

Formalization of Polynomially Bounded Functions in Mizar

Okazaki, Hiroyuki
信州大学学術研究院（工学系）

<https://hdl.handle.net/2324/1520941>

出版情報 : MI lecture note series. 61, pp.38-52, 2015-03-06. 九州大学マス・フォア・インダストリ
研究所
バージョン :
権利関係 :



Formalization of Polynomially Bounded Functions in Mizar

HiroYuki Okazaki *

平成 27 年 1 月 10 日

1 Introduction

A security proof of cryptographic systems is very important. We are trying to formalize many topics about cryptology, such as computational complexity, finite probability[4, 5, 6, 7, 8, 9], algorithms[10], number theory[11, 12], etc. In this report, we introduce our formalization of the polynomially bounded sequences and prove the correctness of the formalization using the Mizar proof checking system as a formal verification tool[2]. Polynomially bounded sequences play an important role in practical computer complexity theory, such as, computer simulations and cryptology, and is very important in proving the security of cryptographic systems. A security proof is performed by proving that an attack on a target cryptosystem is more difficult than a complexity problem. The class P is a fundamental computational complexity class that contains all polynomial-time decision problems[1]. It takes polynomially bounded amount of computation time to solve polynomial-time decision problems by a deterministic Turing machine. Moreover we formalize polynomial sequences and Negligible Functions.

2 Preparation

2.1 Mizar

Mizar[2], which formalizes mathematics, is an advanced project of the Mizar Society led by Andrzej Trybulec. The Mizar project was developed to describe mathematical proofs formally in the Mizar language. The

*Institute of Engineering, School of Science and Technology, Shinshu University, 4-17-1 Wsakasato, Nagano-City, Nagano 380-8553 Japan, okazaki@cs.shinshu-u.ac.jp

Mizar proof checker operates in both Windows and UNIX environments and registers proven definitions and theorems in the Mizar Mathematical Library(MML). The objective of this study is to prove the security of cryptographic systems using the Mizar proof checker.

2.2 Asymptotic notation

In this section, we briefly review an asymptotic notation $\mathcal{O}$, and we then introduce related formal definitions available in Mizar mathematical library.

Definition: 1 *Let $f(\cdot)$ and $g(\cdot)$ be functions from $\mathbb{N}$ to $\mathbb{R}$. $g(\cdot) \in \mathcal{O}(f(\cdot))$ iff $\exists N$ be a natural number and c be a real number s.t. $\forall n \text{ st } N \leq n$ holds $0 \leq g(n) \leq c \cdot f(n)$.*

The $\mathcal{O}$ notation is defined in Mizar as follows:

Definition: 2 (ASYMPT_0:def 9)

```
definition
  let f be eventually-nonnegative Real_Sequence;
  func Big_Oh(f) -> FUNCTION_DOMAIN of NAT,REAL
  equals
  { t where t is Element of Funcs(NAT, REAL)
    : ex c,N st c > 0 &
      for n st n >= N holds t.n <= c*f.n &
        t.n >= 0};
end;
```

In Mizar, Real_Sequence ia an alias for a function from $\mathbb{N}$ to $\mathbb{R}$ “eventually-nonnegative” is an attribute for Real_Sequence defined as follows:

Definition: 3 (ASYMPT_0:def 2)

```
definition
  let f be Real_Sequence;
  attr f is eventually-nonnegative means
  ex N being Nat st for n being Nat
    st n >= N holds f.n >= 0;
end;
```

The sequence of monomial a^n is defined in Mizar as follows:

Definition: 4 (ASYMPT_1:def 3)

```
definition
  let a be Real;
```

```

func seq_n^(a) -> Real_Sequence means
it.0 = 0 &
for n st n > 0 holds
    it.n = n to_power a;
end;

```

The sequence of monomial $a^{b \cdot n + c}$ (a, b, c are given constant numbers) is defined in Mizar as follows:

Definition: 5 (ASYMPT_1:def 1)

definition

```

    let a,b,c be Real;
    func seq_a^(a,b,c) -> Real_Sequence means
        it.n = a to_power (b*n+c);
end;

```

Note that for a given real number x , x^n is represented as “seq_a^(x,1,0)” in Mizar.

2.3 Polynomially-bounded Functions

In this section, we briefly review polynomially-bounded functions.

Definition: 6 *A real valued function $f(\cdot)$ is polynomially-bounded if $\exists n$ be an natural numbet such that*

$$f(x) \in \mathcal{O}(x^n)$$

2.4 Negligible Functions

In this section, we briefly review negligible functions[3]D

Definition: 7 *Let $\mu(\cdot)$ be a function from $\mathbb{N}$ to $\mathbb{R}$. $\mu(\cdot)$ is (negligible function) iff $\exists N$ be a natural number s.t. $\forall n$ be a natural number st $N \leq n$ holds $\forall p(\cdot)$ be a polynomial holds*

$$\mu(n) < \frac{1}{|p(n)|}$$

3 Formalization of Polynomially Bounded Functions and Polynomial Functions

In this section, we introduce our formal definition of polynomially bounded functions and polynomial functions.

3.1 Formalization of Polynomially Bounded Functions

Definition: 8

definition

```
let p be Real_Sequence;  
attr p is polynomially-bounded means  
ex k be Element of NAT st  
  p in Big_Oh(seq_n^(k));  
end;
```

We then introduce some theorems about polynomially bounded functions.

Theorem: 1

theorem

```
for a be Element of NAT st 1 < a holds  
  seq_a^(a,1,0) is non polynomially-bounded;
```

This theorem shows that exponential $f(x) = a^x$ is not polynomially-bounded if $1 < a$.

Theorem: 2

theorem

```
for a,b be Element of INT st b <> 0 holds  
  ex A,B be sequence of NAT,  
    C be Real_Sequence,  
    n be Element of NAT st  
    A.0 = |.a.| & B.0 = |.b.| &  
    (for i be Nat holds  
      A.(i+1) = B.i &  
      B.(i+1) = A.i mod B.i) &  
    n = (min*{i where i is Nat: B.i = 0} ) &  
a  gcd b  = A.n  
  & n <= C.(|. b .|)  
  & C is polynomially-bounded;
```

This theorem shows that Euclidean division algorithm calculates the GCD of two integers in polynomially-bounded repetitions.

3.2 Formalization of Polynomial Functions

Definition: 9

definition

```

let c be XFinSequence of REAL;
func seq_p(c) -> Real_Sequence
means
for x be Element of NAT holds
it.x = Sum(c (#) seq_a^(x,1,0));
end;

```

We then introduce a theorem about polynomial functions.

Theorem: 3

theorem

```

for k be Nat, c be XFinSequence of REAL
st len c = k+1 & 0 < c.k
holds seq_p(c) in Big_Oh( seq_n^(k) );

```

This theorem shows that any polynomial function whose most significant coefficient is positive is polynomially bounded.

4 Formalization of Negligible Functions

In this section, we introduce our formal definition of negligible functions.

Definition: 10

definition

```

let f be Function of NAT,REAL;
attr f is negligible
means@for c be non empty
positive-yielding XFinSequence of REAL holds
ex N be Element of NAT st
for x be Element of NAT
st N <=x holds |. f.x .| < 1/((seq_p(c)).x) ;
end;

```

We then introduce some theorems about negligible functions.

Theorem: 4

theorem

```

for f be Function of NAT,REAL
st for x be Element of NAT holds
@@f.x = 1/ (2 to_power x)@holds
f is negligible;

```

This theorem shows that 2^{-x} is a negligible function.

Theorem: 5

```
theorem
for f,g be Function of NAT,REAL
  st f is negligible & g is negligible
    holds f+g is negligible;
theorem
for f be Function of NAT,REAL,
  a be Real
  st f is negligible
    holds a(#)f is negligible;
theorem
for f,g,h be Function of NAT,REAL
  st f is negligible & g is negligible &
    h =f(#)g
    holds h is negligible;
```

These theorems show that negligible functions are closed with respect to addition, scalar multiplication and multiplication.

5 Conclusion

In this report, we introduce our formalization of polynomially-bounded functions, polynomial functions, and negligible functions in Mizar. We then show some related theorems about them. The definitions and theorems in this study have been verified for correctness using the Mizar proof checker.

参考文献

- [1] Jon Kleinberg, Eva Tardos, “Algorithm Design”, Addison-Wesley, 2005.
- [2] Mizar System: *Available at* <http://mizar.org/>.
- [3] Goldreich, “Foundations of Cryptography : Volume: 1 Basic Tools”, Cambridge University Press, 2001.
- [4] H. Okazaki, Y. Futa, Y. Shidama, “Formal definition of probability on finite and discrete sample space for proving security of cryptographic systems using Mizar”, Artificial Intelligence Research, 2(4), pp.37-48, 2013.
- [5] H. Okazaki, Y. Shidama, “Random Variables and Product of Probability Spaces”, Formalized Mathematics, 21(1), pp.33-39, 2013.

- [6] H. Okazaki, “Posterior Probability on Finite Set”, *Formalized Mathematics*, 20(4), pp.257-263, 2013.
- [7] H. Okazaki, Y. Shidama, “Probability Measure on Discrete Spaces and Algebra of Real Valued Random Variables”, *Formalized Mathematics*, 18(4), pp.213-217, 2010.
- [8] H. Okazaki, “Probability on Finite and Discrete Set and Uniform Distribution” , *Formalized Mathematics*, 17(2), pp.173-178, 2009.
- [9] H. Okazaki, Y. Shidama, “Probability on Finite Set and Real-Valued Random Variables”, *Formalized Mathematics*, 17(2), pp.129-136, 2009.
- [10] H. Okazaki, Y. Aoki, Y. Shidama, “Extended Euclidean Algorithm and CRT Algorithm”, *Formalized Mathematics* 20(2), pp-175-179, 2012.
- [11] Y. Futa, D. Mizushima, H. Okazaki, “ Formalization of Gaussian integers, Gaussian rational numbers, and their algebraic structures with Mizar”, *ISITA 2012*: pp.591-595, 2012.
- [12] Y. Futa, H. Okazaki, Y. Shidama, “Formalization of Definitions and Theorems Related to an Elliptic Curve Over a Finite Prime Field by Using Mizar”, *J. Autom. Reasoning* 50(2), pp.161-172 (2013).

Formalization of Polynomially-bounded Functions in Mizar

HiroYuki Okazaki
Shinhsu Univ.

motivation

Our Aim:

Subset of QED Project for cryptographers

- Proving security of cryptographic protocols
- Formalizing and evaluating cryptographic primitives
- Formalizing performance evaluation of cryptographic algorithms

motivation

- Proving the security of cryptographic systems by using Mizar
 - We want to achieve formal proof in a similar way cryptographers do,
 - Not on a specific model.

Previous research

- automated proof checking
 - CertiCrypt (on Coq)
 - BPW (on Isabelle/HOL)
 - EasyCrypt (on Coq)
- automated theorem proving
 - SCYTH
 - Proverif
 - CryptoVerif
 - AVISPA

Our Aim
Add "Mizar" in this list!

motivation

Our Aim:

Subset of QED Project for cryptographers

- Proving security of cryptographic protocols
- Formalizing and evaluating cryptographic primitives
- Formalizing performance evaluation of cryptographic algorithms

motivation

- Proving the security of cryptographic systems by using Mizar
 - We want to achieve formal proof in a similar way cryptographers do,
 - Not on a specific model.

Game-sequence proof

motivation

Our Aim:

Subset of QED Project for cryptographers

- Proving security of cryptographic protocols
- Formalizing and evaluating cryptographic primitives
- Formalizing performance evaluation of cryptographic algorithms

Related Topics

we must formalize for cryptography

- Probability
- Computational Complexity
- Algorithms
- Number Theory
- Information Theory
- etc.

Computational Complexity

• **Class P** _{easy}

• **Class NP** _{hard}

Class P

- Problem X is in class P if it takes **polynomially bounded** computation time to solve problem X
- There are feasible algorithms to solve X efficiently if problem X in P

Polynomially-bounded Functions

$f(x)$ is *polynomially – boubded*
iff

$$\exists n \in \mathbf{N} \text{ s.t. } f(x) \in O(x^n)$$

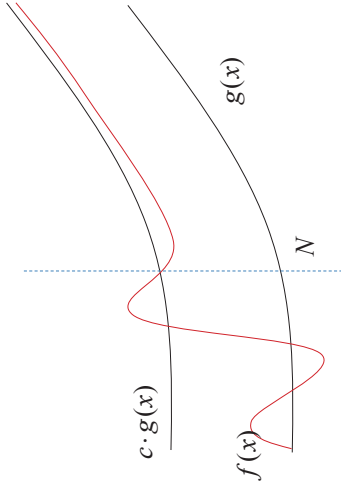
Asymptotic notation $O(*)$

$$f \in O(g)$$

iff

$$\left(\exists c, N \text{ s.t. } 0 < c \text{ \& } \forall x \text{ s.t. } N \leq x \text{ holds } f(x) \leq c \cdot g(x) \right)$$

Asymptotic notation $O(*)$



Related article in MML Asymptotic notation $O(*)$

```

definition
  let f be eventually-nonnegative Real_Sequence;
  func Big_Oh(f) -> FUNCTION_DOMAIN of NAT,
  REAL equals
  :: ASYMPT_0: def 9
  { t where t is Element of Funcs(NAT, REAL) :
    ex c, N st c > 0 & for n st n >= N holds t.n <= c*f.n
    & t.n >= 0 };
end;
```

Related article in MML Monomial sequence

$0, 1^a, 2^a, \dots, n^a, \dots$

```

definition
  let a be Real;
  func seq_n^a(a) -> Real_Sequence means
  :: ASYMPT_1: def 3
  it.0 = 0 & for n st n > 0 holds it.n = n to_power a;
end;
```

Polynomially-bounded Functions in Mizar

```

definition
  let p be Real_Sequence;
  attr p is polynomially-bounded means
  :: ASYMPT_2: def 1
  ex k be Element of NAT st p in
  Big_Oh(seq_n^k(k));
end;
```

Theorem

2^n is non polynomially-bounded

$$\forall x \in \mathbb{N} \text{ st. } 1 < x \text{ holds} \\ \neg \left(\exists c, N \in \mathbb{N} \text{ st. } \forall n \in \mathbb{N} \text{ st. } N \leq n \text{ holds } 2^n \leq c \cdot n^x \right)$$

theorem :: ASYMPT_2:18

for x be Element of NAT st $1 < x$ holds

not ex N,c be Element of NAT st

for n be Element of NAT st $N \leq n$ holds

$2 \text{ to_power } n \leq c * (n \text{ to_power } x)$;

Application to computational complexity

Theorem :: Euclidean division is P

for a,b be Element of INT st $b < 0$ holds

ex A,B be sequence of NAT,

C be Real_Sequence,

n be Element of NAT st

$A.0 = |.a.|$ & $B.0 = |.b.|$ &

(for i be Nat holds

$A.(i+1) = B.i$ &

$B.(i+1) = A.i \bmod B.i$) &

$n = (\min\{i \text{ where } i \text{ is Nat: } B.i = 0\})$ &

$A \text{ gcd } b = A.n$ &

$n \leq C.(|.b.|)$ & C is polynomially-bounded;

Related article in MML

Monomial sequence

$$0, 1^a, 2^a, \dots, n^a, \dots$$

definition

let a be Real;

func seq_n^a(a) -> Real_Sequence means

:: ASYMPT_1:def 3

it.0 = 0 & for n st $n > 0$ holds it.n = $n \text{ to_power } a$;

end;

Univariate Polynomial sequence

definition

let c be XFinSequence of REAL;

func seq_p(c) -> Real_Sequence

means

:: ASYMPT_2:def 2

for x be Element of NAT holds

it.x = $\text{Sum}(c \text{ \#} \text{ seq_a}^x(x,1,0))$;

end;

EXAMPLE

$f(x) = 5 + 4x^1 + 3x^2$
 $c = \langle 5, 4, 3 \rangle; \quad (c.0=5, c.1=4, c.2=3);$
 $c \text{ \#} \text{ seq_a}(x, 1, 0)$
 $= \langle (c.0)^*x^0, (c.1)^*x^1, (c.2)^*x^2 \rangle$
 $= \langle 5^*x^0, 4^*x^1, 3^*x^2 \rangle$
 $(\text{seq_p}(c)).x = 5 + 4^*x^1 + 3^*x^2;$

Polynomial is Polynomially-bounded

theorem :: ASYMPT_2:54
 for k be Nat,
 c be XFinSequence of REAL
 st $\text{len } c = k+1$ & $0 < c.k$
 holds $\text{seq_p}(c)$ in Big_Oh($\text{seq_n}(k)$);

Negligible Functions

ある $N \rightarrow \mathbf{R}$ である関数 $\mu(\cdot)$ について
 任意の多項式 $p(\cdot)$ に対して、
 ある自然数 N が存在し、
 $N \leq n$ なる任意の自然数 n について

$$\mu(n) < \frac{1}{|p(n)|}$$
 であるとき $\mu(\cdot)$ は無視できるほど小さい関数である

Negligible Functions

definition
 let f be Function of NAT, REAL;
 attr f is negligible
 means
 :defneg:
 for c be non empty positive-yielding XFinSequence of REAL
 holds
 ex N be Element of NAT
 st
 for x be Element of NAT
 st $N \leq x$ holds $|f.x| < 1/((\text{seq_p}(c)).x)$;
 end;

$\frac{1}{2^x}$ is negligible

theorem

for f be Function of NAT, REAL st
for x be Element of NAT holds

$$f.x = 1 / (2^{\text{power } x})$$

holds f is negligible

Future Work using negligibility

- Roughly saying, a cryptosystem is secure if the "probability of attack against the cryptosystem succeeds" is negligible.
- Game sequence proof is based on "indistinguishability", and "indistinguishability" is defined using negligibility

Semantic Secure

- Secret key s
- $\text{dec}(\text{enc}(m,s),s) = m$
- Choice message $m = 0$ or 1
- $C = \text{enc}(m,s)$
- Adversary guesses the message is X
- $|\text{Prob}(X=m) - \text{Prob}(X=m) \mid C|$ is negligible then $\{\text{enc}, \text{dec}\}$ is Semantic Secure

Indistinguishability

- Functions f and g are indistinguishable if $|f-g|$ is negligible function

EXAMPLE

Pseudo-random number sequence is defined as a sequence indistinguishable from true-random number sequence

Game based proof

- Design a protocol represented using ideal functionality : GAME0
- Replace Ideal functionality into Feasible algorithm :GAME 1
- Show GAME0 and GAME1 are indistinguishable

Thank you
for your listening