An Anonymous Credential System with Constant-Size Attribute Proofs for CNF Formulas with Negations

Ryo Okishima and Toru Nakanishi(✉)

Graduate School of Engineering, Hiroshima University,
Higashi-Hiroshima, Hiroshima, Japan
{m186746,t-nakanishi}@hiroshima-u.ac.jp

Abstract. To enhance the user's privacy in electronic ID, anonymous credential systems have been researched. In the anonymous credential system, a trusted issuing organization first issues a certificate certifying the user's attributes to a user. Then, in addition to the possession of the certificate, the user can anonymously prove only the necessary attributes. Previously, an anonymous credential system was proposed, where CNF (Conjunctive Normal Form) formulas on attributes can be proved. The advantage is that the attribute proof in the authentication has the constant size for the number of attributes that the user owns and the size of the proved formula. Thus, various expressive logical relations on attributes can be efficiently verified. However, the previous system has a limitation: the proved CNF formulas cannot include any negation. Therefore, in this paper, we propose an anonymous credential system with constant-size attribute proofs such that the user can prove CNF formulas with negations. For the proposed system, we extend the previous accumulator for the limited CNF formulas to verify CNF formulas with negations.

Keywords: Anonymous credentials · Accumulator · Pairing · Attributes

1 Introduction

1.1 Backgrounds

Electronic identity (eID) such as eID card is often used for physical user authentication for entering buildings, use of facilities and so on, and furthermore it can be used for network-based user authentication in Web services. In eID, in addition to the user's ID, the user's attributes such as gender, occupation, and birth date are authorized, and thus the attribute-based authentication using the eID can be performed. However, one of serious problem in the existing eID system is the user's privacy: Since the eID may reveal the user's unique ID, the verifier

N. Attrapadung and T. Yagi (Eds.): IWSEC 2019, LNCS 11689, pp. 89–106, 2019.
https://doi.org/10.1007/978-3-030-26834-3_6

can collect the user's history. As the solution, the anonymous credential system was proposed [9].

In the anonymous credential system, an issuer issues each user a certificate. The certificate is a proof of membership, qualification, or privilege, and ensures the user's own attributes. The user with the certificate can anonymously convince a service provider (SP) of the possession of the certificate. Additionally, the user can prove the possession of attributes, and furthermore a logical relation on the attributes. By the AND relation, the user can prove the possession of all attributes in the relation. By the OR relation, the user can prove the possession of one attribute from the attributes in the relation. As the advantage of the anonymous credential system with attribute proofs, it does not leak any other information beyond the satisfaction of the proved relation.

1.2 Previous Works

In [7,11,14], anonymous credential systems with attribute proofs have been proposed, where the proof size is constant for the number of user's attributes and the size of proved logical relation. However, available relations are only AND or OR relations on attributes. In [12], an anonymous credential system with attribute proofs of constant size has been proposed, where inner product relations on attributes can be proved. This means that CNF (Conjunctive Normal Form) and DNF (Disjunctive Normal Form) formulas are available by using polynomial-based encoding. However, this system has a problem of the computational cost: The proof generation requires the exponentiations depending on the number of OR literals in the proved formula. Thus, when the formula contains lots of OR literals, it requires large time on users' devices such as eID cards.

In the backgrounds, in [4], an efficient anonymous credential system with constant-size attribute proofs was proposed, where the user can prove CNF formulas on attributes. In this system, by newly constructing an efficient accumulator to verify CNF formulas and applying it to the system, the proof generation requires only the multiplications depending on the number of OR literals in the proved formula, and thus it is more efficient than [12]. However, this system has the problem that a user cannot directly prove any CNF relation with negations.

1.3 Our Contributions

In this paper, we construct an accumulator to verify CNF formulas with negations, and we apply it to the previous system [4] with the constant-size attribute proofs. In the proposed system, a user can prove any CNF formula with negations, where the proof generation cost is similar to the previous, i.e., the proof generation needs only multiplications depending on the number of OR literals.

In the previous accumulator [4] for the limited CNF formula without negations, the set relation $U \cap V_\ell \neq \emptyset$ can be verified for the user's attribute set U and the attribute set V_ℓ in the ℓ-th OR clause in the CNF formula, which implies that

the user owns some attribute in each OR clause. In this paper, we consider non-limited CNF formulas of $\bigwedge_i \bigvee_j \breve{a}_{ij}$, where $\breve{a}_{ij}$ is a literal that is a non-negated attribute a_{ij} (the user owns the attribute) or a negated attribute a_{ij} (the user does not own the attribute). Any logical formula can be transformed to a CNF formula. In the proposed accumulator, in addition to $U \cap V_\ell^+ \neq \emptyset$ for the non-negated attribute set V_ℓ^+ in the ℓ-th OR clause, the relation $U \cap V_\ell^- \neq V_\ell^-$ can be verified for the negated attribute set V_ℓ^- in the ℓ-th OR clause. These means that the user owns some non-negated attribute or does not own some negated attribute, which implies the satisfaction of the CNF formula with negations.

1.4 Related Works

In [13], as the extension of [4], an anonymous credential system with the constant-size attribute proofs was proposed. The advantage is that a user can prove any monotone formula on attributes. However, in the system, negations are not available. Our idea is to support negations based on the previous accumulator [4] for the limited CNF formulas, and it does not work well in the accumulator of [13] for monotone formulas.

2 Preliminaries

2.1 Bilinear Maps

In this paper, we use the following bilinear groups with a bilinear map.

1. $\mathcal{G}_1, \mathcal{G}_2, \mathcal{T}$ are cyclic groups of prime order p.
2. g_1, g_2 are randomly chosen generators of $\mathcal{G}_1, \mathcal{G}_2$, respectively.
3. $e : \mathcal{G}_1 \times \mathcal{G}_2 \to \mathcal{T}$ is an efficiently calculated bilinear map satisfying
   (a) **Bilinearity:** for all $u \in \mathcal{G}_1$, $v \in \mathcal{G}_2, a, b \in \mathcal{Z}$, $e(u^a, v^b) = e(u, v)^{ab}$.
   (b) **Non-degeneracy:** $e(g_1, g_2) \neq 1_{\mathcal{T}}$ ($1_{\mathcal{T}}$ is the identity element of group $\mathcal{T}$).

The bilinear map e can be efficiently implemented with a pairing. There are two types of bilinear pairings, symmetric ($\mathcal{G}_1 = \mathcal{G}_2$) and asymmetric ($\mathcal{G}_1 \neq \mathcal{G}_2$). In the following descriptions, for simplicity, we adopt the symmetric one, i.e., e is defined as $\mathcal{G} \times \mathcal{G} \to \mathcal{T}$.

2.2 Assumptions

As in the previous system [4], the security of our system is based on the DLIN (Decision Linear) assumption, the q-SFP (Simultaneous Flexible Pairing) assumption, and n-DHE (DH Exponent) assumption. Hereafter, we use the notation $a \in_R A$ as sampling a from the set A according to the uniform distribution.

Definition 1 (DLIN assumption). *For all PPT algorithm* $\mathcal{A}$*,*

$$|Pr[\mathcal{A}(g, g^a, g^b, g^{ac}, g^{bd}, g^{c+d}) = 1] - Pr[\mathcal{A}(g, g^a, g^b, g^{ac}, g^{bd}, g^z) = 1]|$$

is negligible, where $g \in_R \mathcal{G}$ *and* $a, b, c, d, z \in_R Z_p$*.*

Definition 2 (q-SFP assumption). *For all PPT algorithm* $\mathcal{A}$*, the probability*

$$\begin{aligned} Pr[&\mathcal{A}(g_z, h_z, g_r, h_r, a, \tilde{a}, b, \tilde{b}, \{(z_j, r_j, s_j, t_j, u_j, v_j, w_j)\}_{j=1}^q) \\ &= (z^*, r^*, s^*, t^*, u^*, v^*, w^*) \in \mathcal{G}^7 \\ &\wedge e(a, \tilde{a}) = e(g_z, z^*)e(g_r, r^*)e(s^*, t^*) \\ &\wedge e(b, \tilde{b}) = e(h_z, z^*)e(h_r, u^*)e(v^*, w^*) \\ &\wedge z^* \neq 1_{\mathcal{G}} \wedge z^* \neq z_j \text{ for all } 1 \leq j \leq q] \end{aligned}$$

is negligible, where $(g_z, h_z, g_r, h_r, a, \tilde{a}, b, \tilde{b}) \in \mathcal{G}^8$ *and all tuples* $\{(z_j, r_j, s_j, t_j, u_j, v_j, w_j)\}_{j=1}^q$ *satisfy*

$$e(a, \tilde{a}) = e(g_z, z_j)e(g_r, r_j)e(s_j, t_j) \wedge e(b, \tilde{b}) = e(h_z, z_j)e(h_r, u_j)e(v_j, w_j),$$

and $1_{\mathcal{G}}$ *is the identity element of group* $\mathcal{G}$*.*

Definition 3 (n-DHE assumption). *For all PPT algorithm* $\mathcal{A}$*, the probability*

$$Pr[\mathcal{A}(g, g^a, \ldots, g^{a^n}, g^{a^{n+2}}, \ldots, g^{a^{2n}}) = g^{a^{n+1}}]$$

is negligible, where $g \in_R \mathcal{G}$ *and* $a \in_R Z_p$*.*

2.3 AHO (Abe-Haralambiev-Ohkubo) Signatures

As in the previous system [4], we adopt AHO signatures [2] as the structure-preserving signatures, where multiple messages can be signed, and the verification using pairings can be proved by the following GS proofs. In this paper, we use it for a single message. As proved in [2], this signature is existentially unforgeable against the chosen message attacks under the q-SFP assumption.

AHOKeyGen: Select bilinear groups $\mathcal{G}, \mathcal{T}$ with a prime order p and bilinear map e. Select $g, G_r, H_r \in_R \mathcal{G}$, and $\mu_z, \nu_z, \mu, \nu, \alpha_a, \alpha_b \in_R Z_p$. Compute $G_z = G_r^{\mu_z}, H_z = H_r^{\nu_z}, G = G_r^{\mu}, H = H_r^{\nu}, A = e(G_r, g^{\alpha_a}), B = e(H_r, g^{\alpha_b})$. Output the public key as $pk = (\mathcal{G}, \mathcal{T}, p, e, g, G_r, H_r, G_z, H_z, G, H, A, B)$ and the secret key as $sk = (\alpha_a, \alpha_b, \mu_z, \nu_z, \mu, \nu)$.

AHOSign: The message M given as element of $\mathcal{G}$ is signed with the secret key sk. Choose $\beta, \epsilon, \eta, \iota, \kappa \in_R Z_p$, and compute $\theta_1 = g^{\beta}$ and

$$\begin{aligned} &\theta_2 = g^{\epsilon - \mu_z \beta} M^{-\mu}, && \theta_3 = G_r^{\eta}, && \theta_4 = g^{(\alpha_a - \epsilon)/\eta}, \\ &\theta_5 = g^{\iota - \nu_z \beta} M^{-\nu}, && \theta_6 = H_r^{\kappa}, && \theta_7 = g^{(\alpha_b - \iota)/\kappa}. \end{aligned}$$

Output the signature $\sigma = (\theta_1, \ldots, \theta_7)$.

AHOVerify: Given the message M and the signature $\sigma = (\theta_1, \ldots, \theta_7)$, accept these if the following equations hold:

$$\begin{aligned} A &= e(G_z, \theta_1) \cdot e(G_r, \theta_2) \cdot e(\theta_3, \theta_4) \cdot e(G, M), \\ B &= e(H_z, \theta_1) \cdot e(H_r, \theta_5) \cdot e(\theta_6, \theta_7) \cdot e(H, M). \end{aligned}$$

2.4 GS (Groth-Sahai) Proofs

GS proofs [10] are Non-Interactive Witness Indistinguishable (NIWI) proofs for pairing relations. GS proofs need a CRS (Common Reference String) $(\boldsymbol{f}_1, \boldsymbol{f}_2, \boldsymbol{f}_3) \in (\mathcal{G}^3)^3$, where $\boldsymbol{f}_1 = (f_1, 1, g), \boldsymbol{f}_2 = (1, f_2, g)$ for $f_1, f_2 \in \mathcal{G}$. Two types of CRS are used. In the soundness setting, set $\boldsymbol{f}_3 = \boldsymbol{f_1}^{\xi_1} \cdot \boldsymbol{f_2}^{\xi_2}$ for $\xi_1, \xi_2 \in_R Z_p^*$. Compute the commitment to element X as $\boldsymbol{C} = (1, 1, X) \cdot \boldsymbol{f_1}^r \cdot \boldsymbol{f_2}^s \cdot \boldsymbol{f_3}^t$ for $r, s, t \in_R Z_p^*$. In this case, the commitment $\boldsymbol{C} = (f_1^{r+\xi_1 t}, f_2^{s+\xi_2 t}, X g^{r+s+t(\xi_1+\xi_2)})$ is a linear encryption [3]. Therefore, X can be extracted using the secret keys, $\log_g f_1, \log_g f_2$. On the other hand, in the Witness Indistinguishable (WI) setting, $\boldsymbol{f}_1, \boldsymbol{f}_2, \boldsymbol{f}_3$ are linearly independent, and thus $\boldsymbol{C}$ is perfectly hiding. Under the DLIN assumption, the two types of CRS are computationally indistinguishable.

In order to prove that committed values satisfy the pairing relation, the prover prepares the commitments and replaces variables in the relation with the commitments. By GS proof, we can prove the following pairing product equation.

$$\prod_{i=1}^{n} e(A_i, X_i) \cdot \prod_{i=1}^{n} \prod_{j=1}^{n} e(X_i, X_j)^{a_{ij}} = t$$

for variables $X_1, \ldots, X_n \in \mathcal{G}$ and constants $A_1, \ldots, A_n \in \mathcal{G},\ a_{ij} \in Z_p,\ t \in \mathcal{T}$.

2.5 Set Membership Proof

As in the previous system [4], the set membership proof [6] is used to prove that an element is included in a set of elements, which is constructed from signatures, as follows. An issuer signs all elements of set A and publishes the signatures. To prove that an element a is included in set A, a prover proves the knowledge of a signature on a. Since the issuer does not publish the signatures on elements that are not included in A, $a \in A$ is guaranteed.

3 Accumulator to Verify CNF Formulas with Negations

3.1 Previous Accumulator and Problem

In [8], an efficient pairing-based accumulator using multiplications has been proposed. An accumulator is generated from a set of values, and we can confirm that a single value is included in the set. In the previous work [4], an extended accumulator has been proposed, where we can verify that $U \cap V_\ell \neq \emptyset$ $(1 \leq \ell \leq L)$ for sets U and $V_1, \ldots, V_L$. This verification is applied to the construction of the previous anonymous credential system [4] to verify CNF formulas on attributes. Let $V_1, \ldots, V_L$ be subsets of $\{1, \cdots, n\}$, and $\mathcal{V} = (\mathcal{V}_\infty, \ldots, \mathcal{V}_\mathcal{L})$. Let U be a subset of $\{1, \cdots, n\}$ that satisfies $U \cap V_\ell \neq \emptyset$ $(1 \leq \ell \leq L)$. In the attribute proof, U corresponds to the attribute set of an user. Each V_ℓ corresponds to the ℓ-th OR clause in the proved CNF formula. In the accumulator of [4], we can verify

that $U \cap V_\ell \neq \emptyset$ for $1 \leq \ell \leq L$. This implies that some attribute of the user is included in all OR clauses, and so it can be verified that this user holds the attributes satisfying the CNF formula.

In the accumulator and the attribute proof using it in [4], we can not directly prove any CNF formula including a negation. To solve this problem, we can consider the following simple method without negations: Attributes can be divided into attribute types such as gender, age, and occupation. Then, to prove the non-possession of attribute a in an attribute type can be performed by proving the possession of one of other attributes in the type. However, this is undesirable for two reasons. One is to assign all attributes of the same type to the CNF formula as an OR clause, which increases the overhead of the proof. Secondly, any user must recognize all other attributes, but a user may be not aware of a newly added attribute to a attribute type. Therefore, we need an accumulator to directly verify CNF formulas with negations.

3.2 Construction Idea

In this paper, based on the previous accumulator [4], we extend it to verify the CNF formulas with negations. The accumulator $acc_{\mathcal{V}}$ of the previous scheme is computed as $acc_{\mathcal{V}} = \prod_{1\leq \ell \leq L} \left(\prod_{j \in V_\ell} g_{n+1-j} \right)^{c_\ell}$ for $g_i = g^{\gamma^i}$ (γ is secret) and some integers c_ℓ. On the other hand, in the proposed scheme, for a negated attribute $j \in V_\ell$, $g_{n+1-j}^{-c_\ell}$ is multiplied, instead of $g_{n+1-j}^{c_\ell}$. In the previous one, the verification is successful if $|U \cap V_\ell| \geq 1$ for the attribute set U of the user and the attribute set V_ℓ of the ℓ-th clause of the CNF formula. In the verification, for some witness W,

$$\frac{e(\prod_{i\in U} g_i, acc_{\mathcal{V}})}{e(g, W)} = z^{\delta_1 c_1 + \ldots + \delta_L c_L}, \text{ and } \delta_\ell \geq 1 \text{ for all } 1 \leq \ell \leq L$$

are checked. In the verification, $\delta_\ell = |U \cap V_\ell|$ holds. Thus, when U satisfies the CNF formula, which means $|U \cap V_\ell| \geq 1$, then the above $\delta_\ell \geq 1$ holds for all ℓ. In this previous scheme, c_ℓ is $|U \cap V_\ell|$ times added in the exponent of z for each ℓ in the left side of the verification equation. In the proposed scheme, each V_ℓ is partitioned to the non-negated attribute set V_ℓ^+ and the negated attribute set V_ℓ^-. For the attributes of V_ℓ^+ c_ℓ is added as in the previous scheme, but for the negated attributes of V_ℓ^-, c_ℓ is subtracted. Then, in the verification, the coefficient of c_ℓ in the exponent of z on the left side is $|U \cap V_\ell^+| - |U \cap V_\ell^-|$ for each ℓ, and by checking $\delta_\ell \geq 1 - |V_\ell^-|$, we can verify $|U \cap V_\ell^+| \geq 1$ or $|U \cap V_\ell^-| \neq |V_\ell^-|$. This means $U \cap V_\ell^+ \neq \emptyset$ or $U \cap V_\ell^- \neq V_\ell^-$ (for the detail, see the proof of Theorem 1). Thus, in each OR clause, it means that the user owns a non-negated attribute or does not own a negated attribute, and thus the CNF formula is satisfied. In the proposed scheme, since we only modify c_ℓ in the exponent to $-c_\ell$ for each negated attribute, it is expected that the processing time will remain.

3.3 Proposed Algorithms

AccSetup: This algorithm outputs public parameters. Set η_ℓ as the maximum value of $|V_\ell^+ \cup V_\ell^-|$ for all $1 \leq \ell \leq L$. Let $c_1 = 1$, $c_\ell = (\eta_{\ell-1} + 1) \cdot c_{\ell-1}$ $(2 \leq \ell \leq L)$, $\mathcal{C} = (c_1, \ldots, c_L)$. Here, it is assumed that $(\eta_L + 1) \cdot c_L < p$, as in the previous accumulator [4]. Select bilinear groups $\mathcal{G}, \mathcal{T}$ with prime order p and the bilinear map e. Select $g \in_R \mathcal{G}$. Choose $\gamma \in_R Z_p$. Compute and output the public parameters $(\mathcal{C}, p, \mathcal{G}, \mathcal{T}, e, g, g_1 = g^{\gamma^1}, \ldots, g_n = g^{\gamma^n}, g_{n+2} = g^{\gamma^{n+2}}, \ldots, g_{2n} = g^{\gamma^{2n}}, z = e(g,g)^{\gamma^{n+1}})$.

AccGen: This algorithm, given the public parameters and $\mathcal{V} = (V_1^+, V_1^-, \ldots, V_L^+, V_L^-)$, outputs an accumulator for $\mathcal{V}$. Here, $V_\ell^+ \subseteq \{1, \ldots, n\}$ is the set of non-negated attributes in the ℓ-th OR clause, and $V_\ell^- \subseteq \{1, \ldots, n\}$ is the set of negated attributes. Accumulator $acc_\mathcal{V}$ is calculated as follows.

$$acc_\mathcal{V} = \prod_{1 \leq \ell \leq L} \left(\prod_{j \in V_\ell^+} g_{n+1-j} \right)^{c_\ell} \left(\prod_{j \in V_\ell^-} g_{n+1-j} \right)^{-c_\ell}$$

AccWitGen: This algorithm, given the public parameters, $\mathcal{V}$, and $U \subseteq \{1, \ldots, n\}$, outputs the witness W. W is calculated as follows.

$$W = \prod_{i \in U} \prod_{1 \leq \ell \leq L} \left(\prod_{j \in V_\ell^+}^{j \neq i} g_{n+1-j+i} \right)^{c_\ell} \left(\prod_{j \in V_\ell^-}^{j \neq i} g_{n+1-j+i} \right)^{-c_\ell}$$

Furthermore, $\delta_\ell = |U \cap V_\ell^+| - |U \cap V_\ell^-|$ for all $1 \leq \ell \leq L$ are calculated and outputted as auxiliary parameters.

AccVerify: This algorithm, given the public parameters, $\mathcal{V}, acc_\mathcal{V}, U, W$, $\{\delta_\ell\}_{1 \leq \ell \leq L}$, verifies $U \cap V_\ell^+ \neq \emptyset$ or $U \cap V_\ell^- \neq V_\ell^-$ for all $1 \leq \ell \leq L$. Set $u = \delta_1 c_1 + \ldots + \delta_L c_L$. Accept if the following relations hold.

$$\frac{e(\prod_{i \in U} g_i, acc_\mathcal{V})}{e(g, W)} = z^u, \text{ and } 1 \leq \delta_\ell + |V_\ell^-| \leq \eta_\ell \text{ for all } 1 \leq \ell \leq L.$$

In this case, since $1 - |V_\ell^-| \leq \delta_\ell$, this verification means the check of $1 - |V_\ell^-| \leq |U \cap V_\ell^+| - |U \cap V_\ell^-|$, which implies $|U \cap V_\ell^+| - |U \cap V_\ell^-| \neq -|V_\ell^-|$, and thus $U \cap V_\ell^+ \neq \emptyset$ or $U \cap V_\ell^- \neq V_\ell^-$.

3.4 Security

At first, we show the correctness of the proposed accumulator.

Theorem 1. *Suppose that all parameters of* **AccSetup**, **AccGen**, *and* **Acc WitGen** *are calculated correctly. Then,* **AccVerify** *accepts* $\mathcal{V}$, $acc_\mathcal{V}, U, W$, $\{\delta_\ell\}_{1 \leq \ell \leq L}$ *that those algorithms output, if* $U \cap V_\ell^+ \neq \emptyset$ *or* $U \cap V_\ell^- \neq V_\ell^-$ *for all* $1 \leq \ell \leq L$.

Proof. Assume all parameters of **AccSetup**, **AccGen**, and **AccWitGen** are calculated correctly. Then, the left hand of the verification equation in **AccVerify** is transformed as follows.

$$\begin{aligned}
\frac{e(\prod_{i\in U} g_i, acc_{\mathcal{V}})}{e(g,W)} &= \frac{e(\prod\limits_{i\in U} g_i, \prod\limits_{1\le \ell\le L}\left(\prod\limits_{j\in V_\ell^+} g_{n+1-j}\right)^{c_\ell}\left(\prod\limits_{j\in V_\ell^-} g_{n+1-j}\right)^{-c_\ell})}{e(g, \prod\limits_{i\in U}\prod\limits_{1\le \ell\le L}\left(\prod\limits_{j\in V_\ell^+}^{j\neq i} g_{n+1-j+i}\right)^{c_\ell}\left(\prod\limits_{j\in V_\ell^-}^{j\neq i} g_{n+1-j+i}\right)^{-c_\ell})} \\
&= \frac{e(g, \prod\limits_{i\in U}\prod\limits_{1\le \ell\le L}\left(\prod\limits_{j\in V_\ell^+} g_{n+1-j+i}\right)^{c_\ell}\left(\prod\limits_{j\in V_\ell^-} g_{n+1-j+i}\right)^{-c_\ell})}{e(g, \prod\limits_{i\in U}\prod\limits_{1\le \ell\le L}\left(\prod\limits_{j\in V_\ell^+}^{j\neq i} g_{n+1-j+i}\right)^{c_\ell}\left(\prod\limits_{j\in V_\ell^-}^{j\neq i} g_{n+1-j+i}\right)^{-c_\ell})} \\
&= e(g, \prod_{i\in U}\prod_{1\le \ell\le L}\left(\prod_{j\in V_\ell^+}^{j=i} g_{n+1-j+i}\right)^{c_\ell}\left(\prod_{j\in V_\ell^-}^{j=i} g_{n+1-j+i}\right)^{-c_\ell})
\end{aligned}$$

Set $\delta_\ell = |U \cap V_\ell^+| - |U \cap V_\ell^-|$ for all $1 \le \ell \le L$, and $u = \delta_1 c_1 + \ldots + \delta_L c_L$. Then, the above expression is equal to the right side of the verification equation as follows.

$$e\left(g, \prod_{1\le \ell\le L} g_{n+1}^{\delta_\ell c_\ell}\right) = e(g, g_{n+1})^u = z^u$$

Here, for $|U \cap V_\ell^+|$, the possible range is $0 \le |U \cap V_\ell^+| \le |V_\ell^+|$, and for $|U \cap V_\ell^-|$, it is $0 \le |U \cap V_\ell^-| \le |V_\ell^-|$, and thus

$$-|V_\ell^-| \le |U \cap V_\ell^+| - |U \cap V_\ell^-| \le |V_\ell^+|.$$

On the other hand, we have $U \cap V_\ell^+ \neq \emptyset$ or $U \cap V_\ell^- \neq V_\ell^-$ for all $1 \le \ell \le L$ as the condition in this theorem. In case that the condition of the theorem does not hold, for some ℓ, $|U \cap V_\ell^+| = 0$ and $|U \cap V_\ell^-| = |V_\ell^-|$, which means $|U \cap V_\ell^+| - |U \cap V_\ell^-| = -|V_\ell^-|$. Therefore, in case that the condition in this theorem holds, we obtain

$$1 - |V_\ell^-| \le |U \cap V_\ell^+| - |U \cap V_\ell^-| \le |V_\ell^+|,$$

for all $1 \le \ell \le L$. From $\delta_\ell = |U \cap V_\ell^+| - |U \cap V_\ell^-|$, we have $1 - |V_\ell^-| \le \delta_\ell \le |V_\ell^+|$, and thus

$$1 \le \delta_\ell + |V_\ell^-| \le |V_\ell^+| + |V_\ell^-| \le \eta_\ell,$$

for all $1 \leq \ell \leq L$. Therefore, **AccVerify** accepts these parameters. □

Furthermore, as in the journal version [5] of the previous work [4], using the following lemma (the proof is in [5]), we show the security of the proposed accumulator in Theorem 2.

Lemma 1. *For any $\tilde{\ell}$ s.t. $2 \leq \tilde{\ell} \leq L$, it holds $c_{\tilde{\ell}} > \sum_{1 \leq \ell \leq \tilde{\ell}-1} \eta_\ell \cdot c_\ell$.*

Theorem 2. *Under n-DHE assumption, given the public parameters, any adversary cannot output $U, \mathcal{V} = \{V_\ell^+, V_\ell^-\}_{1 \leq \ell \leq L}, W, \{\delta_\ell\}_{1 \leq \ell \leq L}$ which satisfy the following with a non-negligible probability.*

- *For $acc_\mathcal{V}$ correctly computed from $\mathcal{V}$,* **AccVerify** *accepts $\mathcal{V}, acc_\mathcal{V}, U, W, \{\delta_\ell\}_{1 \leq \ell \leq L}$.*
- *There exists some ℓ s.t. $U \cap V_\ell^+ = \emptyset$ and $U \cap V_\ell^- = V_\ell^-$.*

Proof. Assume an adversary that outputs $U, \mathcal{V} = \{V_\ell^+, V_\ell^-\}_{1 \leq \ell \leq L}, W, \{\delta_\ell\}_{1 \leq \ell \leq L}$ s.t. **AccVerify** accepts them and $U \cap V_\ell^+ = \emptyset$ and $U \cap V_\ell^- = V_\ell^-$ for some ℓ with a non-negligible probability. Since **AccVerify** accepts them, we have

$$\frac{e(\prod_{i \in U} g_i, acc_\mathcal{V})}{e(g, W)} = z^u = e(g, g_{n+1})^u,$$

for $u = \delta_1 c_1 + \ldots + \delta_L c_L$. From the correctly computed

$$acc_\mathcal{V} = \prod_{1 \leq \ell \leq L} \left(\prod_{j \in V_\ell^+} g_{n+1-j} \right)^{c_\ell} \left(\prod_{j \in V_\ell^-} g_{n+1-j} \right)^{-c_\ell},$$

we have

$$\frac{e(\prod_{i \in U} g_i, \prod_{1 \leq \ell \leq L} \left(\prod_{j \in V_\ell^+} g_{n+1-j} \right)^{c_\ell} \left(\prod_{j \in V_\ell^-} g_{n+1-j} \right)^{-c_\ell})}{e(g, W)} = e(g, g_{n+1})^u$$

$$e(g, \prod_{i \in U} \prod_{1 \leq \ell \leq L} \left(\prod_{j \in V_\ell^+} g_{n+1-j+i} \right)^{c_\ell} \left(\prod_{j \in V_\ell^-} g_{n+1-j+i} \right)^{-c_\ell}) = e(g, W {g_{n+1}}^u)$$

Thus, we obtain the followings.

$$\prod_{i \in U} \prod_{1 \leq \ell \leq L} \left(\prod_{j \in V_\ell^+} g_{n+1-j+i} \right)^{c_\ell} \left(\prod_{j \in V_\ell^-} g_{n+1-j+i} \right)^{-c_\ell} = W {g_{n+1}}^u$$

$$\prod_{1 \leq \ell \leq L} \prod_{i \in U} \left(\prod_{j \in V_\ell^+}^{j \neq i} g_{n+1-j+i} \right)^{c_\ell} \cdot \prod_{1 \leq \ell \leq L} {g_{n+1}}^{|U \cap V_\ell^+| c_\ell}$$

$$\cdot \prod_{1 \leq \ell \leq L} \prod_{i \in U} \left(\prod_{j \in V_\ell^-}^{j \neq i} g_{n+1-j+i} \right)^{-c_\ell} \cdot \prod_{1 \leq \ell \leq L} {g_{n+1}}^{-|U \cap V_\ell^-| c_\ell}$$

$$= W {g_{n+1}}^u$$

By setting $\lambda_\ell = |U \cap V_\ell^+| - |U \cap V_\ell^-| + |V_\ell^-|$,

$$\prod_{1\le\ell\le L}\prod_{i\in U}\left(\prod_{j\in V_\ell^+}^{j\neq i} g_{n+1-j+i}\right)^{c_\ell} \cdot \prod_{1\le\ell\le L}\prod_{i\in U}\left(\prod_{j\in V_\ell^-}^{j\neq i} g_{n+1-j+i}\right)^{-c_\ell} = W {g_{n+1}}^{u-\sum_{1\le\ell\le L}(\lambda_\ell - |V_\ell^-|)c_\ell} \tag{1}$$

Define $\Delta = u - \sum_{1\le\ell\le L}(\lambda_\ell - |V_\ell^-|)c_\ell$. Then, we have

$$\begin{aligned}\Delta &= \sum_{1\le\ell\le L}\delta_\ell c_\ell - \sum_{1\le\ell\le L}(\lambda_\ell - |V_\ell^-|)c_\ell \\ &= \sum_{1\le\ell\le L}(\delta_\ell - \lambda_\ell + |V_\ell^-|)c_\ell.\end{aligned}$$

Here, divide $\{1,\ldots,L\}$ into $L^>, L^<$, and $L^=$, where $L^>$ consists of ℓ s.t. $\delta_\ell - \lambda_\ell + |V_\ell^-| > 0$, $L^<$ consists of ℓ s.t. $\delta_\ell - \lambda_\ell + |V_\ell^-| < 0$, and $L^=$ consists of ℓ s.t. $\delta_\ell - \lambda_\ell + |V_\ell^-| = 0$.

Using $L^>, L^<$, and $L^=$, the following equation can be obtained.

$$\Delta = \sum_{\ell\in L^>}(\delta_\ell - \lambda_\ell + |V_\ell^-|)c_\ell + \sum_{\ell\in L^<}(\delta_\ell - \lambda_\ell + |V_\ell^-|)c_\ell$$

Let $\tilde{\ell}$ be the maximum value of ℓ s.t. $\ell \notin L^=$ (i.e., $\tilde{\ell} \in L^>$ or $\tilde{\ell} \in L^<$). From **AccVerify** $= 1$, it holds $\delta_\ell + |V_\ell^-| \ge 1$ for all ℓ. On the other hand, since for some ℓ, $U \cap V_\ell^+ = \emptyset$ and $U \cap V_\ell^- = V_\ell^-$, we have $\lambda_\ell = |U \cap V_\ell^+| - |U \cap V_\ell^-| + |V_\ell^-| = 0$ for the ℓ. This implies that $\delta_\ell - \lambda_\ell + |V_\ell^-| \neq 0$ for the ℓ. Therefore, $\ell \notin L^=$ exists.

Next, we will prove $\Delta \neq 0 \pmod p$ for two cases (i) and (ii).

(i) **Case of** $\tilde{\ell} \in L^<$ $(\delta_{\tilde{\ell}} - \lambda_{\tilde{\ell}} + |V_{\tilde{\ell}}^-| < 0)$:
In this case, $(\delta_{\tilde{\ell}} - \lambda_{\tilde{\ell}} + |V_{\tilde{\ell}}^-|)c_{\tilde{\ell}} \le -c_{\tilde{\ell}}$, which implies

$$\Delta \le -c_{\tilde{\ell}} + \sum_{\ell\in L^>}(\delta_\ell - \lambda_\ell + |V_\ell^-|)c_\ell \ + \sum_{\ell\in L^<, \ell\neq\tilde{\ell}}(\delta_\ell - \lambda_\ell + |V_\ell^-|)c_\ell.$$

For $\ell \in L^>$, since $\lambda_\ell \ge 0$ and $\delta_\ell + |V_\ell^-| \le \eta_\ell$, we have $\delta_\ell - \lambda_\ell + |V_\ell^-| \le \eta_\ell$. For $\ell \in L^<$, we have $\delta_\ell - \lambda_\ell + |V_\ell^-| < 0$. Therefore,

$$\Delta < -c_{\tilde{\ell}} \ + \sum_{\ell\in L^>}\eta_\ell c_\ell.$$

From Lemma 1, we obtain $\Delta < 0$ due to $c_{\tilde{\ell}} > \sum_{\ell\in(L^>\cup L^<)}\eta_\ell c_\ell$.
On the other hand, from $\delta_\ell + |V_\ell^-| > 0$ and $\lambda_\ell = |U\cap V_\ell^+| - |U\cap V_\ell^-| + |V_\ell^-| \le |V_\ell^+ \cup V_\ell^-| \le \eta_\ell$,

$$\Delta = \sum_{1\le\ell\le L}(\delta_\ell + |V_\ell^-|)c_\ell \ - \sum_{1\le\ell\le L}\lambda_\ell c_\ell > \ - \sum_{1\le\ell\le L}\eta_\ell c_\ell$$

From Lemma 1, we obtain $\sum_{1\le\ell\le L-1}\eta_\ell c_\ell < c_L$, and thus

$$\sum_{1\le\ell\le L}\eta_\ell c_\ell \;<\; c_L \;+\; \eta_L c_L \;=\; (\eta_L+1)c_L \;<\; p.$$

This is why $\Delta > -p$. Therefore, in this case, $\Delta \neq 0 \pmod p$.

(ii) **Case of** $\tilde{\ell} \in L^{>}$ $(\delta_{\tilde{\ell}} - \lambda_{\tilde{\ell}} + |V_{\tilde{\ell}}^-| > 0)$:

In this case, $(\delta_{\tilde{\ell}} - \lambda_{\tilde{\ell}} + |V_{\ell}^-|)c_{\tilde{\ell}} \ge c_{\tilde{\ell}}$, which means

$$\Delta \ge c_{\tilde{\ell}} + \sum_{\ell\in L^{>},\ell\neq\tilde{\ell}}(\delta_\ell - \lambda_\ell + |V_\ell^-|)c_\ell + \sum_{\ell\in L^{<}}(\delta_\ell - \lambda_\ell + |V_\ell^-|)c_\ell$$

From $\delta_\ell - \lambda_\ell + |V_\ell^-| > 0$ for any $\ell \in L^{>}$, we have

$$\Delta > c_{\tilde{\ell}} + \sum_{\ell\in L^{<}}(\delta_\ell - \lambda_\ell + |V_\ell^-|)c_\ell.$$

Here, from $\lambda_\ell \le \eta_\ell$ and $\delta_\ell + |V_\ell^-| \ge 0$, we have $\lambda_\ell - \delta_\ell - |V_\ell^-| \le \eta_\ell$. Thus, from $\tilde{\ell} > \ell$ for any $\ell \in L^{<}$ and Lemma 1, we obtain

$$c_{\tilde{\ell}} > \sum_{\ell\in L^{<}}\eta_\ell c_\ell \ge \sum_{\ell\in L^{<}}(\lambda_\ell - \delta_\ell - |V_\ell^-|)c_\ell$$

Therefore,

$$c_{\tilde{\ell}} + \sum_{\ell\in L^{<}}(\delta_\ell - \lambda_\ell + |V_\ell^-|)c_\ell > 0.$$

Namely, we can get $\Delta > 0$.

On the other hand, from $\lambda_\ell \ge 0$, $\delta_\ell + |V_\ell^-| \le \eta_\ell$, and Lemma 1,

$$\begin{aligned}\Delta &= \sum_{1\le\ell\le L}(\delta_\ell + |V_\ell^-|)c_\ell \;-\; \sum_{1\le\ell\le L}\lambda_\ell c_\ell \\ &\le \sum_{1\le\ell\le L}(\delta_\ell + |V_\ell^-|)c_\ell \le \sum_{1\le\ell\le L}\eta_\ell c_\ell = \sum_{1\le\ell\le L-1}\eta_\ell c_\ell + \eta_L c_L \le c_L + \eta_L c_L.\end{aligned}$$

Thus, $\Delta \le (\eta_L+1)c_L < p$. Therefore, it also holds $\Delta \neq 0 \pmod p$ in this case.

Therefore, since $\Delta \neq 0 \pmod p$ in both cases, from Eq. (1), we obtain

$$g_{n+1} = \left(W^{-1}\cdot \prod_{1\le\ell\le L}\prod_{i\in U}\left(\prod_{j\in V_\ell^+}^{j\neq i} g_{n+1-j+i}\right)^{c_\ell} \cdot \prod_{1\le\ell\le L}\prod_{i\in U}\left(\prod_{j\in V_\ell^-}^{j\neq i} g_{n+1-j+i}\right)^{-c_\ell}\right)^{1/\Delta}.$$

For any $i \in U$, any $j \in V_\ell^+$ and $j \in V_\ell^-$ s.t. $j \neq i$, it holds $g_{n+1-j+i} \neq g_{n+1}$. Therefore, we can calculate g_{n+1}, given $g_1, \ldots, g_n, g_{n+2}, \ldots, g_{2n}$, which contradicts the n-DHE assumption. □

4 Syntax and Security Model of Anonymous Credential System

We adopt the syntax and security model of anonymous credential system with attribute proofs in the previous work [4]. It is the non-interactive anonymous credential system, where the user creates the attribute proof from his own certificate issued from an issuer, and the verifier can confirm the proof by himself. Since this concept is similar to the group signature scheme, the security model is derived from the group signature scheme, but concentrates on the security of attribute proofs. This is why the model considers the following two security requirements: misauthentication resistance and the anonymity.

4.1 Syntax

As in [4], each attribute values is indexed by an integer from $\{1, \ldots, n\}$ where n is the total number of attribute values. Use the indexes to describe a CNF formula Ψ (including negations) on attribute, as follows.

$$(\breve{a}_{11} \vee \breve{a}_{12} \vee \ldots) \wedge (\breve{a}_{21} \vee \breve{a}_{22} \vee \ldots) \wedge \cdots,$$

where each literal $\breve{a}_{ij}$ is (non-negated) attribute index $a_{ij} \in \{1, \ldots, n\}$ or its negation $\neg a_{ij}$. The literal a_{ij} means that the user owns the attribute of the index, and the literal $\neg a_{ij}$ means that the user does not own the attribute of the index. Let V_ℓ^+ be the set of non-negated attribute indexes in the ℓ-th clause in CNF formula Ψ (i.e, $V_\ell^+ = \{a_{\ell j} | \breve{a}_{\ell j} = a_{\ell j}\}$). Let V_ℓ^- be the set of negated attribute indexes in the ℓ-th clause in CNF formula Ψ (i.e, $V_\ell^- = \{a_{\ell j} | \breve{a}_{\ell j} = \neg a_{\ell j}\}$).

Let U be a set of attribute indexes that the proving user owns. We assume that the upper bound of each clause size, i.e., $|V_\ell^+ \cup V_\ell^-|$, is η_ℓ for all $1 \leq \ell \leq L$. Also, we assume that the maximum number of clauses of CNF formulas is L.

Then, the satisfaction of the CNF formula Ψ with $(V_1^+, V_1^-, \ldots, V_\ell^+, V_\ell^-)$ by U is shown by $U \cap V_\ell^+ \neq \emptyset$ or $U \cap V_\ell^- \neq V_\ell^-$ for all $1 \leq \ell \leq L$.
The anonymous credential system consists of the following algorithms and protocol.

IssuerKeyGen: This algorithm, given $n, L, \{\eta_\ell\}_{1 \leq \ell \leq L}$, outputs the issuer's public key ipk and the issuer's secret key isk.

CertObtain: This is an interactive protocol between algorithm **CertObtain-$\mathcal{U}_k$** of the k-th user and algorithm **CertObtain-$\mathcal{I}$** of the issuer, where the issuer issues a certificate certifying the attributes to the user. **CertObtain-$\mathcal{U}_k$**'s inputs are ipk and $U_k \subset \{1, \ldots, n\}$ which are the user's attribute indexes, and its output is certificate $cert_k$ that guarantees the attributes of the user. On the other hand, **CertObtain-$\mathcal{I}$** is given ipk, isk and U_k as inputs.

ProofGen: This algorithm for the k-th user, given $ipk, U_k, cert_k$, Ψ that is a proved CNF formula on attributes, outputs the attribute proof σ.

Verify: This algorithm for verification, given ipk, proof σ generated on U_k of the k-th user, and the proved CNF formula Ψ, outputs 'valid' if the attributes U_k satisfy Ψ (i.e., $U_k \cap V_\ell^+ \neq \emptyset$ or $U_k \cap V_\ell^- \neq V_\ell^-$ for all $1 \leq \ell \leq L$), and otherwise 'invalid'.

4.2 Security Model

The security model in [4] consists of *misauthentication resistance* and *anonymity*. The misauthentication resistance means the soundness of attribute proofs, i.e., any adversary $\mathcal{A}$ cannot forge an attribute proof for a CNF formula, where the formula is not satisfied by the attributes of any user who is corrupted by the adversary. The anonymity means the anonymity and unlinkability of proofs, which are similar to those of group signatures. Due to the page limitation, we omit the formal definitions (See [4]).

5 An Anonymous Credential System with Constant-Size Attribute Proofs for CNF Formulas with Negations

We extend the anonymous credential system [4] for limited CNF formulas without negations such that the user can prove any CNF formula with negations. In the previous system, in **IssuerKeyGen**, an issuer publishes the signatures on valid u's in the accumulator verification, which is based on the concept of the set membership proof. In **CertObtain**, to the user, the issuer issues a membership certificate which is the AHO signature on $P_k = \prod_{i \in U_k} g_i$ for the attribute set U_k of the user. In **ProofGen** and **Verify**, the user proves the verification of the AHO signature on P_k, and the equation of the accumulator verification by GS proofs. In addition, to show the range of each δ_ℓ in $u = \delta_1 c_1 + \ldots + \delta_L c_L$ in the accumulator verification, the user proves the verification of the AHO signature on u.

In our extension, **IssuerKeyGen** and **CertObtain** are the almost same as the previous system, where AHO signatures are published for the valid range of $u' = (\delta_1 + |V_1^-|)c_1 + \ldots + (\delta_L + |V_L^-|)c_L$. In **ProofGen** and **Verify**, the used accumulator is modified to our newly constructed accumulator in Sect. 3 for CNF formulas with negations. The user proves the verification equation of the accumulator, and the verification of the AHO signature on u' which means $1 \leq \delta_\ell + |V_\ell^-| \leq \eta_\ell$ for each ℓ. Thus, due to the accumulator, it is ensured that $U \cap V_\ell^+ \neq \emptyset$ or $U \cap V_\ell^- \neq V_\ell^-$ for all $1 \leq \ell \leq L$.

5.1 Construction

The algorithms and protocol of the proposed system is as follows.

IssuerKeyGen: Given n that is the total number of attribute values, L that is the maximum value of clauses of proved CNF formulas, and η_ℓ that is the upper bound of $|V_\ell^+ \cup V_\ell^-|$. This algorithm executes **AccSetup** to generate the public parameters of the proposed accumulator, and generates the key pair of AHO signatures, CRS for GS NIWI proofs, and AHO signatures for the set membership proof.

(i) Select prime order p, bilinear group $\mathcal{G}, \mathcal{T}$ and bilinear map e. Choose $g \in_R \mathcal{G}$.

(ii) Generate public parameters of the proposed accumulator for CNF formulas with negations: Calculate $c_1 = 1, c_\ell = (\eta_{\ell-1} + 1) \cdot c_{\ell-1}$ for $2 \leq \ell \leq L$, and set $\mathcal{C} = (c_1, \ldots, c_L)$. Choose $\gamma \in_R Z_p$ and calculate

$$pk_{acc} = (\mathcal{C}, g_1 = g^{\gamma^1}, \ldots, g_n = g^{\gamma^n}, g_{n+2} = g^{\gamma^{n+2}}, \ldots, g_{2n} = g^{\gamma^{2n}}, z = (g, g)^{\gamma^{n+1}}).$$

(iii) For AHO signatures, generate the following two key pairs for $d = 0, 1$.

$$pk_{AHO}^{(d)} = (G_r^{(d)}, H_r^{(d)}, G_z^{(d)}, H_z^{(d)}, G^{(d)}, H^{(d)}, A^{(d)}, B^{(d)}),$$
$$sk_{AHO}^{(d)} = (\alpha_a^{(d)}, \alpha_b^{(d)}, \mu_z^{(d)}, \nu_z^{(d)}, \mu^{(d)}, \nu^{(d)}).$$

(iv) Generate CRS $\boldsymbol{f} = (\boldsymbol{f_1}, \boldsymbol{f_2}, \boldsymbol{f_3})$ for GS NIWI proof:

$$\boldsymbol{f_1} = (f_1, 1, g),\ \boldsymbol{f_2} = (1, f_2, g),\ \boldsymbol{f_3} = \boldsymbol{f_1}^{\xi_1} \cdot \boldsymbol{f_2}^{\xi_2},$$

where $f_1, f_2 \in_R \mathcal{G},\ \xi_1, \xi_2 \in_R Z_p^*$.

(v) Define set $\Phi = \{u' = \sum_{\ell=1}^{L} \delta'_\ell c_\ell | 1 \leq \delta'_\ell \leq \eta_\ell\}$. Then, $|\Phi| = \prod_{1 \leq \ell \leq L} \eta_\ell$. For all $u' \in \Phi$, we generate an AHO signature on $g_1^{u'}$ using $sk_{AHO}^{(0)}$. The signature is denoted as $\tilde{\sigma}_{u'} = (\tilde{\theta}_{u'1}, \ldots, \tilde{\theta}_{u'7})$.

(vi) As the public key and secret key of the issuer, output

$$ipk = (p, \mathcal{G}, \mathcal{T}, e, g, pk_{AHO}^{(0)}, pk_{AHO}^{(1)}, pk_{acc}, \boldsymbol{f}, \{\tilde{\sigma}_{u'}\}_{u' \in \Phi}),$$
$$isk = (sk_{AHO}^{(0)}, sk_{AHO}^{(1)}).$$

CertObtain: This is the protocol between **CertObtain-$\mathcal{U}_k$** (the k-th user) and **CertObtain-$\mathcal{I}$** (issuer). The input of **CertObtain-$\mathcal{U}_k$** consists of ipk and the set U_k of the user's attribute indexes. The inputs of **CertObtain-$\mathcal{I}$** are ipk, isk, and U_k. In this protocol, the issuer issues the user the certificate $cert_k$. Here, it is assumed that a special attribute value a_{SP} is introduced and all users owns a_{SP}.

(i) **CertObtain-$\mathcal{I}$**: Generate $P_k = \prod_{i \in U_k} g_i$.

(ii) **CertObtain-$\mathcal{I}$**: Use $sk_{AHO}^{(1)}$ to generate the AHO signature $\sigma_k = (\theta_1, \ldots, \theta_7)$ on P_k, where σ_k is sent to **CertObtain-$\mathcal{U}_k$** as the certificate.

(iii) **CertObtain-$\mathcal{U}_k$**: Compute $P_k = \prod_{i \in U_k} g_i$, and verify the AHO signature σ_k on P_k. Then, output $cert_k = (P_k, \sigma_k)$.

ProofGen: Given $ipk, U_k, cert_k$ and CNF formula $\Psi = (\breve{a}_{11} \vee \breve{a}_{12} \vee \ldots) \wedge (\breve{a}_{21} \vee \breve{a}_{22} \vee \ldots) \wedge \cdots (\breve{a}_{L'1} \vee \breve{a}_{L'2} \vee \ldots)$, where each literal $\breve{a}_{ij}$ is (non-negated) attribute index $a_{ij} \in \{1, \ldots, n\}$ or its negation $\neg a_{ij}$. Let V_ℓ^+ be the set of non-negated attributes in the ℓ-th OR clause, and let V_ℓ^- be the set of negated attributes. If $L' < L$, define $V_{L'+1}^+ = \ldots = V_L^+ = \{a_{SP}\}$ and $V_{L'+1}^- = \ldots = V_L^- = \emptyset$. This algorithm generates GS proofs to prove that P_k satisfies the accumulator verification for $acc_{\mathcal{V}}$ corresponding to Ψ and that P_k is signed by the issuer using AHO signatures. In addition, the AHO signature on $g_1^{u'}$ is also used in the accumulator verification.

(i) Compute the accumulator of $\mathcal{V} = (V_1^+, V_1^-, \ldots)$:

$$acc_{\mathcal{V}} = \prod_{1 \le \ell \le L} \left(\prod_{j \in V_\ell^+} g_{n+1-j} \right)^{c_\ell} \left(\prod_{j \in V_\ell^-} g_{n+1-j} \right)^{-c_\ell}$$

(ii) Compute the witness $W_{\mathcal{V}}$:

$$W_{\mathcal{V}} = \prod_{i \in U} \prod_{1 \le \ell \le L} \left(\prod_{j \in V_\ell^+}^{j \ne i} g_{n+1-j+i} \right)^{c_\ell} \left(\prod_{j \in V_\ell^-}^{j \ne i} g_{n+1-j+i} \right)^{-c_\ell}$$

For all $1 \le \ell \le L$, set $\delta_\ell = |U \cap V_\ell^+| - |U \cap V_\ell^-|$ and set $u = \delta_1 c_1 + \ldots + \delta_L c_L$.

(iii) Set $\delta_\ell^{'} = \delta_\ell + |V_\ell^-|$, $u^{'} = \delta_1^{'} c_1 + \ldots + \delta_L^{'} c_L$, and $\tau_{u'} = g_1^{u^{'}}$. From ipk, pick up $\tilde{\sigma}_{u'} = (\tilde{\theta}_{u'1}, \ldots, \tilde{\theta}_{u'7})$ that is the AHO signature on $g_1^{u^{'}}$. Set $\tilde{u} = -(|V_1^-|c_1 + \ldots + |V_L^-|c_L)$ and $\tau_{\tilde{u}} = g_1^{\tilde{u}}$.

(iv) Compute $com_{P_k}, com_{W_{\mathcal{V}}}, com_{\tau_{u'}}$ as the GS commitments to $P_k, W_{\mathcal{V}}, \tau_{u'}$. Then, re-randomize the AHO signature σ_k by the method of [2] to obtain $\sigma_k^{'} = \{\theta_1^{'}, \ldots, \theta_7^{'}\}$. Compute the GS commitments $\{com_{\theta_i^{'}}\}_{i \in \{1,2,5\}}$ to $\{\theta_i^{'}\}_{i \in \{1,2,5\}}$. Similarly, re-randomize the AHO signature $\tilde{\sigma}_{u'}$ to obtain $\tilde{\sigma}_{u'}^{'} = \{\tilde{\theta}^{'}_{u'1}, \ldots, \tilde{\theta}^{'}_{u'7}\}$. Compute the GS commitments $\{com_{\tilde{\theta}^{'}_{u'i}}\}_{i \in \{1,2,5\}}$ to $\{\tilde{\theta}^{'}_{u'i}\}_{i \in \{1,2,5\}}$.

(v) Generate GS proofs $\{\pi_i\}_{i=1}^5$ to prove the following.

$$e(\tau_{\tilde{u}}, g_n)^{-1} = e(P_k, acc_{\mathcal{V}}) \cdot e(g, W_{\mathcal{V}})^{-1} \cdot e(\tau_{u'}, g_n)^{-1}, \quad (2)$$

$$A^{(1)} \cdot e(\theta_3^{'}, \theta_4^{'})^{-1} = e(G_z^{(1)}, \theta_1^{'}) \cdot e(G_r^{(1)}, \theta_2^{'}) \cdot e(G^{(1)}, P_k), \quad (3)$$

$$B^{(1)} \cdot e(\theta_6^{'}, \theta_7^{'})^{-1} = e(H_z^{(1)}, \theta_1^{'}) \cdot e(H_r^{(1)}, \theta_5^{'}) \cdot e(H^{(1)}, P_k), \quad (4)$$

$$A^{(0)} \cdot e(\tilde{\theta}^{'}_{u'3}, \tilde{\theta}^{'}_{u'4})^{-1} = e(G_z^{(0)}, \tilde{\theta}^{'}_{u'1}) \cdot e(G_r^{(0)}, \tilde{\theta}^{'}_{u'2}) \cdot e(G^{(0)}, \tau_{u'}), \quad (5)$$

$$B^{(0)} \cdot e(\tilde{\theta}^{'}_{u'6}, \tilde{\theta}^{'}_{u'7})^{-1} = e(H_z^{(0)}, \tilde{\theta}^{'}_{u'1}) \cdot H_r^{(0)}, \tilde{\theta}^{'}_{u'5}) \cdot e(H^{(0)}, \tau_{u'}) \quad (6)$$

(vi) Output $\sigma = (\{\theta_i^{'}\}_{i=3,4,6,7}, \{\tilde{\theta}^{'}_{u'i}\}_{i=3,4,6,7}, com_{P_k}, com_{W_{\mathcal{V}}}, com_{\tau_{u'}}, \{com_{\theta_i^{'}}\}_{i=1,2,5}, \{com_{\tilde{\theta}^{'}_{u'i}}\}_{i=1,2,5}, \{\pi_i\}_{i=1}^5)$.

By substituting $P_k = \prod_{i \in U_k} g_i$, $\tau_{u'} = g_1^{u^{'}}$, and $\tau_{\tilde{u}} = g_1^{\tilde{u}}$ in Eq. (2), it can be transformed into the verification equation of the accumulator as follows.

$$\frac{e(\prod_{i \in U_k} g_i, acc_{\mathcal{V}})}{e(g, W_{\mathcal{V}})} = e(g_1^{u^{'}}, g_n) \cdot e(g_1^{\tilde{u}}, g_n)^{-1} = z^{u^{'} - \tilde{u}}$$

Equations (3) and (4) prove the verification of the AHO signature on P_k. Equations (5) and (6) show the verification of the AHO signature on $\tau_{u'}$, which ensures that $u^{'} = \delta_1^{'} c_1 + \ldots + \delta_L^{'} c_L$, where $1 \le \delta_\ell^{'} \le \eta_\ell$. Then, we have $z^{u^{'} - \tilde{u}} = z^{(\delta_1^{'} - |V_1^-|)c_1 + \ldots + (\delta_L^{'} - |V_L^-|)c_L}$ from $\tilde{u} = -(|V_1^-|c_1 + \ldots + |V_L^-|c_L)$, and

$1-|V_\ell^-| \leq \delta_\ell^{'}-|V_\ell^-| \leq \eta_\ell-|V_\ell^-|$. By setting $\delta_\ell = \delta_\ell^{'}-|V_\ell^-|$, we obtain $z^{u^{'}-\tilde{u}} = z^u$ and $1-|V_\ell^-| \leq \delta_\ell \leq \eta_\ell-|V_\ell^-|$, i.e, $1 \leq \delta_\ell+|V_\ell^-| \leq \eta_\ell$. It is the verification of the accumulator in Chap. 3. Thus, $U_k \cap V_\ell^+ \neq \emptyset$ or $U_k \cap V_\ell^- \neq V_\ell^-$ for all $1 \leq \ell \leq L$ is verified.

Verify: Given ipk, the proof σ, and the proved CNF formula Ψ, verify the validity of σ as follows.

(i) As in **ProofGen**, compute the accumulator $acc_{\mathcal{V}}$, and set $\tilde{u} = -(|V_1^-|c_1 + \ldots + |V_L^-|c_L)$ and $\tau_{\tilde{u}} = g_1^{\tilde{u}}$.
(ii) If the verification of all GS proofs $\{\pi_i\}_{i=1}^5$ succeeds, accept σ.

5.2 Efficiency Comparisons

Since the proposed system is similar to the previous system [4], it has the similar asymptotic efficiency. The size of the attribute proof σ is $O(1)$, and the size of the certificate $cert_k$ is also $O(1)$. But, the size of the issuer's public key ipk is different from the previous. In the previous system, the maximum number of $\zeta_\ell = |U \cap V_\ell|$ for V_ℓ (the attribute set of the ℓ-th clause in CNF formulas) is fixed in the setup. The number of the AHO signatures for Φ in ipk is $\prod_{1 \leq \ell \leq L} \zeta_\ell$. But, in our system, the number is $\prod_{1 \leq \ell \leq L} \eta_\ell$ where η_ℓ is the maximum number of the attributes in ℓ-th clause which corresponds to $|V_\ell|$. Due to $|U \cap V_\ell| \leq |V_\ell|$, ipk in our system is longer that in the previous system, which is a trade-off to the adaptation to negations in proved CNF formulas.

The computational costs are also similar to the previous system. In **ProofGen**, the computation of the witness $W_{\mathcal{V}}$ depends on the parameters ($acc_{\mathcal{V}}$ also depends on the parameters, but the cost of $W_{\mathcal{V}}$ is heavier). The cost is the same as the previous system, since the exponentiation of the integer c_ℓ is only changed to the exponentiation of $-c_\ell$ for the negated attributes, and the multiplications of OR literals remain.

5.3 Security Considerations

As in the journal version [5] of the previous system [4], we can prove that the proposed system satisfies the misauthentication resistance under the security of the AHO signatures and the proposed accumulator. The security proof of the previous system constructs two types of forgeries by interacting with an adversary winning the misauthentication resistance game and extracting committed secret values in the attribute proof σ forged by the adversary. One forgery is for AHO signatures, and another forgery is for the accumulator. As well as the previous system, in the proposed system, the attribute set U_k of the proving user is ensured by the AHO signature on $P_k = \prod_{i \in U_k} g_i$, and the user proves that U_k satisfies the proved CNF formula Ψ as $U_k \cap V_\ell^+ \neq \emptyset$ or $U_k \cap V_\ell^- \neq V_\ell^-$ for all $1 \leq \ell \leq L$ by the verification of the proposed accumulator, where the correctness of $\tau_{u'} = g_1^{u^{'}}$ is ensured by an AHO signature. Thus, similarly to the proof for the previous system, we can prove the misauthentication resistance.

As for the anonymity, the security proof is also similar to that for the previous system, where the methodology of a sequence of games is used. For the original anonymity game, we can consider the modified game where the GS commitments are replaced by ones using the CRS in the WI setting. In this modified game, since the adversary has no information, the advantage of the adversary in the anonymity game is negligible. Furthermore, this modified game and the original game are indistinguishable due to the indistinguishability of CRS in the real protocol and the WI setting under the DLIN assumption. In our system, the attribute proof σ consists of the same components as those in the previous system, i.e., the re-randomized AHO signatures, GS commitments, and GS proofs. Thus, in the same proof as that for the previous system, we can prove the anonymity.

The security proofs in our system will be shown in the journal version of this paper.

6 Conclusions

In this paper, we have proposed an anonymous credential system with the constant-size attribute proofs, where any CNF formula with negations can be proved. As the key primitive, we have constructed an accumulator to verify the CNF formulas with negations, based on the previous accumulator [4] for limited CNF formulas without negations.

One of our future work is to apply the proposed system to eID systems.

Acknowledgments. This work was partially supported by JSPS KAKENHI Grant Number 19K11964.

References

1. Abe, M., Fuchsbauer, G., Groth, J., Haralambiev, K., Ohkubo, M.: Structure-preserving signatures and commitments to group elements. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 209–236. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14623-7_12
2. Abe, M., Haralambiev, K., Ohkubo, M.: Singing on elements in bilinear groups for modular protocol design. Cryptology ePrint Archive, Report 2010/133 (2010). http://eprint.iacr.org/. (This was merged and presented in [1])
3. Boneh, D., Boyen, X., Shacham, H.: Short group signatures. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 41–55. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-28628-8_3
4. Begum, N., Nakanishi, T., Funabiki, N.: Efficient proofs for CNF formulas on attributes in pairing-based anonymous credential system. In: Kwon, T., Lee, M.-K., Kwon, D. (eds.) ICISC 2012. LNCS, vol. 7839, pp. 495–509. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37682-5_35
5. Begum, N., Nakanishi, T., Funabiki, N.: Efficient proofs for CNF formulas on attributes in pairing-based anonymous credential system. IEICE Trans. Fundam. **96-A**(12), 2422–2433 (2013)

6. Camenisch, J., Chaabouni, R., Shelat, A.: Efficient protocols for set membership and range proofs. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 234–252. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-89255-7_15
7. Camenisch, J., Groß, T.: Efficient attributes for anonymous credentials. In: Proceedings of the ACM Conference on Computer and Communications Security (ACM CCS 2008), pp. 345–356 (2008)
8. Camenisch, J., Kohlweiss, M., Soriente, C.: An accumulator based on bilinear maps and efficient revocation for anonymous credentials. In: Jarecki, S., Tsudik, G. (eds.) PKC 2009. LNCS, vol. 5443, pp. 481–500. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-00468-1_27
9. Camenisch, J., Lysyanskaya, A.: Dynamic accumulators and application to efficient revocation of anonymous credentials. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 61–76. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45708-9_5
10. Groth, J., Sahai, A.: Efficient non-interactive proof systems for bilinear groups. In: Smart, N. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 415–432. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78967-3_24
11. Hanser, C., Slamanig, D.: Structure-preserving signatures on equivalence classes and their application to anonymous credentials. In: Sarkar, P., Iwata, T. (eds.) ASIACRYPT 2014. LNCS, vol. 8873, pp. 491–511. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-45611-8_26
12. Izabachène, M., Libert, B., Vergnaud, D.: Block-wise P-signatures and non-interactive anonymous credentials with efficient attributes. In: Chen, L. (ed.) IMACC 2011. LNCS, vol. 7089, pp. 431–450. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25516-8_26
13. Sadiah, S., Nakanishi, T., Funabiki, N.: Anonymous credential system with efficient proofs for monotone formulas on attributes. In: Tanaka, K., Suga, Y. (eds.) IWSEC 2015. LNCS, vol. 9241, pp. 262–278. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-22425-1_16
14. Sudarsono, A., Nakanishi, T., Funabiki, N.: Efficient proofs of attributes in pairing-based anonymous credential system. In: Fischer-Hübner, S., Hopper, N. (eds.) PETS 2011. LNCS, vol. 6794, pp. 246–263. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22263-4_14