On Zero-Knowledge Identification Based on q-ary Syndrome Decoding

Rong Hu
Graduate School of Mathematics
Kyushu University
Fukuoka, 819-0395, Japan
Email: r-hu@math.kyushu-u.ac.jp

Kirill Morozov
Institute of Mathematics for Industry
Kyushu University
Fukuoka, 819-0395, Japan
Email: morozov@imi.kyushu-u.ac.jp

Tsuyoshi Takagi
Institute of Mathematics for Industry
Kyushu University
Fukuoka, 819-0395, Japan
Email: takagi@imi.kyushu-u.ac.jp

***Abstract*—Cayrel et al at SAC 2010 proposed a zero-knowledge identification scheme based on syndrome decoding of q-ary codes. It is a 5-pass scheme with soundness error $\frac{q}{2(q-1)}$. We propose an alternative to this scheme by generalizing (binary) Stern zero-knowledge identification from CRYPTO 1993 directly to q-ary setting. Our proposal is a 3-pass scheme with soundness error 2/3. We show that it is superior to Cayrel et al scheme in terms of communication cost for the case $q = \{3, 4\}$. A possible application for q-ary code-based identification schemes with small q is a proof of plaintext knowledge for code-based public key encryption.**

***Index Terms*—zero-knowledge protocol; identification protocol; syndrome decoding; information set decoding;**

I. Introduction

Identification protocols serve the goal of entity authentication. Their applications include authentication and access control services such as remote login, credit card purchases and many others. Usually, these are interactive two-party protocols, where one party (called a *prover*) wants to prove a possession of some private identification information to another party (called a *verifier*).

In the public-key setting, on which we focus in our work, most of practically used schemes are challenge-and-response. In this case, *zero-knowledge (ZK)* identification schemes have an advantage in the sense that no information on the private key is released to the verifier. If the eavesdropper observes the communication between the prover and the verifier, then clearly she does not gain any information on the private key as well. Such the scheme is usually constructed as a ZK proof of knowledge with private key as a witness. This approach was pioneered by Fiat and Shamir in [9]. We refer the reader to [26, Ch. 9] for more information on identification protocols.

We focus on code-based identification protocols because their security is based on hardness of decoding random codes – the problem which is not known to have an efficient solution even using quantum computation. Although quantum computers remain at the early prototype stage of development, it is desirable to have a secure *postquantum* alternative for the schemes based on hardness of discrete logarithm or integer factorization [26, Ch. 9].

Our scheme is based on hardness of Syndrome Decoding – a well studied problem – see e.g. [16], [17], [24], [6], [21], [2]. Currently, the most efficient attack against the parameters relevant to our scheme is Information Set Decoding (ISD) [22].

Related Works. The first code-based zero-knowledge identification protocol was proposed by Harari [13] in 1988, however Véron showed it insecure [27], and only recently Malek and Miri [18] fixed the problem. The first secure zero-knowledge identification protocol based on coding was presented by Stern [25]. It is a 3-pass protocol with soundness error $2/3$, based on hardness of syndrome decoding of binary codes. Girault showed a 3-pass identification scheme [11], but it was not practical. Véron proposed a protocol [28] based on (binary) General Decoding problem (a dual of Syndrome Decoding) but this scheme was recently shown insecure by Jain et al [14] who also presented a secure alternative. Gaborit and Girault [10] proposed, in particular, a q-ary code based authentication scheme, but it was based on specific double-circulant binary codes. Kawachi et al proposed a q-ary identification scheme in the context of lattices, which is similar to ours. Xagawa and Tanaka [29] modified the scheme [15] to get a proof of plaintext knowledge for NTRU public key encryption. Recently, Cayrel, Véron and Alaoui [7] presented a ZK identification scheme (we will call it the CVA scheme) based on syndrome decoding of codes over $\mathbb{F}_q$ ($q > 2$) where the soundness error is reduced to $\frac{q}{2(q-1)}$, which is essentially $1/2$ for large q.

Our Contribution. The main motivation for our work was to construct the code-based zero-knowledge identification scheme for small $q > 2$, this is also a new q-ary code based scheme based on stern's ID scheme. Such schemes may be applied for instance for proof of plaintext knowledge [19] for public-key encryption based on codes over $\mathbb{F}_q$ [4].

We constructed a generalization of Stern scheme for q-ary case with soundness error $2/3$ and confirmed that its communication cost is superior as compared to that of CVA scheme for $q = \{3, 4\}$.

In particular, let us consider the 80-bit equivalent security level, and overall soundness error 2^{-16}. Then for $q = 3$, our proposed scheme has 28 rounds and communication cost of 4.79 kilobytes against that of 39 rounds and 7.50 kilobytes, respectively, with CVA scheme. When $q = 4$, both schemes have 28 rounds, and we have 4.33 kilobytes with our scheme against 4.69 kilobytes with CVA scheme.

It is worth noting, however, that already for $q \geq 5$, CVA scheme comes on top in terms of communication cost. In

particular, for $q = 5$, our proposal required 28 rounds and 5.08 kilobytes of communication against 24 rounds and 4.99 kilobytes with CVA scheme.

Section II presents necessary definitions and tools. Our proposed scheme is presented in Section III, its security argued in Section IV. Performance evaluation is given in Section V and Section VI contains concluding remarks and open questions.

II. Preliminaries

We define by $wt(x)$ the Hamming weight of $x \in \mathbb{F}_q^n$. The set of all permutations of n elements is denoted by $\mathcal{S}_n$. For $x, y \in \mathbb{F}_q^n$, we denote by $x + y$ an element-wise addition of x and y over $\mathbb{F}_q$.

An (n, k, d) linear code is a k-dimensional subspace of an n-dimensional vector space over a finite field $\mathbb{F}_q$, and d is the *minimal distance* of the code [23].

Definition 1 (Gilbert-Varshamov Bound [23])**.** *Let $H_q(x) = x\log_q(q-1) - x\log_q(x) - (1-x)\log_q(1-x)$ be the q-ary entropy function. Let $d/n = \zeta$, and the rate of a q-ary linear code $R = k/n$. If $0 \leq \zeta \leq (q-1)q$, then for any n, there exists an (n, k, d) code such that*

$$R \leq 1 - H_q(\zeta).$$

We will use the following generalization of a permutation defined in [7].

Definition 2 ([7])**.** *Let $\Sigma \in \mathcal{S}_n$ and $\gamma = (\gamma_1, ..., \gamma_n) \in \mathbb{F}_q^n$ such that $\forall i, \gamma_i \neq 0$. We define the transformation $\Pi_{\gamma,\Sigma}$ as follows:*

$$\begin{aligned}\Pi_{\gamma,\Sigma} : \mathbb{F}_q^n &\to \mathbb{F}_q^n \\ v &\longmapsto (\gamma_{\Sigma(1)} v_{\Sigma(1)}, ..., \gamma_{\Sigma(n)} v_{\Sigma(n)}).\end{aligned}$$

It was noted in [7] that $\forall v \in \mathbb{F}_q^n$ and $\alpha \in \mathbb{F}_q$, we have $\Pi_{\gamma,\Sigma}(\alpha v) = \alpha\Pi_{\gamma,\Sigma}(v)$, and that this transformation preserves the weight, i.e. $\forall v \in \mathbb{F}_q^n$, $wt(\Pi_{\gamma,\Sigma}(v)) = wt(v)$. In addition, we observe that $\forall v, w \in \mathbb{F}_q^n$: $\Pi_{\gamma,\Sigma}(v + w) = \Pi_{\gamma,\Sigma}(v) + \Pi_{\gamma,\Sigma}(w)$, which implies that $\Pi_{\gamma,\Sigma}$ is a *linear* transformation.

A. Hardness Assumption

Definition 3. *Syndrome Decoding (SD) Problem.*
Input: $H \xleftarrow{\$} \mathbb{F}_q^{(n-k)\times n}$, $y \xleftarrow{\$} \mathbb{F}_q^{n-k}$ and $0 < \omega \in \mathbb{N}$.
Decide: If there exists $e \in \mathbb{F}_q^n$ s.t. $He^T = y$ and $w_H(e) \leq \omega$.

This problem was shown to be NP-complete by Berlekamp et al [5]. The best algorithm for solving it is the Information Set Decoding algorithm by Peters [22].

B. Commitment Schemes

Zero-knowledge proofs use commitment schemes as a building block. For this section, we borrow the presentation of [19]. A commitment scheme consists of two phases: the first one is *committing*, where a sender P provides a receiver V with an evidence about input b. The cheating receiver $\widetilde{V}$ cannot learn b before the second phase, called *opening*, when P reveals b to V. The cheating sender $\widetilde{P}$ cannot successfully open $b' \neq b$.

Let us denote by $\langle P, V\rangle_{A,st}$ the *view* of the party $A \in \{P, V\}$ at the stage st, which is a concatenation of all the messages sent and received by A, along with its local randomness.

Definition 4. *A protocol is said to* securely implement string commitment*, if at the end of its execution by PPT Turing machines P (with input $b \in \mathbb{F}_2^l$, $l \in \mathbb{N}$) and V, the following properties hold:*
(Correctness) $\Pr[\langle P(b), V\rangle_{V,Open} =$ *"ACCEPT"*$]$ *with overwhelming probability.*
(Hiding) *For any PPT $\widetilde{V}$, any $l \in \mathbb{N}$, any $b \in \mathbb{F}_2^l$ and $b' \in \mathbb{F}_2^l$ such that $b' \neq b$, after the committing stage, but before the opening stage, the distributions*

$$\langle P(b), \widetilde{V}\rangle_{\widetilde{V},Commit} \quad \text{and} \quad \langle P(b'), \widetilde{V}\rangle_{\widetilde{V},Commit}$$

are indistinguishable. Depending of the type of indistinguishability, hiding can be statistical *or* computational*.*
(Binding) *For any $\widetilde{P}$, any $l \in \mathbb{N}$, and $b' \in \mathbb{F}_2^l$ there exists $b \in \mathbb{F}_2^l$ which can be computed by P after the committing stage, such that the probability*

$$\Pr[\langle \widetilde{P}(b'), V\rangle_{V,Open} = \textit{"ACCEPT"}]$$

is negligible. If $\widetilde{P}$ restricted to run in PPT, then binding is called computational*, if $\widetilde{P}$'s computing power is not restricted, then the binding is* statistical*.*

Note that committing to binary vectors does not pose a problem, since mathematical objects, which we are working with, will eventually be represented as binary vectors for the actual implementation.

In the random oracle model (ROM) [3], a string commitment which is both computationally hiding and binding can be implemented using (idealized) cryptographic hash function. In the standard model, a computationally hiding and statistically binding code-based commitment commitment schemes are known [8], [14].

We denote by $Com(x_1, x_2, \ldots)$ a commitment to values $(x_1, x_2, \ldots)$.

III. Proposed Protocol

We present our proposed zero-knowledge identification protocol, which is a generalization of (binary) Stern identification scheme [25] to q-ary case. The main difference is that an ordinary permutation cannot be used now, since a permutation of the q-ary witness does indeed release some information on it, hereby violating the zero-knowledge requirement. We avoid this problem by employing the generalized permutation $\Pi_{\gamma,\Sigma}$ introduced in [7] exactly for this purpose. Indeed, it is easy to check that for any input in $\{x \in \mathbb{F}_q^n | wt(x) = \omega\}$, this transformation outputs a vector with uniform distribution in $\{x \in \mathbb{F}_q^n | wt(x) = \omega\}$ given that (uniformly chosen) γ and Σ are unknown. Since transformation $\Pi_{\gamma,\Sigma}$ happens to be linear as noted in Section II, it can be used directly in the construction of [25].

Key Generation.
Input: Given a security parameter and q, choose n, k, and ω, then compute:
$H \xleftarrow{\$} \mathbb{F}_q^{(n-k)\times n}$, $e \xleftarrow{\$} \mathbb{F}_q^n$ such that $wt(e) = \omega$, and $y \leftarrow He^T$.
Output: the private key sk, and the public key (identification) pk: $(\mathrm{sk}, \mathrm{pk}) = (e, (y, H, \omega))$.

Technically, our protocol is an interactive zero-knowledge proof of knowledge [12] for the predcate:

$$\mathcal{P}(sk, pk) = \text{“With respect to } pk,\ e \text{ is such that}$$
$$y = He^T \text{ and } wt(e) = \omega\text{”},$$

where pk is a common data, and sk is a witness.

Our protocol is presented in Table I. It has soundness error $2/3$, which can be reduced to $(2/3)^\delta$ by iterating this protocol independently for δ rounds.

IV. Security Proof

Our proof uses the approach presented in [25], [28]. We also follow [19] by presenting our proof in the standard model.

Let us denote the cheating prover by $\widetilde{P}$ and a cheating verifier by $\widetilde{V}$.

Proposition 1. *The protocol in Table I is an interactive zero-knowldge proof of the predicate $\mathcal{P}(sk, pk)$, in the standard model.*

Proof:
Completeness: If P knows the witness, it is easy to show that she can answer each challenge correctly, in particular one will need to use the fact that the transformation $\Pi_{\gamma,\Sigma}$ is linear and that it preserves the weight.

Soundness:

Lemma 1. *If V accepts $\widetilde{P}$'s proof with probability at least $\left(\frac{2}{3}\right)^r + \epsilon$, then there exists a PPT algorithm which with overwhelming probability computes the witness e.*

Proof: Let T be an execution tree of the protocol between $\widetilde{P}$ and V that corresponds to all possible challenges by V. Verifier V can send 3 possible challenges in each round. Suppose that the binding property of the underlying commitment scheme holds. Then, we present a PPT algorithm (called *witness extractor*) that computes the witness e from a vertex with 3 descendants.

Suppose there exists a vertex with 3 descendants. This implies that all the three challenges were correctly answered. Suppose now that the following responses were provied by $\widetilde{P}$:

- $b = 0 : (u_0, \gamma_0, \Sigma_0)$,
- $b = 1 : (w_1, \gamma_1, \Sigma_1)$ (w_1 corresponds to $u + e$),
- $b = 2 : (z_2, t_2)$ (correspond to $\Pi_{\gamma,\Sigma}(u)$ and $\Pi_{\gamma,\Sigma}(e)$, respectively).

Now, according to the checks perfomed by V, we have : $(\gamma_0, \Sigma_0, Hu_0^T) = Open(c_1) = (\gamma_1, \Sigma_1, Hw_1^T - y)$. Remember that the binding property is assumed to hold, then we have $\gamma_0 = \gamma_1$, $\Sigma_0 = \Sigma_1$, and $Hu_0^T = Hw_1^T - y$. Using correctness of c_2 and c_3, we also have $z_2 = \Pi_{\gamma_0,\Sigma_0}(u_0)$, $z_2 + t_2 = \Pi_{\gamma_1,\Sigma_1}(w_1)$, and $wt(t_2) = \omega$. This implies $t_2 = (t_2 + z_2) - z_2 = \Pi_{\gamma_0,\Sigma_0}(w_1 - u_0)$ where $wt(w_1 - u_0) = \omega$. Therefore, the expression $H(w_1 - u_0)^T = Hw_1^T - Hu_0^T = y$ shows that $w_1 - y_0$ is a valid witness.

The rest of the proof of this lemma is exactly as in [28]. The omitted part shows that the probability for T to have a vertex with 3 descendants is at least ϵ. ∎

Zero-knowledge: This property states that the cheating polynomial-time verifier $\widetilde{V}$ learns no information on the witness irrespective of her cheating strategy.

Lemma 2. *Protocol in Table I is zero-knowledge if the used commitment scheme is hiding.*

Proof: The flavor of zero-knowledge – computational or statistical – depends on the corresponding flavor of the hiding property of the underlying commitment scheme.

Next, we present a PPT algorithm called the *simulator* which works in expected polynomial time, and which constructs a protocol transcript whose distribution is indistinguishable from the transcript of the protocol execution between honest P and V.

Since the zero-knowledge property must hold irrespective of $\widetilde{V}$'s strategy, we denote this strategy by $St(c_1, c_2, c_3)$, and assume that the challenges are chosen according to it. The simulator works as follows:

1) Pick a challenge $b \xleftarrow{\$} \{0, 1, 2\}$.
   - If $b = 0$, choose $u \xleftarrow{\$} \mathbb{F}_q^n$, $\gamma \xleftarrow{\$} (\mathbb{F}_q^*)^n$, $\Sigma \xleftarrow{\$} \mathcal{S}_n$, compute $c_1 = Com(\gamma, \Sigma, Hu^T)$, $c_2 = Com(\Pi_{\gamma,\Sigma}(u))$, $c_3 = Com(0)$, and $Response = (u, \gamma, \Sigma)$.
     It is easy to check that the distributions of c_1, c_2, c_3 and $Response$ are identical to the corresponding distributions in the actual protocol transcript.
   - If $b = 1$, choose $u \xleftarrow{\$} \mathbb{F}_q^n$, $\gamma \xleftarrow{\$} (\mathbb{F}_q^*)^n$, $\Sigma \xleftarrow{\$} \mathcal{S}_n$, and $w = u + z$, where $z \in \mathbb{F}_q^n$ is such that $Hz^T = y$, $z \neq e$, $wt(z) \neq \omega$. Then, compute $c_1 = Com(\gamma, \Sigma, Hu^T)$, $c_2 = Com(0)$, $C_3 = Com(\Pi_{\gamma,\Sigma}(w))$, and $Response = (w, \gamma, \Sigma)$. Again, it is easy to check that the values in $Response$ are consistent with the checks, and also that distributions of the commitments and $Response$ are identical to those in the actual protocol transcript. In particular, a uniform u serves as a one-time pad for z.
   - If $b = 2$, choose $u \xleftarrow{\$} \mathbb{F}_q^n$, $\gamma \xleftarrow{\$} (\mathbb{F}_q^*)^n$, $\Sigma \xleftarrow{\$} \mathcal{S}_n$, and $z \xleftarrow{\$} \{x \in \mathbb{F}_q^n | wt(x) = \omega\}$. Then, compute $c_1 = Com(0)$, $c_2 = Com(\Pi_{\gamma,\Sigma}(u))$, $c_3 = Com(\Pi_{\gamma,\Sigma}(u + z))$, and $Response = (\Pi_{\gamma,\Sigma}(u), \Pi_{\gamma,\Sigma}(z))$. The values in $Response$ are consistent with the checks, and it is easy to see that distributions of the commitments and $Response$ are identical to those in the actual transcript.
2) The simulator computes $b' = St(c_1, c_2, c_3)$.
3) If $b = b'$, then the simulator outputs a transcript which includes H, b and $Response$, otherwise go to Step 1.

Now, in 3δ iterations on the average, the above algorithm constructs the protocol transcript which is indistinguishable

TABLE I
OUR IDENTIFICATION PROTOCOL.

Prover P		Verifier V
$(sk, pk) = (e, (y, H, \omega))$ $u \xleftarrow{\$} \mathbb{F}_q^n$ $\Sigma \xleftarrow{\$} \mathcal{S}_n$, $\gamma \xleftarrow{\$} (\mathbb{F}_q^*)^n$		$pk = (y, H, \omega)$
$c_1 = Com(\gamma, \Sigma, Hu^T)$ $c_2 = Com(\Pi_{\gamma,\Sigma}(u))$ $c_3 = Com(\Pi_{\gamma,\Sigma}(u+e))$		
	$c_1, c_2, c_3 \longrightarrow$	
	$\longleftarrow b$	$b \xleftarrow{\$} \{0,1,2\}$
If $b = 0$	$u, \gamma, \Sigma \longrightarrow$	P opens c_1 and c_2, V checks correctness of c_1, c_2, using H which is public.
If $b = 1$	$u+e, \gamma, \Sigma \longrightarrow$	P opens c_1 and c_3, V checks correctness using $Hu^T = H(u+e)^T - y$.
If $b = 2$	$\Pi_{\gamma,\Sigma}(u), \Pi_{\gamma,\Sigma}(e) \longrightarrow$	P opens c_2 and c_3, and checks correctness using $\Pi_{\gamma,\Sigma}(u) + \Pi_{\gamma,\Sigma}(e) = \Pi_{\gamma,\Sigma}(u+e)$, and checks that $wt(\Pi_{\gamma,\Sigma}(e)) = \omega$.

from the transcript of the protocol execution between the honest parties. ■

Now, Lemmas 1 and 2 conclude the proof of Proposition 1. ■

V. PERFORMANCE EVALUATION

For simplicity of our calculations, we assume that commitments are implemented using random oracles, in the same way as it is done in [25], [7]. Then the commitment function will be computed as some cryptographic hash function h with output size $l_h = 160$ bits. When checking the prover's responses, the verifier will simply check that the output of the hash functions on the values contained in the response are the same as those sent as commitments. For instance, when $b = 0$, the verifier will check if $h(\gamma, \Sigma, Hu^T) = c_1$ and $h(\Pi_{\gamma,\Sigma}(u)) = c_2$.

We will assume that the values u and (γ, Σ) are computed using pseudorandom generators (PRG) with seed of length $l_s = 128$ bits. Hereby, only the seeds can be sent to save on communication, in the same way, as it is done in [25], [7].

The syndrome decoding problem is hardest to solve when $k \approx n/2$ [6] and ω is chosen slightly below the Gilbert-Varshamov bound (see Section II). We take $k = n/2$, and hereby we only need to choose n (as an even integer).

In order to compute the recommended parameters for a given security level, we will use the currently best algorithm for solving q-ary Syndrome Decoding problem that is the Information Set Decoding algorithm by Peters [22]. Since the equations for its parameters do not have a short analytic representation, we use the C program introduced by Peters, which is available at https://bitbucket.org/cbcrypto/isdfq/src.

We let N be the number of bits needed to represent an element of $\mathbb{F}_q$, and we set $N = \lceil \log_2 q \rceil$. Let δ be the number of rounds. We assume that the common matrix H is in the systematic form.

Our scheme has the following parameters.

Size of the common matrix: $k^2 N$.
Size of the public identification: kN.
Size of the secret key: nN.

In fact, the above parameters are the same as in CVA scheme.

Total communication cost, in bits:
$\delta\big(3l_h + 2 + (3l_s + 3nN)/3\big)$.
Prover's computation complexity over $\mathbb{F}_q$, per round:
$(k^2 + n + \omega)$ multiplications and $(k^2 + 2\omega)$ additions.

Concerning calculation of the communication cost, we note that the above values are the average assuming that the challenges are made uniformly at random.

Cayrel et al [7] offer the following estimate for the communication cost of their protocol:

$$\delta(2l_h + N + nN + 1 + (l_\gamma + l_\Sigma + nN)/2),$$

where $l_\gamma = l_\Sigma = l_s$ is the length of a seeds of some PRG used to obtain γ and Σ. For a fair comparison with our scheme, we assume that (γ, Σ) are generated from a single seed in their scheme.

Next, we provide recommended parameters for the equivalent security level of 80 bits and soundness error 2^{-16} – to satisfy the later requirement our scheme needs 28 rounds (independently of q). Our recommended parameters for $q = 3$, along with the corresponding parameters for Cayrel et al scheme for comparison, are presented in Table V. In this case, our scheme requires 4.79 kilobytes of communication cost that is by 36% smaller than the communication cost of CVA scheme being 7.50 kilobytes; besides their scheme requires

TABLE II
PERFORMANCE COMPARISON OF OUR PROPOSAL AND CVA SCHEME FOR $q = 3$.

$q = 3$, $n = 396$, $k = 198$, $\omega = 62$	CVA [7]	Our Proposal
Number of Rounds	39	28
Matrix size (kilobytes)	9.57	9.57
Public key (bits)	396	396
Secret key (bits)	792	792
Communication (kilobytes)	7.50	4.79
Prover's Computation over $\mathbb{F}_3$	$2^{20.58}$ multiplications, $2^{20.54}$ additions	$2^{20.08}$ multiplications, $2^{20.07}$ additions

TABLE III
PERFORMANCE COMPARISON OF OUR PROPOSAL AND CVA SCHEME FOR $q = 4$.

$q = 4$, $n = 328$, $k = 164$, $w = 61$	CVA [7]	Our Proposal
Number of Rounds	28	28
Matrix size (kilobytes)	6.57	6.57
Public key (bits)	328	328
Secret key (bits)	656	656
Communication (kilobytes)	4.69	4.33
Prover's Computation over $\mathbb{F}_4$	$2^{19.56}$ multiplications, $2^{19.53}$ additions	$2^{19.54}$ multiplications, $2^{19.53}$ additions

TABLE IV
PERFORMANCE COMPARISON OF OUR PROPOSAL AND CVA SCHEME FOR $q = 5$.

$q = 5$, $n = 292$, $k = 146$, $w = 60$	CVA [7]	Our Proposal
Number of Rounds	24	28
Matrix size (kilobytes)	7.81	7.81
Public key (bits)	438	438
Secret key (bits)	876	876
Communication (kilobytes)	4.99	5.08
Prover's Computation over $\mathbb{F}_5$	$2^{19.01}$ multiplications, $2^{18.97}$ additions	$2^{19.21}$ multiplications, $2^{19.20}$ additions

39 rounds. When $q = 4$ (see Table III), our scheme requires 4.33 kilobytes of communication that is by 8% smaller than 4.69 kilobytes of communication required for CVA scheme; the number of rounds is 28 in both protocols. For $q = 5$, the performance evaluation results are given in Table IV. Now, our scheme requires 5.08 kilobytes of communication, as compared to 24 rounds and 4.99 kilobytes with CVA, which is by 2% smaller than with ours. For $q > 5$, the CVA scheme is superior to our scheme in terms of communication cost, since the soundness error of their scheme approaches $1/2$, when q is growing. We note that the complexity of Information Set Decoding is growing when q increases, hence larger q require smaller n for the same security level.

VI. CONCLUSION

In this paper, we presented a zero-knowledge identification scheme based on q-ary syndrome decoding with soundness error $2/3$, which is a generalization of (binary) Stern scheme to q-ary case. Our scheme is superior to the CVA scheme [7] in terms of communication cost, but only for $q = \{3, 4\}$.

An open question is to reduce the soundness error to exactly $1/2$ in the case of small q, most importantly for $q = 2$, since a scheme working over the binary field is expected to have a fast implementation. Reducing the size of the public matrix is another natural open question.

ACKNOWLEDGMENTS

The authors would like to thank Pierre-Louis Cayrel and Sidi Mohamed El Yousfi Alaoui for their valuable comments.

The authors would like to thank the anonymous reviewers of AsiaJCIS 2013 for pointing out some errata.

The first author was supported by China Scholarship Council (CSC). The second author was supported by a *kakenhi* Grant-in-Aid for Young Scientists (B) 24700013 from Japan Society for the Promotion of Science.

REFERENCES

[1] A. Barg, Complexity issues in coding theory. In V. S. Pless and W. C. Huffman, editors, Handbook of Coding theory, volume I, chapter 7, pages 649C754. North-Holland, 1998.

[2] A. Becker, A. Joux, A. May, and A. Meurer, Decoding Random Binary Linear Codes in $2^{n/20}$: How 1 + 1 = 0 Improves Information Set Decoding. EUROCRYPT 2012: 520-536.

[3] M. Bellare, and P. Rogaway, Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. ACM Conference on Computer and Communications Security 1993: 62-73.

[4] D.J. Bernstein, T. Lange, and C. Peters, Wild McEliece. Selected Areas in Cryptography 2010: 143-158.

[5] E. Berlekamp, R. McEliece, and H. van Tilborg, On the inherent intractability of certain coding problems, IEEE Trans. on Inf. Theory 24, 1978: 384-386.

[6] A. Canteaut, and F. Chabaud, A new algorithm for finding minimum-weight words in a linear code: application to primitive narrow-sense BCH-codes of length 511. IEEE Transactions on Information Theory 44, 1998: 367-378.

[7] P.-L. Cayrel, and P. Véron, and S.M. El Yousfi Alaoui, A Zero-Knowledge Identification Scheme Based on the q-ary Syndrome Decoding Problem. Selected Areas in Cryptography 2010: 171-186.

[8] R. Dowsley, J. van de Graaf, J. Müller-Quade, and A.C.A. Nascimento, Oblivious Transfer Based on the McEliece Assumptions. ICITS 2008: 107-117.

[9] A. Fiat, and A. Shamir, How to Prove Yourself: Practical Solutions to Identification and Signature Problems. CRYPTO 1986:186-194.

[10] P. Gaborit, and M. Girault, Lightweight code-based identification and signature. ISIT 2007: 191-195.

[11] M. Girault, A (non-practical) three-pass identification protocol using coding theory. AUSCRYPT 1990:265-272.

[12] O. Goldreich, Foundations of Cryptography I: Basic Tools. Cambridge University Press, 2001.

[13] S. Harari, A new authentication algorithm. Coding Theory and Applications 1988: 91-105.

[14] A. Jain, S. Krenn, K. Pietrzak, and Aris Tentes, Commitments and Efficient Zero-Knowledge Proofs from Learning Parity with Noise. ASIACRYPT 2012: 663-680.

[15] A. Kawachi, K. Tanaka, and K. Xagawa, Concurrently Secure Identification Schemes Based on the Worst-Case Hardness of Lattice Problems. ASIACRYPT 2008: 372-389.

[16] P. Lee and E. Brickell, An observation on the security of McEliece's public key cryptosystem. EUROCRYPT 1988: 275-280.

[17] J. Leon, A probabilistic algorithm for computing minimum weights of large error-correcting codes. IEEE Transactions on Information Theory 34, 1988: 1354-1359.

[18] B. Malek, and A. Miri, Securing Harari's Authentication Scheme. I. J. Network Security 14(4): 206-210 (2012).

[19] K. Morozov, and T. Takagi, Zero-Knowledge Protocols for the McEliece Encryption. ACISP 2012: 180-193.

[20] M. Naor, Bit Commitment Using Pseudo-Randomness. CRYPTO 1989: 128-136.

[21] N. Sendrier, On the security of theMcEliece public-key cryptosystem. InM. Blaum, P.G. Farrell, and H. van Tilborg, editors, Information, Coding and Mathematics, pages 141C163. Kluwer, 2002.

[22] C. Peters, Information-Set Decoding for Linear Codes over F_q. PQCrypto 2010: 81-94.

[23] R. Roth, Introduction to coding theory. Cambridge University Press, 2006.

[24] J. Stern, A method for finding codewords of small weight. Coding Theory and Applications 388, 1989: 106-133.

[25] J. Stern, A new paradigm for public key identification. IEEE Transactions on Information Theory 42(6),1996: 1757-1768. A conference version appeared in J. Stern: A New Identification Scheme Based on Syndrome Decoding. CRYPTO 1993: 13-21.

[26] D. R. Stinson, Cryptography Theory and Practice, 3rd ed., Chapman & Hall/CRC, 2006.

[27] P. Vron, Cryptanalysis of Harari's Identification Scheme. IMA Conf. 1995: 264-269.

[28] P. Véron, Improved identification schemes based on error-correcting codes. Appl. Algebra Eng. Commun. Comput. 8(1), 1996: 57-69.

[29] K. Xagawa, and K. Tanaka, Zero-Knowledge Protocols for NTRU: Application to Identification and Proof of Plaintext Knowledge. ProvSec 2009: 198-213.