



شبكة المعلومات الجامعية  
التوثيق الإلكتروني والميكروفيلم

# بسم الله الرحمن الرحيم



**MONA MAGHRABY**



شبكة المعلومات الجامعية  
التوثيق الإلكتروني والميكروفيلم



# شبكة المعلومات الجامعية التوثيق الإلكتروني والميكروفيلم



**MONA MAGHRABY**



شبكة المعلومات الجامعية  
التوثيق الإلكتروني والميكروفيلم

# جامعة عين شمس

## التوثيق الإلكتروني والميكروفيلم

### قسم

نقسم بالله العظيم أن المادة التي تم توثيقها وتسجيلها  
علي هذه الأقراص المدمجة قد أعدت دون أية تغيرات



### يجب أن

تحفظ هذه الأقراص المدمجة بعيدا عن الغبار



**MONA MAGHRABY**



AIN SHAMS UNIVERSITY  
FACULTY OF ENGINEERING  
Computer and Systems Engineering

**Different Consensus Algorithms Protocols used for Blockchain  
Technology and Smart Contract**

A Thesis submitted in partial fulfilment of the requirements of  
Master of Science in Electrical Engineering  
(Computer and Systems Engineering)

by

**Sherif Mohamed Samir Fth-Alla Shebl**

Bachelor of Science in Electrical Engineering  
Communication and Electronics Section

Faculty of Engineering (at Shoubra), Zagazig University, 2003

Supervised By

**Prof. Dr. Hoda Korashy Mohamed**

**Dr. Hazem Said Ahmed Mohamed**

Cairo - (2020)



AIN SHAMS UNIVERSITY  
FACULTY OF ENGINEERING  
Computer and Systems Engineering

**Different Consensus Algorithms Protocols used for Blockchain  
Technology and Smart Contract**

by

**Eng. Sherif Mohamed Samir Fth-Alla**

Bachelor of Science in Electrical Engineering

(Communication and Electronics) Department

Faculty of Engineering (at Shoubra), Zagazig University, 2003

Examiners' Committee

**Name and Affiliation**

**Signature**

Prof. Mostafa Mahmoud Aref

Head of computer science Dept., Computer and  
Information sciences, Ain Shams University

.....

Prof. Mohamed Mahmoud Ahmed Taher  
Computer and Systems, Ain Shams University

.....

Prof. Hoda Korashy Mohamed

Computer and Systems, Ain Shams University

.....

Date: 22 December 2020

# Statement

This thesis is submitted as a partial fulfilment of Master of Science in Electrical Engineering, Faculty of Engineering, Ain shams University.

The author carried out the work included in this thesis, and no part of it has been submitted for a degree or a qualification at any other scientific entity.

**Sherif Mohamed Samir Fth-Alla Shebl**

Sherif M. Samir

.....

Date: 22 December 2020

## Researcher Data

<b>Name</b>	: Sherif Mohamed Samir Fth-Alla Shebl
<b>Date of Birth</b>	: 22/06/1979
<b>Place of Birth</b>	: Cairo, Egypt
<b>Last academic degree :</b>	Diploma of Science in Electrical Engineering.
<b>Field of specialization</b>	: Computer Engineering and Systems.
<b>University issued the degree</b>	: Faculty of Engineering, Ain Shams University
<b>Date of issued degree</b>	: 2014
<b>Current job</b>	: Senior Software Development Engineer.

# Abstract

Secret key and public key are fundamental building blocks in cryptography that are used in numerous protocols. A trusted party is obligated to protect its secret key from any unfaithful party and spread its public key through secure communication channels to all other parties, this notation is called public key encryption.

Node resources scale sub linearly in all respects (storage, disk IO, computation, and bandwidth). So a need for a deeper concept arises. Our goal is to implement a stateless consensus architecture, which means that all blocks can be fully validated without any access to state. The motivation is that this will allow validators to not keep any main chain state, lowering validator hardware requirements and making it more accessible. One of the difficulties with this is that the witness sizes required for this can be very substantial. We can reduce this by using polynomial commitments, in which any number of data elements can be proven using just a single group element as a witness. One such scheme relying on sorted key-value lists. However, it introduces significant complexity in the form of several layers of caching and needing permutation arguments to merge those separate commitments.

This thesis aim is to address these two fundamental issues. First, we scale threshold cryptosystems, which protect secret keys by dividing them up across many parties. We discuss threshold signatures, verifiable secret sharing and distributed key generation protocols that can scale to millions of participants. Our protocols reduce execution time, depending on the scale. For example, at large scales, we reduce time from tens of hours to tens of seconds. At the core of most of our contributions lie new techniques for computing evaluation proofs in constant-sized polynomial commitments. Specifically, we describe how to decrease the time to calculate  $n$  proofs for a degree bound  $n$  polynomial from  $\theta(n^2)$  to  $\theta(n \log n)$ , at the cost of increasing proof size from  $\theta(1)$  to  $\theta(\log n)$ .



# Acknowledgment

First and above all, I sincerely appreciate the almighty God for His graces, strength, sustenance and above all, His faithfulness and love for my life. His benevolence has made me excel and successful in all my academic pursuits. I thank him for providing me this opportunity and granting me the capability to proceed successfully. I am grateful for his provision of joys, challenges and grace for grace that have been bestowed upon me during this research work, and indeed, throughout my life.

There are so many people I want to thank, it would be understandable if you get tired of reading this...

I would like to place on records my heartfelt and sincere thanks to my supervisor, Prof. Hoda Korashy Mohamed, for supporting and guiding me during these years. For the freedom to explore any topic I was interested in. For her guidance when I did not know what to do with that freedom. This thesis may not exist without her valuable advice and useful comments. And my thanks to my thesis committee, Dr. Hazem Said Ahmed Mohamed, for his invaluable Feedback. I appreciate their contribution of time and ideas to make my work productive and stimulating. Their valuable suggestions, comments and guidance encouraged me to learn more day by day. Their deep insights helped me at various stages of my research. Big thanks once again go to them for without them this work would never see the light as it is today.

I would like to take this opportunity to thank my family, for a loving home.

**My mother**, for all her sacrifices. For raising us. For letting us go. For never doubting us. For teaching us independence and responsibility through her faith in us. For being a role model. For her boundless love.

**My wife Rana**, for her love, understanding, and support to finish my academic degree.

Without their help, this thesis would not have been completed. I dedicate all my success to them.

Finally, A Heartfelt thanks to my supportive wonderful family and friends for supporting and encouraging me to keep me motivated to work valuable and harder

in all life moments. You have a constant source of strength and inspiration to me especially at the moment when there was no one to answer my queries.

Sherif Mohamed Samir Fth-Alla Shebl  
Computer and Systems Engineering  
Faculty of Engineering  
Ain Shams University  
Cairo, Egypt  
December 2020

# Table of Contents

<b>LIST OF FIGURES .....</b>	<b>X</b>
<b>LIST OF TABLES .....</b>	<b>XI</b>
<b>List of Abbreviations &amp; Algorithms.....</b>	<b>XII</b>
<b>1. Introduction .....</b>	<b>1</b>
1.1. How to Share a Public Key.....	1
1.2. How to Share a secret Key .....	3
1.3. Thesis Organization .....	6
<b>2. Preliminaries .....</b>	<b>8</b>
2.1. Cryptographic Assumptions .....	9
2.1.1. Bilinear Pairings Groups .....	9
2.2. $\ell$ -Strong Bilinear Diffie-Hellman Assumption ( $\ell$ -SBDH) .....	10
2.3. $\ell$ -Power Knowledge of Exponent .....	11
2.4. Polynomials for Cryptography .....	11
2.5. Polynomial Multipoint Evaluations .....	12
2.6. Lagrange Polynomial Interpolation .....	15
2.7. (Constant-sized) Polynomial Commitments .....	17
2.7.1. Notations and Definitions .....	18
2.7.2. Correctness and Security .....	20
2.8. Homomorphism .....	22
<b>3. Kate-Zaverucha-Goldberg (KZG) Constant-Sized Polynomial Commitments .....</b>	<b>25</b>
3.1. Batch Proofs .....	29
3.1.1. A set of evaluation proofs .....	29
3.1.2. Verifying a batch proof .....	29
3.2. Homomorphism .....	30
3.3. Vector Commitment .....	31
<b>4. Cryptographic Accumulators .....</b>	<b>33</b>
4.1. Notations .....	33
4.2. Constant-Sized Proofs .....	33

4.3. Bilinear Accumulators .....	34
4.3.1.Setup ceremony (Trusted Setup).....	34
4.3.2.Committing to set .....	35
4.3.3.Computing the coefficients of the accumulator Polynomial .....	35
4.3.4.Updating commitments .....	37
4.3.5.Membership Witnesses proof .....	37
4.3.6.Membership Witnesses verify .....	38
4.3.7.Membership Witnesses update .....	39
4.3.8.Non-membership Witnesses proof .....	40
4.3.9.Non-membership Witnesses verification .....	40
4.3.10. Subset witnesses proof .....	41
4.3.11. Subset witnesses verification .....	41
4.3.12. Disjointness Witnesses proof .....	42
4.3.13. Disjointness Witnesses verification .....	42
<b>5. Threshold Cryptosystems .....</b>	<b>44</b>
5.1. Introduction .....	44
5.2. Related Work .....	46
5.2.1.Threshold Signature Schemes (TSS).....	46
5.2.2.Verifiable Secret Sharing (VSS).....	47
5.2.3.Publicly-Verifiable Secret Sharing (PVSS).....	47
5.2.4.Distributed Key Generation (DKG).....	48
5.2.5.DKG Implementations .....	49
<b>6. Threshold Signature Preliminaries .....</b>	<b>51</b>
6.1. Introduction .....	51
6.2. Multi-signature schemes (MSS).....	52
6.3. Preliminaries .....	52
6.3.1.BLS Signature using Asymmetric Pairings .....	52
6.3.2.Aggregate signature from Bilinear maps .....	54
6.3.3.Aggregate public key from Bilinear .....	56
6.3.4.Threshold BLS signatures .....	56

<b>7. Threshold Signature Schemes .....</b>	<b>60</b>
7.1. Communication and Adversarial Model .....	60
7.1.1. Synchronous Communication .....	60
7.1.2. Static, Rushing, Threshold Adversaries .....	60
7.2. $\ell$ -Polynomial Diffie-Hellman (polyDH) Assumption .....	61
7.3. Threshold Signature Schemes (TSS) .....	61
7.3.1. (Threshold) BLS signatures .....	62
7.4. (Verifiable) Secret Sharing (VSS) .....	63
7.4.1. Kate et al.'s eVSS .....	64
7.5. Distributed Key Generation (DKG) .....	66
7.5.1. Kate's eJF-DKG .....	67
<b>8. Scalable Threshold Cryptosystems .....</b>	<b>70</b>
8.1. Scalable Threshold Signatures (TSS) .....	70
8.2. Quasilinear-time Lagrange-based BLS .....	70
8.2.1. Numerators and denominators of Lagrange Coefficient .....	70
8.2.2. Calculating all denominator .....	71
8.2.3. Evaluate some polynomial at $d$ point .....	72
8.2.4. Evaluate $N'(x)$ fast .....	74
8.2.5. Evaluate $N'(x)$ using a Polynomial Multipoint Evaluation .....	74
8.3. Further Speed-ups via Roots of Unity .....	76
8.4. Implementation .....	77
<b>9. Conclusion .....</b>	<b>81</b>
<b>10. Future Work in Threshold Cryptosystems .....</b>	<b>83</b>
10.1. Vector Commitments (VCs) .....	83
10.2. Scaling VSS and DKG .....	84
10.2.1. KZG Evaluation proof .....	84
10.2.2. Scaling VSS and DKG in the Asynchronous Setting .....	84

<b>Bibliography.....</b>	<b>86</b>
--------------------------	-----------

# List of Figures

Figure 2.5:	Polynomial multipoint evaluations	15
Figure 4.1:	Calculate $a_1, a_2, \dots, a_8$ from an $\alpha$ using $e_1, e_2, \dots, e_8$	36
Figure 6.1:	BLS Signature using Asymmetric Pairings	52
Figure 6.2:	BLS threshold signature aggregation code implementation	58
Figure 8.1:	Calculating vanishing polynomials	74
Figure 8.2:	Calculating multipoint evaluation	75
Figure 8.3:	Using a quasilinear-time Lagrange algorithm	77
Figure 8.4:	Aggregate d-over-n BLS TSS of each aggregation phase, (1) Interpolation, (2) multi exponentiation	79