

The Mason–Stothers theorem

Manuel Eberl

February 23, 2021

Abstract

This article provides a formalisation of Snyder’s simple and elegant proof of the Mason–Stothers theorem [2, 1], which is the polynomial analogue of the famous *abc* Conjecture for integers. Remarkably, Snyder found this very elegant proof when he was still a high-school student.

In short, the statement of the theorem is that three non-zero coprime polynomials A, B, C over a field which sum to 0 and do not all have vanishing derivatives fulfil $\max\{\deg(A), \deg(B), \deg(C)\} < \deg(\text{rad}(ABC))$ where $\text{rad}(P)$ denotes the *radical* of P , i. e. the product of all unique irreducible factors of P .

This theorem also implies a kind of polynomial analogue of Fermat’s Last Theorem for polynomials: except for trivial cases, $A^n + B^n + C^n = 0$ implies $n \leq 2$ for coprime polynomials A, B, C over a field.

Contents

1	The Mason–Stother’s Theorem	2
1.1	Auxiliary material	2
1.2	Definition of a radical	2
1.3	Main result	4

1 The Mason–Stoother’s Theorem

theory *Mason-Stoother*

imports

HOL-Computational-Algebra.Computational-Algebra

HOL-Computational-Algebra.Polynomial-Factorial

begin

1.1 Auxiliary material

hide-const (**open**) *Formal-Power-Series.radical*

lemma *degree-div:*

assumes $a \text{ dvd } b$

shows $\text{degree } (b \text{ div } a) = \text{degree } b - \text{degree } a$

using *assms* **by** (*cases* $a = 0$; *cases* $b = 0$) (*auto elim!*: *dvdE simp: degree-mult-eq*)

lemma *degree-pderiv-le:*

shows $\text{degree } (\text{pderiv } p) \leq \text{degree } p - 1$

by (*rule* *degree-le*, *cases* $\text{degree } p = 0$) (*auto simp: coeff-pderiv coeff-eq-0*)

lemma *degree-pderiv-less:*

assumes $\text{pderiv } p \neq 0$

shows $\text{degree } (\text{pderiv } p) < \text{degree } p$

proof –

have $\text{degree } (\text{pderiv } p) \leq \text{degree } p - 1$

by (*rule* *degree-pderiv-le*)

also have $\text{degree } p \neq 0$

using *assms* **by** (*auto intro!*: *Nat.gr0I elim!*: *degree-eq-zeroE*)

hence $\text{degree } p - 1 < \text{degree } p$ **by** *simp*

finally show *?thesis* .

qed

lemma *pderiv-eq-0:*

assumes $\text{degree } p = 0$

shows $\text{pderiv } p = 0$

using *assms* **by** (*auto elim!*: *degree-eq-zeroE*)

1.2 Definition of a radical

The following definition of a radical is generic for any factorial semiring.

context *factorial-semiring*

begin

definition *radical* :: $'a \Rightarrow 'a$ **where**

$\text{radical } x = (\text{if } x = 0 \text{ then } 0 \text{ else } \prod (\text{prime-factors } x))$

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

by (*simp add: radical-def*)

lemma *radical-nonzero*: $x \neq 0 \implies \text{radical } x = \prod (\text{prime-factors } x)$
by (*simp add: radical-def*)

lemma *radical-eq-0-iff* [*simp*]: $\text{radical } x = 0 \iff x = 0$
by (*auto simp: radical-def*)

lemma *prime-factorization-radical* [*simp*]:
assumes $x \neq 0$
shows $\text{prime-factorization } (\text{radical } x) = \text{mset-set } (\text{prime-factors } x)$
proof –
have $\text{prime-factorization } (\text{radical } x) = (\sum_{p \in \text{prime-factors } x} \text{prime-factorization } p)$
unfolding *radical-def* **using** *assms* **by** (*auto intro!: prime-factorization-prod*)
also have $\dots = (\sum_{p \in \text{prime-factors } x} \{ \#p \# \})$
by (*intro Groups-Big.sum.cong*) (*auto intro!: prime-factorization-prime*)
also have $\dots = \text{mset-set } (\text{prime-factors } x)$ **by** *simp*
finally show *?thesis* .
qed

lemma *prime-factors-radical* [*simp*]: $x \neq 0 \implies \text{prime-factors } (\text{radical } x) = \text{prime-factors } x$
by *simp*

lemma *radical-dvd* [*simp, intro*]: $\text{radical } x \text{ dvd } x$
by (*cases x = 0*) (*force intro: prime-factorization-subset-imp-dvd mset-set-set-mset-msubset*)+

lemma *multiplicity-radical-prime*:
assumes $\text{prime } p \ x \neq 0$
shows $\text{multiplicity } p (\text{radical } x) = (\text{if } p \text{ dvd } x \text{ then } 1 \text{ else } 0)$
proof –
have $\text{multiplicity } p (\text{radical } x) = (\sum_{q \in \text{prime-factors } x} \text{multiplicity } p \ q)$
using *assms* **unfolding** *radical-def*
by (*auto simp: prime-elem-multiplicity-prod-distrib*)
also have $\dots = (\sum_{q \in \text{prime-factors } x} \text{if } p = q \text{ then } 1 \text{ else } 0)$
using *assms* **by** (*intro Groups-Big.sum.cong*) (*auto intro!: prime-multiplicity-other*)
also have $\dots = (\text{if } p \in \text{prime-factors } x \text{ then } 1 \text{ else } 0)$ **by** *simp*
also have $\dots = (\text{if } p \text{ dvd } x \text{ then } 1 \text{ else } 0)$
using *assms* **by** (*auto simp: prime-factors-dvd*)
finally show *?thesis* .
qed

lemma *radical-1* [*simp*]: $\text{radical } 1 = 1$
by (*simp add: radical-def*)

lemma *radical-unit* [*simp*]: $\text{is-unit } x \implies \text{radical } x = 1$
by (*auto simp: radical-def prime-factorization-unit*)

lemma *prime-factors-power*:

```

assumes  $n > 0$ 
shows  $\text{prime-factors } (x \wedge n) = \text{prime-factors } x$ 
using assms by (cases  $x = 0$ ) (auto simp: prime-factors-dvd zero-power prime-dvd-power-iff)

lemma radical-power [simp]:  $n > 0 \implies \text{radical } (x \wedge n) = \text{radical } x$ 
  by (auto simp add: radical-def prime-factors-power)

end

context factorial-semiring-gcd
begin

lemma radical-mult-coprime:
  assumes coprime  $a\ b$ 
  shows  $\text{radical } (a * b) = \text{radical } a * \text{radical } b$ 
proof (cases  $a = 0 \vee b = 0$ )
  case False
    with assms have  $\text{prime-factors } a \cap \text{prime-factors } b = \{\}$ 
    using not-prime-unit coprime-common-divisor by (auto simp: prime-factors-dvd)
    hence  $\prod (\text{prime-factors } a \cup \text{prime-factors } b) = \prod (\text{prime-factors } a) * \prod (\text{prime-factors } b)$ 
    by (intro prod.union-disjoint) auto
    with False show ?thesis by (simp add: radical-def prime-factorization-mult)
qed auto

lemma multiplicity-le-imp-dvd':
  assumes  $x \neq 0 \wedge p. p \in \text{prime-factors } x \implies \text{multiplicity } p\ x \leq \text{multiplicity } p\ y$ 
  shows  $x\ \text{dvd}\ y$ 
proof (rule multiplicity-le-imp-dvd)
  fix  $p$  assume prime  $p$ 
  thus  $\text{multiplicity } p\ x \leq \text{multiplicity } p\ y$  using assms(1) assms(2)[of p]
  by (cases  $p\ \text{dvd}\ x$ ) (auto simp: prime-factors-dvd not-dvd-imp-multiplicity-0)
qed fact+

```

end

1.3 Main result

The following proofs are basically a one-to-one translation of Franz Lemmermeyer's presentation [1] of Snyder's proof of the Mason–Stothers theorem.

```

lemma prime-power-dvd-pderiv:
  fixes  $f\ p :: 'a :: \text{field-gcd poly}$ 
  assumes prime-elim  $p$ 
  defines  $n \equiv \text{multiplicity } p\ f - 1$ 
  shows  $p \wedge n\ \text{dvd}\ \text{pderiv } f$ 
proof (cases  $p\ \text{dvd}\ f \wedge f \neq 0$ )
  case True
    hence  $\text{multiplicity } p\ f > 0$  using assms
    by (subst prime-multiplicity-gt-zero-iff) auto

```

hence *Suc-n*: $Suc\ n = multiplicity\ p\ f$ **by** (*simp add: n-def*)
define g **where** $g = f\ div\ p\ \wedge\ Suc\ n$
have $p\ \wedge\ Suc\ n\ dvd\ f$ **unfolding** *Suc-n* **by** (*rule multiplicity-dvd*)
hence $f\text{-eq}$: $f = p\ \wedge\ Suc\ n * g$ **by** (*simp add: g-def*)
also have $pderiv\ \dots = p\ \wedge\ n * (smult\ (of\text{-nat}\ (Suc\ n))\ (pderiv\ p * g) + p * pderiv\ g)$
by (*simp only: pderiv-mult pderiv-power-Suc*) (*simp add: algebra-simps*)
also have $p\ \wedge\ n\ dvd\ \dots$ **by** *simp*
finally show *?thesis* .
qed (*auto simp: n-def not-dvd-imp-multiplicity-0*)

lemma *poly-div-radical-dvd-pderiv*:

fixes $p :: 'a :: field\text{-gcd}\ poly$
shows $p\ div\ radical\ p\ dvd\ pderiv\ p$
proof (*cases pderiv p = 0*)
case *False*
hence $p \neq 0$ **by** *auto*
show *?thesis*
proof (*rule multiplicity-le-imp-dvd'*)
fix $q :: 'a\ poly$ **assume** $q: q \in prime\text{-factors}\ (p\ div\ radical\ p)$
hence $q\ dvd\ p\ div\ radical\ p$ **by** *auto*
also from $\langle p \neq 0 \rangle$ **have** $\dots\ dvd\ p$ **by** (*subst div-dvd-iff-mult*) *auto*
finally have $q\ dvd\ p$.

have $p = p\ div\ radical\ p * radical\ p$ **by** *simp*
also from q **and** $\langle p \neq 0 \rangle$ **have** $multiplicity\ q\ \dots = Suc\ (multiplicity\ q\ (p\ div\ radical\ p))$
by (*subst prime-elem-multiplicity-mult-distrib*)
(*auto simp: dvd-div-eq-0-iff multiplicity-radical-prime $\langle q\ dvd\ p \rangle$ prime-factors-dvd*)
finally have $multiplicity\ q\ (p\ div\ radical\ p) \leq multiplicity\ q\ p - 1$ **by** *simp*
also have $\dots \leq multiplicity\ q\ (pderiv\ p)$ **using** $\langle pderiv\ p \neq 0 \rangle$ **and** q **and** $\langle p \neq 0 \rangle$
by (*intro multiplicity-geI prime-power-dvd-pderiv*)
(*auto simp: prime-factors-dvd dvd-div-eq-0-iff*)
finally show $multiplicity\ q\ (p\ div\ radical\ p) \leq multiplicity\ q\ (pderiv\ p)$.
qed (*insert $\langle p \neq 0 \rangle$, auto simp: dvd-div-eq-0-iff*)
qed *auto*

lemma *degree-pderiv-mult-less*:

assumes $pderiv\ C \neq 0$
shows $degree\ (pderiv\ C * B) < degree\ B + degree\ C$
proof –
have $degree\ (pderiv\ C * B) \leq degree\ (pderiv\ C) + degree\ B$
by (*rule degree-mult-le*)
also from *assms* **have** $degree\ (pderiv\ C) < degree\ C$ **by** (*rule degree-pderiv-less*)
finally show *?thesis* **by** *simp*
qed

lemma *Mason-Stothers-aux*:

fixes $A B C :: 'a :: \text{field-gcd poly}$
assumes $\text{nz}: A \neq 0 B \neq 0 C \neq 0$ **and** $\text{sum}: A + B + C = 0$ **and** $\text{coprime}: \text{Gcd} \{A, B, C\} = 1$
and $\text{deg-ge}: \text{degree } A \geq \text{degree} (\text{radical } (A * B * C))$
shows $\text{pderiv } A = 0 \text{ pderiv } B = 0 \text{ pderiv } C = 0$
proof –
have $C\text{-eq}: C = -A - B - C = A + B$ **using** sum **by** algebra+
from coprime **have** $\text{gcd } A (\text{gcd } B (-C)) = 1$ **by** simp
also note $C\text{-eq}(2)$
finally have $\text{coprime } A B$ **by** $(\text{simp add: gcd.commute add.commute[of } A B] \text{ coprime-iff-gcd-eq-1})$
hence $\text{coprime } A (-C) \text{ coprime } B (-C)$
unfolding $C\text{-eq}$ **by** $(\text{simp-all add: gcd.commute[of } B A] \text{ gcd.commute[of } B A + B] \text{ add.commute coprime-iff-gcd-eq-1})$
hence $\text{coprime } A C \text{ coprime } B C$ **by** simp-all
note $\text{coprime} = \text{coprime } \langle \text{coprime } A B \rangle \text{ this}$
have $\text{coprime1}: \text{coprime } (A \text{ div radical } A) (B \text{ div radical } B)$
by $(\text{rule coprime-divisors[OF - - } \langle \text{coprime } A B \rangle]) (\text{insert nz, auto simp: div-dvd-iff-mult})$
have $\text{coprime2}: \text{coprime } (A \text{ div radical } A) (C \text{ div radical } C)$
by $(\text{rule coprime-divisors[OF - - } \langle \text{coprime } A C \rangle]) (\text{insert nz, auto simp: div-dvd-iff-mult})$
have $\text{coprime3}: \text{coprime } (B \text{ div radical } B) (C \text{ div radical } C)$
by $(\text{rule coprime-divisors[OF - - } \langle \text{coprime } B C \rangle]) (\text{insert nz, auto simp: div-dvd-iff-mult})$
have $\text{coprime4}: \text{coprime } (A \text{ div radical } A * (B \text{ div radical } B)) (C \text{ div radical } C)$
using coprime2 coprime3 **by** $(\text{subst coprime-mult-left-iff}) \text{ auto}$

have $\text{eq}: A * \text{pderiv } B - \text{pderiv } A * B = \text{pderiv } C * B - C * \text{pderiv } B$
by $(\text{simp add: C-eq pderiv-add pderiv-diff pderiv-minus algebra-simps})$

have $A \text{ div radical } A \text{ dvd } (A * \text{pderiv } B - \text{pderiv } A * B)$
using nz **by** $(\text{intro dvd-diff dvd-mult2 poly-div-radical-dvd-pderiv}) (\text{auto simp: div-dvd-iff-mult})$
with eq **have** $A \text{ div radical } A \text{ dvd } (\text{pderiv } C * B - C * \text{pderiv } B)$ **by** simp
moreover have $C \text{ div radical } C \text{ dvd } (\text{pderiv } C * B - C * \text{pderiv } B)$
using nz **by** $(\text{intro dvd-diff dvd-mult2 poly-div-radical-dvd-pderiv}) (\text{auto simp: div-dvd-iff-mult})$
moreover have $B \text{ div radical } B \text{ dvd } (\text{pderiv } C * B - C * \text{pderiv } B)$
using nz **by** $(\text{intro dvd-diff dvd-mult poly-div-radical-dvd-pderiv}) (\text{auto simp: div-dvd-iff-mult})$
ultimately have $(A \text{ div radical } A) * (B \text{ div radical } B) * (C \text{ div radical } C) \text{ dvd } (\text{pderiv } C * B - C * \text{pderiv } B)$ **using** $\text{coprime coprime1 coprime4}$
by $(\text{intro divides-mult}) \text{ auto}$
also have $(A \text{ div radical } A) * (B \text{ div radical } B) * (C \text{ div radical } C) = (A * B * C) \text{ div } (\text{radical } A * \text{radical } B * \text{radical } C)$
by $(\text{simp add: div-mult-div-if-dvd mult-dvd-mono})$
also have $\text{radical } A * \text{radical } B * \text{radical } C = \text{radical } (A * B) * \text{radical } C$

using *coprime* **by** (*subst radical-mult-coprime*) *auto*
also have $\dots = \text{radical } (A * B * C)$
using *coprime* **by** (*subst radical-mult-coprime [symmetric]*) *auto*
finally have *dvd*: $((A * B * C) \text{ div radical } (A * B * C)) \text{ dvd } (pderiv C * B - C * pderiv B)$.

have $pderiv B = 0 \wedge pderiv C = 0$
proof (*rule ccontr*)
assume $\neg(pderiv B = 0 \wedge pderiv C = 0)$
hence $*$: $pderiv B \neq 0 \vee pderiv C \neq 0$ **by** *blast*

have $\text{degree } (pderiv C * B - C * pderiv B) \leq$
 $\text{max } (\text{degree } (pderiv C * B)) (\text{degree } (C * pderiv B))$ **by** (*rule degree-diff-le-max*)
also have $\dots < \text{degree } B + \text{degree } C$
using *degree-pderiv-mult-less*[*of B C*] *degree-pderiv-mult-less*[*of C B*] *
by (*cases pderiv B = 0*; *cases pderiv C = 0*) (*auto simp add: algebra-simps*)
also have $\text{degree } B + \text{degree } C = \text{degree } (B * C)$
using *nz* **by** (*subst degree-mult-eq*) *auto*
also have $\dots = \text{degree } (A * (B * C)) - \text{degree } A$
using *nz* **by** (*subst (2) degree-mult-eq*) *auto*
also have $\dots \leq \text{degree } (A * B * C) - \text{degree } (\text{radical } (A * B * C))$ **unfolding**
mult.assoc
using *assms* **by** (*intro diff-le-mono2*) (*auto simp: mult-ac*)
also have $\dots = \text{degree } ((A * B * C) \text{ div radical } (A * B * C))$
by (*intro degree-div [symmetric]*) *auto*
finally have *less*: $\text{degree } (pderiv C * B - C * pderiv B) <$
 $\text{degree } (A * B * C \text{ div radical } (A * B * C))$ **by** *simp*

have *eq'*: $pderiv C * B - C * pderiv B = 0$
proof (*rule ccontr*)
assume $pderiv C * B - C * pderiv B \neq 0$
hence $\text{degree } (A * B * C \text{ div radical } (A * B * C)) \leq \text{degree } (pderiv C * B - C * pderiv B)$
*C * pderiv B*)
using *dvd* **by** (*intro dvd-imp-degree-le*) *auto*
with less **show** *False* **by** *linarith*

qed
from $*$ **show** *False*
proof (*elim disjE*)
assume [*simp*]: $pderiv C \neq 0$
have $C \text{ dvd } C * pderiv B$ **by** *simp*
also from *eq'* **have** $\dots = pderiv C * B$ **by** *simp*
finally have $C \text{ dvd } pderiv C$ **using** *coprime*
by (*subst (asm) coprime-dvd-mult-left-iff*) (*auto simp: coprime-commute*)
hence $\text{degree } C \leq \text{degree } (pderiv C)$ **by** (*intro dvd-imp-degree-le*) *auto*
moreover have $\text{degree } (pderiv C) < \text{degree } C$ **by** (*intro degree-pderiv-less*)
auto
ultimately show *False* **by** *simp*
next

```

assume [simp]: pderiv B ≠ 0
have B dvd B * pderiv C by simp
also from eq' have ... = pderiv B * C by (simp add: mult-ac)
finally have B dvd pderiv B using coprime
  by (subst (asm) coprime-dvd-mult-left-iff) auto
hence degree B ≤ degree (pderiv B) by (intro dvd-imp-degree-le) auto
moreover have degree (pderiv B) < degree B by (intro degree-pderiv-less)
auto
  ultimately show False by simp
qed
qed
with eq and nz show pderiv A = 0 pderiv B = 0 pderiv C = 0 by auto
qed

```

theorem *Mason-Stothers*:

```

fixes A B C :: 'a :: field-gcd poly
assumes nz: A ≠ 0 B ≠ 0 C ≠ 0 ∃ p ∈ {A,B,C}. pderiv p ≠ 0
  and sum: A + B + C = 0 and coprime: Gcd {A, B, C} = 1
  shows Max {degree A, degree B, degree C} < degree (radical (A * B * C))
proof -
  have degree A < degree (radical (A * B * C))
  if ∃ p ∈ {A,B,C}. p ≠ 0 ∃ p ∈ {A,B,C}. pderiv p ≠ 0 sum-mset {#A,B,C#} = 0
  Gcd {A, B, C} = 1
  for A B C :: 'a poly
  proof (rule ccontr)
  assume ¬(degree A < degree (radical (A * B * C)))
  hence degree A ≥ degree (radical (A * B * C)) by simp
  with Mason-Stothers-aux[of A B C] that show False by (auto simp: add-ac)
  qed
from this[of A B C] this[of B C A] this[of C A B] assms show ?thesis
  by (simp only: insert-commute mult-ac add-ac) (auto simp: add-ac mult-ac)
qed

```

The result can be simplified a bit more in fields of characteristic 0:

corollary *Mason-Stothers-char-0*:

```

fixes A B C :: 'a :: {field-gcd, field-char-0} poly
assumes nz: A ≠ 0 B ≠ 0 C ≠ 0 and deg: ∃ p ∈ {A,B,C}. degree p ≠ 0
  and sum: A + B + C = 0 and coprime: Gcd {A, B, C} = 1
  shows Max {degree A, degree B, degree C} < degree (radical (A * B * C))
proof -
  from deg have ∃ p ∈ {A,B,C}. pderiv p ≠ 0
  by (auto simp: pderiv-eq-0-iff)
  from Mason-Stothers[OF assms(1-3) this assms(5-)] show ?thesis .
qed

```

As a nice corollary, we get a kind of analogue of Fermat's last theorem for polynomials: Given non-zero polynomials A, B, C with $A^n + B^n + C^n = 0$ on lowest terms, we must either have $n \leq 2$ or $(A^n)' = (B^n)' = (C^n)' = 0$.

In the case of a field with characteristic 0, this last possibility is equivalent

to A , B , and C all being constant.

corollary *fermat-poly*:

fixes $A B C :: 'a :: \text{field-gcd poly}$

assumes $\text{sum}: A^n + B^n + C^n = 0$ **and** $\text{cop}: \text{Gcd } \{A, B, C\} = 1$

assumes $\text{nz}: A \neq 0 B \neq 0 C \neq 0$ **and** $\text{deg}: \exists p \in \{A, B, C\}. \text{pderiv } (p^n) \neq 0$

shows $n \leq 2$

proof (*rule ccontr*)

assume $\neg(n \leq 2)$

hence $n > 2$ **by** *simp*

have $\text{Max } \{\text{degree } (A^n), \text{degree } (B^n), \text{degree } (C^n)\} <$

$\text{degree } (\text{radical } (A^n * B^n * C^n))$ (**is** $- < ?d$)

using *assms by (intro Mason-Stothers) (auto simp: degree-power-eq gcd-exp)*

hence $\text{Max } \{\text{degree } (A^n), \text{degree } (B^n), \text{degree } (C^n)\} + 1 \leq ?d$ **by** *linarith*

hence $n * \text{degree } A + 1 \leq ?d \ n * \text{degree } B + 1 \leq ?d \ n * \text{degree } C + 1 \leq ?d$

using *assms by (simp-all add: degree-power-eq)*

hence $n * (\text{degree } A + \text{degree } B + \text{degree } C) + 3 \leq 3 * ?d$

unfolding *ring-distrib by linarith*

also have $A^n * B^n * C^n = (A * B * C)^n$ **by** (*simp add: mult-ac power-mult-distrib*)

also have $\text{radical } \dots = \text{radical } (A * B * C)$

using $\langle n > 2 \rangle$ **by** *simp*

also have $\text{degree } (\text{radical } (A * B * C)) \leq \text{degree } (A * B * C)$

using *nz by (intro dvd-imp-degree-le) auto*

also have $\dots = \text{degree } A + \text{degree } B + \text{degree } C$

using *nz by (simp add: degree-mult-eq)*

finally have $(3 - n) * (\text{degree } A + \text{degree } B + \text{degree } C) \geq 3$

by (*simp add: algebra-simps*)

hence $3 - n \neq 0$ **by** (*intro notI*) *auto*

hence $n < 3$ **by** *simp*

with $\langle n > 2 \rangle$ **show** *False* **by** *simp*

qed

corollary *fermat-poly-char-0*:

fixes $A B C :: 'a :: \{\text{field-gcd, field-char-0}\} \text{ poly}$

assumes $\text{sum}: A^n + B^n + C^n = 0$ **and** $\text{cop}: \text{Gcd } \{A, B, C\} = 1$

assumes $\text{nz}: A \neq 0 B \neq 0 C \neq 0$ **and** $\text{deg}: \exists p \in \{A, B, C\}. \text{degree } p > 0$

shows $n \leq 2$

proof (*rule ccontr*)

assume $*$: $\neg(n \leq 2)$

with *nz* **and** *deg* **have** $\exists p \in \{A, B, C\}. \text{pderiv } (p^n) \neq 0$

by (*auto simp: pderiv-eq-0-iff degree-power-eq*)

from *fermat-poly[OF assms(1-5) this]* **and** $*$ **show** *False* **by** *simp*

qed

end

References

- [1] F. Lemmermeyer. Algebraic Geometry (lecture notes). <http://www.fen.bilkent.edu.tr/~franz/ag05/ag-02.pdf>, 2005.
- [2] N. Snyder. An alternate proof of Mason's theorem. *Elemente der Mathematik*, 55(3):93–94, Aug 2000.