Research Statement

Prashant Nalini Vasudevan

September 3, 2020

My research is centered around the field of cryptography. Modern cryptography, by its very nature, is closely connected to several other areas of computer science – it has as its basis conjectures about algorithms and techniques from complexity theory, it draws motivation from computer and network security and privacy, and models from distributed computing. And, in turn, it gives back in several ways to all of these fields.

Similarly, my research in cryptography uses concepts and techniques from complexity theory and theoretical computer science, while keeping in view the motivations from and relevance to various other areas. The following is a brief description of my contributions to various parts of cryptography. The remaining sections go over these in greater detail.

Useful Computational Hardness: Cryptography is built on hardness assumptions that are used to prove the security of cryptosystems. I have studied these assumptions in various settings, shown how certain existing conjectures can be used for cryptography, and characterised useful assumptions in terms of established concepts from theoretical computer science.

Fine-Grained Cryptography: I have made several fundamental contributions to the area of fine-grained cryptography, which asks for security against adversaries running in time n^k for some constant k. This alternative to requiring security against general polynomial-time adversaries opens up the possibility of meaningful cryptography even in a world where $\mathsf{P} = \mathsf{NP}$.

Data Protection: I have been analysing concepts from recent data protection laws (such as the GDPR and the CCPA) from a cryptographic point of view. I have come up with precise definitions of some of the basic concepts that these laws deal with, as well as techniques to build and analyse systems that seek to satisfy certain requirements in these laws.

Secret Sharing: Secret sharing is a fundamental primitive in information-theoretic cryptography. I have constructed new secret sharing schemes that have additional properties that are essential for their practical applications. I have also expanded our understanding of the complexity of general secret sharing, and its connections to various other concepts.

Complexity Theory: A common thread through a lot of my research is identifying connections to questions and concepts from complexity theory. I have also made direct contributions to various parts of complexity theory itself, such as oracle separations between natural complexity classes, list-decoding algorithms, and hardness amplification theorems.

In the future, I will continue my research in the above directions, while keeping an eye out for other interesting areas of study in which my knowledge, perspective and experience will be useful.

Research Principles. My work has so far been, and continues to be, directed by the following values and considerations that I have developed over the years.

- **Rigour:** My approach to research is based around the rigour of concepts and techniques from mathematics and theoretical computer science. I seek to bring this rigour into any work I do that has a comfortable place for it.
- **Utility:** Even the most rigorous theorem is of little value on its own. Cryptography presents the remarkable balance of being founded on rigour while also dealing with topics of definite utility, be it to real-world systems or to deep theoretical questions from other areas.
- **Learning:** An important part of academic research, to me, is the opportunity to constantly learn new and interesting things, be it a clever proof technique, a new way of looking at a familiar problem, or an adjacent area of research with interesting connections to mine.
- **Collaboration:** Working in cryptography has given me the opportunity to collaborate with a wide variety of researchers. These collaborations have always been educational and rewarding, and I look forward to developing more fruitful collaborations and learning more from them.
- **Broader Societal Impact:** An important aspect of research that I am only beginning to understand is the silent, and often unexpected, effect it can have on several aspects of society (see, for instance, [Win80, Rog15]). I plan to seek out ways in which my work can positively impact society, and maybe even help communities that have been disadvantaged or neglected.

1 Foundations of Cryptography

Assumptions about the computational hardness of specific problems are the foundation of modern cryptography, and are crucial for demonstrating the security of all but the simplest cryptographic primitives. A lot of my work is concerned with studying existing hardness assumptions and identifying new ones that cryptography can be based on. There are three primary motivations for this line of study.

- **Diversity.** In spite of decades of research, we only know a handful of reasonable assumptions that are useful for cryptography, especially for advanced primitives like public-key encryption [Bar17]. This is a precarious situation, especially considering that an important family of such assumptions are broken by quantum computers. There is thus a need for a more varied set of hardness assumptions from which we can construct cryptographic primitives, so that all of our systems are not broken by a handful of algorithmic or technological breakthroughs.
- **Efficiency.** Different assumptions present different tradeoffs in the parameters involved, and starting from a new assumption has the potential to lead to cryptosystems with better efficiency or other desirable properties that are not available in existing ones.
- **Understanding.** Understanding the kinds of problems whose hardness can be used to do cryptography will lead to a better understanding of the landscape of computational hardness from a complexity theoretic point of view, leading in turn to a better understanding of the nature of computation itself.

For these reasons, an important part of my research program is to diversify and reinforce the foundations of cryptography by studying hardness assumptions directly. Along with several collaborators, I made progress in this direction in various settings, some of which I describe below.

1.1 Fine-Grained Cryptography

Efficiency in cryptography, following standard practice in complexity theory, is typically modelled as polynomial running time – algorithms and adversaries are efficient if they run in some polynomial time, and inefficient otherwise. Several real-world circumstances, however, readily accommodate more *fine-grained* models of efficiency. For instance, an encryption scheme where encryption and decryption take n time, and which is secure against n^2-time adversaries, would be realistic and useful for several values of n. In fact, perhaps counterintuitively, such a scheme could be more efficient than ones with exponential security for realistic concrete security levels – in the regime of moderately small parameters n, for a large constant k, security growth at the rate of n^k can easily dominate 2^n.

This opens up the possibility of constructing cryptosystems using problems that would traditionally have been considered too easy for this purpose because they have polynomial-time algorithms, but are still hard for, say, n^2-time algorithms. Intriguingly, this leads to the possibility of a positive answer to the following question, which would have been impossible with standard cryptography:

Can there exist meaningful cryptography in a world where P *is equal to* NP*?*

Along with a number of collaborators, I have made several fundamental contributions to the study of such *fine-grained cryptography.* While it dates back to the early work of Merkle in the 70's [Mer78], fine-grained cryptography has found new ground in the past few years with the recent advances in fine-grained complexity theory (see [Wil15] for a survey). While fine-grained complexity today consists of a substantial body of work, most of it so far has been concerned with the *worst-case* complexities of problems, whereas cryptography requires problems that are hard in the *average-case.*

> **Average-Case Fine-Grained Hardness:** With Marshall Ball, Alon Rosen, and Manuel Sabin, I showed several fine-grained worst-case to average-case reductions [BRSV17]. Assuming well-studied conjectures regarding the worst-case hardness of k-SAT (the Strong Exponential-Time Hypothesis) or any of a handful of other problems, our reductions result in problems that are hard to solve *on average* in time $n^{2-\delta}$. This represents an essential first step in being able to use such hardness assumptions for cryptography.
>
> **Proofs of Work:** Building on the above work, with the same collaborators, I showed how to construct proof-of-work protocols under the same assumptions [BRSV18]. Proof-of-work protocols are inherently fine-grained cryptographic primitives that allow a prover to prove to a verifier that it has done a certain amount of computational work. Among other applications, they are the backbone of the blockchains used in various cryptocurrencies.

Both the above results were obtained using the fact that k-SAT, as well as the other problems we used, could be represented by polynomials of low-degree over finite fields. In fact, our constructions work with any problem that has this property, and this was partially formalised later by Goldreich and Rothblum [GR20]. This represents an interesting sufficient condition for hardness – in this case, fine-grained hardness – to be *useful* for cryptography. An intriguing question here is identifying such useful structure in these problems that is different from what has been found useful for cryptography in the past, and interesting work is already being done in this regard [LLW19].

Cryptography for Low-Depth Circuits: With Akshay Degwekar and Vinod Vaikuntanathan, in [DVV16], I showed constructions of other, more advanced, fine-grained cryptographic primitives with circuit depth (or running time in heavily parallel models of computation) as the computational resource of interest. Some of these were conditional, and others were based on reasonable worst-case assumptions about logspace classes.

1.2 Statistical Zero Knowledge

In the study of such hardness that is useful for cryptography, one concept that comes up repeatedly is that of *Statistical Zero Knowledge (SZK) proofs.* An SZK proof for a language L is an interactive proof involving a computationally unbounded "prover" and an efficient "verifier", where the prover proves to the verifier that a given instance x is contained in L, with the remarkable property that this proof does not reveal anything else to the verifier about x.

The class of computational problems that have such proofs, referred to as SZK, happens to contain a remarkably large fraction of the problems we know to be useful for cryptography, such as Discrete Logarithm, the Diffie Hellman problem, and several lattice problems. These comprise most of the problems that have been used to construct public-key encryption in the past, though these constructions were done in very different ways, using properties specific to the different problems.

Unifying Public-Key Constructions: With Itay Berman, Akshay Degwekar, and Ron Rothblum, I showed that this fact was not a coincidence [BDRV18a]. We identified a common property among these problems that enables the construction of public-key encryption – they all have SZK proofs where the prover, given a witness to the instance, is computationally efficient and only sends very short messages to the verifier. This opens up another means of identifying problems that could be useful for constructing public-key encryption. We also showed that a hard problem with a weaker version of this property was *necessary* in order for public-key encryption to exist.

Multi-Collision Resistance: With the same collaborators, I also showed that the hardness of certain problems closely related to SZK can be used to construct multicollision-resistant hash functions [BDRV18b]. These are a weaker version of a collision-resistant hash functions that can replace them in various applications [KNY17].

1.3 Lossy Algorithms

An algorithm is lossy if its output loses some information about the input (for example, simply by being shorter than the input). Such algorithms generalise concepts from a number of diverse areas in computer science, from kernelisation in the context of parameterised complexity, to randomised encodings from cryptography.

Cryptography from Lossy Reductions: With a number of collaborators, in [BBD+20], I showed that if there is a reduction from the composition of a problem Π with the OR function to another problem Π', and the reduction has certain lossiness properties, then even the worst-case hardness of Π can be used to construct one-way functions. And certain other conditions allow its average-case hardness to be used to construct collision-resistant hash functions. This identifies yet another property that can make hard problems useful for cryptography.

2 Cryptography and Data Protection

Over the past year, I have been looking at certain aspects of recent data protection laws from a cryptographic point of view. The context for this is the growing collection of data about us by corporations and governmental agencies, which is used in various ways that affect several aspects our lives. In the past few years, a number of laws have come into effect in several countries, such as the GDPR in the European Union and the CCPA in California, that regulate how this collected data can be used. These laws are explicitly about and involve technology, and make explicit demands of the systems that handle such data. My objective is to understand these demands as they relate to cryptography, and identify ways in which cryptographic perspectives and ideas can help develop and analyse systems that are subject to these regulations.

2.1 Formalising Data Deletion

An important feature of many of these laws is the *right to erasure* or the *right to be forgotten.* This is the right of an individual to request deletion of their data by an entity that possesses it. But what exactly does the entity need to do to effectively satisfy such a request?

If the data were merely being stored as is, then it could simply be erased. Typically, however, data is not just stored, but also used – to compute statistics, train machine learning models, etc.. The results of these computations could also contain information about the individual's data, and may need to be appropriately modified by the deletion algorithm in order to really honour a deletion request. Thus, the following natural question emerges:

What does it mean for a system to really delete data?

In other terms, can we identify a property of systems that captures correct deletion? Then, if the entity's suite of data-processing algorithms has this property, it could be considered to be deleting data correctly.

> **Defining Data Deletion:** With Sanjam Garg and Shafi Goldwasser, in [GGV20], I formulated technically precise properties that capture certain notions of deletion. These were based on the real-ideal paradigm that is commonly used in cryptographic security definitions. We then showed how existing techniques from various areas, such as history-independent algorithms and differential privacy, could be used to build systems that satisfy these definitions.

Put simply, our definitions compare the state of the system resulting from running the deletion algorithm for a specific individual's data with the state of the system in a hypothetical world where this individual never interacted with the system at all, and ask that these states look similar. These definitions capture natural intuitive notions of what properly deleting data could mean, and we hope that they will be useful both in analysing existing data-processing systems to make sure they are handling deletion correctly, and in designing new systems that do so.

Simplifications and Generalisations: I am currently working on a project (with Aloni Cohen, Adam Smith, and Marika Swanberg) to simplify and generalise the concepts introduced in the above work. Our objective is to come up with a rigorous model of data deletion that is simultaneously precise enough for computer scientists and engineers to use in designing data-processing systems, and also simple and clear enough for policy and law experts, who may not be familiar with cryptographic definitions, to understand and use.

We have so far presented a preliminary draft of the above work at the Privacy Law Scholars Conference 2020, and expect to put out a complete manuscript soon.

2.2 Deletion in Machine Learning

An active area of research in the Machine Learning community is the more specific task of deleting training data from ML models. This is of particular interest today, as a lot of the data collected by companies is indeed used to train ML models. There has been a rapidly growing body of work that studies variants of this question [GGVZ19, GGHvdM19, . . .], but most of them essentially treat re-training on the remaining data as the ideal and try to come up with ways to do this efficiently.

This, however, does not address an important concern with a real-world adversary against deployed ML models – it could have partial or full access to the old model before deletion, in addition to whatever access it has to the updated model after deletion. Even defining security against such adversaries turns out to be non-trivial.

Deletion Privacy against Continual Access: In an ongoing project with Ji Gao, Sanjam Garg, and Mohammad Mahmoody, we are looking into the deletion of training data from machine learning models against adversaries that have access to the models both before and after deletion. We identify a number of necessary properties that deletion algorithms have to satisfy in this regard, and provide experimental evidence to demonstrate that, in several cases, re-training performs rather poorly against such adversaries.

2.3 Verifying Data Deletion

From a cryptographic perspective, an important question to ask in this respect is whether an untrusted server that holds a client's data, and then claims to have deleted it, can indeed *prove* to the client that it has done so. Of course, this is impossible to do soundly if the server has additional space to make backups of the data. But is this the only obstacle to such proofs?

Provably Deleting Data: In ongoing work with Quanquan Liu, a graduate student at MIT, we show that if the client knows, to a good approximation, the amount of storage the server has access to, then the server can indeed prove that (most of) the deletion it claimed to have done was actually done. This involves designing new proof-of-space protocols with interesting additional correctness properties.

3 Secret Sharing

Secret sharing is one of the most basic and important primitives in information-theoretic cryptography (and, indeed, all of cryptography). In its simplest version, called threshold secret sharing, the task is to share a *secret* s into n *shares* such that any collection of at least t shares can be used to reconstruct s (called *correctness*), while any collection of at most $(t-1)$ shares contains no information about the secret (*privacy*).

These properties of correctness and privacy represent security against simple passive adversaries. In many circumstances, however, stronger versions of these properties are desired, and I have been studying such strengthenings and constructions satisfying them.

> **Leakage Resilience:** With Akshayaram Srinivasan, I constructed simple and efficient leakage-resilient secret sharing schemes [SV19]. Leakage-resilience is a strengthening of the privacy property, where the secret has to be hidden from an adversary who is given not just a set of $(t-1)$ shares, but also some small amount of information about each of the other shares, which represents side-channels that the adversary may have access to.
>
> **Robustness:** With Pasin Manurangsi and Akshayaram Srinivasan, I constructed robust secret sharing schemes that are secure against rushing adversaries and have near-optimal overhead in share sizes [MSV20]. Robustness is a strengthening of the correctness property where the secret is recoverable from the shares even if some of them are adversarially corrupted. A rushing adversary is one that is allowed to corrupt its shares *after* looking at all the other shares. Security against such adversaries is an essential property for such schemes to be useful in secure multi-party computation protocols. Our work, by obtaining near-optimal parameters against the strongest adversaries, essentially completes the picture in this line of research.

More generally, secret sharing schemes are described by an *access structure*, which is a specification of the sets of parties that are allowed to recover the secret using their shares.

> **Efficient Secret Sharing and SZK:** Along with Vinod Vaikuntanathan, in [VV15], I showed that access structures that have secret sharing schemes with efficient sharing algorithms correspond in a natural manner to languages that have SZK proofs with logspace verifiers.

Secret sharing is also closely connected to a primitive called *Conditional Disclosure of Secrets (CDS)*, which may be seen as an analogue of SZK proofs in the setting of information-theoretic cryptography. This connection has been behind recent breakthroughs in constructing non-trivial secret sharing schemes for general access structures [LV18].

> **The Complexity of CDS:** With Benny Applebaum and others [AV19, AARV17], I have studied several aspects of the complexity of CDS protocols, found connections to various concepts from the study of communication complexity, and shown new lower bounds, amplification and amortisation procedures, and closure properties.

4 Other Research Highlights

Apart from the aforementioned areas, I have worked on a number of other aspects of cryptography, complexity theory, algorithms, and the connections between them. I describe some of this work briefly below.

The Complexity of SZK: With Adam Bouland, Lijie Chen, Dhiraj Holden, and Justin Thaler, I studied structural properties of the class SZK [BCH$^+$17]. We showed that there is an oracle in the presence of which SZK is not contained in the class PP (problems with unbounded-error randomised algorithms). We also showed that, in the presence of this oracle, SZK is not contained in the class PZK of problems that have *perfect* zero knowledge proofs, a problem that had been open since the work of Aiello and Hastad in the 90's [AH91]. We also showed similar results for the communication complexity analogues of these classes.

Reductions for Sparse Linear Systems: In work with Andrej Bogdanov and Manuel Sabin [BSV19], we gave sample-efficient search-to-decision reductions for the problem of solving/recognising systems of noisy sparse linear equations. In doing so, we developed a new approximate list-decoding algorithm for sparse XOR codes at large distances, resulting in new XOR lemmas in certain parameter regimes.

Verifiable Delay Functions: Together with Nico Döttling, Sanjam Garg, and Giulio Malavolta, I studied Verifiable Delay Functions (VDFs) [DGMV19]. A VDF is a function that takes a certain number of sequential steps to compute, but on computation produces an quickly verifiable proof that it was computed correctly. We studied tight VDFs, where the complexity of computing the function honestly is very close to the guaranteed sequentiality bound. We showed a black-box transformation of non-tight VDFs with a natural additional property into tight VDFs, and also showed that such tight VDFs could not be constructed in a black-box manner from random oracles.

References

[AARV17] Benny Applebaum, Barak Arkis, Pavel Raykov, and Prashant Nalini Vasudevan. Conditional disclosure of secrets: Amplification, closure, amortization, lower-bounds, and separations. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part I*, volume 10401 of *Lecture Notes in Computer Science*, pages 727–757. Springer, 2017.

[AH91] William Aiello and Johan Håstad. Relativized perfect zero knowledge is not BPP. *Inf. Comput.*, 93(2):223–240, 1991.

[AV19] Benny Applebaum and Prashant Nalini Vasudevan. Placing conditional disclosure of secrets in the communication complexity universe. In Avrim Blum, editor, *10th Innovations in Theoretical Computer Science Conference, ITCS 2019, January 10-12, 2019, San Diego, California, USA*, volume 124 of *LIPIcs*, pages 4:1–4:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[Bar17] Boaz Barak. The complexity of public-key cryptograph. *Electronic Colloquium on Computational Complexity (ECCC)*, 24:65, 2017.

[BBD$^+$20] Marshall Ball, Elette Boyle, Akshay Degwekar, Apoorvaa Deshpande, Alon Rosen, Vinod Vaikuntanathan, and Prashant Nalini Vasudevan. Cryptography from information loss. In Thomas Vidick, editor, *11th Innovations in Theoretical Computer*

Science Conference, ITCS 2020, January 12-14, 2020, Seattle, Washington, USA, volume 151 of *LIPIcs*, pages 81:1–81:27. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.

[BCH$^+$17] Adam Bouland, Lijie Chen, Dhiraj Holden, Justin Thaler, and Prashant Nalini Vasudevan. On the power of statistical zero knowledge. In Chris Umans, editor, *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017*, pages 708–719. IEEE Computer Society, 2017.

[BDRV18a] Itay Berman, Akshay Degwekar, Ron D. Rothblum, and Prashant Nalini Vasudevan. From laconic zero-knowledge to public-key cryptography - extended abstract. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part III*, volume 10993 of *Lecture Notes in Computer Science*, pages 674–697. Springer, 2018.

[BDRV18b] Itay Berman, Akshay Degwekar, Ron D. Rothblum, and Prashant Nalini Vasudevan. Multi-collision resistant hash functions and their applications. In Jesper Buus Nielsen and Vincent Rijmen, editors, *Advances in Cryptology - EUROCRYPT 2018 - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29 - May 3, 2018 Proceedings, Part II*, volume 10821 of *Lecture Notes in Computer Science*, pages 133–161. Springer, 2018.

[BRSV17] Marshall Ball, Alon Rosen, Manuel Sabin, and Prashant Nalini Vasudevan. Average-case fine-grained hardness. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 483–496. ACM, 2017.

[BRSV18] Marshall Ball, Alon Rosen, Manuel Sabin, and Prashant Nalini Vasudevan. Proofs of work from worst-case assumptions. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I*, volume 10991 of *Lecture Notes in Computer Science*, pages 789–819. Springer, 2018.

[BSV19] Andrej Bogdanov, Manuel Sabin, and Prashant Nalini Vasudevan. XOR codes and sparse learning parity with noise. In Timothy M. Chan, editor, *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 986–1004. SIAM, 2019.

[DGMV19] Nico Döttling, Sanjam Garg, Giulio Malavolta, and Prashant Nalini Vasudevan. Tight verifiable delay functions. *IACR Cryptol. ePrint Arch.*, 2019:659, 2019.

[DVV16] Akshay Degwekar, Vinod Vaikuntanathan, and Prashant Nalini Vasudevan. Fine-grained cryptography. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part III*, volume 9816 of *Lecture Notes in Computer Science*, pages 533–562. Springer, 2016.

[GGHvdM19] Chuan Guo, Tom Goldstein, Awni Y. Hannun, and Laurens van der Maaten. Certified data removal from machine learning models. *CoRR*, abs/1911.03030, 2019.

[GGV20] Sanjam Garg, Shafi Goldwasser, and Prashant Nalini Vasudevan. Formalizing data deletion in the context of the right to be forgotten. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part II*, volume 12106 of *Lecture Notes in Computer Science*, pages 373–402. Springer, 2020.

[GGVZ19] Antonio Ginart, Melody Y. Guan, Gregory Valiant, and James Zou. Making AI forget you: Data deletion in machine learning. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 3513–3526, 2019.

[GR20] Oded Goldreich and Guy N. Rothblum. Worst-case to average-case reductions for subclasses of P. In Oded Goldreich, editor, *Computational Complexity and Property Testing - On the Interplay Between Randomness and Computation*, volume 12050 of *Lecture Notes in Computer Science*, pages 249–295. Springer, 2020.

[KNY17] Ilan Komargodski, Moni Naor, and Eylon Yogev. Collision resistant hashing for paranoids: Dealing with multiple collisions. *IACR Cryptology ePrint Archive*, 2017:486, 2017.

[LLW19] Rio LaVigne, Andrea Lincoln, and Virginia Vassilevska Williams. Public-key cryptography in the fine-grained setting. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part III*, volume 11694 of *Lecture Notes in Computer Science*, pages 605–635. Springer, 2019.

[LV18] Tianren Liu and Vinod Vaikuntanathan. Breaking the circuit-size barrier in secret sharing. In Ilias Diakonikolas, David Kempe, and Monika Henzinger, editors, *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 699–708. ACM, 2018.

[Mer78] Ralph C. Merkle. Secure communications over insecure channels. *Commun. ACM*, 21(4):294–299, 1978.

[MSV20] Pasin Manurangsi, Akshayaram Srinivasan, and Prashant Nalini Vasudevan. Nearly optimal robust secret sharing against rushing adversaries. In Daniele Micciancio and Thomas Ristenpart, editors, *Advances in Cryptology - CRYPTO 2020 - 40th Annual International Cryptology Conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17-21, 2020, Proceedings, Part III*, volume 12172 of *Lecture Notes in Computer Science*, pages 156–185. Springer, 2020.

[Rog15] Phillip Rogaway. The moral character of cryptographic work. *IACR Cryptol. ePrint Arch.*, 2015:1162, 2015.

[SV19] Akshayaram Srinivasan and Prashant Nalini Vasudevan. Leakage resilient secret sharing and applications. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part II*, volume 11693 of *Lecture Notes in Computer Science*, pages 480–509. Springer, 2019.

[VV15] Vinod Vaikuntanathan and Prashant Nalini Vasudevan. Secret sharing and statistical zero knowledge. In Tetsu Iwata and Jung Hee Cheon, editors, *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part I*, volume 9452 of *Lecture Notes in Computer Science*, pages 656–680. Springer, 2015.

[Wil15] Virginia Vassilevska Williams. Hardness of easy problems: Basing hardness on popular conjectures such as the strong exponential time hypothesis. In *Proc. International Symposium on Parameterized and Exact Computation*, pages 16–28, 2015.

[Win80] Langdon Winner. Do artifacts have politics? *Daedalus*, pages 121–136, 1980.