Ciências Exatas e da Terra - Ciência da Computação - Teoria da Computação

Formalização do Teorema de Fermat Generalizado

Trabalho de Iniciação Científica

Orientador: Prof. Dr. Flávio L. C. de Moura
Departamento de Ciência da Computação
Universidade de Brasília - UnB
e-mail: flaviomoura@unb.br

Aluno: Lucas de Melo Guimarães
Departamento de Ciência da Computação
Universidade de Brasília - UnB
e-mail: lucasmgcic@gmail.com

Introdução

O algoritmo AKS [1] tem como finalidade testar a primalidade de um número inteiro dado. Existem diversos outros testes de primalidade, desde os mais simples, como o crivo de Eratóstenes, até testes probabilísticos que atualmente são os mais eficientes para testar primalidade. A grande vantagem do algoritmo AKS é conseguir aliar o determinismo à execução em tempo polinomial, sendo assim o primeiro teste determinístico de primalidade a funcionar em tempo polinomial.

O foco principal deste projeto é formalizar a correção de uma generalização do teorema de Fermat em Coq, que será necessária na formalização do algoritmo AKS, a ser concluída posteriormente.

Algoritmo AKS

Seja $\mathbb{P}$ o conjunto dos números primos. O algoritmo AKS baseia-se em uma generalização do Teorema de Fermat, onde para todo inteiro n,

$$n \in \mathbb{P} \Leftrightarrow (X+a)^n \equiv X^n + a \quad \text{em} \quad \mathbb{Z}_n[X] \tag{1}$$

O problema do teste (1) é que o mesmo tem ordem $\Omega(n)$, e portanto é exponencial. Para reduzir a complexidade, a solução foi considerar a equação (1) no anel quociente $\frac{\mathbb{Z}_n[X]}{(x^r-1)}$ para um r adequado. Desta forma, o algoritmo AKS é o primeiro teste de primalidade determinístico e polinomial. A apresentação do algoritmo a seguir é baseada em [2].

ENTRADA: Um inteiro ímpar.
Se $n = a^b, a, b \in \mathbb{N}, \quad b > 1$
  Imprime **COMPOSTO e pára.**
$W = 2 \cdot n \cdot (n-1) \cdot (n^2-1) \cdot (n^3-1) \cdots (n^{4 \cdot \lceil \log_2 n \rceil^2} - 1)$
Seja r o menor primo tal que $r \not\equiv 0 \ (mod\ W)$
Para todo q primo, $\quad q < r$
  Se $n \equiv 0 \ (mod\ q)$ E $q = n$
    Imprime **PRIMO e pára.**
  Se $n \equiv 0 \ (mod\ q)$ E $q < n$
    Imprime **COMPOSTO e pára.**
  Se $n \not\equiv 0 \ (mod\ q)$
    Passa para o próximo q
Fim-Para
Se $(x+a)^n \equiv x^n + a$ em $\frac{\mathbb{Z}_n[X]}{(x^r-1)}, \forall a \in S = \{1, 2, \cdots, r\}$
  Imprime **PRIMO e pára.**
Imprime **COMPOSTO e pára.**

Metodologia

Para verificar formalmente a generalização em questão, optou-se por utilizar o assistente de provas Coq [3], que provê uma linguagem formal com o intuito de facilitar a escrita de definições matemáticas, teoremas e especificações em geral, checando sua validade via *software*. Em relação à prova da generalização do teorema de Fermat, realizou-se primeiramente provas com papel e lápis para se analisar a melhor estratégia de se formalizar a generalização em Coq, quebrando a prova em diversos resultados intermediários.

Resultados

```
Lemma binomial_rec :  forall n k, (le k n) ->
(binomial n k) * Z_of_nat(fact k * fact(n - k)) =
Z_of_nat (fact n).
```

Este cabeçalho de Coq indica o seguinte lema: $\binom{n}{k} \cdot k! \cdot (n-k)! = n!\ , \quad \forall n, k \in \mathbb{N},\ k \leq n.$

A prova deste lema utiliza indução sobre n. O passo indutivo baseia-se no fato de que: $(n+1)! = (n+1) \cdot n!$. O mesmo argumento pode ser aplicado em k. No passo indutivo, analisou-se também os possíveis valores de k. A prova para os casos $k = 0$ e $k > n$ é trivial, e para $k \leq n$, utilizou-se diversas táticas para simplificação algébrica e de termos, como por exemplo, *simpl* que busca simplificar alguns termos a partir de sua definição. Esta prova, sem contar com os resultados auxiliares, possui aproximadamente 100 linhas de código.

Ainda sobre os resultados obtidos, foi possível verificar, a partir da aplicação de **binomial_rec**, a seguinte decomposição de binomiais:

$$\binom{n+1}{i+1} = \frac{n+1}{i+1} \cdot \binom{n}{i} \quad \forall n, i \in \mathbb{N},\ i \leq n.$$

```
Lemma dec_bin:  forall (n i :  nat), (le i n) -> binomial (S
n)(S i) = ((Z_of_nat(S n))/(Z_of_nat(S i))) * (binomial n i).
```

Outro resultado importante provado sobre binomiais, diz respeito a divisibilidade e é importante para a formalização proposta. Este resultado é dado pelo teorema:

$$i \mid \binom{n-1}{i-1} \quad \forall n, i \in \mathbb{Z},\ 1 < i < n,.$$

```
Lemma div_bin:  forall (i n:  Z), 1 < i -> i < n ->
(i | binomial (Zabs_nat(n-1)) (Zabs_nat(i-1))).
```

Ainda sobre divisibilidade, outro resultado de suma importância para a verificação da formalização proposta foi provado. Este resultado é dado pelo teorema:

$$i \mid x \ \Rightarrow \ i \ \not| \ x \ + \ k \quad \forall k,\ i,\ x \ \in \mathbb{Z},\ \ 0 < k < i$$

```
Lemma prod_inaux :  forall (x i k :  Z), 0 < k < i -> (i|x)
-> ~(i|(x+k)).
```

Um resultado relevante consiste em provar que o resultado de um binomial qualquer é sempre um número inteiro. Este resultado é dado pelo teorema:

$$\exists k \in \mathbb{Z},\ \binom{n}{i} = k, \quad \forall n, i \in \mathbb{N}.$$

```
Theorem bin_int:forall (n i:nat),(exists k, binomial n i=k).
```

A prova deste teorema baseia-se numa indução sobre n, na qual o passo indutivo está baseado na relação de Stifel que afirma: $\binom{n}{i} + \binom{n}{i+1} = \binom{n+1}{i+1}$. Código fonte disponível em: `http://www.cic.unb.br/~flavio/ic/tfq.tar.gz`

Conclusão

Inicialmente, o teste de primalidade de um número, pode parecer mais uma curiosidade matemática e de interesse apenas teórico do que algo que possua aplicações relevantes. Entretanto, com o desenvolvimento atual das comunicações utilizando redes de computadores, o algoritmo AKS pode representar uma importante contribuição para a segurança dos dados que trafegam nas redes, uma vez que algumas das técnicas de criptografia vigentes se baseiam na utilização de números primos grandes.

Para se verificar tal algoritmo, é necessária a formalização do Teorema de Fermat generalizado proposta neste trabalho. Como continuação deste trabalho, espera-se concluir a verificação formal do algoritmo AKS.

Referências

[1] AGRAWAL, M., KAYAL, N., SAXENA, N. PRIMES is in P. Annals of Mathematics v.160 n.2 p.781-793. 2004.

[2] COUTINHO S.C. Primalidade em Tempo Polinomial: uma introdução ao Algoritmo AKS. Sociedeade Brasileira de Matemática, 2004.

[3] The Coq Development Team, (2008). The Coq Proof Assistant Reference Manual V8.2. INRIA. Disponível em: `http://coq.inria.fr/coq/distrib/current/refman/`

Instituição de Fomento: CNPq