ORIGINAL RESEARCH

Formal definition of probability on finite and discrete sample space for proving security of cryptographic systems using Mizar

Hiroyuki Okazaki[1], Yuichi Futa[2], Yasunari Shidama[3]

1. Graduate School of Science and Technology, Shinshu University, Nagano, Japan. 2. School of Information Science, Japan Advanced Institute of Science and Technology, Ishikawa, Japan. 3. Department of Information Engineering, Faculty of Engineering, Shinshu University, Nagano, Japan

Correspondence: Hiroyuki Okazaki. Address: Graduate School of Science and Technology, Shinshu University, 4-17-1 Wakasato, Nagano, 380-8553, Japan. Telephone: 81-26-269-5503. Email: okazaki@cs.shinshu-u.ac.jp

Received: May 2, 2013 **Accepted:** June 13, 2013 **Online Published:** August 2, 2013
DOI: 10.5430/air.v2n4p37 **URL:** http://dx.doi.org/10.5430/air.v2n4p37

Abstract

Security proofs for cryptographic systems are very important. The ultimate objective of our study is to prove the security of cryptographic systems using the Mizar proof checker. In this study, we formalize the probability on a finite and discrete sample space to achieve our aim. Therefore, we introduce a formalization of the probability distribution and prove the correctness of the formalization using the Mizar proof checking system as a formal verification tool.

Key words

Formal verification, Probability on a finite and discrete sample space, Cryptographic system

1 Introduction

Probability on a finite and discrete sample space plays 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. Recently, lattice cryptosystems that use complexity based on lattice problems have been attracting attention in the cryptographic field. Because lattice cryptosystems have homomorphism of addition and multiplication (full homomorphism), they are used as information protection in cloud computing. Learning with error was introduced to prove the security of lattice cryptosystems [1]. A security proof is performed by proving that an attack on a target cryptosystem is more difficult than a complexity problem. The security proof of lattice cryptosystems uses indistinguishability between probability distributions of cipher texts and random values. Because a modulus of an integer value is used in the implementation of cryptosystems, discrete probability distribution is used in security proofs. Discrete probability distribution is also used in security proofs for hash functions, pseudorandom functions, and the cryptographic schemes that use these functions. Therefore, the formalization of discrete probability distributions is essential for the formalization of cryptosystem security proofs, especially lattice cryptosystems.

Mizar [2, 3], 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 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. To achieve this, we formalize the probability on a finite and discrete sample space.

The remainder of this paper is organized as follows. In Section 2, we review the definitions of probability that have already been formalized in Mizar. In Sections 3-4 we propose a formalization of probability on a finite and discrete sample space. Section 5 describes our proposed definition of posterior probability on a finite and discrete sample space. Section 6 presents our plan to achieve proving security of cryptologic systems using our formalization of probability distributions, and Section 7 presents conclusions and suggestions for future work. The definitions and theorems in this study have been verified for correctness using the Mizar proof checker.

2 Related articles in Mizar mathematical library

This section reviews the definitions of probability that have already been formalized in Mizar. In Mizar, the definition of probability has been formalized as a function from σ-field of Ω to R [4]. However, although the probability of events has been formalized [5], there is still no method to treat the probability of concrete events. Because discussing the probability of concrete events is necessary when analyzing a security proof of cryptographic systems, we propose a formal definition of probability on a finite and discrete sample space that can also treat the probability of concrete events [6, 7]. An additional objective of the Mizar project is to create a check system for mathematical theses. An "article" formalizes and describes a Mizar mathematical proof. When an article is newly described, it is possible to advance it by referring to the articles registered in the MML, which have already been inspected as proofs. Similarly, other articles can refer to an article after it is registered in the MML. Although the Mizar language is based on the description method for general mathematical proofs, the reader should refer to the references for syntactical details, because Mizar uses specific and unique notation [2, 3].

Andrzej Nedzusiak defined probability as follows [4].

DEFINITION 2.1 (Basic definition of probability)

Let Ω be a non-empty set (not necessarily finite and discrete) and σ be a σ-field of over Ω. Then, the mode probability P of σ yielding a function from σ into R is defined by

- For subset A of Ω such that A in σ holds $0 \leqslant P(A)$,
- $P(\Omega) = 1$,
- For all subsets A, B of Ω such that $A,B \in \sigma$ and A misses B holds $P(A \cup B) = P(A) + P(B)$,
- For every sequence ASeq of subsets of Ω such that ASeq. $i \in \sigma$ (for all $i \in N$) and ASeq is non-increasing holds, P*ASeq is convergent and lim(P*ASeq) = P(Intersection ASeq).

Note that P*ASeq is a sequence of real numbers and P(A) is a real number. Various theorems related to Definition 2.1 have already been proven [4, 5].

Furthermore, Mizar has a concept of mode. Mode is a variant of the Pascal language and contains the definition of a general mode. Moreover, Nat, Real, and others are representative modes. Users can also define a mode [2, 3]. Bo Zhang and Yatsuka Nakamura formalized a finite sequence of probability as follows [9].

DEFINITION 2.2 (Definition of finite probability finite sequence)

Let p be a finite sequence of elements of R and i be an element of N. Then, p is a finite probability finite sequence if and only if

For every i such that i ∈ dom p holds p(i) > 0 and $\sum$ p =1.

Various theorems related to Definition 2.2 have also already been proven [9, 10]. However, Definitions 2.1 and 2.2 are considered axiomatic to capture concrete events. Jan Popiolek, in contrast, formalized probability as follows [5].

DEFINITION 2.3. (Definition of prob)

Let E be a finite non-empty set and A be an event of E. Then, the functor prob(A) yields a real number defined as follows:

prob(A) = (card A)/(card E).

Here, an event of E is a subset of E and card A is a cardinal number of A. Various theorems related to Definition 2.3 have already been proven [5]. However, it is difficult for Definition 2.3 to treat the probability of concrete events.

Works related to the present research is classified into two categories. The first category describes the axiom of probability and proves some propositions of probability [4, 8]. The second category assigns probability to all events using a particular principle and proves its correctness [5]. These categories define probability as the principle of equally possible cases.

In this study, we propose a measure function that satisfies both categories. The function holds the property of the probability measure given in Definition 2.1, and its values are real numbers that hold the probability of events given in Definition 2.3.

3 Formalization of probability distribution

Probability distribution means a specification of probabilities that govern the value of random variables. This is either the same as a probability distribution function or is understood as something more fundamental underlying an actual mass or density function. Mathematicians may be more familiar with the former approach than the latter approach. In the former approach, formalizations of probability distribution functions are being actively researched [11, 12]. On the other hand, when discussing probability, cryptographers often use the term "(probability) distribution" to mean something different. For example,

Pick an element x from the uniform distribution of the given cyclic group G.

In this case, computer scientists, including cryptographers, tend to consider probability distribution as the latter approach: "something more fundamental underlying an actual mass or density function."

The latter approach is convenient to capture probability, probabilistic events, and probabilistic functions in computer complexity theory; however, computer scientists sometimes do consider probability distribution as the former approach. Cryptographers sometimes confuse "probability distribution" with something like "random variable," and consider "probability distribution" as a finite sequence.

The aim of this study is to propose a formalization of the cryptographer's recognition of probability distribution, i.e., something to handle random variable, probabilistic events, and a probability distribution function based on experience or number counting.

3.1 Basic concept of probability distribution formalization

In a finite and discrete case, we can intuitively attempt to capture "something about probability" by enumerating all elements. For example, suppose there is a bag that contains red and white balls, and the ratio between the number of red and white balls is 2:1. In this case, we can enumerate the balls in the bag such as {r, r, w}. Thus, we can achieve the

probability distribution function as the ration of red and white balls according to both Definition 2.3 and our understanding. For example, by using sequences of enumerating all elements, we can prove many theorems about cryptology applications, such as "ball and bins" and "birthday problems [13]". Note that there may be many other finite sequences equivalently distributed. For example, {r, r, w, w, r, r} and {w, w, r, r, r, r, r, w}. Cryptographers roughly consider such finite sequences as random variables. On the other hand, in a mathematical sense, finite sequences of S are defined as functions from a subset of I into S, where S is a set. In measure theory, random variables are defined as measurable functions.

In this section, we first introduce the measure-theoretical definition of random variables in the MML and then demonstrate that finite sequences are random variables. Next, we will formalize a method that can systematically determine the probability of the mass function from a given finite sequence.

3.2 Finite sequences as random variables

3.2.1 Finite sequence

Let S be a set; a finite sequence of S, $X_I = (x_1,..., x_i,...)$ is an indexed list of elements whose i-th element is $X_I \in S$ and $i \in I \subset N$. We can consider X_I as a function from I to S. In the MML, a finite sequence is defined as follows [17].

DEFINITION 3.1. (Segment)

Let n be a natural number. Then, the functor Seg(n) yielding a set is defined as follows:

$$\text{func Seg (n)} \rightarrow \text{set equals } \{ k \text{ where k is a natural number: } 1 \leqslant k \,\&\, k \leqslant n \}.$$

DEFINITION 3.2. (Finite Sequence)

Let F be a function. Then, the function f is FinSequence if and only if n exists as a natural number such that dom F = Seg(n).

3.2.2 Random variables

In measure theory, random variables are defined as functions. In the MML, a random variable is defined as follows [14].

DEFINITION 3.3. (Random Variables)

Let Ω_1 and Ω_2 be sets, σ_1 be a σ-Field of Ω_1, σ_2 be a σ-Field of Ω_2, and X be a function. Then, the function X is a random variable of σ_1, σ_2 if and only if:

$$\text{for x being any element of } \sigma_2 \text{ holds } X^{-1}(x) \text{ is an element of } \sigma_1.$$

We then prove the following theorem [15].

THEOREM 3.4. (Relationship between finite sequences and random variables)

The following proposition is true:

Let S be a non-empty finite set, and F be a non- empty finite sequence of elements of S. Then, F is a random variable of Trivial-SigmaField (Seg len F), Trivial-SigmaField (S), where Trivial-SigmaField(X) is the functor that yields the set of subsets of X [23]. Note that the Trivial-SigmaField (X) is the largest σ-Field of X.

Thus, we can conclude that a finite sequence is a random variable.

3.3 Formalization of finite and discrete probabilities

3.3.1 Probability of elements

First, we define the functor "FDprobability". Let S be a non-empty finite set as a sample space and s be a finite sequence of elements of S. Therefore, we consider s as enumerated elements of S in a random order. Next, we define FDprobability(x,s) as the probability Pr(x). FDprobability is defined as follows.

DEFINITION 3.5. (Definition of FDprobability)

Let S be a non-empty finite set, s be a finite sequence of elements of S, and x be a set. Then, the functor FDprobability(x,s) yielding a real number is defined as follows:

$$\text{FDprobability}(x,s) = (\text{card } s^{-1}(x)) / (\text{len } s).$$

Here, len s is the length of s and $s^{-1}(x) = \{i \in \text{dom } s : s(i) = x\}$ [16]. Note that dom s is a domain of s and $s^{-1}(x)$ is a subset of dom s.

We also prove the following theorem to verify the accuracy of our formalization.

LEMMA 3.6.

The following proposition is true:

Let S be a non-empty finite set and s be a finite sequence of elements of S. Then,

$$\text{card } (\text{dom } s) = \text{len } s.$$

THEOREM 3.7. (Relationship between prob and FDprobability)

The following proposition is true:

Let S be a non-empty finite set, s be a finite sequence of elements of S, and x be a set. Then,

$$\text{prob}(s^{-1}(x)) = \text{FDprobability}(x,s).$$

Thus, we can obtain the relationship between the probability defined in Definition 2.3 and FDprobability.

3.3.2 Probability mass functions

Next, we define the probability mass function on a finite and discrete sample space as a finite sequence of FDprobability and denote it as "FDprobSEQ."

DEFINITION 3.8. (Definition of FDprobSEQ)

Let S be a non-empty finite set and s be a finite sequence of elements of S. Then, the functor FDprobSEQ s yielding a finite sequence of elements of R is defined by

$$\text{dom } (\text{FDprobSEQ } s) = \text{Seg}(\text{card } S).$$

For every natural number n such that $n \in \text{dom } (\text{FDprobSEQ } s)$ holds

$$(\text{FDprobSEQ } s)(n) = \text{FDprobability}((\text{canFS}(S))(n),s).$$

Here, Seg(card S) = [1, card S] and canFS(S) is a bijective function from Seg(card S) onto S [17, 18].

We also prove the following theorem to verify the correctness of our formalization.

LEMMA 3.9. (Lemma of theorem 3.10)

The following proposition is true:

For every non-empty finite set S and non-empty finite sequence s of elements of S holds

$$\sum \text{FDprobSEQ } s = 1.$$

THEOREM 3.10. (Relationship between the finite probability finite sequence and FDprobSEQ)

The following proposition is true:

Let S be a non-empty finite set and s be a non-empty finite sequence of elements of S. Then, FDprobSEQ s is a finite probability finite sequence.

Thus, we can obtain the relationship between the finite probability finite sequence defined in Definition 2.2 and FDprobSEQ. Note that (FDprobSEQ s) is a finite sequence of S, i.e., a function from (Seg len s) to S. However, we can achieve the probability mass function as $(\text{FDprobSEQ } s)*(\text{canFS}(S))^{-1}$.

3.4 Probability distribution

A finite sequence, i.e., enumerated elements of a sample space, represents a random variable in our formalization. Naturally, the probability does not depend on the particular enumeration of the elements. Thus, we formalize the following definitions of the equivalence of random variables.

DEFINITION 3.11. (Definition of probability equivalent)

Let S be a non-empty finite set and s and t be finite sequences of elements of S. Then, s and t are probability equivalents if and only if:

For every element x holds FDprobability(x,s) = FDprobability(x,t).

DEFINITION 3.12. (Definition of equivalence class)

Let S be a non-empty finite set and s be a finite sequence of elements of S. Then, the equivalence class of s yielding a non-empty subset of S* is defined by

The equivalence class of s equals

{t; t ranges over finite sequences of elements of S: s and t are probability equivalents}

Here, S* is the set of all finite sequences of S.

Next, we define "distribution family" as an "equivalence class" as follows.

DEFINITION 3.13. (Definition of distribution family)

Let S be a non-empty finite set. Then, the distribution family of S yielding a non-empty family of subsets of S* is defined as follows.

Let A be a subset of S*. Then, A is an element of the distribution family of S if and only if there exists a finite sequence s of elements of S such that A is the equivalence class of s.

In cryptology, the "distribution ensemble" concept is similar to our "distribution family" concept. However, we defined distribution family more generally than distribution ensemble [19]. The definition of distribution ensemble is not sufficiently formal to be defined in Mizar. Distribution ensembles is a family of distributions or random variables X equals $\{X_i\}$ i in I, where I is a (countable) index set, and each X_i is a random variable. Each X_i must have a sufficient large property for i. For example, "i is sufficiently large" is unsuitable for a formal definition; we must define the meaning of sufficiently large. Moreover, we must consider the properties of random variables that are members of a distribution ensemble. In our definition, we employ finite sequences as random variables. We believe that the length of such finite sequences should also be sufficiently large. To handle the concept of sufficiently large, we are attempting to formalize indistinguishability and negligibility. We must redefine distribution ensemble to be better suited for formalization for use in proofs. Thus, we have defined a distribution family more generally than a distribution ensemble. We propose a definition of distribution ensemble as a special case of distribution family.

Note that $\{\} \in S^*$ and {} is the finite sequence whose length is 0. We propose the following definition.

DEFINITION 3.14. (Definition of well distributed)

Let S be a non-empty finite set and D be an element of the distribution family of S. Then, D is well distributed if and only if for every element s of D, s is non-empty.

Thus, "well distributed" means that an element s of an element D of the distribution family of S is non-empty. Consequently, we can achieve the definition of Probability Distribution as follows.

DEFINITION 3.15. (Definition of Probability Distribution)

Let S be a non-empty finite set. Then, the Distribution of S is defined as a well distributed element of the distribution family of S.

4 Formalization of probability

In this section, we propose a definition of a probability measure on a finite and discrete sample space.

4.1 Probability of events

In Section 3, we introduced our definition of FDprobability and the probability mass function on a finite and discrete sample space. However, we have not formalized probability except for the event "X = x", i.e., FDprobability(x,s) = Pr(x). When random variable s is given, we consider the domain of s as the sample space. By Definition 2.3, we consider an event as a subset of sample space of domain of s. However, it is not useful to specify an event as a subset directly because we would sometimes like to express events by conditions. To formalize the probability of any event, we employ a model often used by cryptographers to represent process or communication procedures. For example, randomly pick an element x from Ω and invoke a process, namely a checking oracle machine (CO), to judge whether x meets a certain condition. The CO then outputs either TRUE or FALSE. Note that the composition of a CO and a finite sequence is a finite sequence of BOOLEAN. Thus, we can adopt our definition of FDprobability to formalize the probability of events using the above

mentioned model. Next, we define the functor Prob(CO,s) as the probability of x such that CO(x) = TRUE. The functor Prob is defined as follows.

DEFINITION 4.1. (Definition of Prob)

Let S be a non-empty finite set, D be a well distributed element of the distribution family of S, s be an element of D, and CO be a function from S into BOOLEAN. Then, the functor Prob(CO,s) yielding a real number is defined as follows:

$$\text{Prob(CO, s)} = \text{FDprobability (TRUE, CO * s)}$$

Note that the composition of the CO and s is a finite sequence of BOOLEAN.

We then prove the following theorem to verify the correctness of our formalization.

THEOREM 4.2. (Relationship between Prob and FDprobability)

The following proposition is true:

Let S be a non-empty finite set, D be a well distributed element of the distribution family of S, X be an element of S, s be an element of D, and CO be a function from S into BOOLEAN such that for every set x holds x = X if and only if CO(x) = TRUE. Then, Prob(CO,s) = FDprobability(X,s).

Thus, we can obtain the relationship between Prob defined in Definition 4.1 and FDprobability. As a result, we can formalize the probability of any events by substituting the CO of the functor Prob.

Moreover, we prove the following theorem about the functor Prob(CO,s) in Definition 4.1.

THEOREM 4.3.

The following proposition is true:

Let S be a non-empty finite set, D be a well distributed element of the distribution family of S, X be an element of S, s be an element of D, and f,g be functions from S into BOOLEAN holds:

$$\text{Prob(('not' f),s)=1- Prob(f,s)}$$

$$\text{Prob((f 'or' g),s) = Prob(f,s) + Prob(g,s) - Prob((f '\&' g),s)}$$

$$\text{Prob((f '\&' g),s) = card } ((f^{*}s)^{-1}\text{ (TRUE) } \cup \text{ } (g^{*}s)^{-1}\text{ (TRUE))/(len s)}$$

where 'not', 'or', and '&' are logical operators on BOOLEAN-yielding functions [20, 21]. We have already formalized the product probability measure on the Cartesian product of discrete spaces [22]. However, Theorem 4.3 shows that we can more easily discuss probabilistic events.

Next, we define the functor Prob as follows.

DEFINITION 4.4. (Definition of Prob)

Let S be a non-empty finite set, D be a well distributed element of the distribution family of S, and CO be a function from S into BOOLEAN. Then, the functor Prob(CO,D) yielding a real number is defined as follows:

Prob(CO,D) means that for s being an Element of D holds Prob(CO,s)

 ISSN 1927-6974 E-ISSN 1927-6982

Note that the functor Prob(CO,s) is given in Definition. 4.1.

4.2 Probability measure

Next, we propose a formal definition of a probability measure function over a distribution.

DEFINITION 4.5. (Definition of Probability Measure)

Let S be a non-empty finite set, D be a Distribution of S, and P be a real-valued function on Funcs(S,BOOLEAN). Then, P is the mode Probability of D if and only if:

For every f that is an element of Funcs(S,BOOLEAN) holds P.f = Prob(f,D)

This definition is considerably generalized. However, it is convenient for our future work. Thus, we propose another definition of a probability measure as follows.

DEFINITION 4.6. (Definition of Trivial Probability)

Let S be a non-empty finite set and D be a distribution of S. Then, the functor Trivial-Probability (D) yields a Probability of Trivial-SigmaField (S) defined as follows.

- For every x ∈ Trivial-SigmaField (S) holds chi exists and is the BOOLEAN-valued function on S such that chi(x) = TRUE if x ∈ S, otherwise chi(x) = FALSE.
- P(x) = Prob(chi,D).

Note that we have proven Trivial-Probability (D) has the property of the probability measure given in Definition 2.1.

5 Posterior probability

In this section, we propose a definition of posterior probability on a finite and discrete sample space.

5.1 Preparation

First, we define functors to formalize posterior probability.

DEFINITION 5.1. (Extracting Finite Sequence)

Let S be a set, s be a finite sequence of elements of S, and A be a Subset of (dom s). Then, the functor extract(s,A) yielding a finite sequence of elements of S is defined as follows.

extract(s,A) = s*(canFS(A))

Extract(s,A) is the reduced s with domain limited its domain to A.

DEFINITION 5.2.

Let S be a set, and SS be a subset of S. Then, the functor MembershipDecision(SS) yielding a function from S into BOOLEAN is defined as follows.

For every x ∈ S holds (MembershipDecision(SS))(x) = TRUE if x ∈ SS, otherwise (MembershipDecision(SS))(x) = FALSE.

DEFINITION 5.3.

Let S be a non-empty finite set, D be a Distribution of S, T be a non-empty subset of S. Then, T is a mode samplingRNG of D if and only if:

$$\text{For every } s \in D \text{ holds } T \cap (\text{rng } s) \text{ is non-empty}$$

DEFINITION 5.4. (Conditional Subset)

Let S be a non-empty finite set, D be a Distribution of S, and X be a samplingRNG of D. Then, the functor ConditionalSS(X) yielding a Distribution of S is defined as follows.

$$\text{func ConditionalSS (X)} \rightarrow \text{set equals } \{ t \text{ where } t \text{ is a finite sequence of S: } t = \text{extract}(s,(s^{-1}(X)))\}$$

where s be an finite sequence such that $s \in D$.

5.2 Formalization of posterior probability

In this section, we propose a definition of posterior probability using the functors defined in Section 5.1 and the functor Prob given in Definition 4.4.

DEFINITION 5.5. (Posterior Probability)

Let S be a non-empty finite set, D be a Distribution of S, X be a samplingRNG of D, and f be a function from S into BOOLEAN. Then, the functor Prob(f,X) yielding a real number is defined as follows:

$$\text{Prob}(f,X)=\text{Prob}(f,\text{ConditionalSS}(X))$$

Note that the functor Prob(f, ConditionalSS(X)) on the right side of this equation is given in Definition 4.4.

Moreover, we prove the following theorem about our formalization of posterior probability.

THEOREM 5.6.

The following proposition is true:

Let S be a non-empty finite set, D be a Distribution of S, X be a samplingRNG of D, and f be a function from S into BOOLEAN. Then,

$$\text{Prob}(f,X) * \text{Prob}(\text{MembershipDecision}(X),D) = \text{Prob}(f \text{ '\&' } (\text{MembershipDecision}(X)),D)$$

Note that this theorem is essentially the same as the Bayes theorem.

6 Overview of proving security of cryptologic systems

In this section, we briefly describe our future plans for formalizing the security of cryptologic systems. As explained in Section 3, probability distribution is formalized as a given probability distribution function in other similar proof checking systems. This approach has been used to prove the security of some cryptographic schemes using other similar proof checking systems [24, 25]. However, these proofs are based on given axioms or theorems that must be proven in a different manner. For example, to prove the security of a concrete cryptographic system, we cannot avoid numeric analysis of probabilistic algorithms and functions. Therefore, we proposed a constructive method to formalize probability distribution using finite sequences to analyze the numeric properties of the given probability distribution. Moreover, we can handle a

probability distribution as an element. Consequently, we can formalize a probabilistic function as a function that yields a probability distribution. Cryptologic systems are often expressed using probability functions. Probability distribution is fundamental principal to grasp cryptology. We can define other cryptologic topics using our formalization of probability distribution. Currently, we are attempting to formalize the indistinguishability of probability distributions. In modern cryptology, the security of cryptologic systems is essentially defined by indistinguishability. For example, let enc be a probabilistic encryption function, r be a random value, and c be a cipher texts calculated by enc. If there is no efficient algorithm that can discern between r and c, enc holds indistinguishability, and knowledge of cipher texts does not provide an advantage. Moreover, various cryptologic topics, such as pseudo-random number generator and hash functions, are described using the concept of indistinguishability. We often design a cryptographic scheme by employing ideal functions. However, we must replace such ideal functions with feasible functions when we implement such schemes. Thus, the implemented scheme is not always secure even if its ideal functionality is proven secure in design.

Indistinguishability of functions means the indistinguishability of the distributions of the outputs of functions between an ideal function (e.g., an ideal random function) and a feasible function (e.g., a hash function) to prove security of cryptographic schemes. In future, we intend to define a formalization of indistinguishability using our formalization of probability. Moreover, the results of continuous probability distribution are used in cryptosystem security proofs by regarding discrete probability distribution as continuous. To strictly formalize this, it is important to prove that the difference between discrete and continuous probability distributions is negligible. We plan to prove that this difference is negligible, and the difference between probability distributions of cipher texts and random values is negligible for a security parameter.

We can then show a more formal definition of indistinguishability of probability distribution ensembles as follows.

DEFINITION 6.1. (Computational Indistinguishability)

Let X and Y be distribution ensembles, and n be the given security parameter. X and Y are computationally indistinguishable if, for any x in X and y in Y, the polynomial-time boolean function D holds.

$$| \Pr(A(x) = 1) - \Pr(A(y) = 1) | \leqslant e(n)$$

where e(n) is a real value polynomially bounded by n. However, such a definition is not sufficiently formal to be defined in Mizar. We are attempting to modify these definitions to be more suitable for formal proving systems. To prepare for future work, our formalization of the posterior probability proposed in Section 5 employs a model that is similar to the model of computational indistinguishability mentioned above.

7 Conclusion

In this study, we introduced a formalization of probability distribution on a finite and discrete sample space. We also proposed a formalization of posterior probability. We proved the correctness of our formalization using the Mizar proof checking system as a formal verification tool. These formalizations are very useful for formalizing some cryptologic issues. Currently, we are attempting to formalize the indistinguishability of probability distributions. We will also formalize this negligibility.

Acknowledgements

The first author was partly supported by JSPS KAKENHI 21240001 and the third author was partly supported by JSPS KAKENHI 22300285.

References

[1] Regev. On lattices, learning with errors, random linear codes, and cryptography, Proceedings of the thirty-seventh annual ACM symposium on Theory of computing (Baltimore, MD, USA: ACM, 2005). 2005: 84-93, http://dx.doi.org/10.1145/1060590.1060603

[2] Mizar Proof Checker, Available from: http://mizar.org/

[3] Bonarska. An Introduction to PC Mizar, Mizar Users Group, Fondation Philippele Hodey, Brussels. 1990.

[4] Nedzusiak. σ -Fields and Probability, Formalized Mathematics. 1990; 1(2): 401-407.

[5] Popiolek. Introduction to Probability, Formalized Mathematics. 1990; 1(4): 755-760.

[6] Okazaki. Probability on Finite and Discrete Set and Uniform Distribution, Formalized Mathematics. 2009; 17(2): 173-178. http://dx.doi.org/10.2478/v10037-009-0020-z

[7] Okazaki. Posterior Probability on Finite Set, Formalized Mathematics. 2013; 20(4): 257-263, http://dx.doi.org/10.2478/v10037-012-0030-0

[8] Nedzusiak. Probability. Independence of Events and Conditional Probability, Formalized Mathematics. 1990; 1(4): 745-749.

[9] Zhang & Nakamura. The Definition of Finite Sequences and Matrices of Probability, and Addition of Matrices of Real Elements, Formalized Mathematics. 2006; 14(3): 101-108. http://dx.doi.org/10.2478/v10037-006-0012-1

[10] Zhang & Nakamura. Definition and some Properties of Information Entropy, Formalized Mathematics. 2007; 15(3): 111-119. http://dx.doi.org/10.2478/v10037-007-0012-9

[11] Audebaud & Paulin-Mohring. Proofs of randomized algorithms in Coq, Sci. Comput. Program. 2009; 74(8): 568-589. http://dx.doi.org/10.1016/j.scico.2007.09.002

[12] Hasan & Tahar. Formalization of Continuous Probability Distributions, Automated Deduction CADE-21, Lecture Notes in Computer Science. 2007; 4603: 3-18. http://dx.doi.org/10.1007/978-3-540-73595-3_2

[13] Shoup. A Computational Introduction to Number Theory and Algebra 2nd Edition, Cambridge University Press. 2008.

[14] Jaeger. Elementary Introduction to Stochastic Finance in Discrete Time, Formalized Mathematics. 2012; 20(1): 1-5. http://dx.doi.org/10.2478/v10037-012-0001-5

[15] Okazaki & Shidama (in press), Random Variables and Product of Probability Spaces, Formalized Mathematics. 2013; 21(1): 33-39.

[16] Bylmski. Functions and Their Basic Properties, Formalized Mathematics. 1990; 1(1): 55-65.

[17] Bancerek & Hryniewiecki. Segments of Natural Numbers and Finite Sequences, Formalized Mathematics .1990; 1(1): 107-114.

[18] Rudnicki. Little Bezout Theorem (Factor Theorem), Formalized Mathematics. 2004; 12(1): 49-58.

[19] Goldreich. Foundations of Cryptography: Volume: 1 Basic Tools, Cambridge University Press. 2001.

[20] Woronowicz. Many-Argument Relations, Formalized Mathematics. 1990; 1(4): 733-737.

[21] Kobayashi & Jia. A Theory of Boolean Valued Functions and Partitions, Formalized Mathematics. 1998; 7(2): 249-254.

[22] Okazaki & Shidama. Probability Measure on Discrete Spaces and Algebra of Real-Valued Random Variables, Formalized Mathematics. 2010; 18(4): 213-217. http://dx.doi.org/ 10.2478/v10037-010-0026-6

[23] Okazaki & Shidama. Probability on Finite Set and Real-Valued Random Variables, Formalized Mathematics. 2009; 17(2): 129-136. http://dx.doi.org/10.2478/v10037-009-0014-x

[24] Barthe et al. Interactive Theorem Proving, Lecture Notes in Computer Science. 2012; 7406: 11-27. http://dx.doi.org/10.1007/978-3-642-32347-8-2

[25] Bella. Formal Correctness of Security Protocols, Springer. 2007.