially*
⟨*proof*⟩

lemma *EM-remainder-0* [*simp*]: *EM-remainder n* ($\lambda x.$ *0*) *a* = *0*
  ⟨*proof*⟩

lemma *holomorphic-EM-remainder′*:
  assumes *deriv*:
    $\bigwedge z\ t.\ z \in U \Longrightarrow t \in \{a..x\} \Longrightarrow$
      (($\lambda z.$ *f z t*) *has-field-derivative f′ z t*) (*at z within U*)
  assumes *int*: $\bigwedge b\ c\ z\ e.\ a \leq b \Longrightarrow c \leq x \Longrightarrow z \in U \Longrightarrow$

$(\lambda t.\ \textit{of-real}\ (\textit{bernpoly}\ n\ (t - e)) * f\ z\ t)\ \textit{integrable-on}\ \{b..c\}$
assumes *cont*: $\textit{continuous-on}\ (U \times \{a..x\})\ (\lambda(z,\ t).\ f'\ z\ t)$
assumes *convex U*
shows $(\lambda s.\ \textit{EM-remainder}'\ n\ (f\ s)\ a\ x)\ \textit{holomorphic-on}\ U$
⟨*proof*⟩

lemma
assumes *deriv*: $\bigwedge y.\ a \le y \Longrightarrow (G\ \textit{has-real-derivative}\ g\ y)\ (\textit{at}\ y\ \textit{within}\ \{a..\})$
assumes *deriv'*:
$\bigwedge z\ t\ x.\ z \in U \Longrightarrow x \ge a \Longrightarrow t \in \{a..x\} \Longrightarrow$
$((\lambda z.\ f\ z\ t)\ \textit{has-field-derivative}\ f'\ z\ t)\ (\textit{at}\ z\ \textit{within}\ U)$
assumes *cont*: $\textit{continuous-on}\ (U \times \{\textit{of-int}\ a..\})\ (\lambda(z,\ t).\ f'\ z\ t)$
assumes *int*: $\bigwedge b\ c\ z\ e.\ a \le b \Longrightarrow z \in U \Longrightarrow$
$(\lambda t.\ \textit{of-real}\ (\textit{bernpoly}\ n\ (t - e)) * f\ z\ t)\ \textit{integrable-on}\ \{b..c\}$
assumes *int'*: $\bigwedge a'\ b\ y.\ y \in U \Longrightarrow a \le a' \Longrightarrow a' \le b \Longrightarrow$
$(\lambda t.\ \textit{pbernpoly}\ n\ t *_R f\ y\ t)\ \textit{integrable-on}\ \{a'..b\}$
assumes *conv*: $\textit{convergent}\ (\lambda y.\ G\ (\textit{real}\ y))$
assumes *bound*: $\textit{eventually}\ (\lambda x.\ \forall y{\in}U.\ \textit{norm}\ (f\ y\ x) \le g\ x)\ \textit{at-top}$
assumes *open U*
shows *analytic-EM-remainder*: $(\lambda s{::}\textit{complex}.\ \textit{EM-remainder}\ n\ (f\ s)\ a)\ \textit{analytic-on}\ U$
and *holomorphic-EM-remainder*: $(\lambda s{::}\textit{complex}.\ \textit{EM-remainder}\ n\ (f\ s)\ a)\ \textit{holomorphic-on}\ U$
⟨*proof*⟩

The following lemma is the first step in the proof of the Euler–MacLaurin formula: We show the relationship between the first-order remainder term and the difference of the integral and the sum.

context
fixes $f\ f' :: \textit{real} \Rightarrow {'a} :: \textit{banach}$
fixes $a\ b :: \textit{int}$ **and** $I\ S :: {'a}$
fixes $Y :: \textit{real set}$
assumes $a \le b$
assumes *fin*: *finite Y*
assumes *cont*: $\textit{continuous-on}\ \{\textit{real-of-int}\ a..\textit{real-of-int}\ b\}\ f$
assumes *deriv* [*derivative-intros*]:
$\bigwedge x{::}\textit{real}.\ x \in \{a..b\} - Y \Longrightarrow (f\ \textit{has-vector-derivative}\ f'\ x)\ (\textit{at}\ x)$
defines *S-def*: $S \equiv (\sum i{\in}\{a{<}..b\}.\ f\ i)$ **and** *I-def*: $I \equiv \textit{integral}\ \{a..b\}\ f$
begin

lemma
diff-sum-integral-has-integral-int:
$((\lambda t.\ (\textit{frac}\ t - 1/2) *_R f'\ t)\ \textit{has-integral}\ (S - I - (f\ b - f\ a)\ /_R\ 2))\ \{a..b\}$
⟨*proof*⟩

lemma *diff-sum-integral-has-integral-int'*:
$((\lambda t.\ \textit{pbernpoly}\ 1\ t *_R f'\ t)\ \textit{has-integral}\ (S - I - (f\ b - f\ a)\ /_R\ 2\ ))\ \{a..b\}$
⟨*proof*⟩

lemma *EM-remainder′-Suc-0*: *EM-remainder′* (*Suc 0*) *f′ a b* = *S* − *I* − (*f b* − *f a*) $/_R$ *2*
⟨*proof*⟩

end

Next, we show that the n-th-order remainder can be expressed in terms of the $n + 1$-th-order remainder term. Iterating this essentially yields the Euler–MacLaurin formula.

context
fixes *f f′* :: *real* ⇒ *′a* :: *banach* **and** *a b* :: *int* **and** *n* :: *nat* **and** *A* :: *real set*
assumes *ab*: *a* ≤ *b* **and** *n*: *n* > *0*
assumes *fin*: *finite A*
assumes *cont*: *continuous-on* {*of-int a..of-int b*} *f*
assumes *cont′*: *continuous-on* {*of-int a..of-int b*} *f′*
assumes *deriv*: ⋀*x*. *x* ∈ {*of-int a*<..<*of-int b*} − *A* ⟹ (*f has-vector-derivative f′ x*) (*at x*)
begin

lemma *EM-remainder′-integral-conv-Suc*:
shows *integral* {*a..b*} (λ*t*. *pbernpoly n t* $*_R$ *f t*) =
(*bernoulli* (*Suc n*) / *real* (*Suc n*)) $*_R$ (*f b* − *f a*) −
integral {*a..b*} (λ*t*. *pbernpoly* (*Suc n*) *t* $*_R$ *f′ t*) $/_R$ *real* (*Suc n*)
⟨*proof*⟩

lemma *EM-remainder′-conv-Suc*:
EM-remainder′ n f a b =
((−*1*) ^ *Suc n* ∗ *bernoulli* (*Suc n*) / *fact* (*Suc n*)) $*_R$ (*f b* − *f a*) +
EM-remainder′ (*Suc n*) *f′ a b*
⟨*proof*⟩

end

context
fixes *f f′* :: *real* ⇒ *′a* :: *banach* **and** *a* :: *int* **and** *n* :: *nat* **and** *A* :: *real set* **and** *C*
assumes *n*: *n* > *0*
assumes *fin*: *finite A*
assumes *cont*: *continuous-on* {*of-int a..*} *f*
assumes *cont′*: *continuous-on* {*of-int a..*} *f′*
assumes *lim*: (*f* ⟶ *C*) *at-top*
assumes *deriv*: ⋀*x*. *x* ∈ {*of-int a*<..} − *A* ⟹ (*f has-vector-derivative f′ x*) (*at x*)
begin

lemma
shows *EM-remainder-converges-iff-Suc-converges*:
EM-remainder-converges n f a ⟷ *EM-remainder-converges* (*Suc n*) *f′ a*
and *EM-remainder-conv-Suc*:

EM-remainder-converges n f a $\Longrightarrow$
EM-remainder n f a =
((−*1*) ^ *Suc n* ∗ *bernoulli* (*Suc n*) / *fact* (*Suc n*)) $*_R$ (*C* − *f a*) +
EM-remainder (*Suc n*) *f′ a*

⟨*proof*⟩

end

1.3 The conventional version of the Euler–MacLaurin formula

The following theorems are the classic Euler–MacLaurin formula that can be found, with slight variations, in many sources (e. g. [1, 2, 3]).

context
fixes *f* :: *real* ⇒ *′a* :: *banach*
fixes *fs* :: *nat* ⇒ *real* ⇒ *′a*
fixes *a b* :: *int* **assumes** *ab*: *a* ≤ *b*
fixes *N* :: *nat* **assumes** *N*: *N* > *0*
fixes *Y* :: *real set* **assumes** *fin*: *finite Y*
assumes *fs-0* [*simp*]: *fs 0* = *f*
assumes *fs-cont* [*continuous-intros*]:
⋀*k*. *k* ≤ *N* $\Longrightarrow$ *continuous-on* {*real-of-int a*..*real-of-int b*} (*fs k*)
assumes *fs-deriv* [*derivative-intros*]:
⋀*k x*. *k* < *N* $\Longrightarrow$ *x* ∈ {*a*..*b*} − *Y* $\Longrightarrow$ (*fs k has-vector-derivative fs* (*Suc k*) *x*) (*at x*)
begin

theorem *euler-maclaurin-raw-strong-int*:
defines *S* ≡ (∑ *i*∈{*a*<..*b*}. *f* (*of-int i*))
defines *I* ≡ *integral* {*of-int a*..*of-int b*} *f*
defines *c′* ≡ λ*k*. (*bernoulli′* (*Suc k*) / *fact* (*Suc k*)) $*_R$ (*fs k b* − *fs k a*)
shows *S* − *I* = (∑ *k*<*N*. *c′ k*) + *EM-remainder′ N* (*fs N*) *a b*
⟨*proof*⟩

end

theorem *euler-maclaurin-strong-raw-nat*:
assumes *a* ≤ *b* *0* < *N* *finite Y* *fs 0* = *f*
(⋀*k*. *k* ≤ *N* $\Longrightarrow$ *continuous-on* {*real a*..*real b*} (*fs k*))
(⋀*k x*. *k* < *N* $\Longrightarrow$ *x* ∈ {*real a*..*real b*} − *Y* $\Longrightarrow$
(*fs k has-vector-derivative fs* (*Suc k*) *x*) (*at x*))
shows (∑ *i*∈{*a*<..*b*}. *f* (*real i*)) − *integral* {*real a*..*real b*} *f* =
(∑ *k*<*N*. (*bernoulli′* (*Suc k*) / *fact* (*Suc k*)) $*_R$ (*fs k* (*real b*) − *fs k* (*real a*))) +
EM-remainder′ N (*fs N*) (*real a*) (*real b*)
⟨*proof*⟩

1.4 The "Concrete Mathematics" version of the Euler–MacLaurin formula

As explained in *Concrete Mathematics* [3], the above form of the formula has some drawbacks: When applying it to determine the asymptotics of some concrete function, one is usually left with several different unwieldy constant terms that are difficult to get rid of.

There is no general way to determine what these constant terms are, but in concrete applications, they can often be determined or estimated by other means. We can therefore simply group all the constant terms into a single constant and have the user provide a proof of what it is.

locale *euler-maclaurin-int* =
  fixes *F f* :: *real* ⇒ *'a* :: *banach*
  fixes *fs* :: *nat* ⇒ *real* ⇒ *'a*
  fixes *a* :: *int*
  fixes *N* :: *nat* **assumes** *N*: *N* > *0*
  fixes *C* :: *'a*
  fixes *Y* :: *real set* **assumes** *fin*: *finite Y*
  assumes *fs-0* [*simp*]: *fs 0* = *f*
  assumes *fs-cont* [*continuous-intros*]:
    $\bigwedge k.\ k \le N \Longrightarrow$ *continuous-on* {*real-of-int a..*} (*fs k*)
  assumes *fs-deriv* [*derivative-intros*]:
    $\bigwedge k\ x.\ k < N \Longrightarrow x \in$ {*of-int a..*} − *Y* ⟹ (*fs k has-vector-derivative fs* (*Suc k*) *x*) (*at x*)
  assumes *F-cont* [*continuous-intros*]: *continuous-on* {*of-int a..*} *F*
  assumes *F-deriv* [*derivative-intros*]:
    $\bigwedge x.\ x \in$ {*of-int a..*} − *Y* ⟹ (*F has-vector-derivative f x*) (*at x*)
  assumes *limit*:
    ((λ*b*. (∑*k*=*a..b*. *f k*) − *F* (*of-int b*) −
      (∑*i*<*N*. (*bernoulli'* (*Suc i*) / *fact* (*Suc i*)) $*_R$ *fs i* (*of-int b*))) ⟶ *C*) *at-top*
begin

context
  fixes *C' T*
  defines *C'* ≡ −*f a* + *F a* + *C* + (∑*k*<*N*. (*bernoulli'* (*Suc k*) / *fact* (*Suc k*)) $*_R$ (*fs k* (*of-int a*)))
    and *T* ≡ (λ*x*. ∑*i*<*N*. (*bernoulli'* (*Suc i*) / *fact* (*Suc i*)) $*_R$ *fs i x*)
begin

lemma *euler-maclaurin-strong-int-aux*:
  assumes *ab*: *a* ≤ *b*
  defines *S* ≡ (∑*k*=*a..b*. *f* (*of-int k*))
  shows *S* − *F* (*of-int b*) − *T* (*of-int b*) = *EM-remainder'* *N* (*fs N*) (*of-int a*) (*of-int b*) + (*C* − *C'*)
⟨*proof*⟩

lemma *EM-remainder-limit*:

assumes *ab*: $a \le b$
defines $D \equiv$ *EM-remainder'* N (*fs* N) (*of-int* a) (*of-int* b)
shows *EM-remainder* N (*fs* N) $b = C' - D$
and *EM-remainder-converges*: *EM-remainder-converges* N (*fs* N) b
⟨*proof*⟩

theorem *euler-maclaurin-strong-int*:
assumes *ab*: $a \le b$
defines $S \equiv (\sum k{=}a..b.\ f\ (\textit{of-int}\ k))$
shows $S = F\ (\textit{of-int}\ b) + C + T\ (\textit{of-int}\ b) - \textit{EM-remainder}\ N\ (\textit{fs}\ N)\ b$
⟨*proof*⟩

end
end

The following version of the formula removes all the terms where the associated Bernoulli numbers vanish.

locale *euler-maclaurin-int'* =
fixes $F\ f$ :: *real* ⇒ *'a* :: *banach*
fixes *fs* :: *nat* ⇒ *real* ⇒ *'a*
fixes a :: *int*
fixes N :: *nat*
fixes C :: *'a*
fixes Y :: *real set* **assumes** *fin*: *finite* Y
assumes *fs-0* [*simp*]: *fs* $0 = f$
assumes *fs-cont* [*continuous-intros*]:
$\bigwedge k.\ k \le 2{*}N{+}1 \Longrightarrow$ *continuous-on* {*real-of-int* a..} (*fs* k)
assumes *fs-deriv* [*derivative-intros*]:
$\bigwedge k\ x.\ k \le 2{*}N \Longrightarrow x \in$ {*of-int* a..} $- Y \Longrightarrow$ (*fs* k *has-vector-derivative fs* (*Suc* k) x) (*at* x)
assumes *F-cont* [*continuous-intros*]: *continuous-on* {*of-int* a..} F
assumes *F-deriv* [*derivative-intros*]:
$\bigwedge x.\ x \in$ {*of-int* a..} $- Y \Longrightarrow$ (F *has-vector-derivative* $f\ x$) (*at* x)
assumes *limit*:
$((\lambda b.\ (\sum k{=}a..b.\ f\ k) - F\ (\textit{of-int}\ b) -$
$(\sum i{<}2{*}N{+}1.\ (\textit{bernoulli'}\ (\textit{Suc}\ i)\ /\ \textit{fact}\ (\textit{Suc}\ i)) *_R \textit{fs}\ i\ (\textit{of-int}\ b))) \longrightarrow C)$ *at-top*
begin

sublocale *euler-maclaurin-int* $F\ f$ *fs* $a\ 2{*}N{+}1\ C\ Y$
⟨*proof*⟩

theorem *euler-maclaurin-strong-int'*:
assumes $a \le b$
shows $(\sum k{=}a..b.\ f\ (\textit{of-int}\ k)) =$
$F\ (\textit{of-int}\ b) + C + (1\ /\ 2) *_R f\ (\textit{of-int}\ b) +$
$(\sum i{=}1..N.\ (\textit{bernoulli}\ (2{*}i)\ /\ \textit{fact}\ (2{*}i)) *_R \textit{fs}\ (2{*}i{-}1)\ (\textit{of-int}\ b)) -$
EM-remainder $(2{*}N{+}1)$ (*fs* $(2{*}N{+}1)$) b
⟨*proof*⟩

end

For convenience, we also offer a version where the sum ranges over natural numbers instead of integers.

lemma *sum-atLeastAtMost-of-int-nat-transfer*:
$(\sum k{=}int\ a..int\ b.\ f\ (of\text{-}int\ k)) = (\sum k{=}a..b.\ f\ (of\text{-}nat\ k))$
⟨*proof*⟩

lemma *euler-maclaurin-nat-int-transfer*:
fixes F **and** f :: *real* ⇒ *'a* :: *real-normed-vector*
assumes $((\lambda b.\ (\sum k{=}a..b.\ f\ (real\ k)) - F\ (real\ b) - T\ (real\ b)) \longrightarrow C)$ *at-top*
shows $((\lambda b.\ (\sum k{=}int\ a..b.\ f\ (of\text{-}int\ k)) - F\ (of\text{-}int\ b) - T\ (of\text{-}int\ b)) \longrightarrow C)$ *at-top*
⟨*proof*⟩

locale *euler-maclaurin-nat* =
fixes $F\ f$ :: *real* ⇒ *'a* :: *banach*
fixes *fs* :: *nat* ⇒ *real* ⇒ *'a*
fixes a :: *nat*
fixes N :: *nat* **assumes** N: $N > 0$
fixes C :: *'a*
fixes Y :: *real set* **assumes** *fin*: *finite* Y
assumes *fs-0* [*simp*]: *fs* $0 = f$
assumes *fs-cont* [*continuous-intros*]:
$\bigwedge k.\ k \le N \Longrightarrow$ *continuous-on* {*real a..*} (*fs k*)
assumes *fs-deriv* [*derivative-intros*]:
$\bigwedge k\ x.\ k < N \Longrightarrow x \in \{real\ a..\} - Y \Longrightarrow$ (*fs k has-vector-derivative fs* (*Suc k*) *x*) (*at x*)
assumes *F-cont* [*continuous-intros*]: *continuous-on* {*real a..*} *F*
assumes *F-deriv* [*derivative-intros*]:
$\bigwedge x.\ x \in \{real\ a..\} - Y \Longrightarrow$ (*F has-vector-derivative f x*) (*at x*)
assumes *limit*:
$((\lambda b.\ (\sum k{=}a..b.\ f\ k) - F\ (real\ b) -$
$(\sum i{<}N.\ (bernoulli'\ (Suc\ i)\ /\ fact\ (Suc\ i)) *_R fs\ i\ (real\ b))) \longrightarrow C)$ *at-top*
begin

sublocale *euler-maclaurin-int F f fs int a N C Y*
⟨*proof*⟩

theorem *euler-maclaurin-strong-nat*:
assumes *ab*: $a \le b$
defines $S \equiv (\sum k{=}a..b.\ f\ (real\ k))$
shows $S = F\ (real\ b) + C + (\sum i{<}N.\ (bernoulli'\ (Suc\ i)\ /\ fact\ (Suc\ i)) *_R fs\ i\ (real\ b)) -$
EM-remainder N (*fs N*) (*int b*)
⟨*proof*⟩

end

locale *euler-maclaurin-nat′* =
  fixes *F f* :: *real* ⇒ *′a* :: *banach*
  fixes *fs* :: *nat* ⇒ *real* ⇒ *′a*
  fixes *a* :: *nat*
  fixes *N* :: *nat*
  fixes *C* :: *′a*
  fixes *Y* :: *real set* **assumes** *fin*: *finite Y*
  assumes *fs-0* [*simp*]: *fs 0* = *f*
  assumes *fs-cont* [*continuous-intros*]:
    $\bigwedge k.\ k \leq 2{*}N{+}1 \Longrightarrow$ *continuous-on* {*real a..*} (*fs k*)
  assumes *fs-deriv* [*derivative-intros*]:
    $\bigwedge k\ x.\ k \leq 2{*}N \Longrightarrow x \in$ {*real a..*} − *Y* ⟹ (*fs k has-vector-derivative fs* (*Suc k*) *x*) (*at x*)
  assumes *F-cont* [*continuous-intros*]: *continuous-on* {*real a..*} *F*
  assumes *F-deriv* [*derivative-intros*]:
    $\bigwedge x.\ x \in$ {*real a..*} − *Y* ⟹ (*F has-vector-derivative f x*) (*at x*)
  assumes *limit*:
    ((λ*b*. (∑ *k*=*a..b*. *f k*) − *F* (*real b*) −
      (∑ *i*<*2*∗*N*+*1*. (*bernoulli′* (*Suc i*) / *fact* (*Suc i*)) $*_R$ *fs i* (*real b*))) ⟶ *C*) *at-top*
begin

sublocale *euler-maclaurin-int′ F f fs int a N C Y*
  ⟨*proof*⟩

theorem *euler-maclaurin-strong-nat′*:
  assumes *a* ≤ *b*
  shows (∑ *k*=*a..b*. *f* (*real k*)) =
    F (*real b*) + *C* + (*1* / *2*) $*_R$ *f* (*real b*) +
    (∑ *i*=*1..N*. (*bernoulli* (*2*∗*i*) / *fact* (*2*∗*i*)) $*_R$ *fs* (*2*∗*i*−*1*) (*real b*)) −
    EM-remainder (*2*∗*N*+*1*) (*fs* (*2*∗*N*+*1*)) *b*
  ⟨*proof*⟩

end

1.5 Bounds on the remainder term

The following theorems provide some simple means to bound the remainder terms. In practice, better bounds can often be obtained e. g. for the n-th remainder term by expanding it to the sum of the first discarded term in the expansion and the $n+1$-th remainder term.

lemma
  fixes *f* :: *real* ⇒ *′a* :: {*real-normed-field*, *banach*}
    and *g g′* :: *real* ⇒ *real*
  assumes *fin*: *finite Y*
  assumes *pbernpoly-bound*: ∀ *x*. |*pbernpoly n x*| ≤ *D*
  assumes *cont-f*: *continuous-on* {*a..*} *f*

assumes *cont-g*: *continuous-on* $\{a..\}$ *g*
assumes *cont-g'*: *continuous-on* $\{a..\}$ *g'*
assumes *limit-g*: $(g \longrightarrow C)$ *at-top*
assumes *f-bound*: $\bigwedge x.\ x \geq a \Longrightarrow$ *norm* $(f\ x) \leq g'\ x$
assumes *deriv*: $\bigwedge x.\ x \in \{a..\} - Y \Longrightarrow$ (*g has-field-derivative* $g'\ x$) (*at* x)
shows *norm-EM-remainder-le-strong-int*:
$\forall x.$ *of-int* $x \geq a \longrightarrow$ *norm* (*EM-remainder* $n\ f\ x$) $\leq D$ / *fact* $n * (C - g\ x)$
and *norm-EM-remainder-le-strong-nat*:
$\forall x.$ *real* $x \geq a \longrightarrow$ *norm* (*EM-remainder* $n\ f$ (*int* x)) $\leq D$ / *fact* $n * (C - g\ x)$
⟨*proof*⟩

lemma
fixes f :: *real* ⇒ *'a* :: {*real-normed-field*, *banach*}
and $g\ g'$ :: *real* ⇒ *real*
assumes *fin*: *finite* Y
assumes *pbernpoly-bound*: $\forall x.\ |$*pbernpoly* $n\ x| \leq D$
assumes *cont-f*: *continuous-on* $\{a..\}$ *f*
assumes *cont-g*: *continuous-on* $\{a..\}$ *g*
assumes *cont-g'*: *continuous-on* $\{a..\}$ *g'*
assumes *limit-g*: $(g \longrightarrow 0)$ *at-top*
assumes *f-bound*: $\bigwedge x.\ x \geq a \Longrightarrow$ *norm* $(f\ x) \leq g'\ x$
assumes *deriv*: $\bigwedge x.\ x \in \{a..\} - Y \Longrightarrow$ (*g has-field-derivative* $-g'\ x$) (*at* x)
shows *norm-EM-remainder-le-strong-int'*:
$\forall x.$ *of-int* $x \geq a \longrightarrow$ *norm* (*EM-remainder* $n\ f\ x$) $\leq D$ / *fact* $n * g\ x$
and *norm-EM-remainder-le-strong-nat'*:
$\forall x.$ *real* $x \geq a \longrightarrow$ *norm* (*EM-remainder* $n\ f$ (*int* x)) $\leq D$ / *fact* $n * g\ x$
⟨*proof*⟩

lemma *norm-EM-remainder'-le*:
fixes f :: *real* ⇒ *'a* :: {*real-normed-field*, *banach*}
and $g\ g'$ :: *real* ⇒ *real*
assumes *cont-f*: *continuous-on* $\{a..\}$ *f*
assumes *cont-g'*: *continuous-on* $\{a..\}$ *g'*
assumes *f-bigo*: *eventually* ($\lambda x.$ *norm* $(f\ x) \leq g'\ x$) *at-top*
assumes *deriv*: *eventually* ($\lambda x.$ (*g has-field-derivative* $g'\ x$) (*at* x)) *at-top*
obtains $C\ D$ **where**
eventually ($\lambda x.$ *norm* (*EM-remainder'* $n\ f\ a\ x$) $\leq C + D * g\ x$) *at-top*
⟨*proof*⟩

1.6 Application to harmonic numbers

As a first application, we can apply the machinery we have developed to the harmonic numbers.

definition *harm-remainder* :: *nat* ⇒ *nat* ⇒ *real* **where**
harm-remainder $N\ n$ = *EM-remainder* $(2{*}N{+}1)$ ($\lambda x.$ $-$*fact* $(2{*}N{+}1)$ / x ^ $(2{*}N{+}2)$) (*int* n)

lemma *harm-expansion*:
  assumes *n*: $n > 0$ **and** *N*: $N > 0$
  shows $harm\ n = ln\ n + euler\text{-}mascheroni + 1 / (2*n) -$
    $(\sum i=1..N.\ bernoulli\ (2*i) / ((2*i) * n\ \hat{}\ (2*i))) - harm\text{-}remainder$
$N\ n$
⟨*proof*⟩

lemma *of-nat-ge-1-iff*: $of\text{-}nat\ x \geq (1 :: 'a :: linordered\text{-}semidom) \longleftrightarrow x \geq 1$
  ⟨*proof*⟩

lemma *harm-remainder-bound*:
  fixes $N :: nat$
  assumes *N*: $N > 0$
  shows $\exists C.\ \forall n\geq 1.\ norm\ (harm\text{-}remainder\ N\ n) \leq C / real\ n\ \hat{}\ (2*N+1)$
⟨*proof*⟩

1.7 Application to sums of inverse squares

In the same vein, we can derive the asymptotics of the partial sum of inverse squares.

lemma *sum-inverse-squares-expansion*:
  assumes *n*: $n > 0$ **and** *N*: $N > 0$
  shows $(\sum k=1..n.\ 1 / real\ k\ \hat{}\ 2) =$
    $pi\ \hat{}\ 2 / 6 - 1 / real\ n + 1 / (2 * real\ n\ \hat{}\ 2) -$
      $(\sum i=1..N.\ bernoulli\ (2*i) / n\ \hat{}\ (2*i+1)) -$
      $EM\text{-}remainder\ (2*N+1)\ (\lambda x.\ -fact\ (2*N+2) / x\ \hat{}\ (2*N+3))$
$(int\ n)$
⟨*proof*⟩

lemma *sum-inverse-squares-remainder-bound*:
  fixes $N :: nat$
  assumes *N*: $N > 0$
  defines $R \equiv (\lambda n.\ EM\text{-}remainder\ (2*N+1)\ (\lambda x.\ -fact\ (2*N+2) / x\ \hat{}(2*N+3))$
$(int\ n))$
  shows $\exists C.\ \forall n\geq 1.\ norm\ (R\ n) \leq C / real\ n\ \hat{}\ (2*N+2)$
⟨*proof*⟩

end

2 Connection of Euler–MacLaurin summation to Landau symbols

theory *Euler-MacLaurin-Landau*
imports
  Euler-MacLaurin
  Landau-Symbols.Landau-More
begin

2.1 O-bound for the remainder term

Landau symbols allow us to state the bounds on the remainder terms from the Euler–MacLaurin formula a bit more nicely.

lemma
fixes *f* :: *real* ⇒ *'a* :: {*real-normed-field*, *banach*}
and *g g'* :: *real* ⇒ *real*
assumes *fin*: *finite Y*
assumes *cont-f*: *continuous-on* {*a*..} *f*
assumes *cont-g*: *continuous-on* {*a*..} *g*
assumes *cont-g'*: *continuous-on* {*a*..} *g'*
assumes *limit-g*: (*g* ⟶ *0*) *at-top*
assumes *f-bound*: ⋀*x*. *x* ≥ *a* ⟹ *norm* (*f x*) ≤ *g' x*
assumes *deriv*: ⋀*x*. *x* ∈ {*a*..} − *Y* ⟹ (*g has-field-derivative* −*g' x*) (*at x*)
shows *EM-remainder-strong-bigo-int*: (λ*x*::*int*. *norm* (*EM-remainder n f x*)) ∈ *O*(*g*)
and *EM-remainder-strong-bigo-nat*: (λ*x*::*nat*. *norm* (*EM-remainder n f x*)) ∈ *O*(*g*)
⟨*proof*⟩

2.2 Asymptotic expansion of the harmonic numbers

We can now show the asymptotic expansion

$$H_n = \ln n + \gamma + \frac{1}{2n} - \sum_{i=1}^{m} \frac{B_{2i}}{2i} n^{-2i} + O(n^{-2m-2})$$

lemma *harm-remainder-bigo*:
assumes *N* > *0*
shows *harm-remainder N* ∈ *O*(λ*n*. *1* / *real n* ^ (*2* ∗ *N* + *1*))
⟨*proof*⟩

lemma *harm-expansion-bigo*:
fixes *N* :: *nat*
defines *T* ≡ λ*n*. *ln n* + *euler-mascheroni* + *1* / (*2*∗*n*) −
(∑ *i*=*1*..*N*. *bernoulli* (*2*∗*i*) / ((*2*∗*i*) ∗ *n* ^ (*2*∗*i*)))
defines *S* ≡ (λ*n*. *bernoulli* (*2*∗(*Suc N*)) / ((*2*∗*Suc N*) ∗ *real n* ^ (*2*∗*Suc N*)))
shows (λ*n*. *harm n* − *T n*) ∈ *O*(λ*n*. *1* / *real n* ^ (*2* ∗ *N* + *2*))
⟨*proof*⟩

lemma *harm-expansion-bigo-simple1*:
(λ*n*. *harm n* − (*ln n* + *euler-mascheroni* + *1* / (*2* ∗ *n*))) ∈ *O*(λ*n*. *1* / *n* ^ *2*)
⟨*proof*⟩

lemma *harm-expansion-bigo-simple2*:
(λ*n*. *harm n* − (*ln n* + *euler-mascheroni*)) ∈ *O*(λ*n*. *1* / *n*)
⟨*proof*⟩

lemma *harm-expansion-bigo-simple'*:
harm =o (λ*n. ln n + euler-mascheroni + 1 / (2 ∗ n)) +o O*(λ*n. 1 / n ^ 2*)
⟨*proof*⟩

2.3 Asymptotic expansion of the sum of inverse squares

Similarly to before, we show

$$\sum_{i=1}^{n} \frac{1}{i^2} = \frac{\pi^2}{6} - \frac{1}{n} + \frac{1}{2n^2} - \sum_{i=1}^{m} B_{2i} n^{-2i-1} + O(n^{-2m-3})$$

context
fixes *R* :: *nat* ⇒ *nat* ⇒ *real*
defines *R* ≡ (λ*N n. EM-remainder* (*2∗N+1*) (λ*x.* −*fact* (*2∗N+2*) / *x* ^(*2∗N+3*)) (*int n*))
begin

lemma *sum-inverse-squares-remainder-bigo*:
assumes *N > 0*
shows *R N* ∈ *O*(λ*n. 1 / real n* ^ (*2 ∗ N + 2*))
⟨*proof*⟩

lemma *sum-inverse-squares-expansion-bigo*:
fixes *N* :: *nat*
defines *T* ≡ λ*n. pi ^ 2 / 6 − 1 / n + 1 /* (*2∗n ^ 2*) −
(∑ *i=1..N. bernoulli* (*2∗i*) / (*n* ^ (*2∗i+1*)))
defines *S* ≡ (λ*n. bernoulli* (*2∗*(*Suc N*)) / (*real n* ^ (*2∗N+3*)))
shows (λ*n.* (∑ *i=1..n. 1 / real i ^ 2*) − *T n*) ∈ *O*(λ*n. 1 / real n* ^ (*2 ∗ N + 3*))
⟨*proof*⟩

lemma *sum-inverse-squares-expansion-bigo-simple*:
(λ*n.* (∑ *i=1..n. 1 / real i ^ 2*) − (*pi ^ 2 / 6 − 1 / n + 1 /* (*2∗n^2*))) ∈ *O*(λ*n. 1 / n ^ 3*)
⟨*proof*⟩

lemma *sum-inverse-squares-expansion-bigo-simple'*:
(λ*n.* (∑ *i=1..n. 1 / real i ^ 2*)) *=o* (λ*n. pi ^ 2 / 6 − 1 / n + 1 /* (*2∗n^2*)) *+o O*(λ*n. 1 / n^3*)
⟨*proof*⟩

end

end

References

[1] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables.* Dover, New York, tenth printing edition, 1964.

[2] T. M. Apostol. *An Elementary View of Euler's Summation Formula. The American Mathematical Monthly*, 106(5):409–418, 1999.

[3] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete Mathematics: A Foundation for Computer Science.* Addison–Wesley, Boston, MA, USA, 2nd edition, 1994.