MAT009: Lattice-Based Cryptography

Instructor: Simran Tinani (simran.tinani@math.uzh.ch; Y27H06)

February 15, 2023

1 Introduction

Welcome to MAT009: Lattice-Based Cryptography! This is a seminar course, so each week, one student will prepare a 90-minute talk on a pre-selected topic from the list provided in this document.

As some of you may already know, lattice-based systems comprise the major finalists in the NIST competition for post-quantum cryptography, so the subject of this seminar is both mathematically exciting and practically relevant. In this course, we will explore the structures, methods, and algorithms involved in building lattice-based cryptosystems, and devising cryptanalytic attacks on them. In general, lattice-based systems comprise a vast, deep and rather advanced topic in cryptography. The purpose of this course is by no means to attain expertise in this field, but to obtain a fundamental understanding of the mathematical theory and techniques and the algorithmic aspects of lattice-based cryptography. In order to cover all the main topics of interest, we will avoid overtly detailed and technical definitions and proofs, and stick to the most important results that are feasible to explain and understand, keeping in mind our time limit.

This document contains a comprehensive list of week-wise topics to be covered in this seminar. In the beginning, each student picks their topic and later prepares a 90-minute talk for that week. Since the course progresses from basics to more advanced concepts (so it is very important for us to go in sequence), please do not try to prepare your talk too much in advance, since you would then be missing out on the prerequisite knowledge.

The subtopics mentioned under each week's topics should serve as a baseline for the structure your talk, and as the minimum and necessary concepts you are expected to cover. Feel free to add more material or more detail when preparing your presentation. With each subtopic, I also provide reading resources for the respective concepts as hyperlinks, for you to quickly reference during your reading/preparation. In addition, each topic also contains a general set of reading sources, where you will find information on all the subtopics mentioned. A set of general references for the course is also provided at the beginning of this document.

1.1 Logistics of the course

1. **Attending all the seminars is mandatory** for the completion of the course as per the university's rules. If you have unavoidable reasons to miss one or more of the sessions, please inform me by email (if applicable, provide a doctor's certificate).

2. Please prepare your talks to last 90 minutes, and make sure you cover the subtopics mentioned under each topic, in this document.

3. You are also required to **hand in a written report** of the work you present, on the same day as you give your talk. This report should be typed in LaTeX and should contain all the contents of the talk (certainly all the subtopics listed below), with gaps filled in and more attention to detail (for instance, if you presented the sketch of a proof or idea of an algorithm, you may write the whole proof/algorithm in the report). The report should be written in your own words, and should be a reflection of your understanding of the topic. Please do not copy from the reading material or elsewhere!

4. As the course material is both vast and deep, it is natural that you at some places will feel the need for extra help with planning your presentation and understanding the topic. Please feel

free to contact me any time by email, or to drop by at my office. In the first week, we can also fix certain office hours where you would be able to come ask your questions.

5. As mentioned, you are not expected to attain a mastery of the mentioned topics, and it is understandable if you struggle with certain parts. You will always have the option to take my help, and your presentations do not have to be perfect. Some parts of the reading material may seem intimidating at first, but do not worry, the course is structured so that each topic will be easy to understand in its respective week.

2 Reading Material

1. Lecture notes: Lattices in Computer Science (Oded Regev, NYU)
2. BIU Winter School: Lattice-Based Cryptography and Applications
3. An Introduction to Mathematical Cryptography by Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman
4. Course Textbook. Complexity of Lattice Problems: a Cryptographic Perspective
5. Lecture notes: Lattices algorithms and applications (Daniele Micciancio)
6. Lecture notes: Lattices in Cryptography and Cryptanalysis (Daniele Micciancio)
7. Lecture notes: Lattices algorithms and applications (Daniele Micciancio)
8. Video lecture on lattice-based cryptography

3 Course Structure

3.1 Lattices: Structure and Properties (1 week)

See textbook as a general reference. All the above lecture notes contain this material.

1. Definition of a lattice and equivalence to being a discrete subgroup of R^n. Bases, and description through matrices, and the geometric interpretation. Equivalence of bases, good vs bad bases.
2. Gram-Schmidt orthogonalization and the relation to determinant of a lattice.
3. Fundamental parallelepiped, its volume and geometric interpretation. Proof that it forms a tiling of the entire space R^n, Proof of the theorem: a set of vectors is a basis if and only if the lattice spanned by it intersects the fundamental parallelepiped at 0. See here for a geometric explanation
4. Geometry of numbers and Minkowski's theorem (give at least two proofs)
   (a) Prove that the limit of ratio of lattice points contained in the cube of size $r/2$ is $1/det(L)$ (See claim 1.3 here for proof). Subsequently, Proof of Minkowski using this result.
   (b) Blichfeldt Theorem and Minkowski Theorem
   (c) A standard, comprehensive proof of Minkowski's theorem can be found in Hoffstein's book page 412
   (d) State Minkowski's second theorem (see book)
5. Describe successive minima, and the relationship with the Gram-Schmidt basis. See textbook as a general reference

3.2 Algorithms and Complexity (1 week)

General references: Section 2.1 (Complexity Theory) of textbook, A Gentle Introduction to Algorithm Complexity Analysis.

1. Explain big O notation and time complexity [Slides]
2. Define search vs decision problems; Concepts of P, NP, NP-hard and NP-complete. Resources:
   (a) Introduction, reductions, examples
   (b) Formal definitions with Turing machines
   (c) Comprehensive notes with examples
   (d) More intuitive explanations
3. Define the problems: SVP, CVP (exact and approximate versions), Gap-SVP, SIVP, BDD, SIS, LWE, provide a brief summary of known results on reductions and complexities. [Slides 1, Slides 2].
4. Describe the membership and equivalence problems (easy lattice problems) and provide polynomial time solutions.
5. Solve SVP in dimension 2 using Gauss's algorithm, and prove it is polynomial time. [See Chapter 2, Section 1.2, 1.3 of the textbook].
6. Prove that given an oracle for approx-CVP. we can solve approx-SVP [See section Complexity landscape here]
7. Prove that decisional CVP is NP-complete [See these notes on CVP]

3.3 Cryptography (1 week)

See Hoffstein's book as a general reference

1. Introduce cryptography, distinguish between public and private-key cryptography
2. Describe in simple terms the RSA and Diffie-Hellman protocols
3. Describe one-way functions and "Provable security" with examples (example: discrete log, factoring, modular squaring) (formal definitions not needed)
4. Define pseudorandom functions with examples (formal definitions not needed)
5. Describe hash functions, collision resistance, and digital signatures [Cryptographic functions from worst-case complexity assumptions]
6. Describe the concepts of Average case hardness and worst case hardness, how lattice cryptography is advantageous compared to other forms
7. Describe the Merkle-Hellman subset sum-based cryptosystem (without attacks) [Notes 1, Notes 2]
8. Discuss (without details) Shor's quantum algorithm and the need for post-quantum cryptography, distinguish between classical and post-quantum one-wayness/security. List the well-known post-quantum alternatives for public key cryptography. [Introduction to post-quantum cryptography]
9. Optional topic: PRG from subset sum problem [Private key cryptosystems based on subset-sum]

3.4 The LLL algorithm (2 weeks)

3.4.1 Resources

1. Chapter 2 of the textbook
2. Lecture notes 1
3. Lecture notes 2
4. Lecture notes 3
5. Lecture notes 4 (rather concise explanation, with proofs)

3.4.2 Week 1: Understanding LLL

1. Describe the LLL algorithm and its working, and give a complexity analysis
2. Optional: Discuss (without details) some improvizations of LLL
3. Describe Babai's nearest plane algorithm and its working

3.4.3 Week 2: Applications of LLL

1. In number theory: example, writing a prime $p \equiv 1 \mod 4$ as a square of primes, factoring of polynomials [Notes]
2. As a cryptanalysis tool
   (a) Describe the application of LLL to breaking knapsack cryposystems [Notes 1, Notes 2]
   (b) Give a brief overview of the Coppersmith algorithm [Notes on Coppersmith algorithm]
   (c) Describe an algorithm for finding small solutions to polynomial equations [Notes 1, Notes 2]
   (d) Describe the Attack on RSA with Low Public Exponent

3.5 Results on CVP and SVP (1 week)

See chapters 3 and 4 of the book as a general reference.

1. Describe CVP, SVP, known hardness results, reductions
2. NP-hardness of CVP [See Lecture 10 on this link]
3. NP-hardness of CVP can be easily established by reduction from subset sum, and even approximating CVP within any constant or "almost polynomial" factors is hard for NP [Chapter on CVP and its hardness in the textbook]
4. Describe briefly the exponential algorithm for SVP (Ajtai-Kumar-Sivakumar) [Algorithm for SVP]
5. Describe briefly the Fastest Exponential time algorithm for the shortest vector problem
6. Describe some of the following CVP to SVP and further reductions (optimal, approximate) [Notes]
7. Describe (You may omit details that seem out of scope) Proof of NP-hardness of SVP in l_2-norm

3.6 Some cryptosystems based on hard lattice problems (2 weeks)

See chapter 8 of the textbook and these slides as a general reference.

3.6.1 Week 1: Cryptosystems

1. General introduction: constructing cryptosystems from lattice problems [Notes on lattice trapdoor functions and algorithms]
2. Describe the mathematics of convolutional polynomial rings and the construction of NTRU encrypt, describe decryption and its correctness [Hoffstein book sections 7.9 and 7.10, Lecture Notes, More lecture notes, See chapter 3 of this thesis]
3. Describe how NTRU encrypt is realized as a lattice cryptosystem [Hoffstein sec 7.11]
4. Describe the GGH and NTRU signature schemes [GGH, NTRUSign]
5. Discuss the efficiency and implementation feasibility (time and space costs), parameters, comparison to other systems chapter 3 of this thesis

3.6.2 Week 2: Attacks

1. Security discussion: brute force attacks, meet-in the middle attacks Chapter 4 in this thesis.
2. Describe the NTRU lattice and lattice-based attacks Again see chapter 4 here
3. Explain the application of LLL to attacking NTRU
4. Describe further Lattice attacks on NTRU
5. Discuss the following attack: reduction to Hidden Parallelepiped Problem
6. Optional: go over the following code of the implementation of NTRU and some attacks and/or implement your own version

3.7 Small Integer Solution (SIS) Problem (2 weeks)

3.7.1 Resources

1. Notes
2. Slides
3. Slides 2
4. Slides 3

3.7.2 Week 1

1. Define the SIS problem and realize it as a lattice problem. Explain the importance of the length constraint (Gaussian elimination)
2. Talk about worst case-vs-avg case difficulty in the context of SIS and describe a reduction to SIVP. Explain how to use a SIS oracle to find a short vector in any lattice
3. Describe the hash function defined by SIS and its properties: collision-resistance, compression, regularity, homomorphism
4. Define universal hash function and give a proof of universality of SIS [Notes]
5. State the leftover hash lemma, and describe its application to SIS [see chapter 3 of these lecture notes]
6. Briefly outline some ideas beind the real-world SWIFFT scheme, which constructs a collision-resistant hash function based on SIS [SWIFFT]

3.7.3 Week 2

1. Define a cryptographic commitment scheme and important properties like perfect/statistical hiding and computational binding (You do not have to go into too technical or overly formal definitions, the idea is to explain the concept precisely but in an introductory fashion). Explain the Elgamal and Pedersen commitments [Article, Notes, Slides]
2. Describe a commitment scheme from SIS [see chapter 3 in lecture notes, also these slides]
3. Define homogeneous vs inhomogeneous SIS. Review the definition of a digital signature and explain digital signatures from SIS (show key generation, signing and verification)[ISIS and digital signatures, also these slides]
4. Give a brief introduction to the basic concepts of zero-knowledge proofs with $1-2$ examples [Section 8.3 of Hoffstein, and this blog post, and these lecture notes]
5. Optional: Outline (without technical jargon/proofs) of key ideas of a lattice-based zero-knowledge proof system: See the protocol in figure 4 and Lemma 3.1 of this paper. [For a more general and detailed reference, see this thesis].

3.8 Learning with Errors (LWE) Problem (1 week)

3.8.1 Resources

1. Easy explanation of LWE, secret key and public key encryption, proofs of security
2. Reductions: from SIS to LWE, search to decisional LWE (algorithm, without proof)
3. Survey on LWE
4. Slides
5. Slides 2
6. Example of implementation

3.8.2 Topics

1. Definition of LWE and relationship to lattice problems, search vs decision versions.
2. Properties: Injectivity, Pseudorandomness, Homomorphism [See in particular these slides]
3. SIS vs LWE comparison [Slides]
4. Reductions: from SIS to LWE and from Search to decision version
5. Describe the construction of encryption schemes from LWE: Simple PKE, Regev Encryption, public-Key Cryptosystem and dual system, ost efficient system Slides
6. Describe the Learning With Rounding (LWR) problem and some basic properties and the construction of pseudorandom generator/ functions from LWE/LWR [Slides ,Notes]
7. Optional: go through the LWE implementation in the link above (resource 6).

3.9 Ideal lattices and applications (1 week)

3.9.1 Resources

1. Slides
2. Slides 2
3. Slides 3

3.9.2 Topics

1. Review the mathematics of convolutional polynomial rings (Define number fields, rings of integers, cyclotomic fields, canonical vs polynomial embeddings).
2. Define cyclic lattices and $(x^n + 1)$-ideal lattices
3. State the Ideal-SVP. Prove that a $(x^n + 1)$-ideal lattice cannot have a unique shortest vector
4. State the status of hardness of problems for general and $(x^n + 1)$-Ideal Lattices
5. Prove that GapSVP$_{\sqrt{n}}$ is easy in $(x^n + 1)$-ideal lattices
6. State the Ring-SIS ad Ring-LWE problems. State the relationship/reduction between Ring-SIS and Ring-LWE and Ideal-SVP [See these slides]
7. Describe the computational advantages of using structured/ideal lattices, and potential security disadvantages
8. Define module lattices and module-SIS

3.10 The NIST finalists (1-2 weeks)

There is no fixed set of topics/results to be covered. The task in this week is to best desribe the four NIST finalists for encryption and digital signatures with the concepts learned in the seminar course. Try to stay as mathematically precise as possible, without going too much into the technical details and proofs. Some important aspects to talk about are

1. Which lattice problem is the system based on and what is the underlying structure/lattice? How does the encryption/decryption or signing/verification function?
2. What are the key mathematical ideas/methods used in the construction?
3. What other features (apart from the underlying one-way function) have been used?
4. What are the key sizes and running times and how do these compare with classical systems (e.g. RSA signatures, Elgamal encryption)? What methods have been employed to optimize these?
5. Talk more about the security (if applicable, talk about whether it is CPA or CCA. You may briefly define these terms in your talk.) of the system.
6. Without going into details, what methods of attacks have been attempted on (similar) systems and why are they ineffective for these?
7. Without going into details, why is this proposal more secure/efficient than other lattice-based proposals? Has it undergone modifications and improvisations over time?

3.10.1 Resources

NIST Announces First Four Quantum-Resistant Cryptographic Algorithms

1. CRYSTALS-Kyber (PKE) Overview, Paper
2. CRYSTALS-Dilithium (Digital Signatures) Overview, Paper
3. Falcon Overview, Paper
4. SPHINCS+ Overview, Paper Blog post

3.11 Optional: Integer Optimization and Lattices (1-2 weeks)

You must pick and present all material in at least two of the chapters in Integer Optimization and Lattices