MPC for Group Reconstruction Circuits

Lúcás Críostóir Meier

June 8, 2022

Abstract

In this paper, we present a thing.

1 Introduction

Write the introduction

2 Background

Throughout this paper, we let $\mathbb{G}$ denote a group of prime order q, with generators G and H. Let $\mathbb{F}_q$ denote the scalar field associated with this group, and let $\mathbb{Z}/(q)$ denote the additive group of elements in this field.

We make heavy use of group homomorphisms throughout this paper. We let

$$\varphi(P_1, \ldots, P_m) : \mathbb{A} \to \mathbb{B}$$

denote a homomorphism from $\mathbb{A}$ to $\mathbb{B}$, parameterized by some public values $P_1, \ldots, P_m$. Commonly $\mathbb{A}$ will be a product of several groups $\mathbb{G}_1, \ldots, \mathbb{G}_n$, in which case we'd write:

$$\varphi(P_1, \ldots, P_m)(x_1, \ldots, x_n)$$

to denote the application of φ to an element $(x_1, \ldots, x_n)$ of the product group. We also often leave the public values P_i implicit.

2.1 Pedersen Commitments

2.2 Sigma Protocols

2.3 Maurer's φ-Proof

In [Mau09], Maurer generalized Schnorr's sigma protocol for knowledge of the discrete logarithm [Sch90] to a much larger class of relations. In particular, Maurer provided a sigma protocol for proving knowledge of the pre-image of a group homomorphism φ. We denote this protocol as a "φ-proof", and recapitulate the scheme here.

Given a homomorphism $\varphi : \mathbb{A} \to \mathbb{B}$, and a public value $X \in \mathbb{B}$, the prover wants to demonstrate knowledge of a private value $x \in \mathbb{A}$ such that $\varphi(x) = X$. The prover does this by means of Protocol 2.1:

Protocol 2.1: φ**-Proof**

Prover		**Verifier**
knows $x \in \mathbb{A}$		public $X \in \mathbb{B}$
$k \xleftarrow{R} \mathbb{A}$		
$K \leftarrow \varphi(k)$		
	$\xrightarrow{K}$	
		$c \xleftarrow{R} \mathbb{Z}/(p)$
	$\xleftarrow{c}$	
$s \leftarrow k + c \cdot x$		
	$\xrightarrow{s}$	
		$\varphi(s) \stackrel{?}{=} K + c \cdot X$

Here, p is chosen such that $\forall B \in \mathbb{B}.\ p \cdot B = 0$. In practice, we'll set $p = q$, which will work perfectly for the groups we use, which are all products of $\mathbb{G}$ or $\mathbb{Z}/(q)$.

Claim 2.1. Protocol 2.1 is a valid sigma protocol.

Completeness follows directly from the fact that φ is a homomorphism.

For the HVZK property, the simulator $\mathcal{S}(X, c)$ works by generating a random $s \xleftarrow{R} \mathbb{A}$, and then setting $K := \varphi(S) - c \cdot X$.

Finally, we prove 2-extractability. Given two verifying transcripts (K, c, s) and (K, c', s') sharing the first message, we extract a value $\hat{x}$ satisfying $\varphi(\hat{x}) = X$ as follows:

$$\begin{aligned}
\varphi(s) - c \cdot X = K &= \varphi(s') - c' \cdot X \\
\varphi(s) - \varphi(s') &= c \cdot X - c' \cdot X \\
\frac{1}{c - c'} \cdot \varphi(s - s') &= X \\
\varphi\left(\frac{s - s'}{c - c'}\right) &= X
\end{aligned}$$

Thus, defining $\hat{x} := (s - s')/(c - c')$, we successfully extract a valid pre-image.

We conclude that the protocol is a valid sigma protocol.

■

Maurer's protocol can also work even in the case where the order of the groups are not known, but this makes the challenge generation a bit more complicated, and we don't need this functionality in this work.

2.4 UC Security and the Hybrid Model

2.5 Ideal Functionalities for Sigma Protocols

Functionality 2.1: Zero-Knowledge Functionality $\mathcal{F}(\texttt{ZK}, \varphi)$

A functionality $\mathcal{F}$ for parties $P_1, \ldots, P_n$.

On receiving $(\texttt{prove}, \texttt{sid}, X, x)$ from P_i:
$\mathcal{F}$ checks that $\texttt{sid}$ has not been used by P_i before.
$\mathcal{F}$ checks that $\varphi(x) \stackrel{?}{=} X$.
$\mathcal{F}$ sends $(\texttt{proof}, \texttt{pid}_i, \texttt{sid}, X)$ to every party P_j.

2.6 Broadcast Functionalities

Functionality 2.2: Authenticated Broadcast Functionality C

A functionality C for parties $P_1, \ldots, P_n$.

On receiving $(\texttt{broadcast-in}, \texttt{sid}, m)$ from P_i:
C checks that $\texttt{sid}$ has not been used by P_i before.
C sends $(\texttt{broadcast-out}, \texttt{pid}_i, \texttt{sid}, m)$ to every party P_j.

3 Group Reconstruction Circuits

3.1 Formal Definition

3.2 Normalized Form

4 MPC Protocol for GRCs

5 Security Analysis

6 Applications

7 Limitations and Further Work

8 Conclusion

References

[Mau09] Ueli Maurer. Unifying Zero-Knowledge Proofs of Knowledge. In *AFRICACRYPT 2009*, volume 5580 of *LNCS*, pages 272–286. Springer, Berlin, Heidelberg, 2009.

[Sch90] C. P. Schnorr. Efficient Identification and Signatures for Smart Cards. In *CRYPTO 1989*, volume 435 of *LNCS*, pages 239–252, New York, NY, 1990. Springer.