

# Young's Inequality for Increasing Functions

Lawrence C. Paulson

March 24, 2023

## Abstract

Young's inequality states that

$$ab \leq \int_0^a f(x)dx + \int_0^b f^{-1}(y)dy$$

where  $a \geq 0$ ,  $b \geq 0$  and  $f$  is strictly increasing and continuous. Its proof is formalised following the development by Cunningham and Grossman [1]. Their idea is to make the intuitive, geometric folklore proof rigorous by reasoning about step functions. The lack of the Riemann integral makes the development longer than one would like, but their argument is reproduced faithfully.

## Contents

<b>1</b>	<b>Young's Inequality for Increasing Functions</b>	<b>3</b>
1.1	Library Extras: already added to the repository . . . . .	3
1.2	Toward Young's inequality . . . . .	6
1.3	Regular divisions . . . . .	7
1.4	Special cases of Young's inequality . . . . .	10

**Acknowledgements** The author was supported by the ERC Advanced Grant ALEXANDRIA (Project 742178) funded by the European Research Council.

# 1 Young's Inequality for Increasing Functions

From the following paper: Cunningham, F., and Nathaniel Grossman. On Youngs Inequality. The American Mathematical Monthly 78, no. 7 (1971): 78183. <https://doi.org/10.2307/2318018>

```
theory Youngs imports
  HOL-Analysis.Analysis
```

```
begin
```

## 1.1 Library Extras: already added to the repository

In fact, strict inequality is required only at a single point within the box.

```
lemma integral-less:
```

```
  fixes f :: 'n::euclidean-space  $\Rightarrow$  real
  assumes cont: continuous-on (cbox a b) f continuous-on (cbox a b) g and ne:
    box a b  $\neq$  {}
  and fg:  $\bigwedge x. x \in \text{box } a \ b \implies f \ x < g \ x$ 
  shows integral (cbox a b) f < integral (cbox a b) g
proof -
  obtain int: f integrable-on (cbox a b) g integrable-on (cbox a b)
  using cont integrable-continuous by blast
  then have integral (cbox a b) f  $\leq$  integral (cbox a b) g
  by (metis fg integrable-on-open-interval integral-le integral-open-interval less-eq-real-def)
  moreover have integral (cbox a b) f  $\neq$  integral (cbox a b) g
  proof (rule ccontr)
    assume  $\neg$  integral (cbox a b) f  $\neq$  integral (cbox a b) g
    then have 0: (( $\lambda x. g \ x - f \ x$ ) has-integral 0) (cbox a b)
    by (metis (full-types) cancel-comm-monoid-add-class.diff-cancel has-integral-diff
      int
        integrable-integral)
    have cgf: continuous-on (cbox a b) ( $\lambda x. g \ x - f \ x$ )
    using cont continuous-on-diff by blast
    show False
    using has-integral-0-cbox-imp-0 [OF cgf - 0] ne box-subset-cbox fg by fastforce
  qed
  ultimately show ?thesis
  by linarith
qed
```

```
lemma integral-less-real:
```

```
  fixes f :: real  $\Rightarrow$  real
  assumes continuous-on {a..b} f continuous-on {a..b} g and {a<..}  $\neq$  {}
  and  $\bigwedge x. x \in \{a<..\} \implies f \ x < g \ x$ 
  shows integral {a..b} f < integral {a..b} g
  by (metis assms box-real integral-less)
```

```
lemma has-integral-UN:
```

```

fixes f :: 'n::euclidean-space => 'a::banach
assumes finite I
  and int:  $\bigwedge i. i \in I \implies (f \text{ has-integral } (g \ i)) \ (\mathcal{T} \ i)$ 
  and neg: pairwise  $(\lambda i \ i'. \text{negligible } (\mathcal{T} \ i \cap \mathcal{T} \ i')) \ I$ 
shows  $(f \text{ has-integral } (\text{sum } g \ I)) \ (\bigcup i \in I. \mathcal{T} \ i)$ 
proof -
let  $\mathcal{U} = ((\lambda(a,b). \mathcal{T} \ a \cap \mathcal{T} \ b) \ \{(a,b). a \in I \wedge b \in I - \{a\}\})$ 
have  $((\lambda x. \text{if } x \in (\bigcup i \in I. \mathcal{T} \ i) \text{ then } f \ x \text{ else } 0) \text{ has-integral } \text{sum } g \ I) \ \text{UNIV}$ 
proof (rule has-integral-spike)
  show negligible  $(\bigcup \mathcal{U})$ 
  proof (rule negligible-Union)
    have finite  $(I \times I)$ 
    by (simp add:  $\langle \text{finite } I \rangle$ )
    moreover have  $\{(a,b). a \in I \wedge b \in I - \{a\}\} \subseteq I \times I$ 
    by auto
    ultimately show finite  $\mathcal{U}$ 
    by (simp add: finite-subset)
  show  $\bigwedge t. t \in \mathcal{U} \implies \text{negligible } t$ 
  using neg unfolding pairwise-def by auto
qed
next
show  $(\text{if } x \in (\bigcup i \in I. \mathcal{T} \ i) \text{ then } f \ x \text{ else } 0) = (\sum i \in I. \text{if } x \in \mathcal{T} \ i \text{ then } f \ x \text{ else } 0)$ 
  if  $x \in \text{UNIV} - (\bigcup \mathcal{U})$  for  $x$ 
proof clarsimp
  fix i assume  $i: i \in I \ x \in \mathcal{T} \ i$ 
  then have  $\forall j \in I. x \in \mathcal{T} \ j \longleftrightarrow j = i$ 
  using that by blast
  with i show  $f \ x = (\sum i \in I. \text{if } x \in \mathcal{T} \ i \text{ then } f \ x \text{ else } 0)$ 
  by (simp add: sum.delta[OF  $\langle \text{finite } I \rangle$ ])
qed
next
show  $((\lambda x. (\sum i \in I. \text{if } x \in \mathcal{T} \ i \text{ then } f \ x \text{ else } 0)) \text{ has-integral } \text{sum } g \ I) \ \text{UNIV}$ 
  using int by (simp add: has-integral-restrict-UNIV has-integral-sum[OF  $\langle \text{finite } I \rangle$ ])
qed
then show ?thesis
  using has-integral-restrict-UNIV by blast
qed

lemma integrable-mono-on-nonneg:
  fixes f :: real => real
  assumes mon: mono-on  $\{a..b\}$  f and 0:  $\bigwedge x. 0 \leq f \ x$ 
  shows integrable (lebesgue-on  $\{a..b\}$ ) f
proof -
  have space lborel = space lebesgue sets borel  $\subseteq$  sets lebesgue
  by force+
  then have fborel:  $f \in \text{borel-measurable } (\text{lebesgue-on } \{a..b\})$ 
  by (metis mon borel-measurable-mono-on-fnc borel-measurable-subalgebra mono-restrict-space
    space-lborel space-restrict-space)

```

**then obtain**  $g$  **where**  $g$ : *incseq*  $g$  **and** *simple*:  $\bigwedge i$ . *simple-function* (*lebesgue-on*  $\{a..b\}$ ) ( $g$   $i$ )  
**and** *bdd*:  $(\forall x$ . *bdd-above* (*range* ( $\lambda i$ .  $g$   $i$   $x$ ))) **and** *nonneg*:  $\forall i$   $x$ .  $0 \leq g$   $i$   $x$   
**and** *fsup*:  $f = (\text{SUP } i$ .  $g$   $i$ )  
**by** (*metis borel-measurable-implies-simple-function-sequence-real*  $0$ )  
**have**  $f$  ‘  $\{a..b\} \subseteq \{f$   $a..f$   $b\}$   
**using** *assms* **by** (*auto simp: mono-on-def*)  
**have** *g-le-f*:  $g$   $i$   $x \leq f$   $x$  **for**  $i$   $x$   
**proof** –  
**have** *bdd-above* ( $(\lambda h$ .  $h$   $x$ ) ‘ *range*  $g$ )  
**using** *bdd cSUP-lessD linorder-not-less* **by** *fastforce*  
**then show** *?thesis*  
**by** (*metis SUP-apply UNIV-I bdd cSUP-upper fsup*)  
**qed**  
**then have** *gfb*:  $g$   $i$   $x \leq f$   $b$  **if**  $x \in \{a..b\}$  **for**  $i$   $x$   
**by** (*smt (verit, best) mon atLeastAtMost-iff mono-on-def that*)  
**have** *g-le*:  $g$   $i$   $x \leq g$   $j$   $x$  **if**  $i \leq j$  **for**  $i$   $j$   $x$   
**using**  $g$  **by** (*simp add: incseq-def le-funD that*)  
**show** *integrable* (*lebesgue-on*  $\{a..b\}$ ) ( $f$ )  
**proof** (*rule integrable-dominated-convergence*)  
**show**  $f \in$  *borel-measurable* (*lebesgue-on*  $\{a..b\}$ )  
**using** *fborel* **by** *blast*  
**have**  $\bigwedge x$ .  $(\lambda i$ .  $g$   $i$   $x$ )  $\longrightarrow$  ( $\text{SUP } h \in$  *range*  $g$ .  $h$   $x$ )  
**proof** (*rule order-tendstoI*)  
**show**  $\forall_F$   $i$  **in** *sequentially*.  $y < g$   $i$   $x$   
**if**  $y < (\text{SUP } h \in$  *range*  $g$ .  $h$   $x$ ) **for**  $x$   $y$   
**proof** –  
**from** *that* **obtain**  $h$  **where**  $h$ :  $h \in$  *range*  $g$   $y < h$   $x$   
**using** *g-le-f* **by** (*subst (asm)less-cSUP-iff*) *fastforce*+  
**then show** *?thesis*  
**by** (*smt (verit, ccfv-SIG) eventually-sequentially g-le imageE*)  
**qed**  
**show**  $\forall_F$   $i$  **in** *sequentially*.  $g$   $i$   $x < y$   
**if**  $(\text{SUP } h \in$  *range*  $g$ .  $h$   $x$ )  $< y$  **for**  $x$   $y$   
**by** (*smt (verit, best) that Sup-apply g-le-f always-eventually fsup image-cong*)  
**qed**  
**then show** *AE*  $x$  **in** *lebesgue-on*  $\{a..b\}$ .  $(\lambda i$ .  $g$   $i$   $x$ )  $\longrightarrow$   $f$   $x$   
**by** (*simp add: fsup*)  
**fix**  $i$   
**show**  $g$   $i \in$  *borel-measurable* (*lebesgue-on*  $\{a..b\}$ )  
**using** *borel-measurable-simple-function simple* **by** *blast*  
**show** *AE*  $x$  **in** *lebesgue-on*  $\{a..b\}$ . *norm* ( $g$   $i$   $x$ )  $\leq f$   $b$   
**by** (*simp add: gfb nonneg Measure-Space.AE-I' [of {}]*)  
**qed** *auto*  
**qed**

**lemma** *integrable-mono-on*:  
**fixes**  $f$  :: *real*  $\Rightarrow$  *real*

```

assumes mono-on {a..b} f
shows integrable (lebesgue-on {a..b}) f
proof -
  define f' where f' ≡ λx. if x ∈ {a..b} then f x - f a else 0
  have mono-on {a..b} f'
    by (smt (verit, best) assms f'-def mono-on-def)
  moreover have 0: ∧x. 0 ≤ f' x
    by (smt (verit, best) assms atLeastAtMost-iff f'-def mono-on-def)
  ultimately have integrable (lebesgue-on {a..b}) f'
    using integrable-mono-on-nonneg by presburger
  then have integrable (lebesgue-on {a..b}) (λx. f' x + f a)
    by force
  moreover have space lborel = space lebesgue sets borel ⊆ sets lebesgue
    by force+
  then have fborel: f ∈ borel-measurable (lebesgue-on {a..b})
    using borel-measurable-mono-on-fnc [OF assms]
    by (metis borel-measurable-subalgebra mono-restrict-space space-lborel space-restrict-space)
  ultimately show ?thesis
    by (rule integrable-cong-AE-imp) (auto simp: f'-def)
qed

```

```

lemma integrable-on-mono-on:
  fixes f :: real ⇒ real
  assumes mono-on {a..b} f
  shows f integrable-on {a..b}
  by (simp add: assms integrable-mono-on integrable-on-lebesgue-on)

```

```

lemma strict-mono-image-endpoints:
  fixes f :: 'a::linear-continuum-topology ⇒ 'b::linorder-topology
  assumes strict-mono-on {a..b} f and f: continuous-on {a..b} f and a ≤ b
  shows f ' {a..b} = {f a..f b}
proof
  show f ' {a..b} ⊆ {f a..f b}
    using assms(1) strict-mono-on-leD by fastforce
  show {f a..f b} ⊆ f ' {a..b}
    using assms IVT'[OF - - f] by (force simp: Bex-def)
qed

```

## 1.2 Toward Young's inequality

Generalisations of the type of  $f$  are not obvious

```

lemma strict-mono-continuous-invD:
  fixes f :: real ⇒ real
  assumes sm: strict-mono-on {a..} f and contf: continuous-on {a..} f
    and fm: f ' {a..} = {f a..} and g: ∧x. x ≥ a ⇒ g (f x) = x
  shows continuous-on {f a..} g
proof (clarsimp simp add: continuous-on-eq-continuous-within)
  fix y
  assume f a ≤ y

```

**then obtain**  $u$  **where**  $u: y+1 = f u \ u \geq a$   
**by** (*smt (verit, best) atLeast-iff fim imageE*)  
**have** *continuous-on*  $\{f a..y+1\} g$   
**proof** –  
**obtain** *continuous-on*  $\{a..u\} f$  *strict-mono-on*  $\{a..u\} f$   
**using** *contf sm continuous-on-subset* **by** (*force simp add: strict-mono-on-def*)  
**moreover have** *continuous-on*  $(f \ ' \ \{a..u\}) g$   
**using** *assms continuous-on-subset*  
**by** (*intro continuous-on-inv*) *fastforce+*  
**ultimately show** *?thesis*  
**using** *strict-mono-image-endpoints* [*of - - f*]  
**by** (*simp add: strict-mono-image-endpoints u*)  
**qed**  
**then have**  $*$ : *continuous* (*at y within*  $\{f a..y+1\}) g$   
**by** (*simp add: <f a ≤ y> continuous-on-imp-continuous-within*)  
**show** *continuous* (*at y within*  $\{f a..\}) g$   
**proof** (*clarsimp simp add: continuous-within-topological Ball-def*)  
**fix**  $B$   
**assume** *open*  $B$  **and**  $g y \in B$   
**with**  $*$  **obtain**  $A$  **where**  $A$ : *open*  $A \ y \in A$  **and**  $\bigwedge x. f a \leq x \wedge x \leq y+1 \implies x \in A \longrightarrow g x \in B$   
**by** (*force simp: continuous-within-topological*)  
**then have**  $\forall x \geq f a. x \in A \cap \text{ball } y \ 1 \longrightarrow g x \in B$   
**by** (*smt (verit, ccfv-threshold) IntE dist-norm mem-ball real-norm-def*)  
**then show**  $\exists A. \text{open } A \wedge y \in A \wedge (\forall x \geq f a. x \in A \longrightarrow g x \in B)$   
**by** (*metis Elementary-Metric-Spaces.open-ball Int-iff A centre-in-ball open-Int zero-less-one*)  
**qed**  
**qed**

### 1.3 Regular divisions

Our lack of the Riemann integral forces us to construct explicitly the step functions mentioned in the text.

**definition** *segment*  $\equiv \lambda n k. \{\text{real } k / \text{real } n..(1 + k) / \text{real } n\}$

**lemma** *segment-nonempty*: *segment*  $n \ k \neq \{\}$   
**by** (*auto simp: segment-def divide-simps*)

**lemma** *segment-Suc*: *segment*  $n \ ' \ \{..<\text{Suc } k\} = \text{insert } \{k / n..(1 + \text{real } k) / n\}$   
(*segment*  $n \ ' \ \{..<k\}$ )  
**by** (*simp add: segment-def lessThan-Suc*)

**lemma** *Union-segment-image*:  $\bigcup (\text{segment } n \ ' \ \{..<k\}) = (\text{if } k=0 \text{ then } \{\} \text{ else } \{0.. \text{real } k / \text{real } n\})$

**proof** (*induction k*)

**case** (*Suc k*)

**then show** *?case*

**by** (*simp add: divide-simps segment-Suc Un-commute ivl-disj-un-two-touch split*)

*if-split-asm*)

**qed** (*auto simp: segment-def*)

**definition** *segments*  $\equiv \lambda n. \text{segment } n \text{ ' } \{..<n\}$

**lemma** *card-segments* [*simp*]:  $\text{card } (\text{segments } n) = n$

**by** (*simp add: segments-def segment-def card-image divide-right-mono inj-on-def*)

**lemma** *segments-0* [*simp*]:  $\text{segments } 0 = \{\}$

**by** (*simp add: segments-def*)

**lemma** *Union-segments*:  $\bigcup (\text{segments } n) = (\text{if } n=0 \text{ then } \{\} \text{ else } \{0..1\})$

**by** (*simp add: segments-def Union-segment-image*)

**definition** *regular-division*  $\equiv \lambda a \ b \ n. (\text{image } ((+) \ a \ \circ \ (*) \ (b-a))) \text{ ' } (\text{segments } n)$

**lemma** *translate-scale-01*:

**assumes**  $a \leq b$

**shows**  $(\lambda x. a + (b - a) * x) \text{ ' } \{0..1\} = \{a..b::\text{real}\}$

**using** *closed-segment-real-eq* [*of a b*] *assms closed-segment-eq-real-ivl* **by** *auto*

**lemma** *finite-regular-division* [*simp*]:  $\text{finite } (\text{regular-division } a \ b \ n)$

**by** (*simp add: regular-division-def segments-def*)

**lemma** *card-regular-division* [*simp*]:

**assumes**  $a < b$

**shows**  $\text{card } (\text{regular-division } a \ b \ n) = n$

**proof** –

**have** *inj-on*  $((\text{'}) \ ((+) \ a \ \circ \ (*) \ (b - a))) \ (\text{segments } n)$

**proof**

**fix**  $x \ y$

**assume**  $((+) \ a \ \circ \ (*) \ (b - a)) \text{ ' } x = ((+) \ a \ \circ \ (*) \ (b - a)) \text{ ' } y$

**then have**  $(+) \ (-a) \text{ ' } ((+) \ a \ \circ \ (*) \ (b - a)) \text{ ' } x = (+) \ (-a) \text{ ' } ((+) \ a \ \circ \ (*) \ (b - a)) \text{ ' } y$

**by** *simp*

**then have**  $((*) \ (b - a)) \text{ ' } x = ((*) \ (b - a)) \text{ ' } y$

**by** (*simp add: image-comp*)

**then have**  $(*) \ (\text{inverse}(b - a)) \text{ ' } (*) \ (b - a) \text{ ' } x = (*) \ (\text{inverse}(b - a)) \text{ ' } (*) \ (b - a) \text{ ' } y$

**by** *simp*

**then show**  $x = y$

**using** *assms* **by** (*simp add: image-comp mult-ac*)

**qed**

**then show** *?thesis*

**by** (*metis card-image card-segments regular-division-def*)

**qed**

**lemma** *Union-regular-division*:

**assumes**  $a \leq b$



**shows**  $\bigcup (\text{regular-division } a \ b \ n) = (\text{if } n=0 \text{ then } \{\} \text{ else } \{a..b\})$   
**using** *assms*  
**by** (*auto simp: regular-division-def Union-segments translate-scale-01 simp flip: image-Union*)

**lemma** *regular-division-eqI*:

**assumes**  $K: K = \{a + (b-a) * (\text{real } k / n) .. a + (b-a) * ((1 + \text{real } k) / n)\}$   
**and**  $a < b \ k < n$   
**shows**  $K \in \text{regular-division } a \ b \ n$   
**unfolding** *regular-division-def segments-def image-comp*  
**proof**  
**have**  $K = (\lambda x. (b-a) * x + a) \ ` \ \{ \text{real } k / \text{real } n .. (1 + \text{real } k) / \text{real } n \}$   
**using**  $K \ \langle a < b \rangle$  **by** (*simp add: image-affinity-atLeastAtMost divide-simps*)  
**then show**  $K = ((\cdot) ((+) a \circ (*) (b - a)) \circ \text{segment } n) \ k$   
**by** (*simp add: segment-def add.commute*)  
**qed** (*use assms in auto*)

**lemma** *regular-divisionE*:

**assumes**  $K \in \text{regular-division } a \ b \ n \ a < b$   
**obtains**  $k$  **where**  $k < n \ K = \{a + (b-a) * (\text{real } k / n) .. a + (b-a) * ((1 + \text{real } k) / n)\}$   
**proof** –  
**have**  $\text{eq}: (\lambda x. a + (b - a) * x) = (\lambda x. a + x) \circ (\lambda x. (b - a) * x)$   
**by** (*simp add: o-def*)  
**obtain**  $k$  **where**  $k < n \ K = ((\lambda x. a+x) \circ (\lambda x. (b-a) * x)) \ ` \ \{ k/n .. (1 + \text{real } k) / n \}$   
**using** *assms* **by** (*auto simp: regular-division-def segments-def segment-def*)  
**with** *that*  $\langle a < b \rangle$  **show** *?thesis*  
**unfolding** *image-comp [symmetric]* **by** *auto*  
**qed**

**lemma** *regular-division-division-of*:

**assumes**  $a < b \ n > 0$   
**shows**  $(\text{regular-division } a \ b \ n) \ \text{division-of} \ \{a..b\}$   
**proof** (*rule division-ofI*)  
**show** *finite*  $(\text{regular-division } a \ b \ n)$   
**by** (*simp add: regular-division-def segments-def*)  
**show**  $\S: \bigcup (\text{regular-division } a \ b \ n) = \{a..b\}$   
**using** *Union-regular-division assms* **by** *simp*  
**fix**  $K$   
**assume**  $K: K \in \text{regular-division } a \ b \ n$   
**then obtain**  $k$  **where**  $K \text{eq}: K = \{a + (b-a) * (k/n) .. a + (b-a) * ((1 + \text{real } k) / n)\}$   
**using**  $\langle a < b \rangle$  *regular-divisionE* **by** *meson*  
**show**  $K \subseteq \{a..b\}$   
**using**  $K$  *Union-regular-division*  $\langle n > 0 \rangle$  **by** (*metis Union-upper*  $\S$ )  
**show**  $K \neq \{\}$   
**using**  $K$  **by** (*auto simp: regular-division-def segment-nonempty segments-def*)  
**show**  $\exists a \ b. K = \text{cbox } a \ b$

```

  by (metis K ‹a<b› box-real(2) regular-divisionE)
fix K'
assume K': K' ∈ regular-division a b n and K ≠ K'
then obtain k' where Keq': K' = {a + (b-a)*(k'/n) .. a + (b-a)*((1 + real
k') / n)}
  using K ‹a<b› regular-divisionE by meson
consider 1 + real k ≤ k' | 1 + real k' ≤ k
  using Keq Keq' ‹K ≠ K'› by force
then show interior K ∩ interior K' = {}
proof cases
  case 1
  then show ?thesis
    by (simp add: Keq Keq' min-def max-def divide-right-mono assms)
next
  case 2
  then have interior K' ∩ interior K = {}
    by (simp add: Keq Keq' min-def max-def divide-right-mono assms)
  then show ?thesis
    by (simp add: inf-commute)
qed
qed

```

## 1.4 Special cases of Young's inequality

**lemma** *weighted-nesting-sum*:

```

fixes g :: nat ⇒ 'a::comm-ring-1
shows (∑ k<n. (1 + of-nat k) * (g (Suc k) - g k)) = of-nat n * g n - (∑ i<n.
g i)
by (induction n) (auto simp: algebra-simps)

```

**theorem** *Youngs-exact*:

```

fixes f :: real ⇒ real
assumes sm: strict-mono-on {0..} f and cont: continuous-on {0..} f and a:
a ≥ 0
and f: f 0 = 0 f a = b
and g: ∧x. [0 ≤ x; x ≤ a] ⇒ g (f x) = x
shows a*b = integral {0..a} f + integral {0..b} g
proof (cases a=0)
  case False
  with ‹a ≥ 0› have a > 0 by linarith
  then have b ≥ 0
    by (smt (verit, best) atLeast-iff f sm strict-mono-onD)
  have sm-0a: strict-mono-on {0..a} f
    by (metis atLeastAtMost-iff atLeast-iff sm strict-mono-on-def)
  have cont-0a: continuous-on {0..a} f
    using cont continuous-on-subset by fastforce
  with sm-0a have continuous-on {0..b} g
    by (metis a atLeastAtMost-iff compact-Icc continuous-on-inv f g strict-mono-image-endpoints)
  then have intgb-g: g integrable-on {0..b}

```

```

using integrable-continuous-interval by blast
have intgb-f:  $f$  integrable-on  $\{0..a\}$ 
using cont-0a integrable-continuous-real by blast

have f-iff [simp]:  $f x < f y \longleftrightarrow x < y$   $f x \leq f y \longleftrightarrow x \leq y$ 
if  $x \geq 0$   $y \geq 0$  for  $x y$ 
using that by (smt (verit, best) atLeast-iff sm strict-mono-onD)+
have fim:  $f \text{ ` } \{0..a\} = \{0..b\}$ 
by (simp add:  $\langle a \geq 0 \rangle$  cont-0a strict-mono-image-endpoints strict-mono-on-def
f)
have uniformly-continuous-on  $\{0..a\}$   $f$ 
using compact-uniformly-continuous cont-0a by blast
then obtain del where del-gt0:  $\bigwedge e. e > 0 \implies \text{del } e > 0$ 
and del:  $\bigwedge e x x'. [|x'-x| < \text{del } e; e > 0; x \in \{0..a\}; x' \in \{0..a\}] \implies |f x' - f x| < e$ 
unfolding uniformly-continuous-on-def dist-real-def by metis

have *:  $|a * b - \text{integral } \{0..a\} f - \text{integral } \{0..b\} g| < 2 * \varepsilon$  if  $\varepsilon > 0$  for  $\varepsilon$ 
proof -
define  $\delta$  where  $\delta = \min a (\text{del } (\varepsilon/a)) / 2$ 
have  $\delta > 0$   $\delta \leq a$ 
using  $\langle a > 0 \rangle \langle \varepsilon > 0 \rangle$  del-gt0 by (auto simp:  $\delta$ -def)
define  $n$  where  $n \equiv \text{nat} \lfloor a / \delta \rfloor$ 
define a-seg where a-seg  $\equiv \lambda u::\text{real}. u * a/n$ 
have  $n > 0$ 
using  $\langle a > 0 \rangle \langle \delta > 0 \rangle \langle \delta \leq a \rangle$  by (simp add: n-def)
have a-seg-ge-0 [simp]: a-seg  $x \geq 0 \longleftrightarrow x \geq 0$ 
and a-seg-le-a [simp]: a-seg  $x \leq a \longleftrightarrow x \leq n$  for  $x$ 
using  $\langle n > 0 \rangle \langle a > 0 \rangle$  by (auto simp: a-seg-def zero-le-mult-iff divide-simps)
have a-seg-le-iff [simp]: a-seg  $x \leq a$   $\longleftrightarrow x \leq y$ 
and a-seg-less-iff [simp]: a-seg  $x < a$   $\longleftrightarrow x < y$  for  $x y$ 
using  $\langle n > 0 \rangle \langle a > 0 \rangle$  by (auto simp: a-seg-def zero-le-mult-iff divide-simps)
have strict-mono a-seg
by (simp add: strict-mono-def)
have a-seg-eq-a-iff: a-seg  $x = a \longleftrightarrow x = n$  for  $x$ 
using  $\langle 0 < n \rangle \langle a > 0 \rangle$  by (simp add: a-seg-def nonzero-divide-eq-eq)
have fa-eq-b:  $f (a\text{-seg } n) = b$ 
using a-seg-eq-a-iff  $f$  by fastforce

have  $a/d < \text{real-of-int } \lfloor a * 2 / \min a d \rfloor$  if  $d > 0$  for  $d$ 
by (smt (verit)  $\langle 0 < \delta \rangle \langle \delta \leq a \rangle$  add-divide-distrib divide-less-eq-1-pos floor-eq-iff
that)
then have an-less-del:  $a/n < \text{del } (\varepsilon/a)$ 
using  $\langle a > 0 \rangle \langle \varepsilon > 0 \rangle$  del-gt0 by (simp add: n-def  $\delta$ -def field-simps)

define lower where lower  $\equiv \lambda x. a\text{-seg} \lfloor (\text{real } n * x) / a \rfloor$ 
define f1 where f1  $\equiv f \circ \text{lower}$ 
have f1-lower: f1  $x \leq f x$  if  $0 \leq x$   $x \leq a$  for  $x$ 
proof -

```

```

have lower  $x \leq x$ 
  using  $\langle n > 0 \rangle$  floor-divide-lower [OF  $\langle a > 0 \rangle$ ]
  by (auto simp: lower-def a-seg-def field-simps)
moreover have lower  $x \geq 0$ 
  unfolding lower-def using  $\langle n > 0 \rangle \langle a \geq 0 \rangle \langle 0 \leq x \rangle$  by force
ultimately show ?thesis
  using sm strict-mono-on-leD by (fastforce simp add: f1-def)
qed
define upper where upper  $\equiv \lambda x. a\text{-seg}[real\ n * x / a]$ 
define f2 where f2  $\equiv f \circ upper$ 
have f2-upper: f2  $x \geq f\ x$  if  $0 \leq x$   $x \leq a$  for  $x$ 
proof -
  have  $x \leq upper\ x$ 
    using  $\langle n > 0 \rangle$  ceiling-divide-upper [OF  $\langle a > 0 \rangle$ ] by (simp add: upper-def
a-seg-def field-simps)
  then show ?thesis
    using sm strict-mono-on-leD  $\langle 0 \leq x \rangle$  by (force simp: f2-def)
qed
let ?D = regular-division 0 a n
have div: ?D division-of  $\{0..a\}$ 
  using  $\langle a > 0 \rangle \langle n > 0 \rangle$  regular-division-division-of zero-less-nat-eq by pres-
burger

have int-f1-D: (f1 has-integral  $f(\text{Inf } K) * (a/n)$ )  $K$ 
  and int-f2-D: (f2 has-integral  $f(\text{Sup } K) * (a/n)$ )  $K$  and less:  $|f(\text{Sup } K) -$ 
 $f(\text{Inf } K)| < \varepsilon/a$ 
  if  $K \in ?D$  for  $K$ 
proof -
  from regular-divisionE [OF that]  $\langle a > 0 \rangle$ 
  obtain  $k$  where  $k < n$  and  $k: K = \{a\text{-seg}(real\ k)..a\text{-seg}(\text{Suc } k)\}$ 
  by (auto simp: a-seg-def mult.commute)
  define  $u$  where  $u \equiv a\text{-seg } k$ 
  define  $v$  where  $v \equiv a\text{-seg } (\text{Suc } k)$ 
  have  $u < v$   $0 \leq u$   $0 \leq v$   $u \leq a$   $v \leq a$  and  $Kuv: K = \{u..v\}$ 
  using  $\langle n > 0 \rangle \langle k < n \rangle \langle a > 0 \rangle$  by (auto simp: k u-def v-def divide-simps)
  have  $\text{Inf } K: \text{Inf } K = u$  and  $\text{Sup } K: \text{Sup } K = v$ 
  using  $Kuv$   $\langle u < v \rangle$  apply force
  using  $\langle n > 0 \rangle \langle a > 0 \rangle$  by (auto simp: divide-right-mono k u-def v-def)
  have f1: f1  $x = f(\text{Inf } K)$  if  $x \in K - \{v\}$  for  $x$ 
  proof -
    have  $x \in \{u..<v\}$ 
      using that  $Kuv$  atLeastLessThan-eq-atLeastAtMost-diff by blast
    then have  $\lfloor real\text{-of-int } n * x / a \rfloor = \text{int } k$ 
      using  $\langle n > 0 \rangle \langle a > 0 \rangle$  by (simp add: field-simps u-def v-def a-seg-def
floor-eq-iff)
    then show ?thesis
      by (simp add: InfK f1-def lower-def a-seg-def mult.commute u-def)
  qed
  have  $((\lambda x. f(\text{Inf } K)) \text{ has-integral } (f(\text{Inf } K) * (a/n))) K$ 

```

```

using has-integral-const-real [of  $f$  ( $\text{Inf } K$ )  $u$   $v$ ]
   $\langle n > 0 \rangle \langle a > 0 \rangle$  by (simp add: Kuv field-simps a-seg-def u-def v-def)
then show ( $f1$  has-integral ( $f$  ( $\text{Inf } K$ ) * ( $a/n$ )))  $K$ 
  using has-integral-spike-finite-eq [of  $\{v\}$   $K$   $\lambda x. f$  ( $\text{Inf } K$ )  $f1$ ]  $f1$  by simp
have  $f2$ :  $f2\ x = f$  ( $\text{Sup } K$ ) if  $x \in K - \{u\}$  for  $x$ 
proof –
  have  $x \in \{u <..v\}$ 
    using that Kuv greaterThanAtMost-eq-atLeastAtMost-diff by blast
  then have  $\lceil x * \text{real-of-int } n / a \rceil = 1 + \text{int } k$ 
    using  $\langle n > 0 \rangle \langle a > 0 \rangle$  by (simp add: field-simps u-def v-def a-seg-def
ceiling-eq-iff)
  then show ?thesis
    by (simp add: mult.commute f2-def upper-def a-seg-def SupK v-def)
qed
have ( $\lambda x. f$  ( $\text{Sup } K$ )) has-integral ( $f$  ( $\text{Sup } K$ ) * ( $a/n$ )))  $K$ 
  using  $\langle n > 0 \rangle \langle a > 0 \rangle$  has-integral-const-real [of  $f$  ( $\text{Sup } K$ )  $u$   $v$ ]
  by (simp add: Kuv field-simps u-def v-def a-seg-def)
then show ( $f2$  has-integral ( $f$  ( $\text{Sup } K$ ) * ( $a/n$ )))  $K$ 
  using has-integral-spike-finite-eq [of  $\{u\}$   $K$   $\lambda x. f$  ( $\text{Sup } K$ )  $f2$ ]  $f2$  by simp
have  $|v - u| < \text{del } (\varepsilon/a)$ 
  using  $\langle n > 0 \rangle \langle a > 0 \rangle$  by (simp add: v-def u-def a-seg-def field-simps
an-less-del)
  then have  $|f\ v - f\ u| < \varepsilon/a$ 
    using  $\langle \varepsilon > 0 \rangle \langle a > 0 \rangle \langle 0 \leq u \rangle \langle u \leq a \rangle \langle 0 \leq v \rangle \langle v \leq a \rangle$ 
    by (intro del) auto
  then show  $|f(\text{Sup } K) - f(\text{Inf } K)| < \varepsilon/a$ 
    using InfK SupK by blast
qed

have int-21-D: ( $\lambda x. f2\ x - f1\ x$ ) has-integral ( $f(\text{Sup } K) - f(\text{Inf } K)$ ) * ( $a/n$ ))
 $K$  if  $K \in ?\mathcal{D}$  for  $K$ 
  using that has-integral-diff [OF int-f2-D int-f1-D] by (simp add: algebra-simps)

have D-ne:  $?\mathcal{D} \neq \{\}$ 
  by (metis  $\langle 0 < a \rangle \langle n > 0 \rangle$  card-gt-0-iff card-regular-division)
have  $f12$ : ( $\lambda x. f2\ x - f1\ x$ ) has-integral ( $\sum K \in ?\mathcal{D}. (f(\text{Sup } K) - f(\text{Inf } K)) *$ 
 $(a/n)$ ))  $\{0..a\}$ 
  by (intro div int-21-D has-integral-combine-division)
moreover have ( $\sum K \in ?\mathcal{D}. (f(\text{Sup } K) - f(\text{Inf } K)) * (a/n)$ )  $< \varepsilon$ 
proof –
  have ( $\sum K \in ?\mathcal{D}. (f(\text{Sup } K) - f(\text{Inf } K)) * (a/n)$ )  $\leq$  ( $\sum K \in ?\mathcal{D}. |f(\text{Sup } K) -$ 
 $f(\text{Inf } K)| * (a/n)$ )
    using  $\langle n > 0 \rangle \langle a > 0 \rangle$ 
    by (smt (verit) divide-pos-pos of-nat-0-less-iff sum-mono zero-le-mult-iff)
  also have  $\dots < (\sum K \in ?\mathcal{D}. \varepsilon/n)$ 
    using  $\langle n > 0 \rangle \langle a > 0 \rangle$  less
  by (intro sum-strict-mono finite-regular-division D-ne) (simp add: field-simps)
  also have  $\dots = \varepsilon$ 
    using  $\langle n > 0 \rangle \langle a > 0 \rangle$  by simp

```

**finally show** *?thesis* .  
**qed**  
**ultimately have** *f2-near-f1*:  $\text{integral } \{0..a\} (\lambda x. f2\ x - f1\ x) < \varepsilon$   
**by** (*simp add: integral-unique*)

**define** *yidx* **where**  $yidx \equiv \lambda y. \text{LEAST } k. y < f\ (a\text{-seg}\ (Suc\ k))$   
**have** *fa-yidx-le*:  $f\ (a\text{-seg}\ (yidx\ y)) \leq y$  **and** *yidx-gt*:  $y < f\ (a\text{-seg}\ (Suc\ (yidx\ y)))$   
**if**  $y \in \{0..b\}$  **for** *y*  
**proof** –  
**obtain** *x* **where**  $x: f\ x = y\ x \in \{0..a\}$   
**using** *Topological-Spaces.IVT'* [*OF - - cont-0a*] *assms*  
**by** (*metis*  $\langle y \in \{0..b\} \rangle$  *atLeastAtMost-iff*)  
**define** *k* **where**  $k \equiv \text{nat } \lfloor x/a * n \rfloor$   
**have** *x-lims*:  $a\text{-seg}\ k \leq x\ x < a\text{-seg}\ (Suc\ k)$   
**using**  $\langle n > 0 \rangle \langle 0 < a \rangle$  *floor-divide-lower floor-divide-upper* [*of a x\*n*] *x*  
**by** (*auto simp: k-def a-seg-def field-simps*)  
**with that** *x* **obtain** *f-lims*:  $f\ (a\text{-seg}\ k) \leq y\ y < f\ (a\text{-seg}\ (Suc\ k))$   
**using** *strict-mono-onD* [*OF sm*] **by** *force*  
**then have**  $a\text{-seg}\ (yidx\ y) \leq a\text{-seg}\ k$   
**by** (*simp add: Least-le*  $\langle \text{strict-mono } a\text{-seg} \rangle$  *strict-mono-less-eq yidx-def*)  
**then have**  $f\ (a\text{-seg}\ (yidx\ y)) \leq f\ (a\text{-seg}\ k)$   
**using** *strict-mono-onD* [*OF sm*] **by** *simp*  
**then show**  $f\ (a\text{-seg}\ (yidx\ y)) \leq y$   
**using** *f-lims* **by** *linarith*  
**show**  $y < f\ (a\text{-seg}\ (Suc\ (yidx\ y)))$   
**by** (*metis LeastI f-lims(2) yidx-def*)  
**qed**

**have** *yidx-equality*:  $yidx\ y = k$  **if**  $y \in \{0..b\}\ y \in \{f\ (a\text{-seg}\ k)..<f\ (a\text{-seg}\ (Suc\ k))\}$  **for** *y k*  
**proof** (*rule antisym*)  
**show**  $yidx\ y \leq k$   
**unfolding** *yidx-def* **by** (*metis atLeastLessThan-iff that(2) Least-le*)  
**have**  $(a\text{-seg}\ (\text{real } k)) < a\text{-seg}\ (1 + \text{real}\ (yidx\ y))$   
**using** *yidx-gt* [*OF that(1)*] *that(2) strict-mono-onD* [*OF sm*] *order-le-less-trans*  
**by** *fastforce*  
**then have**  $\text{real } k < 1 + \text{real}\ (yidx\ y)$   
**by** (*simp add:*  $\langle \text{strict-mono } a\text{-seg} \rangle$  *strict-mono-less*)  
**then show**  $k \leq yidx\ y$   
**by** *simp*  
**qed**

**have**  $yidx\ b = n$   
**proof** –  
**have**  $a < (1 + \text{real } n) * a / \text{real } n$   
**using**  $\langle 0 < n \rangle \langle 0 < a \rangle$  **by** (*simp add: divide-simps*)  
**then have**  $b < f\ (a\text{-seg}\ (1 + \text{real } n))$   
**using** *f*  $\langle a \geq 0 \rangle$  *a-seg-def sm strict-mono-onD* **by** *fastforce*  
**then show** *?thesis*

```

    using ⟨0 ≤ b⟩ by (auto simp: f a-seg-def yidx-equality)
  qed
  moreover have yidx-less-n: yidx y < n if y < b for y
    by (metis ⟨0 < n⟩ fa-eq-b gr0-conv-Suc less-Suc-eq-le that Least-le yidx-def)
  ultimately have yidx-le-n: yidx y ≤ n if y ≤ b for y
    by (metis dual-order.order-iff-strict that)

  have zero-to-b-eq: {0..b} = (⋃ k<n. {f(a-seg k)..f(a-seg (Suc k))}) (is ?lhs =
  ?rhs)
  proof
    show ?lhs ⊆ ?rhs
    proof
      fix y assume y: y ∈ {0..b}
      have fn: f (a-seg n) = b
        using a-seg-eq-a-iff ⟨f a = b⟩ by fastforce
      show y ∈ ?rhs
      proof (cases y=b)
        case True
          with fn ⟨n>0⟩ show ?thesis
            by (rule-tac a=n-1 in UN-I) auto
        next
          case False
            with y show ?thesis
              apply (simp add: subset-iff Bex-def)
              by (metis atLeastAtMost-iff of-nat-Suc order-le-less yidx-gt fa-yidx-le
  yidx-less-n)
      qed
    qed
    show ?rhs ⊆ ?lhs
    apply clarsimp
    by (smt (verit, best) a-seg-ge-0 a-seg-le-a f f-iff(2) nat-less-real-le of-nat-0-le-iff)
  qed

  define g1 where g1 ≡ λy. if y=b then a else a-seg (Suc (yidx y))
  define g2 where g2 ≡ λy. if y=0 then 0 else a-seg (yidx y)
  have g1: g1 y ∈ {0..a} if y ∈ {0..b} for y
    using that ⟨a > 0⟩ yidx-less-n [of y] by (auto simp: g1-def a-seg-def di-
  vide-simps)
  have g2: g2 y ∈ {0..a} if y ∈ {0..b} for y
    using that ⟨a > 0⟩ yidx-le-n [of y] by (simp add: g2-def a-seg-def divide-simps)

  have g2-le-g: g2 y ≤ g y if y ∈ {0..b} for y
  proof -
    have f (g2 y) ≤ y
      using ⟨f 0 = 0⟩ g2-def that fa-yidx-le by presburger
    then have f (g2 y) ≤ f (g y)
      using that g by (smt (verit, best) atLeastAtMost-iff fim image-iff)
    then show ?thesis
      by (smt (verit, best) atLeastAtMost-iff fim g g2 imageE sm-0a strict-mono-onD

```

that)

qed

have  $g\text{-le-}g1: g\ y \leq g1\ y$  if  $y \in \{0..b\}$  for  $y$

proof –

  have  $y \leq f\ (g1\ y)$

  by (smt (verit, best)  $\langle f\ a = b \rangle$   $g1\text{-def that yidx-gt}$ )

  then have  $f\ (g\ y) \leq f\ (g1\ y)$

  using that  $g$  by (smt (verit, best)  $atLeastAtMost\text{-iff fim image-iff}$ )

  then show ?thesis

  by (smt (verit, ccfv-threshold)  $atLeastAtMost\text{-iff f-iff}(1)\ g1$  that)

qed

define  $DN$  where  $DN \equiv \lambda K. \text{nat } [\text{Inf } K * \text{real } n / a]$

have [simp]:  $DN\ \{a * \text{real } k / n..a * (1 + \text{real } k) / n\} = k$  for  $k$

  using  $\langle n > 0 \rangle \langle a > 0 \rangle$  by (simp add:  $DN\text{-def divide-simps}$ )

have  $DN: \text{bij-betw } DN\ ?\mathcal{D}\ \{..<n\}$

proof (intro  $\text{bij-betw-imageI}$ )

  show  $\text{inj-on } DN\ (\text{regular-division } 0\ a\ n)$

  proof

    fix  $K\ K'$

    assume  $K \in \text{regular-division } 0\ a\ n$

    with  $\langle a > 0 \rangle$  obtain  $k$  where  $k: K = \{a * (\text{real } k / n) .. a * (1 + \text{real } k) / n\}$

      by (force elim:  $\text{regular-divisionE}$ )

    assume  $K' \in \text{regular-division } 0\ a\ n$

    with  $\langle a > 0 \rangle$  obtain  $k'$  where  $k': K' = \{a * (\text{real } k' / n) .. a * (1 + \text{real } k') / n\}$

      by (force elim:  $\text{regular-divisionE}$ )

    assume  $DN\ K = DN\ K'$

    then show  $K = K'$  by (simp add:  $k\ k'$ )

  qed

  have  $\exists K \in \text{regular-division } 0\ a\ n. k = \text{nat } [\text{Inf } K * \text{real } n / a]$  if  $k < n$  for  $k$

  using  $\langle n > 0 \rangle \langle a > 0 \rangle$  that

  by (force simp:  $\text{divide-simps intro: regular-division-eqI [OF refl]}$ )

  with  $\langle a > 0 \rangle$  show  $DN\ \text{regular-division } 0\ a\ n = \{..<n\}$

  by (auto simp:  $DN\text{-def bij-betw-def image-iff frac-le elim!: regular-divisionE}$ )

qed

have  $\text{int-f1}: (f1\ \text{has-integral } (\sum k < n. f(a\text{-seg } k)) * (a/n))\ \{0..a\}$

proof –

  have  $a\text{-seg } (\text{real } (DN\ K)) = \text{Inf } K$  if  $K \in ?\mathcal{D}$  for  $K$

  using that  $\langle a > 0 \rangle$  by (auto simp:  $DN\text{-def field-simps a-seg-def elim: regular-divisionE}$ )

  then have  $(\sum K \in ?\mathcal{D}. f(\text{Inf } K) * (a/n)) = (\sum k < n. (f(a\text{-seg } k)) * (a/n))$

  by (simp flip:  $\text{sum.reindex-bij-betw [OF DN]}$ )

  moreover have  $(f1\ \text{has-integral } (\sum K \in ?\mathcal{D}. f(\text{Inf } K) * (a/n)))\ \{0..a\}$

  by (intro  $\text{div int-f1-D has-integral-combine-division}$ )

  ultimately show ?thesis

  by (metis  $\text{sum-distrib-right}$ )



qed

The claim ( $f_2$  has-integral  $(\sum k < n. f (a\text{-seg } (\text{real } (\text{Suc } k)))) * (a / \text{real } n)$ )  $\{0..a\}$  can similarly be proved

have *int-g1-D*: ( $g_1$  has-integral  $a\text{-seg } (\text{Suc } k) * (f (a\text{-seg } (\text{Suc } k)) - f (a\text{-seg } k))$ )

$\{f(a\text{-seg } k)..f(a\text{-seg } (\text{Suc } k))\}$

and *int-g2-D*: ( $g_2$  has-integral  $a\text{-seg } k * (f (a\text{-seg } (\text{Suc } k)) - f (a\text{-seg } k))$ )

$\{f(a\text{-seg } k)..f(a\text{-seg } (\text{Suc } k))\}$

if  $k < n$  for  $k$

proof –

define  $u$  where  $u \equiv f (a\text{-seg } k)$

define  $v$  where  $v \equiv f (a\text{-seg } (\text{Suc } k))$

obtain  $u < v$   $0 \leq u$   $0 \leq v$

unfolding *u-def v-def assms*

by (*smt (verit, best) a-seg-ge-0 a-seg-le-iff f(1) f-iff(1) of-nat-0-le-iff of-nat-Suc*)

have  $u \leq b$   $v \leq b$

using  $\langle k < n \rangle \langle a \geq 0 \rangle$  by (*simp-all add: u-def v-def flip: \langle f a = b \rangle*)

have *yidx-eq*:  $yidx\ x = k$  if  $x \in \{u..<v\}$  for  $x$

using  $\langle 0 \leq u \rangle \langle v \leq b \rangle$  that *u-def v-def yidx-equality* by *auto*

have  $g_1\ x = a\text{-seg } (\text{Suc } k)$  if  $x \in \{u..<v\}$  for  $x$

using that  $\langle v \leq b \rangle$  by (*simp add: g1-def yidx-eq*)

moreover have  $((\lambda x. a\text{-seg } (\text{Suc } k))$  has-integral  $(a\text{-seg } (\text{Suc } k) * (v-u))$ )  $\{u..v\}$

using *has-integral-const-real*  $\langle u < v \rangle$

by (*metis content-real-if less-eq-real-def mult.commute real-scaleR-def*)

ultimately show  $(g_1$  has-integral  $(a\text{-seg } (\text{Suc } k) * (v-u))$ )  $\{u..v\}$

using *has-integral-spike-finite-eq* [*of*  $\{v\}$   $\{u..v\}$   $\lambda x. a\text{-seg } (\text{Suc } k)$   $g_1$ ] by

*simp*

have  $g_2: g_2\ x = a\text{-seg } k$  if  $x \in \{u<..<v\}$  for  $x$

using that  $\langle 0 \leq u \rangle$  by (*simp add: g2-def yidx-eq*)

moreover have  $((\lambda x. a\text{-seg } k)$  has-integral  $(a\text{-seg } k * (v-u))$ )  $\{u..v\}$

using *has-integral-const-real*  $\langle u < v \rangle$

by (*metis content-real-if less-eq-real-def mult.commute real-scaleR-def*)

ultimately show  $(g_2$  has-integral  $(a\text{-seg } k * (v-u))$ )  $\{u..v\}$

using *has-integral-spike-finite-eq* [*of*  $\{u,v\}$   $\{u..v\}$   $\lambda x. a\text{-seg } k$   $g_2$ ] by *simp*

qed

have *int-g1*: ( $g_1$  has-integral  $(\sum k < n. a\text{-seg } (\text{Suc } k) * (f (a\text{-seg } (\text{Suc } k)) - f (a\text{-seg } k)))$ )  $\{0..b\}$

and *int-g2*: ( $g_2$  has-integral  $(\sum k < n. a\text{-seg } k * (f (a\text{-seg } (\text{Suc } k)) - f (a\text{-seg } k)))$ )  $\{0..b\}$

unfolding *zero-to-b-eq* using *int-g1-D int-g2-D*

by (*auto simp: min-def pairwise-def intro!:* has-integral-UN negligible-atLeastAtMostI)

have  $(\sum k < n. a\text{-seg } (\text{Suc } k) * (f (a\text{-seg } (\text{Suc } k)) - f (a\text{-seg } k)))$

$$= (\sum k < n. (Suc\ k) * (f\ (a-seg\ (Suc\ k)) - f\ (a-seg\ k))) * (a/n)$$
**unfolding** *a-seg-def sum-distrib-right sum-divide-distrib* **by** (*simp add: mult-ac*)  
**also have**  $\dots = (n * f\ (a-seg\ n) - (\sum k < n. f\ (a-seg\ k))) * a / n$   
**using** *weighted-nesting-sum* [**where**  $g = f \circ a-seg$ ] **by** *simp*  
**also have**  $\dots = a * b - (\sum k < n. f\ (a-seg\ k)) * a / n$   
**using**  $\langle n > 0 \rangle$  **by** (*simp add: fa-eq-b field-simps*)  
**finally have** *int-g1'*: (*g1 has-integral*  $a * b - (\sum k < n. f\ (a-seg\ k)) * a / n$ )  
 $\{0..b\}$   
**using** *int-g1* **by** *simp*

The claim (*g2 has-integral*  $a * b - (\sum k < n. f\ (a-seg\ (real\ (Suc\ k)))) * a / real\ n$ )  $\{0..b\}$  can similarly be proved.

**have** *a-seg-diff*:  $a-seg\ (Suc\ k) - a-seg\ k = a/n$  **for**  $k$   
**by** (*simp add: a-seg-def field-split-simps*)  
**have** *f-a-seg-diff*:  $|f\ (a-seg\ (Suc\ k)) - f\ (a-seg\ k)| < \varepsilon/a$  **if**  $k < n$  **for**  $k$   
**using** *that*  $\langle a > 0 \rangle$  *a-seg-diff an-less-del*  $\langle \varepsilon > 0 \rangle$   
**by** (*intro del*) *auto*

**have** ( $(\lambda x. g1\ x - g2\ x)$  *has-integral*  $(\sum k < n. (f\ (a-seg\ (Suc\ k)) - f\ (a-seg\ k)) * (a/n))$ )  $\{0..b\}$

**using** *has-integral-diff* [*OF int-g1 int-g2*] *a-seg-diff*  
**apply** (*simp flip: sum-subtractf left-diff-distrib*)  
**apply** (*simp add: field-simps*)  
**done**

**moreover have**  $(\sum k < n. (f\ (a-seg\ (Suc\ k)) - f\ (a-seg\ k)) * (a/n)) < \varepsilon$

**proof** –

**have**  $(\sum k < n. (f\ (a-seg\ (Suc\ k)) - f\ (a-seg\ k)) * (a/n))$   
 $\leq (\sum k < n. |f\ (a-seg\ (Suc\ k)) - f\ (a-seg\ k)| * (a/n))$   
**by** *simp*

**also have**  $\dots < (\sum k < n. (\varepsilon/a) * (a/n))$

**proof** (*rule sum-strict-mono*)

**fix**  $k$  **assume**  $k \in \{..<n\}$

**with**  $\langle n > 0 \rangle$   $\langle a > 0 \rangle$  *divide-strict-right-mono f-a-seg-diff pos-less-divide-eq*

**show**  $|f\ (a-seg\ (Suc\ k)) - f\ (a-seg\ k)| * (a/n) < \varepsilon/a * (a/n)$  **by** *fastforce*

**qed** (*use*  $\langle n > 0 \rangle$  **in** *auto*)

**also have**  $\dots = \varepsilon$

**using**  $\langle n > 0 \rangle$   $\langle a > 0 \rangle$  **by** *simp*

**finally show** *?thesis* .

**qed**

**ultimately have** *g2-near-g1*: *integral*  $\{0..b\}$   $(\lambda x. g1\ x - g2\ x) < \varepsilon$

**by** (*simp add: integral-unique*)

**have** *ab1*: *integral*  $\{0..a\}$  *f1* + *integral*  $\{0..b\}$  *g1* =  $a*b$

**using** *int-f1 int-g1'* **by** (*simp add: integral-unique*)

**have** *integral*  $\{0..a\}$   $(\lambda x. f\ x - f1\ x) \leq$  *integral*  $\{0..a\}$   $(\lambda x. f2\ x - f1\ x)$

**proof** (*rule integral-le*)

**show**  $(\lambda x. f\ x - f1\ x)$  *integrable-on*  $\{0..a\}$   $(\lambda x. f2\ x - f1\ x)$  *integrable-on*  $\{0..a\}$

```

    using Henstock-Kurzweil-Integration.integrable-diff int-f1 intgb-f f12 by
blast+
  qed (auto simp: f2-upper)
  with f2-near-f1 have integral {0..a} (λx. f x - f1 x) < ε
    by simp
  moreover have integral {0..a} f1 ≤ integral {0..a} f
    by (intro integral-le has-integral-integral intgb-f has-integral-integrable [OF
int-f1])
    (simp add: f1-lower)
  ultimately have f-error: |integral {0..a} f - integral {0..a} f1| < ε
    using Henstock-Kurzweil-Integration.integrable-diff int-f1 intgb-f by fastforce

  have integral {0..b} (λx. g1 x - g x) ≤ integral {0..b} (λx. g1 x - g2 x)
  proof (rule integral-le)
    show (λx. g1 x - g x) integrable-on {0..b} (λx. g1 x - g2 x) integrable-on
{0..b}
      using Henstock-Kurzweil-Integration.integrable-diff int-g1 int-g2 intgb-g by
blast+
    qed (auto simp: g2-le-g)
    with g2-near-g1 have integral {0..b} (λx. g1 x - g x) < ε
      by simp
    moreover have integral {0..b} g ≤ integral {0..b} g1
      by (intro integral-le has-integral-integral intgb-g has-integral-integrable [OF
int-g1])
      (simp add: g-le-g1)
    ultimately have g-error: |integral {0..b} g1 - integral {0..b} g| < ε
      using integral-diff int-g1 intgb-g by fastforce
    show ?thesis
      using f-error g-error ab1 by linarith
  qed
  show ?thesis
    using * [of |a * b - integral {0..a} f - integral {0..b} g| / 2] by fastforce
qed (use assms in force)

```

**corollary** *Youngs-strict*:

```

  fixes f :: real ⇒ real
  assumes sm: strict-mono-on {0..} f and cont: continuous-on {0..} f and a>0
b≥0
    and f: f 0 = 0 f a ≠ b and fim: f ' {0..} = {0..}
    and g: ∧x. 0 ≤ x ⇒ g (f x) = x
  shows a*b < integral {0..a} f + integral {0..b} g
  proof -
    have f-iff [simp]: f x < f y ⇔ x < y f x ≤ f y ⇔ x ≤ y
      if x ≥ 0 y ≥ 0 for x y
    using that by (smt (verit, best) atLeast-iff sm strict-mono-onD)+
    let ?b' = f a
    have ?b' ≥ 0

```

```

  by (smt (verit, best) ⟨0 < a⟩ atLeast-iff f sm strict-mono-onD)
then have sm-gx: strict-mono-on {0..} g
  unfolding strict-mono-on-def
  by (smt (verit, best) atLeast-iff f-iff(1) f-inv-into-f fim g inv-into-into)
show ?thesis
proof (cases ?b' < b)
  case True
  have gt-a: a < g y if y ∈ {?b'<..b} for y
  proof -
    have a = g ?b'
      using ⟨a > 0⟩ g by force
    also have ... < g y
      using ⟨0 ≤ ?b'⟩ sm-gx strict-mono-onD that by fastforce
    finally show ?thesis .
  qed
  have continuous-on {0..} g
    by (metis cont f(1) fim g sm strict-mono-continuous-invD)
  then have contg: continuous-on {?b'..b} g
    by (meson Icc-subset-Ici-iff ⟨0 ≤ f a⟩ continuous-on-subset)
  have mono-on {0..} g
    by (simp add: sm-gx strict-mono-on-imp-mono-on)
  then have int-g0b: g integrable-on {0..b}
    by (simp add: integrable-on-mono-on mono-on-subset)
  then have int-gb'b: g integrable-on {?b'..b}
    by (simp add: ⟨0 ≤ ?b'⟩ integrable-on-subinterval)
  have a * (b - ?b') = integral {?b'..b} (λy. a)
    using True by force
  also have ... < integral {?b'..b} g
    using contg True gt-a by (intro integral-less-real) auto
  finally have *: a * (b - ?b') < integral {?b'..b} g .
  have a*b = a * ?b' + a * (b - ?b')
    by (simp add: algebra-simps)
  also have ... = integral {0..a} f + integral {0..?b'} g + a * (b - ?b')
    using Youngs-exact ⟨a > 0⟩ cont ⟨f 0 = 0⟩ g sm by force
  also have ... < integral {0..a} f + integral {0..?b'} g + integral {?b'..b} g
    by (simp add: *)
  also have ... = integral {0..a} f + integral {0..b} g
    by (smt (verit) Henstock-Kurzweil-Integration.integral-combine True ⟨0 ≤ ?b'⟩
int-g0b)
  finally show ?thesis .
next
  case False
  with f have b < ?b' by force
  obtain a' where f a' = b a' ≥ 0
    using fim ⟨b ≥ 0⟩ by force
  then have a' < a
    using ⟨b < f a⟩ ⟨a > 0⟩ by force
  have gt-b: b < f x if x ∈ {a'<..a} for x
    using ⟨0 ≤ a'⟩ ⟨f a' = b⟩ that by fastforce

```

```

have int-f0a: f integrable-on {0..a}
  by (simp add: integrable-on-mono-on mono-on-def)
then have int-fa'a: f integrable-on {a'..a}
  by (simp add: ⟨0 ≤ a'⟩ integrable-on-subinterval)
have cont-f': continuous-on {a'..a} f
  by (meson Icc-subset-Ici-iff ⟨0 ≤ a'⟩ cont continuous-on-subset)
have b * (a - a') = integral {a'..a} (λx. b)
  using ⟨a' < a⟩ by simp
also have ... < integral {a'..a} f
  using cont-f' ⟨a' < a⟩ gt-b by (intro integral-less-real) auto
finally have *: b * (a - a') < integral {a'..a} f .
have a*b = a' * b + b * (a - a')
  by (simp add: algebra-simps)
also have ... = integral {0..a'} f + integral {0..b} g + b * (a - a')
  by (simp add: Youngs-exact ⟨0 ≤ a'⟩ ⟨f a' = b⟩ cont f g sm)
also have ... < integral {0..a'} f + integral {0..b} g + integral {a'..a} f
  by (simp add: *)
also have ... = integral {0..a} f + integral {0..b} g
  by (smt (verit) Henstock-Kurzweil-Integration.integral-combine ⟨0 ≤ a'⟩ ⟨a'
< a⟩ int-f0a)
  finally show ?thesis .
qed
qed

```

**corollary** *Youngs-inequality:*

```

fixes f :: real ⇒ real
assumes sm: strict-mono-on {0..} f and cont: continuous-on {0..} f and a ≥ 0
b ≥ 0
  and f: f 0 = 0 and fim: f ' {0..} = {0..}
  and g: ∧x. 0 ≤ x ⇒ g (f x) = x
shows a*b ≤ integral {0..a} f + integral {0..b} g
proof (cases a=0)
  case True
    have g x ≥ 0 if x ≥ 0 for x
      by (metis atLeast-iff fim g imageE that)
    then have 0 ≤ integral {0..b} g
      by (metis Henstock-Kurzweil-Integration.integral-nonneg atLeastAtMost-iff
not-integrable-integral order-refl)
    then show ?thesis
      by (simp add: True)
  next
  case False
    then show ?thesis
      by (smt (verit) assms Youngs-exact Youngs-strict)
qed
end

```

## References

- [1] F. Cunningham and N. Grossman. On Young's inequality. *The American Mathematical Monthly*, 78(7):781–783, 1971.