Joe Kilian (Ed.)

LNCS 3378

Theory of Cryptography

Second Theory of Cryptography Conference, TCC 2005
Cambridge, MA, USA, February 2005
Proceedings

Springer

Joe Kilian (Ed.)

Theory of Cryptography

Second Theory of Cryptography Conference, TCC 2005
Cambridge, MA, USA, February 10-12, 2005
Proceedings

Springer

Volume Editor

Joe Kilian
Yianilos Labs
707 State Rd., Rt. 206, Suite 212, Princeton, NJ 08540, USA
E-mail: joe@pnylab.com

Library of Congress Control Number: 2005920136

CR Subject Classification (1998): E.3, F.2.1-2, C.2.0, G, D.4.6, K.4.1, K.4.3, K.6.5

ISSN 0302-9743
ISBN 3-540-24573-1 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11390305 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3378

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

Lecture Notes in Computer Science

For information about Vols. 1–3290

please contact your bookseller or Springer

Vol. 3412: X. Franch, D. Port (Eds.), COTS-Based Software Systems. XVI, 312 pages. 2005.

Vol. 3406: A. Gelbukh (Ed.), Computational Linguistics and Intelligent Text Processing. XVII, 829 pages. 2005.

Vol. 3403: B. Ganter, R. Godin (Eds.), Formal Concept Analysis. XI, 419 pages. 2005. (Subseries LNAI).

Vol. 3398: D.-K. Baik (Ed.), Systems Modeling and Simulation: Theory and Applications. XIV, 733 pages. 2005. (Subseries LNAI).

Vol. 3397: T.G. Kim (Ed.), Artificial Intelligence and Simulation. XV, 711 pages. 2005. (Subseries LNAI).

Vol. 3391: C. Kim (Ed.), Information Networking. XVII, 936 pages. 2005.

Vol. 3388: J. Lagergren (Ed.), Comparative Genomics. VIII, 133 pages. 2005. (Subseries LNBI).

Vol. 3387: J. Cardoso, A. Sheth (Eds.), Semantic Web Services and Web Process Composition. VIII, 148 pages. 2005.

Vol. 3386: S. Vaudenay (Ed.), Public Key Cryptography - PKC 2005. IX, 436 pages. 2005.

Vol. 3385: R. Cousot (Ed.), Verification, Model Checking, and Abstract Interpretation. XII, 483 pages. 2005.

Vol. 3382: J. Odell, P. Giorgini, J.P. Müller (Eds.), Agent-Oriented Software Engineering V. X, 239 pages. 2004.

Vol. 3381: P. Vojtáš, M. Bieliková, B. Charron-Bost, O. Sýkora (Eds.), SOFSEM 2005: Theory and Practice of Computer Science. XV, 448 pages. 2005.

Vol. 3379: M. Hemmje, C. Niederee, T. Risse (Eds.), From Integrated Publication and Information Systems to Information and Knowledge Environments. XXIII, 321 pages. 2005.

Vol. 3378: J. Kilian (Ed.), Theory of Cryptography. XII, 621 pages. 2005.

Vol. 3376: A. Menezes (Ed.), Topics in Cryptology – CT-RSA 2005. X, 385 pages. 2004.

Vol. 3375: M.A. Marsan, G. Bianchi, M. Listanti, M. Meo (Eds.), Quality of Service in Multiservice IP Networks. XIII, 656 pages. 2005.

Vol. 3368: L. Paletta, J.K. Tsotsos, E. Rome, G. Humphreys (Eds.), Attention and Performance in Computational Vision. VIII, 231 pages. 2005.

Vol. 3366: I. Rahwan, P. Moraitis, C. Reed (Eds.), Argumentation in Multi-Agent Systems. XII, 263 pages. 2005. (Subseries LNAI).

Vol. 3363: T. Eiter, L. Libkin (Eds.), Database Theory - ICDT 2005. XI, 413 pages. 2004.

Vol. 3362: G. Barthe, L. Burdy, M. Huisman, J.-L. Lanet, T. Muntean (Eds.), Construction and Analysis of Safe, Secure, and Interoperable Smart Devices. IX, 257 pages. 2005.

Vol. 3361: S. Bengio, H. Bourlard (Eds.), Machine Learning for Multimodal Interaction. XII, 362 pages. 2005.

Vol. 3360: S. Spaccapietra, E. Bertino, S. Jajodia, R. King, D. McLeod, M.E. Orlowska, L. Strous (Eds.), Journal on Data Semantics II. XI, 223 pages. 2004.

Vol. 3359: G. Grieser, Y. Tanaka (Eds.), Intuitive Human Interfaces for Organizing and Accessing Intellectual Assets. XIV, 257 pages. 2005. (Subseries LNAI).

Vol. 3358: J. Cao, L.T. Yang, M. Guo, F. Lau (Eds.), Parallel and Distributed Processing and Applications. XXIV, 1058 pages. 2004.

Vol. 3357: H. Handschuh, M.A. Hasan (Eds.), Selected Areas in Cryptography. XI, 354 pages. 2004.

Vol. 3356: G. Das, V.P. Gulati (Eds.), Intelligent Information Technology. XII, 428 pages. 2004.

Vol. 3355: R. Murray-Smith, R. Shorten (Eds.), Switching and Learning in Feedback Systems. X, 343 pages. 2005.

Vol. 3353: J. Hromkovič, M. Nagl, B. Westfechtel (Eds.), Graph-Theoretic Concepts in Computer Science. XI, 404 pages. 2004.

Vol. 3352: C. Blundo, S. Cimato (Eds.), Security in Communication Networks. XI, 381 pages. 2004.

Vol. 3350: M. Hermenegildo, D. Cabeza (Eds.), Practical Aspects of Declarative Languages. VIII, 269 pages. 2005.

Vol. 3349: B.M. Chapman (Ed.), Shared Memory Parallel Programming with Open MP. X, 149 pages. 2005.

Vol. 3348: A. Canteaut, K. Viswanathan (Eds.), Progress in Cryptology - INDOCRYPT 2004. XIV, 431 pages. 2004.

Vol. 3347: R.K. Ghosh, H. Mohanty (Eds.), Distributed Computing and Internet Technology. XX, 472 pages. 2004.

Vol. 3346: R.H. Bordini, M. Dastani, J. Dix, A.E.F. Seghrouchni (Eds.), Programming Multi-Agent Systems. XIV, 249 pages. 2005. (Subseries LNAI).

Vol. 3345: Y. Cai (Ed.), Ambient Intelligence for Scientific Discovery. XII, 311 pages. 2005. (Subseries LNAI).

Vol. 3344: J. Malenfant, B.M. Østvold (Eds.), Object-Oriented Technology. ECOOP 2004 Workshop Reader. VIII, 215 pages. 2004.

Vol. 3342: E. Şahin, W.M. Spears (Eds.), Swarm Robotics. IX, 175 pages. 2004.

Vol. 3341: R. Fleischer, G. Trippen (Eds.), Algorithms and Computation. XVII, 935 pages. 2004.

Vol. 3340: C.S. Calude, E. Calude, M.J. Dinneen (Eds.), Developments in Language Theory. XI, 431 pages. 2004.

Vol. 3339: G.I. Webb, X. Yu (Eds.), AI 2004: Advances in Artificial Intelligence. XXII, 1272 pages. 2004. (Subseries LNAI).

Vol. 3338: S.Z. Li, J. Lai, T. Tan, G. Feng, Y. Wang (Eds.), Advances in Biometric Person Authentication. XVIII, 699 pages. 2004.

Vol. 3337: J.M. Barreiro, F. Martin-Sanchez, V. Maojo, F. Sanz (Eds.), Biological and Medical Data Analysis. XI, 508 pages. 2004.

Vol. 3336: D. Karagiannis, U. Reimer (Eds.), Practical Aspects of Knowledge Management. X, 523 pages. 2004. (Subseries LNAI).

Vol. 3335: M. Malek, M. Reitenspieß, J. Kaiser (Eds.), Service Availability. X, 213 pages. 2005.

Vol. 3334: Z. Chen, H. Chen, Q. Miao, Y. Fu, E. Fox, E.-p. Lim (Eds.), Digital Libraries: International Collaboration and Cross-Fertilization. XX, 690 pages. 2004.

Vol. 3333: K. Aizawa, Y. Nakamura, S. Satoh (Eds.), Advances in Multimedia Information Processing - PCM 2004, Part III. XXXV, 785 pages. 2004.

Vol. 3332: K. Aizawa, Y. Nakamura, S. Satoh (Eds.), Advances in Multimedia Information Processing - PCM 2004, Part II. XXXVI, 1051 pages. 2004.

Vol. 3331: K. Aizawa, Y. Nakamura, S. Satoh (Eds.), Advances in Multimedia Information Processing - PCM 2004, Part I. XXXVI, 667 pages. 2004.

Vol. 3330: J. Akiyama, E.T. Baskoro, M. Kano (Eds.), Combinatorial Geometry and Graph Theory. VIII, 227 pages. 2005.

Vol. 3329: P.J. Lee (Ed.), Advances in Cryptology - ASIACRYPT 2004. XVI, 546 pages. 2004.

Vol. 3328: K. Lodaya, M. Mahajan (Eds.), FSTTCS 2004: Foundations of Software Technology and Theoretical Computer Science. XVI, 532 pages. 2004.

Vol. 3327: Y. Shi, W. Xu, Z. Chen (Eds.), Data Mining and Knowledge Management. XIII, 263 pages. 2004. (Subseries LNAI).

Vol. 3326: A. Sen, N. Das, S.K. Das, B.P. Sinha (Eds.), Distributed Computing - IWDC 2004. XIX, 546 pages. 2004.

Vol. 3323: G. Antoniou, H. Boley (Eds.), Rules and Rule Markup Languages for the Semantic Web. X, 215 pages. 2004.

Vol. 3322: R. Klette, J. Žunić (Eds.), Combinatorial Image Analysis. XII, 760 pages. 2004.

Vol. 3321: M.J. Maher (Ed.), Advances in Computer Science - ASIAN 2004. XII, 510 pages. 2004.

Vol. 3320: K.-M. Liew, H. Shen, S. See, W. Cai (Eds.), Parallel and Distributed Computing: Applications and Technologies. XXIV, 891 pages. 2004.

Vol. 3319: D. Amyot, A.W. Williams (Eds.), Telecommunications and beyond: Modeling and Analysis of Reactive, Distributed, and Real-Time Systems. XII, 301 pages. 2005.

Vol. 3318: E. Eskin, C. Workman (Eds.), Regulatory Genomics. VIII, 115 pages. 2005. (Subseries LNBI).

Vol. 3317: M. Domaratzki, A. Okhotin, K. Salomaa, S. Yu (Eds.), Implementation and Application of Automata. XII, 336 pages. 2005.

Vol. 3316: N.R. Pal, N.K. Kasabov, R.K. Mudi, S. Pal, S.K. Parui (Eds.), Neural Information Processing. XXX, 1368 pages. 2004.

Vol. 3315: C. Lemaître, C.A. Reyes, J.A. González (Eds.), Advances in Artificial Intelligence – IBERAMIA 2004. XX, 987 pages. 2004. (Subseries LNAI).

Vol. 3314: J. Zhang, J.-H. He, Y. Fu (Eds.), Computational and Information Science. XXIV, 1259 pages. 2004.

Vol. 3313: C. Castelluccia, H. Hartenstein, C. Paar, D. Westhoff (Eds.), Security in Ad-hoc and Sensor Networks. VIII, 231 pages. 2004.

Vol. 3312: A.J. Hu, A.K. Martin (Eds.), Formal Methods in Computer-Aided Design. XI, 445 pages. 2004.

Vol. 3311: V. Roca, F. Rousseau (Eds.), Interactive Multimedia and Next Generation Networks. XIII, 287 pages. 2004.

Vol. 3310: U.K. Wiil (Ed.), Computer Music Modeling and Retrieval. XI, 371 pages. 2005.

Vol. 3309: C.-H. Chi, K.-Y. Lam (Eds.), Content Computing. XII, 510 pages. 2004.

Vol. 3308: J. Davies, W. Schulte, M. Barnett (Eds.), Formal Methods and Software Engineering. XIII, 500 pages. 2004.

Vol. 3307: C. Bussler, S.-k. Hong, W. Jun, R. Kaschek, D.. Kinshuk, S. Krishnaswamy, S.W. Loke, D. Oberle, D. Richards, A. Sharma, Y. Sure, B. Thalheim (Eds.), Web Information Systems – WISE 2004 Workshops. XV, 277 pages. 2004.

Vol. 3306: X. Zhou, S. Su, M.P. Papazoglou, M.E. Orlowska, K.G. Jeffery (Eds.), Web Information Systems – WISE 2004. XVII, 745 pages. 2004.

Vol. 3305: P.M.A. Sloot, B. Chopard, A.G. Hoekstra (Eds.), Cellular Automata. XV, 883 pages. 2004.

Vol. 3303: J.A. López, E. Benfenati, W. Dubitzky (Eds.), Knowledge Exploration in Life Science Informatics. X, 249 pages. 2004. (Subseries LNAI).

Vol. 3302: W.-N. Chin (Ed.), Programming Languages and Systems. XIII, 453 pages. 2004.

Vol. 3300: L. Bertossi, A. Hunter, T. Schaub (Eds.), Inconsistency Tolerance. VII, 295 pages. 2005.

Vol. 3299: F. Wang (Ed.), Automated Technology for Verification and Analysis. XII, 506 pages. 2004.

Vol. 3298: S.A. McIlraith, D. Plexousakis, F. van Harmelen (Eds.), The Semantic Web – ISWC 2004. XXI, 841 pages. 2004.

Vol. 3296: L. Bougé, V.K. Prasanna (Eds.), High Performance Computing - HiPC 2004. XXV, 530 pages. 2004.

Vol. 3295: P. Markopoulos, B. Eggen, E. Aarts, J.L. Crowley (Eds.), Ambient Intelligence. XIII, 388 pages. 2004.

Vol. 3294: C.N. Dean, R.T. Boute (Eds.), Teaching Formal Methods. X, 249 pages. 2004.

Vol. 3293: C.-H. Chi, M. van Steen, C. Wills (Eds.), Web Content Caching and Distribution. IX, 283 pages. 2004.

Vol. 3292: R. Meersman, Z. Tari, A. Corsaro (Eds.), On the Move to Meaningful Internet Systems 2004: OTM 2004 Workshops. XXIII, 885 pages. 2004.

Vol. 3291: R. Meersman, Z. Tari (Eds.), On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE, Part II. XXV, 824 pages. 2004.

Preface

TCC 2005, the 2nd Annual Theory of Cryptography Conference, was held in Cambridge, Massachusetts, on February 10–12, 2005. The conference received 84 submissions, of which the program committee selected 32 for presentation. These proceedings contain the revised versions of the submissions that were presented at the conference. These revisions have not been checked for correctness, and the authors bear full responsibility for the contents of their papers.

The conference program also included a panel discussion on the future of theoretical cryptography and its relationship to the real world (whatever that is). It also included the traditional "rump session," featuring short, informal talks on late-breaking research news.

Much as hatters of old faced mercury-induced neurological damage as an occupational hazard, computer scientists will on rare occasion be afflicted with egocentrism, probably due to prolonged CRT exposure. Thus, you must view with pity and not contempt my unalloyed elation at having my name on the front cover of this LNCS volume, and my deep-seated conviction that I fully deserve the fame and riches that will surely come of it. However, having in recent years switched over to an LCD monitor, I would like to acknowledge some of the many who contributed to this conference.

First thanks are due to the many researchers from all over the world who submitted their work to this conference. Lacking shrimp and chocolate-covered strawberries, TCC has to work hard to be a good conference. As a community, I think we have.

Shafi Goldwasser, the general chair, and Joanne Talbot Hanley, her administrative assistant, went far beyond the call of duty in their support for this conference. It is a matter of debate whether temporary insanity is a prerequisite for volunteering to be general chair, or a consequence. But, certainly, volunteering twice consecutively qualifies one for academic sainthood, if not martyr status. I wish them both several months of well-deserved peace and quiet.

Evaluating submissions requires deep knowledge of the literature, razor-sharp analytical skills, impeccable taste, wisdom and common sense. For my part, I have some pretty good Python scripts. The rest was filled in by my committee. I picked twelve people, and every last one of them did a great job. That just doesn't happen any more, not even in the movies. They supported me far more than I led them.

Like everyone else these days, we outsourced. Our deliberations benefited greatly from the expertise of the many outside reviewers who assisted us in our deliberations. My thanks to all those listed in the following pages, and my thanks and apologies to any I have missed.

I have had the pleasure of working with our publisher, Springer, and in particular with Alfred Hofmann, Ursula Barth, and Erika Siebert-Cole. Although this was my second time working with Springer, I am sure I have not lost my

amateur status. It is wrong to prejudge based on nationality, so forgive me, but I did sleep easier knowing that in Germany people spell "Kilian" correctly.

I am grateful to Mihir Bellare, the steering committee chair, and the steering committee in general for making this conference possible.

The time I spent on this project was graciously donated by my places of employment and by my family. I thank NEC and Peter Yianilos for their support and understanding. I thank Dina, Gersh and Pearl for their support, understanding and love.

Finally, I wish to acknowledge the lives and careers of Shimon Even and Larry Stockmeyer, who left us much too soon. Looking at my own work, I can point to specific papers and research directions where their influence is direct. On a deeper level, both shaped their fields by their work and by their interactions with others. Many are their heirs without knowing it. Thank you.

December 2004

Joe Kilian
Program Chair
TCC 2005

TCC 2005

February 10–12, 2005, Cambridge, Massachusetts, USA

General Chair

Shafi Goldwasser, Massachusetts Institute of Technology, USA
Weizmann Institute, Israel
Administrative Assistant: Joanne Talbot Hanley

Program Committee

Boaz Barak ..... IAS and Princeton University, USA
Amos Beimel ..... Ben-Gurion University, Israel
Rosario Gennaro ..... IBM, USA
Joe Kilian (Chair) ..... Yianilos Labs, USA
Anna Lysyanskaya ..... Brown University, USA
Tal Malkin ..... Columbia University, USA
Rafail Ostrovsky ..... UCLA, USA
Erez Petrank ..... Technion Institute, Israel
Tal Rabin ..... IBM, USA
Leonid Reyzin ..... Boston University, USA
Alon Rosen ..... MIT, USA
Amit Sahai ..... UCLA, USA
Louis Salvail ..... Aarhus University, Denmark

Steering Committee

Mihir Bellare (Chair) (UCSD, USA), Ivan Damgård (Aarhus University, Denmark), Oded Goldreich (Weizmann Institute, Israel), Shafi Goldwasser (MIT, USA and Weizmann Institute, Israel), Johan Håstad (Royal Institute of Technology, Sweden), Russell Impagliazzo (UCSD, USA), Ueli Maurer (ETHZ, Switzerland), Silvio Micali (MIT, USA), Moni Naor (Weizmann Institute, Israel), Tatsuaki Okamoto (NTT, Japan)

Table of Contents

Steganography and Zero Knowledge

Secure Computation I

Secure Computation II

Quantum Cryptography and Universal Composability

Cryptographic Primitives and Security

Encryption and Signatures

Information Theoretic Cryptography

Optimal Error Correction Against Computationally Bounded Noise

Silvio Micali, Chris Peikert, Madhu Sudan, and David A. Wilson

MIT CSAIL, 77 Massachusetts Ave,
Building 32, Cambridge, MA, 02139
{silvio, cpeikert, madhu, dwilson}@mit.edu

Abstract. For computationally bounded adversarial models of error, we construct appealingly simple, efficient, cryptographic encoding and unique decoding schemes whose error-correction capability is much greater than classically possible. In particular:

1. *For binary alphabets, we construct positive-rate coding schemes which are uniquely decodable from a* $1/2 - \gamma$ *error rate for any constant* $\gamma > 0$.
2. *For large alphabets, we construct coding schemes which are uniquely decodable from a* $1 - \sqrt{R}$ *error rate for any information rate* $R > 0$.

Our results are qualitatively stronger than related work: the construction works in the public-key model (requiring no shared secret key or joint local state) and allows the channel to know everything that the receiver knows. In addition, our techniques can potentially be used to construct coding schemes that have information rates approaching the Shannon limit. Finally, our construction is qualitatively optimal: we show that unique decoding under high error rates is *impossible* in several natural relaxations of our model.

1 Introduction

The theory of error correction is concerned with sending information reliably over a "noisy channel" that introduces errors into the transmitted data. In this setting, a *sender* starts with some *message*, which is a fixed-length string of symbols over some alphabet. The sender *encodes* the message into a longer string over the same alphabet, then transmits the block of data over a *channel*. The channel introduces *errors* (or *noise*) by changing some of the symbols of the transmitted block, then delivers the corrupted block to the *recipient*. Finally, the recipient *decodes* the block (hopefully to the intended message). Whenever the sender wants to transmit a new message, the process is repeated.

Two quantities are of special interest in this setting: the *information rate* (i.e., the ratio of the message length to the encoded block length) and the *error rate* (i.e., the ratio of the number of errors to the block length). Coding schemes having high information rate and tolerating high error rate are, of course, the

J. Kilian (Ed.): TCC 2005, LNCS 3378, pp. 1–16, 2005.

most desirable. Small alphabets are desirable too, and in particular most natural channels are indeed best at transmitting only bits.

But the question remains: *how should we model a noisy channel?*

Standard Channels. There are two historically popular ways to model a noisy channel. Shannon's *symmetric channel* independently changes each symbol to a random different one, with some fixed probability. Hamming's *adversarial channel* changes symbols in a worst-case fashion, subject only to an upper bound on the number of errors per block of data. In particular — though this is not often stated explicitly — the adversarial channel is computationally *unbounded.* Working with this "pessimistic" model certainly ensures the robustness of the resulting coding scheme, but it also severely restricts the information and error rates. For instance, when the alphabet is binary, the error rate must be less than 1/4 for unique decoding to be possible (unless the blocks are exponentially longer than the messages).

One way to recover from a higher error rate is to relax the task of decoder, allowing it to output a short *list* of messages which contains the intended one. To tolerate adversarial channels with high error rates, list decoding seems to be the best one can do — but under a more *"realistic"* model of an adversarial channel, is it possible to *uniquely* decode under high error rates?

Computationally Bounded Channels. In 1994, Lipton [9] put forward the notion of a computationally bounded channel, which is essentially a Hamming channel restricted to *feasible* computation. That is, the channel still introduces errors adversarially (always subject to a given error rate), but must do so in time polynomial in the block length.

We posit that natural processes can be implemented by efficient computation, so *all* real-world channels are, in fact, computationally bounded. We therefore have confidence that results in this model will be as meaningful and applicable as classical codes. Indeed, the nature of the model is such that if some malicious (or natural!) process is capable of causing incorrect decoding, then that process can be efficiently harnessed to break standard hardness assumptions. In contrast to coding schemes which are only guaranteed to work against channels that are modelled by *very specific, limited* probabilistic processes, results in this model apply *universally* to any channel which can be modelled by efficient computation.

Remarkably, under standard cryptographic assumptions and assuming that sender and receiver share secret randomness, Gopalan, Lipton, and Ding [3] proved that for such a bounded channel, it is possible to decode correctly from higher error rates. Unfortunately, their result requires the communicating parties to share a secret key which is unknown to the channel.

More significantly, though the bounded-channel model was first envisioned over a decade ago, nobody has yet shown an *essential* use of this assumption to yield any unique benefits over an unbounded channel. That is, previous constructions still work when the channel is computationally unbounded, as long as the sender and receiver share some secret randomness. The bounded-channel

assumption is used to reduce the *amount* of shared randomness that is needed, but not to eliminate it altogether. This computational assumption is thus an *additional* one, and does not supplant the assumption of secret randomness shared between the sender and receiver.

Our goal is to provide a general method for optimal error correction, exploiting the bounded-channel assumption in an essential way.

1.1 Our Contributions

Our Setting. We work in a very simple cryptographic setting: we assume that a one-way function exists (the "minimal" cryptographic assumption) and that the sender has a public key known to the receiver (and, perhaps, to the channel as well).

The sender (but *not* the receiver) keeps a small amount of state information, which he uses when encoding messages. Because the sender keeps state, our constructions are actually *dynamic* coding schemes, in which the same message results in a different encoding each time it is sent.

Our Results. Our setting yields great benefits in error correction for both binary and large alphabets. Namely,

1. *For binary alphabets, we construct positive-rate dynamic coding schemes which are uniquely decodable from a* $1/2 - \gamma$ *error rate for any constant* $\gamma > 0$.
   Classically, a $1/4 - \gamma$ error rate is the best possible for unique decoding (and positive information rate). We stress that *in any reasonable model*, decoding of *any kind* (even list decoding) is impossible under an error rate of $1/2$. Therefore this result is optimal in a very strong sense, and matches the best possible error rates in the weaker Shannon model.
2. *For large alphabets, we construct dynamic coding schemes which are uniquely decodable from a* $1 - \sqrt{R}$ *error rate for any information rate* $R > 0$.
   The $1 - \sqrt{R}$ error rate is actually a consequence of known list decoding algorithms, and not imposed by our technique. Note that when $R < 1/4$, we can uniquely decode from error rates much greater than $1/2$, which is impossible in the Hamming model.

To achieve these results, we actually prove a very general *reduction*, namely,

> If one-way functions exist, *(dynamic) unique decoding* from e errors in the bounded-channel model reduces to efficient *(static) list decoding* from e errors in the Hamming model (with no asymptotic loss in information rate).

We obtain results 1 and 2 above by applying this reduction to the classical Guruswami-Sudan [7] and Reed-Solomon codes.

Optimality of Our Model. There are three defining characteristics of our model: (1) the sender is stateful (the amount of state required is minimal; either a single counter value or a local clock would suffice) while the receiver is stateless, (2) the sender keeps a secret key which is unknown to the channel, and (3) the channel is assumed to be computationally bounded.

We show that our model is qualitatively optimal: relaxing any of these three requirements makes the task of unique decoding under high error rates *impossible*. Thus our construction can be seen as the "best possible" use of the bounded-channel assumption for error correction. See Section 4.3 for details.

Overview of the Construction. Starting with any static code, we specify a *cryptographic sieving* procedure, which only certain "authentic" codewords will pass. Authentic words are hard for the adversary to compute (even after seeing other authentic codewords), but easy for the sender to generate and for the recipient to sieve out.

Upon receiving a corrupted word, the recipient first *list decodes* it. Of course, list decoding only provides a set of candidate codewords. In order to uniquely decode, the recipient next uses the cryptographic sieve to filter out only the authentic word(s). Provided that the number of errors is suitably limited, the intended codeword is guaranteed to appear in the decoded list and pass the sieve. However, it may not be alone: though the bounded channel cannot produce any *new* authentic codewords, it may be able to cause *prior* ones to appear in the decoded list. This is where the sender's state comes into play: dynamic encoding allows the receiver to choose the "freshest" word that passes the sieve, resulting (with overwhelming probability) in correct, unique decoding.

2 Related Work

We wish to contrast our results with several other models and techniques for tolerating high error rates.

2.1 List Decoding

One of the best-known methods of decoding beyond classical limits under adversarial error is known as *list decoding*. In list decoding, a received word is not decoded to a unique message, but rather to a short *list* of possible messages. If the number of errors is within the *list-decoding radius*, the original message will appear in the list.

There exist codes with rate approaching the Shannon capacity of the channel and yielding constant-size lists (cf. [5]); however, no efficient list decoding algorithms are known for such codes. Still, many popular codes have efficient list-decoding algorithms that can decode significantly beyond the half-the-distance bound.

The obvious drawback of list decoding is that one typically desires to know the *unique* message that was sent, rather than a list of possible messages. The works presented below, as well as our cryptographic sieve, use list decoding as a