May 31, 2011.

Rafail Ostrovsky – Research Statement

My research interests are in the areas of Cryptography, Distributed Algorithms, and Search Algorithms for Streaming and High-Dimensional Data. My extensive experience in industry prior to joining UCLA has influenced the type of problems that I choose to work on: my research emphasis has always been on problems that are strongly driven by their practical significance and their practical applications. I find these topics exciting to work on, not only due to their practical importance, but also since they are concerned with fundamental issues in computation, including fault-tolerance, randomization, knowledge, and interaction. I intend to continue to work in these fields, improving the efficiency of and our understanding of various protocol problems, as well as proving lower bounds for these problems. I often find underlying ties between the techniques from the three diverse areas of my studies. I consider it to be an important effort to bring together approaches and tools from different fields.

1 Cryptography and Related Areas.

With the rapid technological advances in wireless and wired media and fast affordable hardware, the Internet and other communication networks have become pervasive. As networks become ever larger and more ubiquitous – distributed protocols begin to play a central role in both the scientific and business environments of today. Hand-in-hand with these developments come the questions of *security*, *privacy*, and *fault-tolerance*. For example, how do we ensure that distributed computations and protocols are not compromised, that passwords are not broken, or that we can cope with a virus spread or overloaded network traffic? Luckily, cryptography and distributed algorithms provide some of the answers to these questions. The answers can be evaluated in terms of the assumptions needed, the number of bits of communication required, the number of rounds of interaction, and the running time. The goal, of course, is to minimize the necessary assumptions (or to show that they cannot be further reduced), and to obtain the most efficient solutions. Below, I have highlighted some of the topics that I worked on in these areas.

Public Key Encryption: The notion of probabilistic public-key encryption was put forward in a seminal paper of Goldwasser and Micali in 1982. Before 2008, (see [20]) it was known that a plaintext message to be encrypted should not in any way depend on the secret decryption keys themselves. The danger of encrypting messages that the adversary could not find on his own has already been noted more than two decades earlier by Goldwasser and Micali's original 1982 paper. However, over the past few years, researchers observed that in some situations the plaintext messages do depend on the secret keys. Such situations may arise due to careless key management. For example, a backup system may store the backup encryption key on a disc and then encrypt the entire disc, including the key, and back up the result. Another example is the BitLocker disc encryption utility (used in Windows 7 and Windows Vista), where the disc encryption can end up on the disc itself and be encrypted along with the disc contents. There are also situations where circular security is needed by design, where multiple users are required to publish a cycle of their encrypted private keys so that if one of the users leaks anyone's key, all keys in the cycle become compromised, including his own key. Finally, in the formal methods community, the notion of key-dependent security was used to prove equivalence between computational security and axiomatic security. Despite wide interest, until 2008, the problem of constructing a cryptosystem under standard cryptographic assumptions that can depend on private keys remained unresolved. The first such construction, under a standard cryptographic assumption, was shown in [20]. The techniques proposed there also opened the flood-gates for other applications, where the [20] cryptosystem was later shown to be the first "leakage-resilient" cryptosystem as well. Other highly-cited works of mine on pubic-key

encryption include encryptions with additional properties: e.g., one that allows "selective" keyword search [19] and "deniable" public-key encryption [26].

Zero-Knowledge: Zero-knowledge interactive proofs, introduced by Goldwasser, Micali and Rackoff, involve two parties, a *prover* and a *verifier*, who talk back and forth. The prover tries to convince the probabilistic polynomial time verifier that a given theorem is true (for example, that a CNF formula is satisfiable). A *zero-knowledge proof* is an interactive proof with an additional privacy constraint: informally, the verifier does not learn why the theorem is true. Zero-knowledge proofs are central to the implementation of any protocol problem with maximum privacy, as was shown by Goldreich, Micali and Wigderson. My contributions to our understanding of zero-knowledge include:

- [35, 9]**Near-optimal communication complexity for Zero-Knowledge *proofs***: For an arbitrary bounded fan-in circuit C with s gates, we show (under any one-way function with security parameter k) a zero-knowledge protocol for proving circuit-satisfiability of C with communication complexity $O(s) + poly(k; \log s)$. Thus, for large circuits, the ratio between the communication complexity and the circuit size approaches a constant. This improves over the $O(ks)$ complexity of the best previous protocols. The paper also introduced an important connection between two-party protocols and multi-party secure protocols "in the head". (The idea, very informally, is that a single player executes "in his head" a multi-party protocol and then sends messages to the other player that reveal partial view of his "in the head" simulation. This allows the encoding of messages between two players using error-correcting codes that come from multi-party computation literature.) The major impact of this novel proof method allowed the porting of techniques developed for secure multi-party computation into the two-party setting. This led to multiple follow-up works using this paradigm.
- [34] The first **Statistical** Non-Interactive Zero-Knowledge (NIZK) for all of NP. The question of Statistical NIZK, where the witness must remain hidden information-theoretically, remained open since the inception of NIZK by Blum, Feldman and Micali in 1988. In addition, the paper developed *dual encryption/lossy-encryption* novel proof method that led to multiple additional papers. In [33] we showed the existence of the first non-interactive ZAP without the need of common reference string.
- [43, 50] The equivalence between zero-knowledge (for any non-trivial language outside BPP) and one-way functions.
- [2, 3] Introduction of an **interactive hashing** technique that later led to Statistical Zero-Knowledge arguments for NP based on one-way permutations [49, 42, 14]. Interactive hashing was later used as a key stepping-stone to construct Statistical Zero-Knowledge arguments for NP based on general one-way functions.

Secure Multi-Party Computation: Two-Party and Multi-Party secure Computation (MPC) was introduced by Yao, and by Goldwasser, Micali and Wigderson in the late 1980s. The goal is to distributively compute an output without revealing players' individual inputs. My contributions include:

- [36] a method for performing two-party computation with constant computational and communication overhead, i.e. only constant times larger then the size of the circuit. The paper also constructed (without any unproven assumptions) the first linear-size circuit for pair-wise independent hashing, disproving the STOC 1990 conjecture of Mansour, Nisan, and Tiwari.

- [27] The first Universally-Composable MPC, where multiple invocations can be executed in an arbitrary interleaved manner.
- [37] Matching upper and lower bounds on round complexity of 2-party secure computation.
- [31] We defined security notion and provided protocols for secure MPC on sparse networks. We further improved the efficiency of MPC on sparse networks in [28], where we reduced the degree and improved tolerance for sparse network. This resolved an open question of Dwork, Peleg, Pippenger, and Upfal posed in 1986.
- [13, 41] We showed an unexpected tight connection between circuit lower bounds and randomness needed for private computation.

Password Security: If two players have a short secret common password, can they jointly generate a secure session-key in a way that prevents the man in the middle from mounting a "dictionary attack" as well as guarantees security in a concurrent setting, where multiple invocations of the protocol run in an arbitrary interleaved manner? This question attracted considerable attention in practice, but it remained open until [38, 10]. The work turned out to be very influential, with over 250 citations. In [32] we showed how to achieve these results without the common reference string. Additionally, in [8] we considered how to generate session-keys using biometric data and in [6] we showed how to extract optimal amount of common randomness from weak correlated sources.

Proactive security: Together with Yung, in 1991 I put forward a model of malicious faults which can spread through the network, analogous to the spread of a virus [51]. Assuming that the detection and infection rates are comparable (i.e., assuming that the virus does not take over the entire network), we show how a distributed network can maintain computation even if a constant fraction of the machines are infected, and even if faults move around. This work was the first to initiate an area of study called *proactive security*, with over 400 papers subsequently written by various researchers on this subject.

Private Information Retrieval: I worked on the problem of communication-efficient retrieval of data from a remote database in such a way that even a database administrator would not know which particular data-item had been retrieved. Disproving a widely held belief within the community, I showed, with Kushilevitz, that replication of databases is not necessary for this task [39] (all prior works required replication).The paper became influential, with over 400 citations. I have worked on other extensions and generalizations of this problem [29, 7, 48] as well as connections to other cryptographic primitives [40] and minimal assumptions needed for solving this problem [30].

2 Algorithms

Fault-tolerant protocols and computations play an important role in distributed networks: as networks (such as the Internet or cellular networks) become larger and larger, the need for distributed algorithms which are resilient against errors is apparent. By examining different types of faults, different questions about fault-tolerance can be studied. I am interested in finding efficient fault-tolerant algorithms resilient against errors in various settings. For brevity, only a few of my research directions in this area are described below:

Routing and Distributed Protocols: In [45] with Rabani we show a nearly-optimal (in terms of packet delivery time) randomized routing strategy for arbitrary topology networks, answering the problem posed

in 1988 by Leighton, Maggs and Rao. In [4] we give a routing algorithm competitive against bursty adversarial traffic where the algorithm has to choose packet-routes. As another example, we have shown that on overloaded networks (such as the Internet) it is beneficial to study network *throughput* [18], deriving results that are incomparable with adversarial queuing theory. In [25], with Bunn we established (via competitive analysis) a matching upper and lower bound for online routing in an asynchronous adversarial setting.

Distributed Communication Complexity: In the Combinatorica paper [11] with Kushilevitz and Linial we show that in some distributed protocols, a "smart" intermediary who does not have any inputs can nevertheless help to reduce the total cost of communication, disproving a conjecture of Tiwari from 1982.

Geometric Search Problems on High-Dimensional Data: One of the central issues in computational geometry and data-mining protocols is the challenge of dealing with high-dimensional data. In computational geometry, many problems suffer from the so-called "curse of dimensionality" which basically says that work needed to solve the problem grows exponentially with the problem's dimension. The key challenge in that area (which many other problems are reducible to) is the problem of "nearest-neighbor search". In [12] with Kushilevitz and Rabani we show a very efficient protocol for solving an approximate version of this problem, considerably strengthening the previous results of Kleinberg. (The paper became highly cited, with over 300 citations). We showed results for the hamming cube and for Euclidian space. For Euclidian space, our results were independently and contemporaneously achieved by Indyk and Motwani. At the heart of our construction is the notion of distance-preserving dimension reduction for the hamming cube in a certain range of distance. Our bounds for dimension reduction in the hamming cube were further sharpened in our JACM paper [16] that showed a PTAS for k-clustering in high dimensions. (Our dimension-reduction technique for the cube is an analog of the celebrated Johnson-Lindenstrauss lemma.) The technique proved to be a powerful new tool for many high-dimensional problems, such as finding approximate MST [5] in high dimensions. On the lower-bound front, with Borodin and Rabani we showed a lower bound for nearest-neighbor as well as partial-match problems using tools from communication complexity [1]. More recently, I became interested in the edit distance and showed with Rabani how to *embed* edit distance metric into l_1 with small distortion [17, 46].

Streaming Algorithms: Several years ago Indyk and McGregor asked if it was possible to obtain an effective approximation of how correlated the data in a stream is (say of internet packets of database entries). Finding correlations in huge volumes of data (when there is no memory to store it all) is akin to finding the proverbial needle in the haystack; yet, correlations must be found in real time. In [23] with Braverman, we showed that this is possible. In 1996 Alon, Matias and Szegedi published a seminal paper on frequency functions and stated an important open question: *what class of functions on frequency moments can be computed on streams?* The problem has remained unresolved since 1996 despite hundreds of publications by other researchers. In [24] we came up with a zero-one law for computing functions on streams that gives a litmus test for what could be done, thus answering the question of Alon, et al from 1996. Another important question in steaming was asked by Indyk and Motwani in 2002: *what can be computed in sliding windows (i.e., when data entries eventually expire)?* Despite dozens of publications on this subject, little was known until in our work with Braverman [22] where we resolved this question optimally. With Skeith, I showed how to perform private keyword search on *streaming* data [44, 15]. This work received much attention from the *intelligence community*. In [21], we considered whether our clustering algorithm proposed in [47] (that introduced a new *seeding* procedure for k-clustering) can be performed in streaming environment. Using a novel construction, we answered this question in the affirmative.

.

Rafail Ostrovsky – Selected Publications

Book Chapters

[1] Allan Borodin, Rafail Ostrovsky, and Yuval Rabani. *In Discrete and Computational Geometry - The Goodman-Pollack Festschrift. Algorithms and Combinatorics Series 3143*, chapter Lower Bounds for High Dimensional Nearest Neighbor Search and Related Problems, pages 255–276. Springer Verlag, Berlin, 2003.

[2] Rafail Ostrovsky, Ramarathnam Venkatesan, and Moti Yung. Fair Games Against an All-Powerful Adversary (preliminary version). In Renato M. Capocelli, Alfredo De Santis, and Ugo Vaccaro, editors, *Sequences II: Communication, Security, and Computer Science*, pages 418–429. Springer-Verlag, 1991. From International Advanced Workshop. Sequences II, Positano, Italy, June 1991. Prior to Positano, this work was first presented at DIMACS Complexity and Cryptography Workshop, October 1990, Princeton, NJ.

[3] Rafail Ostrovsky, Ramarathnam Venkatesan, and Moti Yung. Fair Games Against an All-Powerful Adversary (full version). In Jin-Yi Cai, editor, *DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol. 13*, pages 155–169. AMS, 1993. This work was first presented at DIMACS Complexity and Cryptography Workshop, October 1990, Princeton, NJ.

Journal Publications

[4] William Aiello, Eyal Kushilevitz, Rafail Ostrovsky, and Adi Rosén. Adaptive packet routing for bursty adversarial traffic. *J. Comput. Syst. Sci.*, 60(3):482–509, 2000.

[5] Allan Borodin, Rafail Ostrovsky, and Yuval Rabani. Subquadratic approximation algorithms for clustering problems in high dimensional spaces. *Machine Learning*, 56(1-3):153–167, 2004.

[6] Ran Canetti, Eyal Kushilevitz, Rafail Ostrovsky, and Adi Rosén. Randomness versus fault-tolerance. *J. Cryptology*, 13(1):107–142, 2000.

[7] Giovanni Di Crescenzo, Yuval Ishai, and Rafail Ostrovsky. Universal service-providers for private information retrieval. *J. Cryptology*, 14(1):37–74, 2001.

[8] Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM J. Comput.*, 38(1):97–139, 2008.

[9] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Zero-knowledge proofs from secure multiparty computation. *SIAM J. Comput.*, 39(3):1121–1152, 2009.

[10] Jonathan Katz, Rafail Ostrovsky, and Moti Yung. Efficient and secure authenticated key exchange using weak passwords. *J. ACM*, 57(1), 2009.

[11] Eyal Kushilevitz, Nathan Linial, and Rafail Ostrovsky. The linear-array conjecture in communication complexity is false. *Combinatorica*, 19(2):241–254, 1999.

[12] Eyal Kushilevitz, Rafail Ostrovsky, and Yuval Rabani. Efficient search for approximate nearest neighbor in high dimensional spaces. *SIAM J. Comput.*, 30(2):457–474, 2000.

[13] Eyal Kushilevitz, Rafail Ostrovsky, and Adi Rosén. Characterizing linear size circuits in terms of pricacy. *J. Comput. Syst. Sci.*, 58(1):129–136, 1999.

[14] Moni Naor, Rafail Ostrovsky, Ramarathnam Venkatesan, and Moti Yung. Perfect zero-knowledge arguments for *np* using any one-way permutation. *J. Cryptology*, 11(2):87–108, 1998.

[15] Rafail Ostrovsky and William E. Skeith III. Private searching on streaming data. *J. Cryptology*, 20(4):397–430, 2007.

[16] Rafail Ostrovsky and Yuval Rabani. Polynomial-time approximation schemes for geometric min-sum median clustering. *J. ACM*, 49(2):139–156, 2002.

[17] Rafail Ostrovsky and Yuval Rabani. Low distortion embeddings for edit distance. *J. ACM*, 54(5), 2007.

Refereed Conference Proceedings

[18] William Aiello, Eyal Kushilevitz, Rafail Ostrovsky, and Adi Rosén. Dynamic routing on networks with fixed-size buffers. In *SODA*, pages 771–780, 2003.

[19] Dan Boneh, Giovanni Di Crescenzo, Rafail Ostrovsky, and Giuseppe Persiano. Public key encryption with keyword search. In *EUROCRYPT*, pages 506–522, 2004.

[20] Dan Boneh, Shai Halevi, Michael Hamburg, and Rafail Ostrovsky. Circular-secure encryption from decision diffie-hellman. In *CRYPTO*, pages 108–125, 2008.

[21] Vladimir Braverman, Adam Meyerson, Rafail Ostrovsky, Alan Roytman, Michael Shindler, and Brian Tagiku. Streaming k-means on well-clusterable data. In *SODA*, pages 26–40, 2011.

[22] Vladimir Braverman and Rafail Ostrovsky. Smooth histograms for sliding windows. In *FOCS*, pages 283–293, 2007.

[23] Vladimir Braverman and Rafail Ostrovsky. Measuring independence of datasets. In *STOC*, pages 271–280, 2010.

[24] Vladimir Braverman and Rafail Ostrovsky. Zero-one frequency laws. In *STOC*, pages 281–290, 2010.

[25] Paul Bunn and Rafail Ostrovsky. Asynchronous throughput-optimal routing in malicious networks. In *ICALP (2)*, pages 236–248, 2010.

[26] Ran Canetti, Cynthia Dwork, Moni Naor, and Rafail Ostrovsky. Deniable encryption. In *CRYPTO*, pages 90–104, 1997.

[27] Ran Canetti, Yehuda Lindell, Rafail Ostrovsky, and Amit Sahai. Universally composable two-party and multi-party secure computation. In *STOC*, pages 494–503, 2002.

[28] Nishanth Chandran, Juan A. Garay, and Rafail Ostrovsky. Improved fault tolerance and secure computation on sparse networks. In *ICALP (2)*, pages 249–260, 2010.

[29] Giovanni Di Crescenzo, Yuval Ishai, and Rafail Ostrovsky. Non-interactive and non-malleable commitment. In *STOC*, pages 141–150, 1998.

[30] Giovanni Di Crescenzo, Tal Malkin, and Rafail Ostrovsky. Single database private information retrieval implies oblivious transfer. In *EUROCRYPT*, pages 122–138, 2000.

[31] Juan A. Garay and Rafail Ostrovsky. Almost-everywhere secure computation. In *EUROCRYPT*, pages 307–323, 2008.

[32] Vipul Goyal, Abhishek Jain, and Rafail Ostrovsky. Password-authenticated session-key generation on the internet in the plain model. In *CRYPTO*, pages 277–294, 2010.

[33] Jens Groth, Rafail Ostrovsky, and Amit Sahai. Non-interactive zaps and new techniques for nizk. In *CRYPTO*, pages 97–111, 2006.

[34] Jens Groth, Rafail Ostrovsky, and Amit Sahai. Perfect non-interactive zero knowledge for np. In *EUROCRYPT*, pages 339–358, 2006.

[35] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Zero-knowledge from secure multiparty computation. In *STOC*, pages 21–30, 2007.

[36] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Cryptography with constant computational overhead. In *STOC*, pages 433–442, 2008.

[37] Jonathan Katz and Rafail Ostrovsky. Round-optimal secure two-party computation. In *CRYPTO*, pages 335–354, 2004.

[38] Jonathan Katz, Rafail Ostrovsky, and Moti Yung. Efficient password-authenticated key exchange using human-memorable passwords. In *EUROCRYPT*, pages 475–494, 2001.

[39] Eyal Kushilevitz and Rafail Ostrovsky. Replication is not needed: Single database, computationally-private information retrieval. In *FOCS*, pages 364–373, 1997.

[40] Eyal Kushilevitz and Rafail Ostrovsky. One-way trapdoor permutations are sufficient for non-trivial single-server private information retrieval. In *EUROCRYPT*, pages 104–121, 2000.

[41] Eyal Kushilevitz, Rafail Ostrovsky, and Adi Rosén. Characterizing linear size circuits in terms of privacy. In *STOC*, pages 541–550, 1996.

[42] Moni Naor, Rafail Ostrovsky, Ramarathnam Venkatesan, and Moti Yung. Perfect zero-knowledge arguments for np can be based on general complexity assumptions. In *CRYPTO*, pages 196–214, 1992.

[43] Rafail Ostrovsky. One-way functions, hard on average problems, and statistical zero-knowledge proofs. In *Structure in Complexity Theory Conference*, pages 133–138, 1991.

[44] Rafail Ostrovsky and William E. Skeith III. Private searching on streaming data. In *CRYPTO*, pages 223–240, 2005.

[45] Rafail Ostrovsky and Yuval Rabani. Universal (congestion + dilation + $\log^{(1+\epsilon)} n$) local control packet switching algorithms. In *STOC*, pages 644–653, 1997.

[46] Rafail Ostrovsky and Yuval Rabani. Low distortion embeddings for edit distance. In *STOC*, pages 218–224, 2005.

[47] Rafail Ostrovsky, Yuval Rabani, Leonard J. Schulman, and Chaitanya Swamy. The effectiveness of lloyd-type methods for the k-means problem. In *FOCS*, pages 165–176, 2006.

[48] Rafail Ostrovsky and Victor Shoup. Private information storage. In *STOC*, pages 294–303, 1997.

[49] Rafail Ostrovsky, Ramarathnam Venkatesan, and Moti Yung. Interactive hashing simplifies zero-knowledge protocol design. In *EUROCRYPT*, pages 267–273, 1993.

[50] Rafail Ostrovsky and Avi Wigderson. One-way fuctions are essential for non-trivial zero-knowledge. In *ISTCS*, pages 3–17, 1993.

[51] Rafail Ostrovsky and Moti Yung. How to withstand mobile virus attacks. In *PODC*, pages 51–59, 1991.