Çetin Kaya Koç *Editor*

Open Problems in Mathematics and Computational Science

Springer

Çetin Kaya Koç
Editor

Open Problems in Mathematics and Computational Science

Springer

Editor
Çetin Kaya Koç
Department of Computer Science
University of California, Santa Barbara
Santa Barbara, CA
USA

ISBN 978-3-319-10682-3 ISBN 978-3-319-10683-0 (eBook)
DOI 10.1007/978-3-319-10683-0
Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2014957413

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

A selected group of invited speakers and more than 150 students and researchers attended a special conference on September 18–20, 2013, in "Said Halim Pasha Palace" in Istanbul. There had never been a conference of this kind in Turkey, where "open" or "unsolved" problems are discussed, and even in the world there have only been a few examples.

In principle, mathematicians, scientists, and engineers attend conferences to speak about problems they have solved and to "impress" and inform the academic community about their methods and the final solution. It is not generally expected that a researcher would take the stand in a conference to talk about a problem she or he could not (yet) solve. However, all scientific processes start with hypotheses whose ramifications we do not know or problems whose solutions are not clear yet. Either for personal reasons or in accordance with the expectations of scientific conferences and their attendees, researchers tend to push the open/unsolved problems to the back burner and talk about what they have solved, understood, or proved. Still, once in a while (perhaps every 5–10 years), some researchers come together to discuss problems they have not solved yet or problems whose solutions seem rather challenging. Since the 1970s, there have been 7 such conferences.

Therefore, I am very happy that we were able to organize this *Open Problems in Mathematical and Computational Sciences Conference* with support from the Scientific and Technological Research Council (TÜBİTAK) of Turkey.

A large number of young researchers, MSc, and PhD candidates from Turkey, as well as several from neighboring countries, attended the conference. The invited scientists of the conference are among the most prolific mathematical and computational scientists in the world. They come from various countries, demonstrating that science and engineering are culturally very diverse now. The list of countries and number of scientists from each country were a good reminder of this fact: Belgium (2), Brazil (1), Canada (2), China (2), France (3), Germany (2), Japan (1), Norway (1), Romania (1), Turkey (3), and the USA (2).

The Open Problems Conference was held in Said Halim Pasha Palace, one of the most beautiful seaside palaces in Istanbul, whose history goes back at least 150 years and as far as Egypt!

Said Halim Pasha was the son of Mehmet Abdülhalim Pasha who was one of the four sons of Mehmet Ali Pasha from Kavala, the second largest city in Northern Greece. Mehmet Ali Pasha (Muhammad Ali of Egypt) was an Ottoman commander of Albanian origin and is regarded as the founder of modern Egypt because of the dramatic reforms in the military, economic, and cultural spheres he instituted. Said Halim Pasha was born in Cairo in the year 1863 and completed his education in private lessons in Cairo, where he learned Arabic, Persian, English, and French. He studied politics for 5 years in Switzerland. The palace had become the property of Prince Abdülhalim Pasha in the year 1876 and was reconstructed to its current appearance by the travelling architect, Petraki Adamandidis of the Dardanelles. The property was inherited by the nine children of the Abdülhalim Pasha after his death in 1890. After going through several owners, the Said Halim Pasha Palace was restored following a fire in 1995 under the name "Prime Ministry Official Guest House."

Several peoples' names need to be mentioned with gratitude, they made both the Open Problems Conference and the Open Problems Book possible.

First of all, I sincerely thank Ronan Nugent for his valuable advice and the Editorial Office of Springer for their help in getting the book published.

On behalf of the invited speakers, I am also sincerely grateful to TÜBİTAK for agreeing with us about the vision of the Open Problems Conference and their subsequent work that produced this book and for providing the financial support. I would also like to thank to Şükran Külekci, İsa Sertkaya, Birnur Ocaklı, Mehmet Sabır Kiraz, and Osmanbey Uzunkol for working around the clock several days before, during, and after the conference.

Santa Barbara, CA, USA Çetin Kaya Koç

Contents

About Open Problems

Çetin Kaya Koç

Abstract A small group of computer scientists and mathematicians from industry and academia convened in a historical home ("Said Halim Pasha Palace") overlooking the Bosphorus Straits to discuss several difficult problems they and others in similar fields are tackling. The motivation of the *Open Problems in Mathematical and Computational Sciences Conference* was to enable and encourage the academic community, particularly young researchers and Ph.D. candidates, to hear about unsolved, open problems in mathematical and computation sciences, directly from the scientists who are rigorously investigating them.

1 The Conference

In general, scientists go to conferences to present discoveries that are already made, to explain results or to expose and excite the community about connections within various theories or structures, and to share their insights and proofs. Conferences are places where we get to see and hear about solutions, ask questions about them, and hope to understand them better in this process. Rarely is there an opportunity to talk about problems that have not been solved yet or solutions which are not yet satisfactory, except during the lunches, coffee breaks, or at other quiet times.

In many instances, scientists working on problems whose solutions are difficult to obtain will state that asking the right question is the real challenge. It is imperative to stop and think once in a while in order to understand the background of the tools and the mechanisms needed for tackling the problems we are working on. Conferences that deal with open problems are rare, but they are useful avenues for such objectives. Almost all conferences are for presenting the solutions to certain classes of problems whose origins we may not have any idea about.

Ç.K. Koç (✉)
University of California Santa Barbara, Santa Barbara, CA 93106, USA

Mathematical and Computational Sciences Labs, TÜBİTAK BİLGEM, Gebze, Kocaeli, Turkey
e-mail: koc@cs.ucsb.edu

Ç.K. Koç (ed.), *Open Problems in Mathematics and Computational Science*,
DOI 10.1007/978-3-319-10683-0_1

In a world replete with information, what matters most is sometimes not the answers but rather the context, the origin and the body of questions for which answers are sought or obtained.

This conference was planned with these ideas in mind. One purpose of the *Open Problems in Mathematical and Computational Sciences Conference* is to encourage, motivate, and excite the mathematical and computational sciences community to discuss open problems. We would like to hear them formulate the questions and present processes which will be helpful in the quest for answers.

Of course, we all know about certain open problems or conjectures in mathematics such as the Goldbach conjecture or the twin primes conjecture or the Riemann hypothesis. Some well-known problems have been resolved during the last 20 years, three excellent examples being Fermat's last theorem by Andrew Weil in 1995, the Poincaré conjecture by Grigori Perelman in 2003, and the prime gap problem by Yitang Zhang (and later by the Polymath Project participants) in 2013. The list of difficult problems in mathematics is pretty long, and solutions come in decades or even centuries. And when they come, they are deservedly celebrated, and the international media and thus the public pay attention; stories are made and impressions are created. Furthermore, mathematics institutes around the world, for example, the Clay Institute, publish problem lists and offer prizes which further publicize the phenomena.

However, we are limiting our attention to computational problems in this conference; there is also a long list of unsolved problems in computer science, such as:

- $P = NP$ problem
- Existence of one-way functions
- Is the graph isomorphism problem in P?
- Is factoring in P?
- Is primality testing in P?
- What is the fastest algorithm for the multiplication of integers?
- What is the fastest algorithm for matrix multiplication?

The list is not complete, and our intention is not to complete the list, but to bring the best minds to describe, elucidate, and explain some of these open problems in the mathematical and computational sciences, particularly the problems they themselves are interested in or working on or for which they have formulated partial or near-complete solutions. We want them to tell us how they approach such problems and what are the mechanisms and tools they are using and share with us and excite us with the creative energy they are applying to such problems.

A perfect example from the above list was the question "Is Primality Testing in P?" This was affirmatively answered by Manindra Agrawal, Neeraj Kayal, and Nitin Saxena of the Indian Institute of Technology Kanpur, by giving the first deterministic polynomial time algorithm for primality testing. The implications of this development are indeed great for cryptography, coding, and finite fields, where primality plays a central role.

To summarize, one of the underlying purposes of our 3-day conference was to encourage young researchers, particularly Ph.D. candidates, to learn about exciting, interesting, and important (yet) unsolved problems in mathematical and computation sciences, directly from the researchers who are thinking about them. I believe the informal atmosphere of the conference allowed them to listen to the seminars, ask questions, interact, and discuss possible answers or pose new questions to the invited speakers.

We believe such a close interactive environment served as a catalyzing event and hopefully will synchronize local research communities with the best, most challenging, and perhaps most useful problems the world's best minds are working on. Hopefully, in several years, perhaps even as early as the next Open Problems Conference, a few of these challenging problems will find their partial or complete solutions.

2 The Participants

The following people attended the conference as invited speakers:

- Paulo Barreto, Universidade de Sao Paulo
- Claude Carlet, Université Paris 8
- Guanrong Chen, City University of Hong Kong
- Ömer Eğecioğlu, University of California, Santa Barbara
- Gerhard Frey, Göttingen Academy of Sciences
- Tor Helleseth, University of Bergen
- Antoine Joux, Université de Versailles Saint-Quentin-en-Yvelines
- Andrew Klapper, University of Kentucky
- Alfred Menezes, University of Waterloo
- David Naccache, Université Paris II
- Koji Nakano, Hiroshima University
- Ferruh Özbudak, Middle East Technical University
- Daniel Panario, Carleton University
- Bart Preneel, KU Leuven
- Gheorghe Pãun, Romanian Academy
- Jean-Jacques Quisquater, Université catholique de Louvain
- Henning Stichtenoth, Sabancı University
- Murat Tekalp, Koç University
- Han Vinck, University of Duisburg-Essen

We thank our speakers for taking time to come to Istanbul to talk about problems that excite them and to share them with us. There were more than 150 participants, most of whom were from Turkey, as expected; however, about 10 % of the participants were from other European countries, including Bulgaria, Denmark, France, and Romania.

3 The Book

As we were planning the conference, we also developed a plan to publish a book arising from the presentations.

This book contains *selected and revised* papers from the conference. We gave a window of about 6 months to the speakers to create the chapters in this book, revising and expanding their work by adding an introduction section and an annotated bibliography. The introduction section of each chapter is intended to provide the background of the topic of the chapter, assuming the reader is a first-year graduate student who has the general knowledge of electrical engineering, computer science, programming, and computational mathematics via his/her undergraduate education and has just started reading books and papers in the area of the chapter. Therefore, the chapters attempt to give all basic definitions, introduce the context, and summarize algorithms, theorems, and proofs. On the other hand, the bibliography aims to introduce the most important references to follow up, giving a short description of these papers and books, and their importance to the field. I hope you will find these chapters to your liking.

The Past, Evolving Present, and Future of the Discrete Logarithm

Antoine Joux, Andrew Odlyzko, and Cécile Pierrot

Abstract The first practical public key cryptosystem ever published, the Diffie–Hellman key exchange algorithm, relies for its security on the assumption that discrete logarithms are hard to compute. This intractability hypothesis is also the foundation for the security of a large variety of other public key systems and protocols.

Since the introduction of the Diffie–Hellman key exchange more than three decades ago, there have been substantial algorithmic advances in the computation of discrete logarithms. However, in general the discrete logarithm problem is still considered to be hard. In particular, this is the case for the multiplicative groups of finite fields with medium to large characteristic and for the additive group of a general elliptic curve.

This chapter presents a survey of the state of the art concerning discrete logarithms and their computation.

1 Introduction

1.1 The Discrete Logarithm Problem

Many popular public key cryptosystems are based on discrete exponentiation. If G is a multiplicative group, such as the group of invertible elements in a finite

A. Joux
CryptoExperts, Paris, France

Chaire de Cryptologie de la Fondation de l'UPMC, Paris, France

Sorbonne Universités, LIP6, UMR 7606, UPMC Univ Paris 06, Paris, France
e-mail: Antoine.Joux@m4x.org

A. Odlyzko
School of Mathematics, University of Minnesota, Minneapolis, MN 55455, USA
e-mail: odlyzko@umn.edu

C. Pierrot (✉)
DGA/CNRS, Sorbonne Universités, LIP6, UMR 7606, UPMC Univ Paris 06, Paris, France
e-mail: Cecile.Pierrot@lip6.fr

Ç.K. Koç (ed.), *Open Problems in Mathematics and Computational Science*,
DOI 10.1007/978-3-319-10683-0_2

field or the group of points on an elliptic curve, and g is an element of G, then g^x is the discrete exponentiation of base g to the power x. This operation shares basic properties with ordinary exponentiation, for example, $g^{x+y} = g^x \cdot g^y$. The inverse operation is, given h in G, to determine a value of x, if it exists, such that $h = g^x$. Such a number x is called a discrete logarithm of h to the base g, since it shares many properties with the ordinary logarithm. If, in addition, we require some normalization of x to limit the possible answers to single valid value, we can then speak of *the* discrete logarithm of h. Indeed, without such a normalization, x is not unique and is only determined modulo the order of the element g.

Assume for simplicity that G is a cyclic group generated by g and that the notation $\log_g(h)$ denotes a value such that $h = g^{\log_g(h)}$. Then, as with ordinary logarithms, there is a link between multiplication of elements and addition of logarithms. More precisely, we have:

$$\log_g(h \cdot j) \equiv \log_g(h) + \log_g(j) \mod |G|.$$

We say that we solve the discrete logarithm problem (DLP) in G if given any element g^x in G, we are able to recover x. To normalize the result, we usually ask for x to be taken in the range $0 \leqslant x < |G|$. In many applications, in particular in cryptography, it is sufficient to be able to solve this problem in a substantial fraction of cases. (The usual theoretical standard is that this fraction should be at least the inverse of a polynomial in the logarithm of the size of the group.)

The main interest of discrete logarithm for cryptography is that, in general, this problem is considered to be hard. The aim of this chapter is to provide state-of-the-art information about the DLP in groups that are used for cryptographic purposes. It gives pointers to the latest results and presents observations about the current status and likely future of the DLP.

1.2 Applications of Discrete Logarithms

In some sense, the discrete logarithm has a long history in number theory. It is just an explicit way to state that an arbitrary cyclic group containing N elements is isomorphic to $(\mathbb{Z}_N, +)$. Still, before the invention of the Diffie–Hellman protocol, the problem of efficiently computing discrete logarithms attracted little attention. Perhaps the most common application was in the form of Zech's logarithm, as a way to precompute tables allowing faster execution of arithmetic in small finite fields.

The role of the DLP in cryptography predates Diffie–Hellman. Indeed, the security of secret-key cryptosystem involving linear feedback shift registers (LFSR) is closely related to the computation of discrete logarithms in finite fields of characteristic two. More precisely, locating the position where a given subsequence appears in the output of an LFSR is, in fact, a DLP in the finite field defined by the feedback polynomial.[1]

The main impetus to intensive study of discrete logarithms came from the invention of the Diffie–Hellman method in 1976 [DH76]. Much later, the introduction of pairing in cryptography in 2000 (journal versions [Jou04, BF03]) increased the level of attention on some atypical finite fields, with composite extension degrees and/or medium-sized characteristic.

1.2.1 Diffie–Hellman Key Exchange

Let us recall the first practical public key technique to be published, which is still widely used, the Diffie–Hellman key exchange algorithm. The basic approach is as follows. If Alice and Bob wish to create a common secret key, they first agree, on a cyclic group G and a generator g of this group.[2] Then, Alice chooses a random integer a, computes g^a, and sends it to Bob over a public channel, while Bob chooses a random integer b and sends g^b to Alice. Now Alice and Bob can both compute a common value, which then serves as their shared secret:

$$(g^b)^a = g^{a \cdot b} = (g^a)^b.$$

The security of this system depends on the assumption that an eavesdropper who overhears the exchange, and thus knows g, g^a, and g^b, will not be able to compute the shared secret. In particular, this hypothesis assumes that the eavesdropper is unable to solve the DLP in G. Indeed, if the DLP for this group is solvable, he can compute either a or b and recover the shared secret $g^{a \cdot b}$. However, it is not known whether the problem of computing g^{ab} given g, g^a, and g^b, which is known as the computational Diffie–Hellman problem (CDH), is equivalent to the computation of discrete logarithms. Moreover, to prove the security of many cryptographic protocols, it is often necessary to consider the associated decision problem: given g, g^a, g^b, and h, decide whether h is the correct value of g^{ab} or not. This latest problem is called the decision Diffie–Hellman problem (DDH).

There are also many generalized computational and decision problems somehow related to the DLP that have been introduced as possible foundations for various cryptosystems. Since it is not easy to compare all these assumptions, in an attempt to simplify the situation, Boneh et al. [BBG05] have proposed the *uber-assumption* which subsumes all these variations and can be proven secure in the generic group model (see Sect. 2.5).

However, the DLP itself remains fundamental. Indeed from a mathematical viewpoint, it is a much more natural question than the other related problems, and

[1] Assuming that it is irreducible, which is usually the case.

[2] The group G and generator g can be the same for many users and can be part of a public standard. However, that can lead to a reduction in security of the system.

in practice, none of these other problems has ever been broken independently of the DLP. Since the introduction of the Diffie–Hellman key exchange, this concern has motivated a constant flow of research on the computation of discrete logarithms.

Another extremely important assumption in the above description is that the eavesdropper is passive and only listens to the traffic between Alice and Bob. If the attacker becomes active, then the security may be totally lost, for example, if he can mount a *man-in-the-middle* attack where he impersonates Bob when speaking to Alice and conversely. This allows him to listen to the decrypted traffic. To avoid detection, the attacker forwards all messages to their intended recipient after reencrypting with the key that this recipient has shared with him during the initial phase. One essential issue when devising cryptosystems based on discrete logarithms is to include safety measures preventing such active attacks.

1.2.2 Other Protocols

After the invention of the RSA cryptosystems, it was discovered by El Gamal [Gam85] that the DLP can be used not only for the Diffie–Hellman key exchange, but also for encryption and signature. Later Schnorr [Sch89] gave an identification protocol based on a zero-knowledge proof of knowledge of a discrete logarithm, which can be turned into Schnorr's signature scheme using the Fiat–Shamir transform [FS86].

There are many more cryptosystems based on the DLP which will not be covered here. However, let us mention the Paillier encryption [Pai99]. This system works in the group $\mathbb{Z}_{N^2}^*$, where $N = pq$ is an RSA number of unknown factorization. In particular, this is an example of a discrete logarithm-based cryptosystem that works within a group of unknown order. This system possesses an interesting property, in that it is additively homomorphic; the product of the Paillier encryption of two messages is an encryption of their sum.

Another very interesting feature of discrete logarithms is the ability to construct key exchange protocols with additional properties, such as authenticated key exchange, which embed the verification of the other party identity within the key exchange protocol. Perfect forward secrecy, in which disclosure of long-term secrets does not allow for decryption of earlier exchanges, is also easy to provide with schemes based on discrete logarithms. For example, in the Diffie–Hellman key exchange, Alice's secret a and Bob's secret b are ephemeral, and so is the shared secret they used to create, and (if proper key management is used) are discarded after the interaction is completed. Thus, an intruder who manages to penetrate either Alice's or Bob's computer would still be unable to obtain those keys and decrypt their earlier communications. It is also possible to mix long-term secrets, i.e., private keys, and ephemeral secrets in order to simultaneously provide perfect forward secrecy and identity verification.

1.2.3 A Powerful Extension: Pairing-Based Cryptography

Besides the Diffie–Hellman key exchange, a natural question to ask is whether there exists a three-party one-round key agreement protocol that is secure against eavesdroppers. This question remained open until 2000 when Joux [Jou04] devised a simple protocol that settles this question using bilinear pairings. Until then, building a common key between more than two users required two rounds of interaction. A typical solution for an arbitrary number of users is the Burmester–Desmedt protocol [BD94].

The one-round protocol based on pairing works as follows. If Alice, Bob, and Charlie wish to create a common secret key, they first agree on $G_1 = \langle P \rangle$ an additive group with identity $\mathcal{O}$, a multiplicative group G_2 of the same order with identity 1, and a bilinear pairing from G_1 to G_2. Let us recall the definition

Definition 1.1 A symmetric bilinear pairing[3] on (G_1, G_2) is a map

$$e : G_1 \times G_1 \to G_2$$

satisfying the following conditions:

1. e is bilinear: $\forall\, R, S, T \in G_1,\ e(R + S, T) = e(R, T) \cdot e(S, T)$,
   and $e(R, S + T) = e(R, S) \cdot e(R, T)$.
2. e is non-degenerate: If $\forall\, R \in G_1,\ e(R, S) = 1$, then $S = \mathcal{O}$.

Alice randomly selects a secret integer a modulo the order of G_1 and broadcasts the value aP to the other parties. Similarly and simultaneously, Bob and Charlie select their one secret integer b and c and broadcast bP and cP. Alice (and Bob and Charlie, respectively) can now compute the shared secret key

$$K = e(bP, cP)^a = e(P, P)^{abc}$$

We know that the security of DH-based protocols often relies on the hardness of the CDH and DDH problems. Likewise, the security of pairing-based protocols depends on the problem of computing $e(P, P)^{abc}$ given P, aP, bP, and cP, which is known as the computational bilinear Diffie–Hellman problem (CBDH or simply BDH). This problem also exists in its decision form (DBDH). However, little is known about the exact intractability of the BDH, and the problem is generally assumed to be as hard as the DLP in the easier of the groups G_1 and G_2. Indeed, if the DLP in G_1 can be efficiently solved, then an eavesdropper who wishes to compute K can recover a from aP and then compute $e(bP, cP)^a$. Similarly, if the DLP in G_2 can be efficiently

[3] In general, asymmetric pairings are also considered. For simplicity of presentation, we only describe the symmetric case.

solved, he could recover bc from $e(bP, cP) = e(P, P)^{bc}$, then compute bcP, and finally obtain K as $e(aP, bcP)$.

One consequence of the bilinearity property is that the DLP in G_1 can be efficiently reduced to the DLP in G_2. More precisely, assume that Q is an element of G_1 such that $Q = xP$, then we see that $e(P, Q) = e(P, xP) = e(P, P)^x$. Thus, computing the logarithm of $e(P, Q)$ in G_2 (to the base $e(P, P)$) yields x. This reduction was first described by Menezes et al. [MOV93] to show that supersingular elliptic curves are much weaker than random elliptic curves, since the DLP can be transferred from a supersingular curve to a relatively small finite field using pairings.

After the publication of the Menezes, Okamoto, and Vanstone result, cryptographers started investigating further applications of pairings. The next two important applications were the identity-based encryption scheme of Boneh and Franklin [BF03] and the short signature scheme of Boneh et al. [BLS04]. Since then, there has been a tremendous activity in the design, implementation, and analysis of cryptographic protocols using bilinear pairings on elliptic curves and also on more general abelian varieties, for example, on hyperelliptic curves.

1.3 Advantages of Discrete Logarithms

A large fraction of the protocols that public key cryptography provides, such as digital signatures and key exchange, can be accomplished with RSA and its variants. Pairing-based cryptosystems are a notable exception to this general rule. However, even for classical protocols, using discrete logarithms instead of RSA as the underlying primitive offers some notable benefits.

1.3.1 Technical Advantages

Smaller Key Sizes The main advantage of discrete logarithms comes from the fact that the complexity of solving the elliptic curve discrete logarithm problem (ECDLP) on a general elliptic curve is, as far as we know, much higher than factoring an integer of comparable size. As a direct consequence, elliptic curve cryptosystems currently offer the option of using much smaller key sizes than would be required by RSA or discrete logarithms on finite fields to obtain a comparable security level.

In truth, the key size reduction is so important that it more than offsets the additional complexity level of elliptic curve arithmetic. Thus, for the same overall security level, elliptic curve systems currently outperform more classical systems.

Perfect Forward Secrecy When using RSA to set up a key exchange, the usual approach is for one side to generate a random secret key and send it to the other encrypted with his RSA public key. This grants, to an adversary that records all the