Lucas's Theorem

Chelsea Edmonds

February 23, 2021

Abstract

This work presents a formalisation of a generating function proof for Lucas's theorem. We first outline extensions to the existing Formal Power Series (FPS) library, including an equivalence relation for coefficients modulo n, an alternate binomial theorem statement, and a formalised proof of the Freshman's dream (mod p) lemma.

The second part of the work presents the formal proof of Lucas's Theorem. Working backwards, the formalisation first proves a well known corollary of the theorem which is easier to formalise and then applies induction to prove the original theorem statement. The proof of the corollary aims to provide a good example of a formalised generating function equivalence proof using the FPS library. The final theorem statement is intended to be integrated into the formalised proof of Hilbert's 10th Problem [1].

Contents

theory *Lucas-Theorem*
imports *Main HOL−Computational-Algebra.Computational-Algebra*
begin

notation *fps-nth* (**infixl** \$ *75*)

1 Extensions on Formal Power Series (FPS) Library

This section presents a few extensions on the Formal Power Series (FPS) library, described in [2]

1.1 FPS Equivalence Relation

This proof requires reasoning around the equivalence of coefficients mod some prime number. This section defines an equivalence relation on FPS using the pattern described by Paulson in [4], as well as some basic lemmas for reasoning around how the equivalence holds after common operations are applied

definition *fpsmodrel p* ≡ { (*f*, *g*). ∀ *n*. (*f* $ *n*) *mod p* = (*g* $ *n*) *mod p* }

lemma *fpsrel-iff* [*simp*]: (*f*, *g*) ∈ *fpsmodrel p* ⟷ (∀ *n*. (*f* $ *n*) *mod p* = (*g* $ *n*) *mod p*)
⟨*proof*⟩

lemma *fps-equiv*: *equiv UNIV* (*fpsmodrel p*)
⟨*proof*⟩

Equivalence relation over multiplication

lemma *fps-mult-equiv-coeff*:
fixes *f g* :: ('*a* :: {*euclidean-ring-cancel*}) *fps*
assumes (*f*, *g*) ∈ *fpsmodrel p*
shows (*f*∗*h*)$*n mod p* = (*g*∗*h*)$*n mod p*
⟨*proof*⟩

lemma *fps-mult-equiv*:
fixes *f g* :: ('*a* :: {*euclidean-ring-cancel*}) *fps*
assumes (*f*, *g*) ∈ *fpsmodrel p*
shows (*f*∗*h*, *g*∗*h*) ∈ *fpsmodrel p*
⟨*proof*⟩

Equivalence relation over power operator

lemma *fps-power-equiv*:
fixes *f g* :: ('*a* :: {*euclidean-ring-cancel*}) *fps*
fixes *x* :: *nat*
assumes (*f*, *g*) ∈ *fpsmodrel p*
shows (*f*^*x*, *g*^*x*) ∈ *fpsmodrel p*
⟨*proof*⟩

1.2 Binomial Coefficients

The *fps-binomial* definition in the formal power series uses the *n gchoose k* operator. It's defined as being of type '*a fps*, however the equivalence

relation requires a type $'a$ that supports the modulo operator. The proof of the binomial theorem based on FPS coefficients below uses the choose operator and does not put bounds on the type of *fps-X*.

lemma *binomial-coeffs-induct*:
  fixes *n k* :: *nat*
  shows *(1 + fps-X)^n $ k = of-nat(n choose k)*
⟨*proof*⟩

1.3 Freshman's Dream Lemma on FPS

The Freshman's dream lemma modulo a prime number p is a well known proof that $(1 + x^p) \equiv (1 + x)^p \mod p$

First prove that $\binom{p^n}{k} \equiv 0 \mod p$ for $k \geq 1$ and $k < p^n$. The eventual proof only ended up requiring this with $n = 1$

lemma *pn-choose-k-modp-0*:
  fixes *n k*::*nat*
  assumes *prime p*
        k ≥ 1 ∧ k ≤ p^n − 1
        n > 0
  shows *(p^n choose k) mod p = 0*
⟨*proof*⟩

Applying the above lemma to the coefficients of $(1 + X)^p$, it is easy to show that all coefficients other than the 0th and pth will be 0

lemma *fps-middle-coeffs*:
  assumes *prime p*
        n ≠ 0 ∧ n ≠ p
  shows *((1 + fps-X :: int fps) ^p) $ n mod p = 0 mod p*
⟨*proof*⟩

It follows that $(1 + X)^p$ is equivalent to $(1 + X^p)$ under our equivalence relation, as required to prove the freshmans dream lemma.

lemma *fps-freshmans-dream*:
  assumes *prime p*
  shows *(((1 + fps-X :: int fps) ^p), (1 + (fps-X)^(p))) ∈ fpsmodrel p*
⟨*proof*⟩

2 Lucas's Theorem Proof

A formalisation of Lucas's theorem based on a generating function proof using the existing formal power series (FPS) Isabelle library

2.1 Reasoning about Coefficients Helpers

A generating function proof of Lucas's theorem relies on direct comparison between coefficients of FPS which requires a number of helper lemmas to prove formally. In particular it compares the coefficients of $(1+X)^n \mod p$ to $(1+X^p)^N * (1+X)^r n \mod p$, where $N = n/p$, and $rn = n \mod p$. This section proves that the kth coefficient of $(1+X^p)^N * (1+X)^r n = (NchooseK) * (rnchooserk)$

Applying the (oo) operator enables reasoning about the coefficients of $(1+X^p)^n$ using the existing binomial theorem proof with X^p instead of X.

lemma *fps-binomial-p-compose*:
assumes $p \neq 0$
shows $(1 + (fps\text{-}X :: ('a :: \{idom\}\ fps))\hat{}p)\hat{}n = ((1 + fps\text{-}X)\hat{}n)\ oo\ (fps\text{-}X\hat{}p)$
⟨*proof*⟩

Next the proof determines the value of the kth coefficient of $(1+X^p)^N$.

lemma *fps-X-pow-binomial-coeffs*:
assumes *prime p*
shows $(1 + (fps\text{-}X :: int\ fps)\hat{}p)\hat{}N\ \$k = (if\ p\ dvd\ k\ then\ (N\ choose\ (k\ div\ p))\ else\ 0)$
⟨*proof*⟩

The final helper lemma proves the kth coefficient is equivalent to $\binom{?N}{?K} * \binom{?rn}{?rk}$ as required.

lemma *fps-div-rep-coeffs*:
assumes *prime p*
shows $((1 + (fps\text{-}X :: int\ fps)\hat{}p)\hat{}(n\ div\ p) * (1 + fps\text{-}X)\hat{}(n\ mod\ p))\ \$\ k =$
$((n\ div\ p)\ choose\ (k\ div\ p)) * ((n\ mod\ p)\ choose\ (k\ mod\ p))$
(**is** $((1 + (fps\text{-}X :: int\ fps)\hat{}p)\hat{}?N * (1 + fps\text{-}X)\hat{}?rn)\ \$\ k = (?N\ choose\ ?K) * (?rn\ choose\ ?rk)$)
⟨*proof*⟩

2.2 Lucas Theorem Proof

The proof of Lucas's theorem combines a generating function approach, based off [3] with induction. For formalisation purposes, it was easier to first prove a well known corollary of the main theorem (also often presented as an alternative statement for Lucas's theorem), which can itself be used to backwards prove the the original statement by induction. This approach was adapted from P. Cameron's lecture notes on combinatorics [5]

2.2.1 Proof of the Corollary

This step makes use of the coefficient equivalence arguments proved in the previous sections

corollary *lucas-corollary*:
fixes *n k* :: *nat*
assumes *prime p*
shows (*n choose k*) *mod p* = (((*n div p*) *choose* (*k div p*)) ∗ ((*n mod p*) *choose* (*k mod p*))) *mod p*
(**is** (*n choose k*) *mod p* = ((*?N choose ?K*) ∗ (*?rn choose ?rk*)) *mod p*)
⟨*proof*⟩

2.2.2 Proof of the Theorem

The theorem statement requires a formalised way of referring to the base p representation of a number. We use a definition that specifies the ith digit of the base p representation. This definition is originally from the Hilbert's 10th Problem Formalisation project [1] which this work contributes to.

definition *nth-digit-general* :: *nat* ⇒ *nat* ⇒ *nat* ⇒ *nat* **where**
nth-digit-general num i base = (*num div* (*base* ^ *i*)) *mod base*

Applying induction on d, where d is the highest power required in either n or k's base p representation, *prime ?p* ⟹ (*?n choose ?k*) *mod ?p* = (*?n div ?p choose ?k div ?p*) ∗ (*?n mod ?p choose ?k mod ?p*) *mod ?p* can be used to prove the original theorem.

theorem *lucas-theorem*:
fixes *n k d*::*nat*
assumes *n* < *p* ^ (*Suc d*)
assumes *k* < *p* ^ (*Suc d*)
assumes *prime p*
shows (*n choose k*) *mod p* = (∏ *i*≤*d*. ((*nth-digit-general n i p*) *choose* (*nth-digit-general k i p*))) *mod p*
⟨*proof*⟩

end

References

[1] J. Bayer, M. David, A. Pal, B. Stock, and D. Schleicher. The DPRM Theorem in Isabelle (Short Paper). In J. Harrison, J. O'Leary, and A. Tolmach, editors, *10th International Conference on Interactive Theorem Proving (ITP 2019)*, volume 141 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 33:1–33:7, Dagstuhl, Germany, 2019. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.

[2] A. Chaieb. Formal power series. *Journal of Automated Reasoning*, 47(3):291–318, Oct. 2011.

[3] N. J. Fine. Binomial coefficients modulo a prime. *The American Mathematical Monthly*, 54(10):589–592, 1947.

[4] L. C. Paulson. Defining Functions on Equivalence Classes. *ACM Transactions on Computational Logic (TOCL)*, 7(4):658–675, Oct. 2006.

[5] Peter Cameron. Notes on Combinatorics. http://www.maths.qmul.ac.uk/~pjc/notes/comb.pdf, 2007.