

Distribution-Aware Encryption with Optimal Recovery and Semantic Security

Haijie Su¹, Haibo Cheng^{1,2}, and Ping Wang¹

¹ Peking University

suhaijie@pku.edu.cn, hbcheng@pku.edu.cn, pwang@pku.edu.cn

² Minzu University of China

Abstract. Distribution-aware encryption seeks guarantees tailored to the probability law of the protected data and the secret key. We study simultaneous message-recovery, key-recovery, and target-distribution semantic security for arbitrary finite joint distributions, without assuming that messages and keys are independent. We represent encryption by a decomposition of the joint probability matrix and construct this decomposition through capacity-controlled randomized rounding. A stepwise conditional sampler turns the rounding process into a correct encryption algorithm without enumerating its output distribution or rejecting complete samples. Let $p_\star$ and $q_\star$ be the largest message and key probabilities. Our construction attains the exact message-recovery optimum $\max\{p_\star, q_\star\}$, has key-recovery success below $2q_\star$, and provides $O(\sqrt{q_\star})$ advantage against prediction of every fixed Boolean message predicate. The semantic guarantee follows from negative correlation of the binary row-rounding remainders. We prove that the key-recovery factor cannot be uniformly improved while preserving exact message-recovery optimality, and that the semantic bound has the optimal worst-case order. We also establish additive robustness under distribution error and short-ciphertext existence by preserving only the moments needed for the security proof. The general sampler runs in expected polynomial time in the explicit rational input size; short ciphertexts do not by themselves imply efficient preprocessing or compact public descriptions. Structured sources admit additional efficient implementations.

Keywords: Distribution-aware encryption · Message recovery · Semantic security · Dependent rounding.

1 Introduction

Modern encryption is usually analyzed against computationally bounded adversaries, but several settings call for a more distribution-aware information-theoretic view. Long-lived secrets may need protection against future computational advances, and password-based systems already expose an offline guessing surface in which an adversary can test likely keys and discard implausible decryptions. Similar issues arise for PINs, biometric-derived keys, and other source-generated secrets: the key material may be non-uniform, predictable, or reconstructed from noisy measurements [13]. In such settings, the effective security of an encryption scheme is shaped not only by its algebraic structure, but also by the prior distributions of the messages and keys it protects.

Shannon’s theory of secrecy systems gives the cleanest information-theoretic ideal: a ciphertext should reveal no information about the message [27]. The cost of this ideal is equally clear. Perfect secrecy requires key uncertainty comparable to message uncertainty, which is often unrealistic when secrets are protected by user-chosen passwords or other low-entropy key material.

Later work on entropic security relaxed the security requirement [25,14]. Instead of demanding perfect secrecy for every message distribution, it asks whether short keys can protect high-min-entropy messages against unbounded adversaries who try to learn predicates of the message. These results provide a security framework below the one-time-pad regime and clarify how distributional assumptions should guide encryption design, but they do not by themselves strengthen encryption for sources whose effective entropy is already too small.

Honey encryption takes a more active approach to the low-entropy setting. A conventional ciphertext with a recognizable plaintext space lets an offline attacker reject guessed keys once keys can be enumerated: wrong keys tend to decrypt to invalid-looking plaintexts. For long-lived ciphertexts, such enumeration may become realistic as computational assumptions age; for password-based encryption, low-entropy user-chosen keys can make enumeration effective already today. Honey encryption changes the encryption mechanism so that wrong keys decrypt to plausible decoys, making it harder to tell whether a guessed key is correct [20]. The quality of the distribution model is therefore central: poorly modeled decoys may again become distinguishable from real plaintexts [17].

The canonical construction of Juels and Ristenpart realizes this idea through a distribution-transforming encoder driven by the message prior, together with a password-based encryption layer. This makes the message distribution central. Although its security analysis accounts for non-uniform keys, the full joint message-key distribution is not the object of the construction.

Honey-encryption security distinguishes complete recovery from partial leakage. *Message-recovery security* (MRS) measures the success probability of recovering the complete message after seeing one ciphertext [20]; the analogous *key-recovery security* (KRS) objective concerns recovery of the complete key. Small recovery probabilities can nevertheless coexist with substantial partial leakage: many messages may remain plausible even when the ciphertext reveals whether the message belongs to a sensitive category. Jaeger, Ristenpart, and Tang [19] address this gap through *target-distribution semantic security* (TDSS), with further information-theoretic and repeated-encryption results due to Li, Tang, and Zhang [22]. For a Boolean message property fixed before encryption, TDSS measures the improvement in prediction success after seeing a ciphertext over the best prediction using the prior alone. Recovery resistance and partial-information protection are therefore complementary goals.

This motivates considering two distributional effects jointly. First, in password-based encryption the key itself is sampled from a highly non-uniform password distribution [8,30,23,24,29], so likely keys are an additional attack surface. A scheme that accounts for the message prior but ignores key-side probabilities may fail to reach the best possible one-shot security. Second, the protected data and the key material may be statistically related. Unlike standard key-dependent-message security, which studies

messages of the form $M = f(K)$ for adversarially specified function classes [5,7,2], our model takes a prior-level joint distribution $Q(M, K)$ as the object of design and analysis.

Password vaults make this dependence concrete. A vault is encrypted under a user-chosen master password, while the protected object consists of the collection of account passwords; therefore the relation between the master password and the vault contents is part of the object being modeled. Both sides may contain related names, languages, dates, words, or numerical suffixes chosen by the same user [12]. Honey-vault systems model the probability of an entire vault as a structured password sequence: early natural-language encoders used n -gram and PCFG models [9,17], while more recent work studies updateable and practically secure honey vaults with stronger learned password models [11,10]. The same modeling idea can be converted into a joint distribution by including the master password as the first component of the sequence and the vault contents as the remaining components. Such a model can represent dependence,

$$p_{M,K}(m, k) \neq p_M(m)p_K(k),$$

so an independent message-key model would discard part of the security-relevant prior information. Similar motivations arise in other low-entropy or long-term protection settings, including password-based encryption and genomic-data protection [18].

This paper develops distribution-aware encryption for every finite, publicly specified joint distribution of messages and keys. The key construction is a randomized, capacity-controlled decomposition of the joint probability matrix. An expectation-preserving cycle process generates a decryption pattern describing the candidate message-key pairs for a ciphertext. Conditioning this process on the true input turns the decomposition into an encryption algorithm, without first enumerating all possible patterns. This approach adopts the security framework of honey encryption without relying on its distribution-transforming-encoder architecture; in particular, some wrong keys may decrypt to $\perp$.

The construction simultaneously attains the information-theoretic optimum for message recovery, keeps key-recovery success within a factor of two of the best prior key guess, and protects every fixed Boolean message property through a TDSS bound of optimal worst-case order. The latter is a worst-case guarantee across sources, not a claim of TDSS optimality for each individual source. These results hold for dependent as well as independent sources and concern one challenge ciphertext against an unbounded adversary. We also quantify robustness to source-estimation error and distinguish the existence of short ciphertexts from the costs of implementing the construction.

1.1 Contributions

Write n_m and n_k for the message and key space sizes, and $p_\star$ and $q_\star$ for their largest probabilities. Let $L = \max\{p_\star, q_\star\}$. Every correct scheme has message-recovery success at least L : an observer can guess the most likely message, or guess the most likely key and decrypt. Key-recovery success is at least $q_\star$, not necessarily L . Following the advantage notation of [19], write Adv^{mr} , Adv^{kr} , and Adv^{tdss} for the optimal adversary's recovery probabilities and TDSS advantage, respectively; Section 2 gives the formal definitions. Our contributions are threefold.

1. *Simultaneous security for arbitrary joint sources.* For every finite joint distribution, we construct a correct scheme satisfying

$$\begin{aligned} \text{Adv}^{\text{mr}} &= L, & \text{Adv}^{\text{kr}} &= \beta_* < 2q_*, \\ \text{Adv}^{\text{tdss}} &\leq \min\{1/2, \sqrt{\beta_*/2}\} = O(\sqrt{q_*}). \end{aligned}$$

Here β_* is an explicit function of the two largest message probabilities and q_* (Theorem 4). Thus MRS is exactly optimal, KRS is within a factor of two of its prior lower bound, and every fixed Boolean message property receives a semantic guarantee. The same construction covers dependent and independent sources. We also show that the KRS factor cannot be uniformly improved while preserving MRS optimality, and that the TDSS bound has optimal worst-case order (Section 4); these are not claims of per-source KRS or TDSS optimality.

2. *Robustness to distribution-estimation error.* The security analysis remains useful when the supplied distribution is only an estimate. For a fixed scheme designed using $\widehat{Q}$, replacing the source by Q changes each recovery probability by at most $\delta = \text{TV}(Q, \widehat{Q})$, and TDSS by at most $\delta + \text{TV}(Q_M, \widehat{Q}_M) \leq 2\delta$, where $\widehat{Q}_M, Q_M$ are the message marginals. These bounds quantify how error in the joint model affects security under the true source, without requiring the scheme’s structural properties to hold for that source (Section 5).
3. *Constructive realization and implementation costs.* We turn the joint-source matrix decomposition into explicit encryption and decryption algorithms. Encryption follows a randomized rounding path conditioned on the true input, without enumerating every possible output. For an explicitly listed rational source, the algorithm runs in expected polynomial time in the input size. Its matrix-valued ciphertext, however, can be large to transmit. We show that the same security guarantees can also be realized with ciphertext indices of $O(\log(n_m + n_k))$ bits, while making clear that the required public table and preprocessing may still be large. For specified structured sources, both independent and correlated, we additionally give efficient algorithms with compact descriptions (Appendix E). This separates the general construction, the existence of small ciphertexts, and the stronger efficiency available under additional source structure.

All guarantees are one-shot: there is one challenge ciphertext, and the predicate is fixed before its fresh randomness is sampled. An unbounded adversary may examine every key internally; the restriction is on the challenge and final answer, not its computation. The guarantees do not extend automatically to chosen-plaintext attacks, arbitrary auxiliary information, or key reuse. Likewise, the public source is supplied rather than learned for free, and polynomial time in its explicit probability matrix need not be polynomial in message and key bit lengths.

1.2 Related Work

Information-theoretic and distribution-dependent encryption. Unlike perfect secrecy [27], our target-distribution experiment permits bounded improvement over a particular prior. Unlike entropic security [25, 14], the construction is tailored to a specified joint law

rather than providing one scheme for a class of sources satisfying an entropy threshold. Source assumptions are also central to deterministic and efficiently searchable encryption [3], which pursues computational guarantees and different functionality.

Honey encryption and semantic security. The progression from recovery security [20] to semantic security [19] provides our closest security framework. Li, Tang, and Zhang [22] already obtain square-root-in- $q_\star$ single-message bounds for independent sources, as well as studying repeated encryption. The present contribution is therefore not the TDSS notion or that asymptotic scale in isolation, but their combination with exact MRS optimality and near-prior KRS for arbitrary joint sources. The conditional-rounding construction and sharp recovery tradeoffs explain how these guarantees can be attained together.

Honey vaults, source models, and stronger attacks. Decoy-based password defenses predate honey encryption: Kamouflage [6] uses decoys in password management, while Honeywords [21] aims to detect the use of compromised passwords. Honeywords targets attack detection rather than the ciphertext-only inference guarantees studied here. Building on the vault setting [9,17], later research addresses incremental updates [11], analyzes practical distinguishing and guessing attacks [15], and develops new designs and online-guessing evaluations [10]. These works motivate both our robustness analysis and the distinction between one-ciphertext prediction and attacks using vault snapshots, known passwords, or online validation. Our generic construction is a theoretical source-dependent scheme, not a complete vault system addressing all these attacks.

Joint priors versus key-dependent-message security. Key-dependent-message (KDM) security typically studies encryptions of adversarially selected functions of secret keys from a specified class [5]. We instead fix a public joint prior and allow the scheme to depend on it. Although deterministic dependence $M = f(K)$ is a special source in our model, the different quantifiers and adversarial capabilities prevent interpreting our result as a KDM guarantee.

Guessing measures and combinatorial tools. MRS and KRS are Bayesian guessing probabilities, related to Bayes vulnerability and quantitative information flow [28,1]. TDSS addresses partial information not captured by these complete-recovery objectives, using a predicate-specific prior baseline. On the construction side, matrix and matching decompositions are classical [4,26], as is expectation-preserving dependent rounding [16]. We do not attribute novelty to these tools. Their role here is to connect a joint-source encryption decomposition to input-conditioned sampling and to the candidate-count properties needed by all three security objectives.

2 Ciphertext-Indexed Characterization of Encryption

This section reduces the design of encryption for a fixed joint source Q to the construction of a matrix decomposition. We first define the security objectives and their basic limits. We then establish an equivalence: every correct scheme yields a column-unique

decomposition of Q , and, crucially, every such decomposition defines correct encryption and decryption. Its component masses also determine all three security measures. Thus, from Section 3 onward, we can focus on constructing a decomposition with the required properties; the correspondence supplies the scheme and its security interpretation.

2.1 Notation and Preliminaries

Let $\mathcal{M} = \{m_1, \dots, m_{n_m}\}$ and $\mathcal{K} = \{k_1, \dots, k_{n_k}\}$ be finite message and key spaces, with cardinalities n_m and n_k , respectively. The public joint source is a nonnegative matrix Q of total mass one, with

$$Q_{ij} = \Pr[M = m_i, K = k_j], \quad p_i = \sum_j Q_{ij}, \quad q_j = \sum_i Q_{ij}. \quad (1)$$

Thus p and q are the message and key marginals. We delete zero-mass rows and columns, but impose no ordering on the remaining labels. Write $p_\star = \max_i p_i$, $q_\star = \max_j q_j$, and $L = \max\{p_\star, q_\star\}$. The term *marginal* denotes a probability distribution; row and column sums of auxiliary matrices below need not have that interpretation.

An encryption scheme $\text{SE} = (\text{Enc}, \text{Dec})$ consists of randomized Enc and deterministic Dec , both allowed to depend on the public source. We require

$$\Pr[\text{Dec}(k_j, \text{Enc}(k_j, m_i)) = m_i] = 1 \quad \text{whenever } Q_{ij} > 0.$$

If correctness on every pair is desired, a disjoint ciphertext tag (*fallback*, i, j) can handle $Q_{ij} = 0$: it decrypts to m_i under k_j and to $\perp$ under other keys. Such ciphertexts have zero probability under the design source. They become relevant when comparing different sources in Section 5.

The construction has a finite ciphertext space. We state the matrix identities for discrete ciphertexts. A scheme with a more general ciphertext space can be coarsened by its complete decryption vector, which has at most $(n_m + 1)^{n_k}$ values including failure. Coarsening preserves correctness and cannot increase any of the observer's optimal success probabilities. Thus the existence claims and universal lower bounds do not rely on an additional restriction to finite ciphertext spaces.

2.2 Recovery and Semantic Security

The experiment draws $(M, K) \sim Q$ and outputs $C \leftarrow \text{Enc}(K, M)$. The adversary knows the source, algorithms, and public parameters. Following the advantage notation of honey encryption [19], define the message- and key-recovery advantages of adversaries A and B by

$$\begin{aligned} \text{Adv}_{\text{SE}, Q}^{\text{mr}}(A) &= \Pr[A(C) = M], \\ \text{Adv}_{\text{SE}, Q}^{\text{kr}}(B) &= \Pr[B(C) = K]. \end{aligned} \quad (2)$$

For a fixed scheme, we abbreviate their optimal values as

$$\text{Adv}_Q^{\text{mr}} = \max_A \text{Adv}_{\text{SE}, Q}^{\text{mr}}(A), \quad \text{Adv}_Q^{\text{kr}} = \max_B \text{Adv}_{\text{SE}, Q}^{\text{kr}}(B).$$

The maximizations are over computationally unbounded adversaries. Despite the term *advantage*, these recovery quantities are success probabilities, with no prior term subtracted, as in the message-recovery definition of [19]. We use the analogous convention for key recovery. One-shot refers to the challenge and final guess, not to a restriction on internal computation.

For a Boolean predicate $f : \mathcal{M} \rightarrow \{0, 1\}$, let $\theta_f = \Pr[f(M) = 1]$ and $b_f = \max\{\theta_f, 1 - \theta_f\}$. Following the TDSS prediction formulation [19,22], define

$$\text{Adv}_{\text{SE},Q}^{\text{tdss}}(A, f) = \Pr[A(C) = f(M)] - b_f. \quad (3)$$

Its optimal fixed-predicate and overall values are

$$\begin{aligned} \text{Adv}_Q^{\text{tdss}}(f) &= \max_A \text{Adv}_{\text{SE},Q}^{\text{tdss}}(A, f), \\ \text{Adv}_Q^{\text{tdss}} &= \max_f \text{Adv}_Q^{\text{tdss}}(f). \end{aligned}$$

Here f may depend on the public source and scheme, but is fixed before the challenge ciphertext and its fresh coins. Our change to the usual independent-source experiment is the joint draw $(M, K) \sim Q$. Unlike recovery, TDSS subtracts the best prior-only prediction probability. When the scheme and source are clear, we write Adv^{mr} , Adv^{kr} , Adv^{tdss} , and $\text{Adv}_Q^{\text{tdss}}(f)$ for these optimal values. When comparing sources under a fixed encryption channel W , we use W in the scheme subscript, writing $\text{Adv}_{W,Q}^{\text{mr}}$, $\text{Adv}_{W,Q}^{\text{kr}}$, and $\text{Adv}_{W,Q}^{\text{tdss}}$. Thus a channel is distinguished from the predicate argument f .

The probability averages over the true input and fresh encryption randomness, not over predicates. It is also not a guarantee for every individual ciphertext. For example, revealing a balanced predicate completely with probability $1/3$ and providing no information otherwise gives optimal prediction success $2/3$, hence advantage $1/6$. No challenge-correlated auxiliary information is present unless explicitly included in the model. The results below do not automatically extend to known-plaintext attacks or key reuse.

Basic recovery limits. Guessing the most probable message gives success $p_\star$. Guessing the most probable key and decrypting succeeds at least whenever that key is the true one, an event of probability $q_\star$. Hence every correct scheme satisfies

$$\text{Adv}_Q^{\text{mr}} \geq L, \quad \text{Adv}_Q^{\text{kr}} \geq q_\star. \quad (4)$$

More generally, any key-recovery strategy gives a message-recovery strategy by decrypting with the guessed key and returning an arbitrary message on failure. Whenever the key guess is correct, so is the message guess. Thus $\text{Adv}_Q^{\text{mr}} \geq \text{Adv}_Q^{\text{kr}}$.

There is an additional constraint if a ciphertext and a candidate message uniquely identify the candidate key. For discrete ciphertexts, call a pair a *candidate* for c when it has positive joint probability of occurring with c in the experiment; pairs of probability zero are not counted. Call the property that, for every c , each message has at most one such key *message-side uniqueness*. A message guess then identifies a unique candidate key whenever it is correct; if no candidate exists, output any key. The reverse inequality follows, giving

$$\text{message-side uniqueness} \implies \text{Adv}_Q^{\text{mr}} = \text{Adv}_Q^{\text{kr}}. \quad (5)$$

To lower KRS below L when $p_\star > q_\star$, a scheme must therefore allow several candidate keys for the same message and ciphertext. Message-side uniqueness is sufficient for equality of the recovery probabilities, not necessary; the argument does not assume equally likely candidates.

2.3 Equivalence of encryption and matrix decomposition

Representing an existing scheme by its ciphertext-indexed probability masses is natural. The important design step is the converse: the decomposition constraint alone suffices to define a correct scheme, with no additional encryption structure required. We give this conversion explicitly so that finding a suitable decomposition becomes the remaining construction problem.

From encryption to a decomposition. Fix a scheme. Encryption distributes the probability mass Q_{ij} of each input pair among the ciphertexts that this pair can produce. Collect the mass assigned to ciphertext c into a matrix A_c :

$$\begin{aligned} A_c(i, j) &= \Pr[M = m_i, K = k_j, C = c] \\ &= Q_{ij} \Pr[\text{Enc}(k_j, m_i) = c], \quad Q = \sum_c A_c. \end{aligned} \quad (6)$$

The last identity holds entry by entry, since encryption must output some ciphertext. A component A_c is generally *not* a probability distribution: its entries sum to $\Pr[C = c]$, not to one.

Correct deterministic decryption imposes a simple constraint. Once c and k_j are fixed, the decryptor can return only one message. Thus at most one row in column j of A_c can have positive mass. We call this *column uniqueness*. A row, by contrast, may contain several positive entries: different keys are allowed to decrypt the same ciphertext to the same message.

From a decomposition to encryption. Now suppose we are given only nonnegative matrices $\{A_c\}_c$ that sum to Q , each with column uniqueness. Use the component labels c as ciphertexts and make the matrices public. On an input (m_i, k_j) with $Q_{ij} > 0$, choose the label using

$$\Pr[\text{Enc}(k_j, m_i) = c] = A_c(i, j)/Q_{ij}. \quad (7)$$

These are valid probabilities because $\sum_c A_c(i, j) = Q_{ij}$. Only components containing positive mass at the true input can be chosen. To decrypt c with k_j , read column j of A_c and return the message at its unique positive entry, or $\perp$ if the column is empty. Every ciphertext produced on the true input has that entry in row i , so decryption returns m_i with probability one. Moreover, multiplying (7) by Q_{ij} recovers exactly $A_c(i, j)$.

Concretely, the ciphertext c labels a public weighted component A_c . Its positive positions form a *decryption pattern*: the message–key pairs still possible after observing c . A holder of key k_j selects the unique candidate in column j , or obtains failure if that column is empty. Encryption announces the chosen component and its pattern, without separately marking the true pair within it. Distinct weighted components can have the same pattern, so the pattern alone need not specify a general ciphertext or its

probabilities. The construction below will specialize this representation so that a binary pattern itself serves as ciphertext.

Proposition 1 (Encryption–decomposition equivalence). *On the positive support of Q , the joint laws induced by randomized encryption with deterministic correct decryption are exactly the nonnegative decompositions $Q = \sum_c A_c$ with column uniqueness. The induced encryption rule is (7).*

This characterization is a description of a scheme, not yet an efficient way to store or sample a potentially large family of components. Nor does it require that the positive entries of one component have equal weights. If a decomposition is written as $Q = \sum_c \lambda_c T_c$, simply set $A_c = \lambda_c T_c$. Unless the T_c are normalized, the coefficients λ_c need not sum to one and are not themselves ciphertext probabilities. The ciphertext probability is the total mass of A_c .

In this representation, message-side uniqueness adds at most one positive entry per row to column uniqueness. Together, these two restrictions say exactly that the decryption pattern is a matching between messages and keys. Without the additional row restriction, several keys may lead to the same message, so a decryption pattern is not generally a matching. As shown in (5), this additional row restriction forces equality of the recovery probabilities, even for unequal entry weights. It is not required for correctness.

2.4 Computing security from the components

For every positive-probability ciphertext, the total mass of its component is

$$\Pr[C = c] = \sum_{i,j} A_c(i, j).$$

The conditional, or *posterior*, probability matrix after observing c is $A_c / \Pr[C = c]$. Its row sums give the message probabilities and its column sums give the key probabilities:

$$\begin{aligned} \Pr[M = m_i \mid C = c] &= \frac{\sum_j A_c(i, j)}{\Pr[C = c]}, \\ \Pr[K = k_j \mid C = c] &= \frac{\sum_i A_c(i, j)}{\Pr[C = c]}. \end{aligned} \tag{8}$$

For message recovery, the best observer chooses the row of largest mass. Its conditional success is that row mass divided by $\Pr[C = c]$. Averaging over ciphertexts multiplies it by $\Pr[C = c]$ again, so the denominator cancels. The same argument applies to the largest column for key recovery. Consequently,

$$\text{Adv}_Q^{\text{mr}} = \sum_c \max_i \sum_j A_c(i, j), \quad \text{Adv}_Q^{\text{kr}} = \sum_c \max_j \sum_i A_c(i, j). \tag{9}$$

Thus we can evaluate recovery security directly from the decomposition, without deriving the posterior distribution separately for every input.

For a predicate f , let $S = \{i : f(m_i) = 1\}$ and write $\bar{S} = \{1, \dots, n_m\} \setminus S$. The relevant quantities are the joint probability of ciphertext c and a true predicate, and the prior probability that the predicate is true:

$$r_c(S) := \Pr[f(M) = 1, C = c] = \sum_{i \in S, j} A_c(i, j),$$

$$p(S) := \Pr[f(M) = 1] = \sum_{i \in S} p_i = \theta_f.$$

Thus $r_c(S)$ is the mass of the rows in S within component A_c , not a conditional probability; dividing by $\Pr[C = c]$ would give the posterior probability. The complementary mass $r_c(\bar{S})$ is $\Pr[f(M) = 0, C = c]$, and summing $r_c(S)$ over c gives $p(S)$. Now the observer compares two groups of rows, not two individual messages: it predicts 1 if $r_c(S) \geq r_c(\bar{S})$ and 0 otherwise. Its optimal prediction success is

$$\max_A \Pr[A(C) = f(M)] = \sum_c \max\{r_c(S), r_c(\bar{S})\}. \quad (10)$$

TDSS subtracts the success of the best prediction before seeing the ciphertext. Using $\max\{x, y\} = (x + y + |x - y|)/2$ gives the exact identity

$$\begin{aligned} \text{Adv}_Q^{\text{tdss}}(f) &= \sum_c \max\{r_c(S), r_c(\bar{S})\} - \max\{p(S), 1 - p(S)\} \\ &= \frac{1}{2} \left(\sum_c |r_c(S) - r_c(\bar{S})| - |2p(S) - 1| \right). \end{aligned} \quad (11)$$

The first line measures the improvement over the prior-only prediction; the second expresses this in terms of the mass imbalance between the two groups. A useful decomposition must therefore control more than its largest row or column: it must also control these imbalances for every fixed set S .

A worked decomposition. Consider two messages and three keys, with

$$Q = \frac{1}{12} \begin{pmatrix} 2 & 0 & 5 \\ 3 & 2 & 0 \end{pmatrix} = A_0 + A_1, \quad (12)$$

$$A_0 = \frac{1}{12} \begin{pmatrix} 2 & 0 & 3 \\ 0 & 1 & 0 \end{pmatrix}, \quad A_1 = \frac{1}{12} \begin{pmatrix} 0 & 0 & 2 \\ 3 & 1 & 0 \end{pmatrix}.$$

Both components have column uniqueness. Encrypting (m_1, k_3) outputs $c = 0$ with probability $3/5$ and $c = 1$ with probability $2/5$; encrypting (m_2, k_2) uses probabilities $1/2$ and $1/2$. The other positive-probability inputs each use their only compatible component. For $c = 0$, decryption under (k_1, k_2, k_3) returns (m_1, m_2, m_1) ; for $c = 1$, it returns (m_2, m_2, m_1) . Correctness follows directly.

Each component has total mass $1/2$, so the two ciphertexts are equally likely overall, even though their probabilities depend on the input. Their largest row masses are $5/12$ and $4/12$; their largest column masses are both $3/12$. Hence

$$\text{Adv}_Q^{\text{mr}} = \frac{3}{4}, \quad \text{Adv}_Q^{\text{kr}} = \frac{1}{2}.$$

For the predicate $f(m_1) = 1$, $f(m_2) = 0$, the prior best guess is 1, with success $7/12$. After observing $c = 0$ the best guess remains 1, whereas after $c = 1$ it becomes 0. Its success is $3/4$, giving $\text{Adv}_Q^{\text{tdss}}(f) = 3/4 - 7/12 = 1/6$. This example illustrates how to read a scheme and its security from a decomposition; it is not an optimal construction for this source.

We have therefore reduced the design problem to finding a decomposition with three properties: column uniqueness for correctness, controlled largest row and column masses for recovery, and controlled group-mass imbalances for TDSS. Section 3 constructs such a decomposition through a random matrix whose expectation recovers Q .

3 Encryption via Capacity-Controlled Random Decomposition

We now build the decomposition required by Section 2 through a random process whose outputs specify decryption patterns. One execution produces one pattern; the distribution over all executions supplies the decomposition. We then turn this process into encryption by conditioning it on including the true message–key pair. Theorem 1 summarizes the resulting construction and its correctness.

3.1 An expectation as a decomposition

The characterization in Section 2 suggests a probabilistic way to build a decomposition: construct a finitely supported random nonnegative matrix Y such that $\mathbb{E}Y = Q$ and every realization has at most one positive entry per column. If Y takes values Y_c with probabilities w_c , then

$$A_c = w_c Y_c, \quad \sum_c A_c = \mathbb{E}Y = Q. \quad (13)$$

Thus an expectation identity supplies exactly the component matrices required for encryption. An individual Y_c need not have total mass one. The expectation fixes each pair’s average contribution, while leaving freedom in which pairs occur together; we will use this freedom to control security.

The algorithmic advantage is an *implicit* representation of the decomposition. An explicit table of all components and weights may be very large, making storage and input-dependent sampling costly. Instead, we seek a short sequence of random updates that generates one output matrix at a time. A process with t binary updates can have as many as 2^t branch histories, yet sampling an output follows only one history of at most t updates. There is no need to store the whole branching tree or the full terminal distribution. This does not mean every decomposition requires a table, or that an expectation identity alone gives efficiency: we must still design inexpensive updates and a way to sample correctly for the true input.

One way to obtain such a Y is to start at $Y_0 = Q$ and repeatedly make random changes with no average displacement. If each update satisfies

$$\mathbb{E}[Y_{t+1} \mid Y_0, \dots, Y_t] = Y_t,$$

then averaging over the previous choices gives $\mathbb{E}Y_{t+1} = \mathbb{E}Y_t = Q$. If the process reaches a column-unique matrix within a bounded number of steps, and leaves completed outputs unchanged, its final output Y has the required expectation. The challenge is to choose updates that reach this simple form while also controlling the row masses relevant to security. The construction below implements this idea by rounding a scaled matrix with auxiliary entries.

Sampling Y independently of the input is not yet encryption: the sampled matrix may not contain the pair to be encrypted. The conversion from Section 2 instead gives

$$\Pr[\text{Enc}(k_j, m_i) = c] = \frac{w_c(Y_c)_{ij}}{Q_{ij}} \quad (Q_{ij} > 0).$$

In words, reweight each possible matrix by its entry at the actual input pair. Incompatible matrices receive probability zero, and $\mathbb{E}Y = Q$ ensures that the remaining probabilities sum to one. This is generally more than conditioning on compatibility: different positive entries receive different weights. The actual ciphertext law is $\Pr[C = c] = w_c \sum_{i,j} (Y_c)_{ij}$, which need not equal w_c . Our algorithmic task is therefore twofold: produce a suitable random matrix and implement the input-dependent sampling rule.

A one-step illustration. Consider the correlated source

$$Q = \frac{1}{8} \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix}. \quad (14)$$

Starting at $Y_0 = Q$, consider a single random update to one of

$$Y_d = \frac{1}{2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad Y_o = \frac{1}{2} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

with probabilities $3/4$ and $1/4$, respectively. Its output Y_1 satisfies $\mathbb{E}Y_1 = (3/4)Y_d + (1/4)Y_o = Q$, so the update itself specifies a decomposition. Here there are only two outcomes and we can write both down. For larger matrices, the same principle will be realized by successive local updates rather than a precomputed list of outcomes. Section 3.3 explains how to obtain this very update by moving around a cycle.

The distinction from encryption remains important even here. The base process chooses the diagonal pattern with probability $3/4$, but encryption of (m_1, k_1) must choose it with probability one because the other pattern omits that pair. For (m_1, k_2) it must choose the off-diagonal pattern. Larger instances can retain randomness after the input is fixed; Section 3.4 implements the required reweighting along a single path, without evaluating all terminal probabilities w_c .

3.2 Capacity choice and integer completion

We choose outputs whose positive entries all have a common weight: $Y = \beta R$, with $q_\star \leq \beta \leq L$. Here the zero-one matrix R records a decryption pattern: $R_{ij} = 1$ means that (m_i, k_j) is a candidate pair. The restriction of at most one 1 per column is exactly what makes decryption under a fixed key unambiguous. We require a nonempty

pattern because an actual ciphertext must admit at least one honest input; the completion below will ensure this automatically. We impose no one-per-row restriction, so several candidate keys may decrypt to the same message.

Equal positive weights are a design choice, not a correctness requirement. They make the posterior uniform over candidate pairs, so security can be analyzed by counting candidates, and they turn input reweighting into conditioning on the event $R_{ij} = 1$. Both facts will be derived below. In this construction we use the binary pattern R itself as the ciphertext; an index can also represent it if a public table is available.

Write μ for the base distribution of R . The expectation condition and resulting decomposition become

$$A_R = \beta\mu(R)R, \quad \mathbb{E}_{R \sim \mu} R = Q/\beta. \quad (15)$$

Entrywise, edge (i, j) is selected with probability Q_{ij}/β . The expected number of selected entries in column j is q_j/β ; the choice $\beta \geq q_\star$ keeps this at most one, as column uniqueness requires. Under this base distribution, the expected number of selected edges is $1/\beta$, so a smaller β allows more candidate keys per output. We will prove that key-recovery advantage is exactly β . Several keys may then decrypt to the same message, however, so controlling message recovery also requires controlling the number assigned to each row. At $\beta = L$, the row capacities below are one and the construction reduces to matchings.

To realize the required expectation, we start from the fractional matrix Q/β and apply random updates that preserve each entry's expectation until all entries are zero or one. An auxiliary row and column let us preserve integer sums throughout this process and thereby control the final row counts. Let

$$b_i = \lceil p_i/\beta \rceil$$

be the capacity of message row i . Index an auxiliary row and column by 0. Complete the real block Q/β to an $(n_m + 1) \times (n_k + 1)$ matrix X by setting

$$\begin{aligned} X_{ij} &= Q_{ij}/\beta, & X_{i0} &= b_i - p_i/\beta, \\ X_{0j} &= 1 - q_j/\beta, & X_{00} &= 1/\beta - \lfloor 1/\beta \rfloor. \end{aligned} \quad (16)$$

All entries lie in $[0, 1]$: in particular, $Q_{ij} \leq q_j \leq \beta$. The row and column sums are integers:

$$\begin{aligned} \sum_{j=0}^{n_k} X_{ij} &= b_i & (i \geq 1), & \sum_{i=0}^{n_m} X_{ij} = 1 & (j \geq 1), \\ \sum_{j=0}^{n_k} X_{0j} &= n_k - \lfloor 1/\beta \rfloor, & \sum_{i=0}^{n_m} X_{i0} &= \sum_{i=1}^{n_m} b_i - \lfloor 1/\beta \rfloor. \end{aligned} \quad (17)$$

The auxiliary entries are bookkeeping devices, not new messages, keys, or secret information. In particular, integer sums in the completed matrix do not imply that the real block will have row and column sums equal to one. After rounding, auxiliary entries account precisely for the deficit. When $\beta = L$, all $b_i = 1$; smaller parameters may give a high-probability message capacity greater than one without introducing cloned message nodes.

3.3 Expectation-preserving cycle rounding

We now implement the random process promised in Section 3: starting from the completed matrix X , we round its entries to zero or one while preserving their expectations. The real block of the final matrix will be a decryption pattern. It is the law of these output patterns, not a single pattern, that defines the decomposition.

What a cycle does. View rows and columns as two kinds of vertices. Connect row i to column j whenever X_{ij} is strictly between 0 and 1. This is a bipartite graph, including the auxiliary row and column. A *simple cycle* is a closed path that alternates between the two kinds of vertices and does not revisit a vertex except to close the path. It therefore has an even number of edges. The smallest example is

$$\text{row 1} \rightarrow \text{column 1} \rightarrow \text{row 2} \rightarrow \text{column 2} \rightarrow \text{row 1}.$$

Its four edges are the four entries of a two-by-two rectangle. Adding an amount to two opposite entries and subtracting it from the other two leaves each row and column sum unchanged. The same alternating adjustment works around any even cycle, including longer cycles or ones involving auxiliary vertices. A four-edge cycle need not always be available.

For the example in (14), take $\beta = 1/2$. All auxiliary entries are zero, and the real block and an alternating direction are

$$2Q = \begin{pmatrix} 3/4 & 1/4 \\ 1/4 & 3/4 \end{pmatrix}, \quad H_{\text{real}} = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}.$$

Moving by $+1/4$ along this direction reaches the diagonal pattern $2Y_d$; moving by $-3/4$ reaches the off-diagonal pattern $2Y_o$. Choose these endpoints with probabilities $3/4$ and $1/4$, respectively. The average displacement is zero because

$$\frac{3}{4} \cdot \frac{1}{4} - \frac{1}{4} \cdot \frac{3}{4} = 0.$$

Thus the row and column sums are preserved on *each* branch, whereas the entries are preserved *in expectation*. Scaling the output by β gives exactly the one-step process with expectation Q described above. The unequal branch probabilities are essential to preserve this expectation.

The general update. A fractional cycle exists whenever rounding is unfinished. Indeed, every nonisolated vertex in the fractional graph has degree at least two: a single fractional incident entry together with integer entries could not sum to an integer. A finite graph with this property contains a simple cycle.

Choose such a cycle and form a direction matrix H with alternating $+1$ and -1 on its edges and zero elsewhere. Each affected row and column receives one plus and one minus, so moving along H preserves all sums. Let $u_+, u_- > 0$ be the largest step lengths in the two directions that keep every entry in $[0, 1]$. The base transition is

$$X' = \begin{cases} X + u_+ H, & \text{with probability } u_- / (u_+ + u_-), \\ X - u_- H, & \text{with probability } u_+ / (u_+ + u_-). \end{cases} \quad (18)$$

Conditional on the history and selected cycle, its mean displacement is zero, so $\mathbb{E}[X' \mid \text{history, cycle}] = X$. Each endpoint makes at least one fractional entry integral, and integral entries are never changed again. Let F_0 be the number of strictly fractional entries in the completed initial matrix. The process therefore terminates after at most F_0 updates. This is the classical dependent-rounding mechanism on cycles [16]; our completion and subsequent conditioning connect it to encryption.

Let Z denote the final zero-one matrix, let R be its real block, and put

$$d_i(R) = \sum_{j=1}^{n_k} R_{ij}, \quad D(R) = \sum_{i=1}^{n_m} d_i(R).$$

The row count $d_i(R)$ is the number of candidate keys that decrypt this pattern to message m_i . The total count $D(R)$ is the number of candidate message-key pairs; by column uniqueness, it also equals the number of candidate keys. These are counts, not probabilities, and $D(R)$ need not equal the number of distinct candidate messages. For example, if two keys lead to m_1 and one to m_2 , then $d_1 = 2$, $d_2 = 1$, and $D = 3$, although there are only two candidate messages. These multiplicities will determine the posterior probabilities and recovery guarantees below. We omit the argument R when the pattern is clear from context.

Preservation of expectations and integer sums gives

$$\begin{aligned} \mathbb{E}R &= Q/\beta, & \sum_i R_{ij} &= 1 - Z_{0j} \in \{0, 1\}, \\ d_i(R) &= b_i - Z_{i0} \in \{\lfloor p_i/\beta \rfloor, \lceil p_i/\beta \rceil\}, \\ D(R) &= \lfloor 1/\beta \rfloor + Z_{00}. \end{aligned} \tag{19}$$

If p_i/β is an integer, $X_{i0} = 0$ initially and remains zero, so the row count is exactly p_i/β . Similarly, if $1/\beta$ is an integer, $Z_{00} = 0$ and $D = 1/\beta$ exactly. The formula for D follows from the auxiliary column sum in (17). Since $\beta \leq 1$, $\lfloor 1/\beta \rfloor \geq 1$ and R is nonempty. Initially zero entries never change, so $\text{supp}(R) \subseteq \text{supp}(Q)$. Writing μ for the output law, the expectation identity therefore gives the column-unique decomposition $Q = \sum_R \beta \mu(R) R$ required in Section 2. No comparison between the numbers of messages and keys is required.

3.4 Encryption and decryption

Fix a public policy for selecting a legal fractional cycle from the current state and history. The policy must be the same for every input: its choices may reflect earlier conditioned updates, but may not use the true message-key pair separately. A deterministic policy suffices; the same construction also allows a randomized public policy, as explained in Appendix A. Let μ be the output law of this base process. For a positive-probability input edge $e = (i, j)$, the desired encryption law is

$$\Pr[\text{Enc}(k_j, m_i) = R] = \Pr_\mu[R \mid R_e = 1] = \frac{\beta \mu(R) R_{ij}}{Q_{ij}}. \tag{20}$$

Algorithm 1 Capacity-rounding encryption and decryption

Require: Public $Q, \beta \in [q_\star, L]$, and a fixed public cycle-selection policy

```

1: function ENC( $k_j, m_i$ )  $\triangleright Q_{ij} > 0$ 
2:    $e \leftarrow (i, j)$ ; initialize  $X$  by (16)
3:   while  $X$  has a fractional entry do
4:     Use the public policy to select a simple fractional cycle
5:     Form its alternating direction  $H$ 
6:     Compute  $u_+, u_-$  and states  $X^+ = X + u_+H, X^- = X - u_-H$ 
7:     Set  $w_+ = u_-/(u_+ + u_-)$  and  $\tilde{w}_+ = w_+X_e^+/X_e$ 
8:     Replace  $X$  by  $X^+$  with probability  $\tilde{w}_+$ , otherwise by  $X^-$ 
9:   end while
10:  return the real block  $R$  of  $X$ 
11: end function
12: function DEC( $k_j, R$ )
13:  if column  $j$  has a unique selected entry  $R_{ij} = 1$  then
14:    return  $m_i$ 
15:  else
16:    return  $\perp$ 
17:  end if
18: end function

```

Repeatedly running the base process until it selects e would implement this law, but would require β/Q_e trials in expectation. We instead condition each transition, avoiding rejection of complete matrices.

At any history with current matrix X , the probability of eventually selecting e is exactly X_e . Indeed, future transitions preserve the conditional expectation of this entry, and its terminal value is an indicator. If a base transition has branches b with probabilities w_b and next states $X^{(b)}$, choose branch b with probability

$$\tilde{w}_b = w_b \frac{X_e^{(b)}}{X_e}. \quad (21)$$

These probabilities sum to one because the base transition preserves X_e in expectation. A branch setting X_e to zero receives probability zero. The resulting process therefore ends with $R_e = 1$. For instance, base branches of probability 1/2 that change $X_e = 0.2$ to 0.3 and 0.1 become conditional branches of probabilities 0.75 and 0.25. Only the next states are needed, not a search over future paths. This stepwise application of conditional probability is a Doob transform.

Algorithm 1 combines these conditioned updates with decryption by column lookup. Only one rounding path is followed; neither procedure needs a table of all terminal patterns.

Theorem 1 (Capacity-rounding construction). *For any finite joint source Q , parameter $\beta \in [q_\star, L]$, and fixed public cycle-selection policy, Algorithm 1 encrypts every input (m_i, k_j) with $Q_{ij} > 0$ in at most F_0 rounding updates and samples exactly $\mu(\cdot \mid R_{ij} = 1)$. Together with its decryption procedure, it implements the decomposition $Q = \sum_R \beta \mu(R)R$ and is correct with probability one.*

Proof. Conditioning changes only branch probabilities, so each update still makes a fractional entry integral and the F_0 bound is unchanged. Along a terminating path, the ratios in (21) telescope: the conditioned path probability is its base probability multiplied by $R_e/X_e^{(0)}$. Since $X_e^{(0)} = Q_{ij}/\beta$, summing paths with the same output gives (20). Every sampled pattern contains e , and column uniqueness then gives correct decryption.

The tagged fallback from Section 2 extends the scheme to inputs outside the support of Q when all-pairs correctness is required. For an efficient cycle-selection policy and rational inputs, the explicit-input implementation has expected cost polynomial in the matrix dimensions and input bit length; Appendix A gives precise time, space, and ciphertext costs. This does not imply efficiency in message and key bit lengths when the matrix itself is exponentially large.

Ciphertext law and posteriors. To analyze security, we must distinguish the base distribution μ from the ciphertext distribution produced by encryption under the source Q .

The joint mass induced by encryption is

$$\Pr[M = m_i, K = k_j, C = R] = \beta\mu(R)R_{ij}. \quad (22)$$

Consequently, the actual ciphertext distribution is the size-biased law

$$\nu(R) = \Pr[C = R] = \beta D(R)\mu(R). \quad (23)$$

It is normalized because $\mathbb{E}_\mu D = 1/\beta$. More edges mean that more possible honest inputs can generate the matrix. For example, at $\beta = 3/5$, the base probabilities of $D = 1$ and $D = 2$ are $1/3$ and $2/3$; their actual ciphertext probabilities are $1/5$ and $4/5$. The two laws agree if D is constant, but not in general.

For every positive-probability ciphertext,

$$\Pr[K = k_j \mid C = R] = \frac{\sum_i R_{ij}}{D(R)}, \quad \Pr[M = m_i \mid C = R] = \frac{d_i(R)}{D(R)}. \quad (24)$$

Selected keys are equally likely. Selected messages need not be: their posterior probabilities are proportional to their row counts. The next section chooses β and uses these posteriors to analyze recovery and semantic security.

4 Security Guarantees of Our Scheme

Throughout this section, expectations and covariances are under the unconditioned base law μ , unless explicitly stated otherwise. This convention is essential: the real ciphertext law is $\nu(R) = \beta D(R)\mu(R)$, and an encryption of a fixed input follows a third law, obtained by conditioning on the input edge.

4.1 Exact recovery probabilities and the choice of capacity

Equations (22) and (24) yield

$$\text{Adv}^{\text{kr}} = \beta, \quad \text{Adv}^{\text{mr}} = \beta \mathbb{E}_{\mu} \max_i d_i(R). \quad (25)$$

For key recovery, the conditional success $1/D$ cancels the factor D in the ciphertext law. Message recovery instead involves the largest row count. This is the precise role of β : it directly sets KRS, while integer row counts determine whether the MRS lower bound is attained.

If $p_{\star} \leq q_{\star}$, choosing $\beta = q_{\star} = L$ gives nonempty matchings, hence $\text{Adv}^{\text{mr}} = \text{Adv}^{\text{kr}} = L$. For the remaining case $p_{\star} > q_{\star}$, relabel messages locally so that $p_1 \geq p_2 \geq \dots$, with $p_1 = p_{\star}$ and $p_2 = 0$ if $n_m = 1$. This convention only simplifies the analysis of the capacity choice: neither running the construction at a given β nor proving its general TDSS bound requires the rows to be sorted. It suffices to ensure

$$p_2 \leq \beta \lfloor p_1 / \beta \rfloor. \quad (26)$$

Then row 1 always has at least $\lfloor p_1 / \beta \rfloor$ selected edges, whereas every other row has at most that many. Therefore $\text{Adv}^{\text{mr}} = \beta \mathbb{E} d_1 = p_1 = L$.

Set

$$s = \lfloor p_1 / q_{\star} \rfloor, \quad \beta_{\text{dom}} = \max\{q_{\star}, p_2 / s\}. \quad (27)$$

Both arguments of the maximum are at most p_1 / s . Consequently,

$$s \leq p_1 / \beta_{\text{dom}} \leq p_1 / q_{\star} < s + 1, \quad \beta_{\text{dom}} \leq p_1 / s < (1 + 1/s) q_{\star} \leq 2q_{\star}.$$

Thus (26) holds. The factor two comes from choosing a scale at which the leading message dominates in *every* rounded output, not from assuming a favorable average rounding outcome.

If $p_2 \leq q_{\star} \lfloor p_1 / q_{\star} \rfloor$, one may take $\beta = q_{\star}$ and attain both recovery lower bounds exactly. This is a sufficient condition, not a characterization of all sources permitting simultaneous optimality. Without that condition, taking $\beta = q_{\star}$ still gives

$$\text{Adv}^{\text{mr}} \leq q_{\star} \lceil p_{\star} / q_{\star} \rceil < p_{\star} + q_{\star}. \quad (28)$$

Table 1 summarizes these choices. Its dominance condition is sufficient, not necessary; all guarantees apply to arbitrary dependence between the message and key.

The factor-two loss in the message-dominant choice cannot be improved uniformly over sources while preserving optimal MRS. The following exact result makes the obstruction explicit.

Theorem 2 (Exact recovery cost for uniform marginals). *Let Q have a uniform message marginal on n_m values and a uniform key marginal on $n_k \geq n_m$ values, with arbitrary dependence between them. Among all correct schemes satisfying $\text{Adv}^{\text{mr}} = 1/n_m$, the minimum KRS is*

$$\frac{1}{n_m \lfloor n_k / n_m \rfloor}. \quad (29)$$

This minimum is achievable with $\text{Adv}^{\text{tdss}} = 0$.

Table 1. Recovery guarantees for different capacity choices. Entries without an inequality are exact. In the message-dominant rows, p_2 is the second-largest message probability and $s = \lfloor p_\star/q_\star \rfloor$; β_{dom} is defined in (27) and satisfies $\beta_{\text{dom}} < 2q_\star$.

Source condition	Capacity β	Adv^{mr}	Adv^{kr}
Any source (matching choice)	L	L	L
Key-dominant: $p_\star \leq q_\star$	$q_\star$	$L = q_\star$	$q_\star$
Message-dominant: $p_\star > q_\star$	β_{dom}	$L = p_\star$	β_{dom}
Message-dominant and $p_2 \leq sq_\star$	$q_\star$	$L = p_\star$	$q_\star$
Any source (minimum KRS)	$q_\star$	$\leq q_\star \lceil p_\star/q_\star \rceil$	$q_\star$

The proof is in Appendix F.2. Relative to the prior key-recovery probability $q_\star = 1/n_k$, the necessary ratio is $n_k/(n_m \lfloor n_k/n_m \rfloor)$. Taking $n_k = 2n_m - 1$ gives $2 - 1/n_m$, approaching two. Thus no fixed constant below two can replace the universal factor; this does not make β_{dom} optimal for each nonuniform source. Appendix B gives an example where unequal component weights outperform this sufficient parameter rule, and Appendix F.1 gives an exact two-message tradeoff.

4.2 Semantic Security

We now bound the improvement in predicting a fixed Boolean message property after seeing a ciphertext. The proof first controls the joint fluctuations of the candidate counts, then translates this control into a TDSS bound.

Negative correlation of row counts. Separate each row count into a fixed integer part and a binary remainder:

$$\ell_i = \lfloor p_i/\beta \rfloor, \quad r_i = p_i/\beta - \ell_i, \quad B_i = d_i - \ell_i. \quad (30)$$

Here $B_i \in \{0, 1\}$ has mean r_i and records whether row i receives one edge beyond its guaranteed integer share. If $r_i > 0$, then $B_i = 1 - Z_{i0}$. If $r_i = 0$, then $B_i = 0$ identically; the preceding formula must not be used for that case.

Lemma 1 (Negative correlation of row remainders). *For every set of rows A ,*

$$\mathbb{E}_\mu \prod_{i \in A} B_i \leq \prod_{i \in A} r_i. \quad (31)$$

In particular, $\text{Cov}_\mu(d_i, d_h) \leq 0$ for distinct rows i, h .

Proof. It is enough to consider sets containing only rows with $r_i > 0$; otherwise both sides vanish. On the current completed matrix define

$$F_A(X) = \prod_{i \in A} (1 - X_{i0}).$$

Its initial value is $\prod_{i \in A} r_i$, and its terminal value is $\prod_{i \in A} B_i$. A simple cycle changes at most two entries in the auxiliary column, in opposite directions. If both affected entries belong to A , denote their present complementary values by v, w and the signed random step by u . Conditional on the current history and chosen cycle, v, w are constants, while u takes values u_+ and $-u_-$ with mean zero. The product of the changed factors is

$$(v - u)(w + u) = vw + (v - w)u - u^2.$$

Its conditional expectation is $vw - \mathbb{E}[u^2] \leq vw$. All other factors remain fixed and nonnegative. If only one factor is affected, the product is affine in u and its conditional expectation is unchanged; if none is affected, the product itself is unchanged. Thus the conditional expectation of F_A cannot increase at any step. Iterating over the bounded number of rounding steps proves (31). Taking $A = \{i, h\}$ gives the covariance claim because adding the fixed integer parts ℓ_i, ℓ_h does not change covariance.

The proof concerns decreasing *conditional expectations*, not monotonicity along each execution: one branch may increase the product. Nor does the lemma assert negative correlation of arbitrary posterior probabilities d_i/D , or of the remainders after conditioning on an honest input. Only the base-law row counts are needed. Negative correlation also does not require fewer keys than messages; a pair of indicators satisfying $B_1 + B_2 = 1$ is a simple example of such dependence.

From count fluctuations to predicate advantage. Fix a predicate before the challenge, let S be its set of true messages, and let $\theta = \sum_{i \in S} p_i$. For a base output define

$$T = \sum_{i \in S} d_i, \quad U = \sum_{i \notin S} d_i, \quad G = T - U, \quad D = T + U.$$

The counts T and U enumerate candidate keys decrypting to true and false messages. Given the ciphertext, the optimal predictor selects the larger side and succeeds with probability $\max(T, U)/D$. Multiplying by the actual ciphertext mass cancels the denominator:

$$\Pr[\text{optimal prediction succeeds}] = \beta \mathbb{E}_\mu \max(T, U) = \frac{1}{2} + \frac{\beta}{2} \mathbb{E}_\mu |G|.$$

Since $\mathbb{E}_\mu G = (2\theta - 1)/\beta$, subtracting the best prior-only prediction gives the exact identity

$$\text{Adv}^{\text{tdss}}(f) = \frac{\beta}{2} (\mathbb{E}_\mu |G| - |\mathbb{E}_\mu G|). \quad (32)$$

This is the general component identity (11) specialized to $A_R = \beta\mu(R)R$. The observer can choose the majority separately for each ciphertext; without a ciphertext, only the average majority is available. Their gap is controlled by fluctuations of G around its mean. Specifically,

$$\mathbb{E}|G| - |\mathbb{E}G| \leq \mathbb{E}|G - \mathbb{E}G| \leq \sqrt{\text{Var}(G)}. \quad (33)$$

One must not apply negative covariance directly to the signed sum G : its minus signs reverse some covariance terms. Instead, bound the two nonnegative sums separately, using Lemma 1:

$$\text{Var}(T) \leq \sum_{i \in S} r_i(1 - r_i), \quad \text{Var}(U) \leq \sum_{i \notin S} r_i(1 - r_i).$$

Now use the identity

$$\text{Var}(T - U) = 2 \text{Var}(T) + 2 \text{Var}(U) - \text{Var}(T + U).$$

Write $\eta = 1/\beta - \lfloor 1/\beta \rfloor$ for the fractional part of the expected total count. By (19), $D = \lfloor 1/\beta \rfloor + Z_{00}$ and $\Pr[Z_{00} = 1] = \eta$, so $\text{Var}(D) = \eta(1 - \eta)$. It follows that

$$\text{Var}(G) \leq V_\beta(Q) := 2 \sum_i r_i(1 - r_i) - \eta(1 - \eta). \quad (34)$$

The right-hand side is independent of the predicate. It is nonnegative, as the construction exists and the displayed argument bounds a variance.

Theorem 3 (TDSS of capacity rounding). *For every $\beta \in [q_\star, L]$ and every legal public base cycle policy, capacity-rounding encryption satisfies*

$$\text{Adv}^{\text{tdss}} \leq \mathcal{B}_\beta(Q) := \min \left\{ \frac{1}{2}, \frac{\beta}{2} \sqrt{V_\beta(Q)} \right\} \leq \min \left\{ \frac{1}{2}, \sqrt{\frac{\beta}{2}} \right\}. \quad (35)$$

Proof. Combine (32)–(34) and maximize over fixed predicates. The simplified bound uses $\sum_i r_i(1 - r_i) \leq \sum_i p_i/\beta = 1/\beta$. The universal cap $1/2$ holds because the best prior-only Boolean prediction already succeeds with probability at least $1/2$.

For a particular predicate, its prior bias gives the additional bound

$$\text{Adv}^{\text{tdss}}(f) \leq \frac{1}{2} \left(\sqrt{(2\theta - 1)^2 + \beta^2 V_\beta(Q)} - |2\theta - 1| \right), \quad (36)$$

by applying $\mathbb{E}|G| \leq \sqrt{\mathbb{E}G^2}$ to (32). None of these bounds claims that every ciphertext has a nearly unchanged posterior; the success probability averages over the actual challenge input and fresh encryption randomness.

Worst-case order. The square-root scale cannot be improved for arbitrary sources, even without imposing either recovery guarantee. For uniform independent messages and keys with $n_k \rightarrow \infty$ and $n_m/n_k \rightarrow \infty$, every correct scheme with deterministic decryption satisfies

$$\text{Adv}^{\text{tdss}} \geq \frac{1 - o(1)}{\sqrt{2\pi n_k}} = \Omega(\sqrt{q_\star}). \quad (37)$$

The reason is that one ciphertext can leave at most n_k possible messages; a suitable fixed predicate benefits from this restriction on posterior support. Theorem 7 in Appendix F.3 gives the formal bound and proof. Together with the simultaneous guarantee below, this establishes the worst-case order, not an optimal leading constant or a positive square-root lower bound for every source. Different cycle rules can still improve actual TDSS on particular sources while sharing the same certificate; Appendix F.4 discusses this distinction.

4.3 The simultaneous guarantee

Theorem 4 (Arbitrary finite joint sources). *For a finite message–key source Q , with zero-probability rows and columns removed, let p_2 in the following parameter formula denote the second-largest message probability, counting ties, and set $p_2 = 0$ if there is only one message. This value does not depend on the original labeling of the messages. Define*

$$\beta_* = \begin{cases} q_*, & p_* \leq q_*, \\ \max\{q_*, p_2/\lfloor p_*/q_* \rfloor\}, & p_* > q_*. \end{cases} \quad (38)$$

Capacity rounding with stepwise conditioning is correct and satisfies

$$\text{Adv}^{\text{mr}} = L, \quad \text{Adv}^{\text{kr}} = \beta_* < 2q_*, \quad \text{Adv}^{\text{tdss}} \leq \mathcal{B}_{\beta_*}(Q) \leq \min\{1/2, \sqrt{\beta_*/2}\}. \quad (39)$$

For explicit rational input, an efficient cycle policy yields expected bit complexity polynomial in the input length. For arbitrary real input, the algorithm is understood in an ideal exact-arithmetic model.

Proof. The parameter lies in $[q_*, L]$. Theorem 1 gives correctness. The dominance argument proves $\text{Adv}^{\text{mr}} = L$ and $\beta_* < 2q_*$, while (25) proves $\text{Adv}^{\text{kr}} = \beta_*$. Apply Theorem 3; the complexity statement is established in Appendix A.

In particular, the general TDSS scale is $O(\sqrt{q_*})$ even when p_* is substantially larger. For comparison, choosing $\beta = L$ preserves row uniqueness, gives $\text{Adv}^{\text{mr}} = \text{Adv}^{\text{kr}} = L$, and specializes the bound to

$$\text{Adv}^{\text{tdss}} \leq \min \left\{ \frac{1}{2}, \frac{1}{2} \sqrt{2L - 2 \sum_i p_i^2 - L^2 \eta(1 - \eta)} \right\}. \quad (40)$$

Here $\eta = 1/L - \lfloor 1/L \rfloor$ is specialized to this choice of parameter. A smaller parameter improves the simplified scale bound, but neither the fractional-remainder certificate nor actual TDSS is generally monotone in β .

A correlated three-by-four example. Let

$$Q = \frac{1}{16} \begin{pmatrix} 3 & 1 & 2 & 2 \\ 1 & 2 & 0 & 1 \\ 0 & 1 & 2 & 1 \end{pmatrix}, \quad p = (1/2, 1/4, 1/4), \quad q = (1/4, 1/4, 1/4, 1/4). \quad (41)$$

The zero entries with positive row and column marginals witness dependence. The matching choice $\beta = L = 1/2$ gives $\text{Adv}^{\text{mr}} = \text{Adv}^{\text{kr}} = 1/2$. At $\beta = q_* = 1/4$, the row counts are always $(2, 1, 1)$ and every column is selected. Thus every ciphertext has message posterior $(1/2, 1/4, 1/4)$, and

$$\text{Adv}^{\text{mr}} = 1/2, \quad \text{Adv}^{\text{kr}} = 1/4, \quad \text{Adv}^{\text{tdss}} = 0.$$

This is message perfect secrecy by direct equality of posterior and prior, not merely an inference from zero TDSS. More generally, if all p_i/β are integers, all row counts and D are fixed and the same conclusion holds. Attaining $\text{Adv}^{\text{kr}} = q_*$ alone does not generally imply key perfect secrecy; here the uniform key distribution and selection of every column also give $C \perp K$.

5 Robustness and Ciphertext Representation

The construction so far assumes a known joint distribution and describes a ciphertext by its decoding matrix. We now separate two further questions: how sensitive is security to the design distribution, and how much of the random matrix distribution must be retained to obtain short ciphertexts?

5.1 Security under a misspecified distribution

Let Q be the true joint distribution and $\widehat{Q}$ a normalized design distribution on the same message and key spaces. Their message marginals are $p_i = \sum_j Q_{ij}$ and $\widehat{p}_i = \sum_j \widehat{Q}_{ij}$, respectively. Let

$$\delta = \text{TV}(Q, \widehat{Q}) = \frac{1}{2} \sum_{i,j} |Q_{ij} - \widehat{Q}_{ij}|, \quad \delta_M = \text{TV}(p, \widehat{p}) \leq \delta. \quad (42)$$

Total variation is the largest difference between the probabilities assigned to the same event by two distributions. The comparison below fixes an encryption channel W ; in particular, a channel designed for $\widehat{Q}$ is not silently replaced by one designed for Q . We write $\text{Adv}_{W,Q}^{\text{tdss}}$ for its TDSS advantage under Q .

Theorem 5 (Distributional stability). *For a fixed channel W defined on $\text{supp}(\widehat{Q}) \cup \text{supp}(Q)$,*

$$|\text{Adv}_{W,Q}^{\text{tdss}} - \text{Adv}_{W,\widehat{Q}}^{\text{tdss}}| \leq \delta + \delta_M \leq 2\delta, \quad (43)$$

$$|\text{Adv}_{W,Q}^{\text{mr}} - \text{Adv}_{W,\widehat{Q}}^{\text{mr}}| \leq \delta, \quad |\text{Adv}_{W,Q}^{\text{kr}} - \text{Adv}_{W,\widehat{Q}}^{\text{kr}}| \leq \delta. \quad (44)$$

Proof. For each fixed adversary, its conditional probability of success is a function of the input pair taking values in $[0, 1]$. Its expectation changes by at most δ when the input distribution changes. Taking the maximum over adversaries preserves this bound. For a fixed predicate, the optimal prior-only prediction probability changes by at most δ_M . Subtracting this baseline, maximizing over predicates, and interchanging $\widehat{Q}$ and Q proves (43). Recovery success probabilities have no subtracted baseline and hence incur only δ .

Let $\widehat{\beta}$ be the parameter of Theorem 4 selected using $\widehat{Q}$, and let $\widehat{W}$ be the resulting channel. The theorem implies

$$\begin{aligned} \text{Adv}_{\widehat{W},Q}^{\text{tdss}} &\leq \min\{1/2, \mathcal{B}_{\widehat{\beta}}(\widehat{Q}) + \delta + \delta_M\}, \\ \text{Adv}_{\widehat{W},Q}^{\text{mr}} &\leq \min\{1, L(\widehat{Q}) + \delta\}, \quad \text{Adv}_{\widehat{W},Q}^{\text{kr}} \leq \min\{1, \widehat{\beta} + \delta\}. \end{aligned} \quad (45)$$

Here $\mathcal{B}$ is the certificate in Theorem 3. Negative correlation and the posterior formulas need hold only under the design source; stability transfers the resulting bounds to the true source. If $\widehat{Q}_{ij} = 0 < Q_{ij}$, a fallback encryption rule must first be defined for that input. It may reveal the input completely. Its total probability under Q is at most δ and is already covered by the stability bound.

5.2 Design matrices and one-sided errors

A design source may be chosen deliberately to simplify implementation, not only to estimate Q as accurately as possible. The appropriate objective balances the design-source certificate, the distributional error, and the implementation cost. Entrywise overestimation alone is not sufficient to avoid a TDSS penalty.

More generally, let $A \geq 0$ be a design matrix of total mass $t > 0$, and set $\widehat{Q} = A/t$. If $a_i = \sum_j A_{ij}$, define

$$u = \sum_{i,j} (Q_{ij} - A_{ij})_+, \quad v_M = \sum_i (a_i - p_i)_+. \quad (46)$$

For the fixed channel $\widehat{W}$ designed for $\widehat{Q}$,

$$\text{Adv}_{\widehat{W}, Q}^{\text{tdss}} \leq t \text{Adv}_{\widehat{W}, \widehat{Q}}^{\text{tdss}} + u + v_M. \quad (47)$$

Indeed, the optimal post-ciphertext success mass under Q exceeds its unnormalized value under A by at most u . Conversely, the optimal prior prediction mass under A exceeds that under Q by at most v_M . Subtracting proves (47). For the same fixed channel, the argument also gives $\text{Adv}_Q^{\text{mr}} \leq t \text{Adv}_{\widehat{Q}}^{\text{mr}} + u$ and $\text{Adv}_Q^{\text{kr}} \leq t \text{Adv}_{\widehat{Q}}^{\text{kr}} + u$. Even if $A \geq Q$ and $u = 0$, the term v_M may be positive. Examples in Appendix C show that both the normalized 2δ bound and this one-sided caveat are necessary.

If a normalized approximation preserves the message marginal, then $\delta_M = 0$. In particular, using the true marginals p, q to form the independent design source $\widehat{Q} = pq^\top$ gives

$$\text{Adv}_{\widehat{W}, Q}^{\text{tdss}} \leq \text{Adv}_{\widehat{W}, \widehat{Q}}^{\text{tdss}} + \text{TV}(Q, pq^\top). \quad (48)$$

This can replace explicit processing of Q when an efficient independent source construction is available and the correlation residual is small. Strongly correlated sources need not admit such an approximation.

5.3 A small support preserving the security certificate

Short ciphertexts, a small public description, and fast encryption are different requirements. We first prove an existence result for ciphertext length alone. Let $\mathcal{R}_0$ be the support of the base distribution μ of the main construction, and let $s_Q = \text{nnz}(Q)$ be the number of positive entries of Q . Define the feature vector

$$\Phi(R) = ((R_{ij})_{Q_{ij} > 0}, (d_i(R)d_h(R))_{i < h}), \quad T_0 = 1 + s_Q + \binom{n_m}{2}. \quad (49)$$

The first features describe individual edge inclusion; the remaining features describe joint fluctuations of message-row counts.

Theorem 6 (Short indices preserving the main guarantee). *For $\beta = \beta_*$, there is a distribution $\widehat{\mu}$ supported on at most T_0 members of $\mathcal{R}_0$ such that $\mathbb{E}_{\widehat{\mu}} \Phi = \mathbb{E}_{\mu} \Phi$. The induced encryption scheme has $\text{Adv}^{\text{mr}} = L$, $\text{Adv}^{\text{kr}} = \beta_*$, and the same TDSS certificate $\mathcal{B}_{\beta_*}(Q)$ as in Theorem 3. On the source support, a ciphertext can be represented by an index of at most $\lceil \log_2 T_0 \rceil$ bits.*

Proof. Whenever more than T_0 support points have positive weight, their vectors $(1, \Phi(R))$ are linearly dependent. Choose nonzero coefficients c_R with $\sum_R c_R = 0$ and $\sum_R c_R \Phi(R) = 0$. They have both signs. With $\gamma = \min_{c_R > 0} \mu(R)/c_R$, replace each weight by $\mu(R) - \gamma c_R$. The weights remain nonnegative, their sum and all feature expectations are unchanged, and at least one weight becomes zero. Iteration proves the support bound.

Preserving the edge expectations preserves the decomposition and all row means. Let $\ell_i = \lfloor p_i/\beta \rfloor$. Since each support point satisfies $d_i \in \{\ell_i, \ell_i + 1\}$, with the integer case fixed,

$$d_i^2 = (2\ell_i + 1)d_i - \ell_i(\ell_i + 1). \quad (50)$$

Thus first moments also determine diagonal second moments. The additional features preserve off-diagonal second moments and hence the variance of every signed sum of row counts. All supportwise count constraints remain valid, so the TDSS variance certificate is preserved.

For the main parameter, if $p_\star > q_\star$, fix a row attaining $p_\star$. This row dominates every other row on every original support point. Consequently its preserved mean still determines $\text{Adv}^{\text{mr}} = L$. If $p_\star \leq q_\star$, every support point is a nonempty matching, so its maximum row count remains one. Finally, nonemptiness, column uniqueness, and normalization of the base distribution preserve $\text{Adv}^{\text{kr}} = \beta_\star$.

For retained matrices $R_1, \dots, R_t$, encryption of an input edge e uses

$$\Pr[C = a \mid e] = \frac{\beta_\star \hat{\mu}(R_a)(R_a)_e}{Q_e}. \quad (51)$$

These probabilities sum to one, but evaluating and sampling them still requires the retained matrices and their weights. The theorem does not assert that edge marginals alone specify an efficient sampler. A singleton support needs no index bits. Tagged fallback ciphertexts extend correctness to all inputs with $O(\log(n_m + 1) + \log(n_k + 1))$ bits overall.

For other choices of β , the listed moments need not preserve the actual MRS value; adding the feature $\max_i d_i(R)$ preserves it at the cost of one additional support point. Nor do two moments determine an absolute first moment: actual TDSS may change under compression, even though the proved certificate remains valid.

Since $s_Q \leq n_m n_k$, the index length is logarithmic in the alphabet sizes. The public table may nevertheless be large, and finding it may be expensive. Appendix D gives a rational-weight version with bounded description length and analyzes replayable paths. Appendix E gives special sources with genuinely succinct algorithms. These statements do not imply efficient short-description encryption for arbitrary succinctly specified joint distributions.

6 Discussion and Limitations

The matrix viewpoint separates three issues that are easily conflated. Column uniqueness and input-edge conditioning ensure correctness, while edge expectations enforce the joint source; the way edges occur together controls recovery and predicate prediction; and the

description of the output family determines communication and implementation costs. The same source can admit many decompositions, with different semantic behavior and different representations.

Independence and structured efficiency. Independence is not required for the main theorem, but it remains useful for lower-bound instances and efficient special cases. An explicit matrix may have exponentially many entries in the message or key bit length even when it has rank one. A compact probability model likewise does not automatically supply fast conditional sampling. Appendix E gives short affine descriptions for uniform independent sources and an efficient correlated-source construction when both inputs determine a common label, under the balanced uniform-index and efficient exact-codec assumptions specified there.

The common-label assumption is substantive. Having an unobserved common cause is not enough: the label must be recoverable from each input separately. In a full-support joint source any such deterministic common label is constant. Arbitrary semantic clustering, shared neural-network architectures, or similar embeddings do not establish exact agreement, balanced groups, conditional independence, or reversible encoding. Approximate structural models can instead be assessed using the fixed-channel transfer theorem, provided their distribution error is quantified. Sparse nonnegative mixtures require a separate accounting of information revealed by the component label.

What optimality does and does not mean. Our MRS guarantee is exactly optimal for every finite joint source. The KRS approximation factor is tight as a uniform worst-case guarantee subject to that requirement. The TDSS guarantee has optimal worst-case order on a specified family of sources. These statements neither solve the minimum KRS for every individual source nor identify the exact minimum TDSS. Unequal component weights can improve particular instances beyond the capacity construction, as shown in Appendix B. Furthermore, zero TDSS alone does not imply equality of every posterior and its prior, and optimal KRS alone does not imply independence of the key and ciphertext.

Scope of the security experiment. TDSS here concerns one ciphertext and a predicate fixed before fresh encryption randomness. Observations of previous ciphertexts, known plaintexts, public labels, or other challenge-related information change the relevant conditional source and require their own analysis. Fresh encryption coins alone do not establish secure key reuse. The repeated-message results of prior honey-encryption work [22] are not consequences of our one-shot theorem.

Open algorithmic questions. The main sampler is efficient for an explicitly listed rational source, whereas its output may be large. Moment preservation establishes short ciphertext existence but may shift the cost to a large public table or expensive preprocessing. Finding broader compact source families with efficient conditional sampling, computing small-support certificates without enumerating their candidates, and optimizing the full recovery frontier for a fixed source remain separate problems. These distinctions are important when interpreting a distribution-aware security theorem as the basis for an implementation.

References

1. Alvim, M.S., Chatzikokolakis, K., McIver, A., Morgan, C., Palamidessi, C., Smith, G.: The Science of Quantitative Information Flow. *Information Security and Cryptography*, Springer International Publishing (2020). <https://doi.org/10.1007/978-3-319-96131-6>
2. Applebaum, B.: Key-dependent message security: Generic amplification and completeness. In: *Advances in Cryptology—EUROCRYPT 2011*. pp. 527–546. Springer (2011)
3. Bellare, M., Boldyreva, A., O’Neill, A.: Deterministic and efficiently searchable encryption. In: *Advances in Cryptology—CRYPTO 2007*. pp. 535–552. Springer (2007)
4. Birkhoff, G.: Tres observaciones sobre el álgebra lineal. *Universidad Nacional de Tucumán Revista, Serie A* **5**, 147–151 (1946)
5. Black, J., Rogaway, P., Shrimpton, T.: Encryption-scheme security in the presence of key-dependent messages. In: *Selected Areas in Cryptography*. pp. 62–75. Springer (2003). https://doi.org/10.1007/3-540-36492-7_6
6. Bojinov, H., Bursztein, E., Boyen, X., Boneh, D.: Kamouflage: Loss-resistant password management. In: *ESORICS 2010*. pp. 286–302. Springer (2010)
7. Boneh, D., Halevi, S., Hamburg, M., Ostrovsky, R.: Circular-secure encryption from decision Diffie–Hellman. In: *Advances in Cryptology—CRYPTO 2008*. pp. 108–125. Springer (2008)
8. Bonneau, J.: The science of guessing: Analyzing an anonymized corpus of 70 million passwords. In: *2012 IEEE Symposium on Security and Privacy*. pp. 538–552. IEEE (2012)
9. Chatterjee, R., Bonneau, J., Juels, A., Ristenpart, T.: Cracking-resistant password vaults using natural language encoders. In: *2015 IEEE Symposium on Security and Privacy*. pp. 481–498. IEEE (2015). <https://doi.org/10.1109/SP.2015.36>
10. Cheng, H., Huang, F., Yang, J., Li, W., Wang, P.: Practically secure honey password vaults: New design and new evaluation against online guessing. In: *34th USENIX Security Symposium*. pp. 7781–7798. USENIX Association (2025), <https://www.usenix.org/conference/usenixsecurity25/presentation/cheng-haibo>
11. Cheng, H., Li, W., Wang, P., Chu, C.H., Liang, K.: Incrementally updateable honey password vaults. In: *30th USENIX Security Symposium*. pp. 857–874. USENIX Association (2021), <https://www.usenix.org/conference/usenixsecurity21/presentation/cheng-haibo>
12. Das, A., Bonneau, J., Caesar, M., Borisov, N., Wang, X.: The tangled web of password reuse. In: *Network and Distributed System Security Symposium*. Internet Society (2014). <https://doi.org/10.14722/ndss.2014.23357>
13. Dodis, Y., Reyzin, L., Smith, A.: Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In: *Advances in Cryptology—EUROCRYPT 2004*. pp. 523–540. Springer (2004)
14. Dodis, Y., Smith, A.: Entropic security and the encryption of high entropy messages. In: *Theory of Cryptography*. *Lecture Notes in Computer Science*, vol. 3378, pp. 556–577. Springer (2005). https://doi.org/10.1007/978-3-540-30576-7_30
15. Duan, F., Wang, D., Jia, C.: A security analysis of honey vaults. In: *2024 IEEE Symposium on Security and Privacy*. pp. 1424–1442. IEEE (2024). <https://doi.org/10.1109/SP54263.2024.00219>
16. Gandhi, R., Khuller, S., Parthasarathy, S., Srinivasan, A.: Dependent rounding and its applications to approximation algorithms. *Journal of the ACM* **53**(3), 324–360 (2006). <https://doi.org/10.1145/1147954.1147956>
17. Golla, M., Beuscher, B., Dürmuth, M.: On the security of cracking-resistant password vaults. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. pp. 1230–1241. ACM (2016). <https://doi.org/10.1145/2976749.2978416>

18. Huang, Z., Ayday, E., Fellay, J., Hubaux, J.P., Juels, A.: GenoGuard: Protecting genomic data against brute-force attacks. In: 2015 IEEE Symposium on Security and Privacy. pp. 447–462. IEEE (2015). <https://doi.org/10.1109/SP.2015.34>
19. Jaeger, J., Ristenpart, T., Tang, Q.: Honey encryption beyond message recovery security. In: Advances in Cryptology—EUROCRYPT 2016. pp. 758–788. Lecture Notes in Computer Science, Springer (2016). https://doi.org/10.1007/978-3-662-49890-3_29
20. Juels, A., Ristenpart, T.: Honey encryption: Security beyond the brute-force bound. In: Advances in Cryptology—EUROCRYPT 2014. Lecture Notes in Computer Science, vol. 8441, pp. 293–310. Springer (2014). https://doi.org/10.1007/978-3-642-55220-5_17
21. Juels, A., Rivest, R.L.: Honeywords: Making password-cracking detectable. In: Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security. pp. 145–160. ACM (2013)
22. Li, X., Tang, Q., Zhang, Z.: Fooling an unbounded adversary with a short key, repeatedly: The honey encryption perspective. In: Tessaro, S. (ed.) 2nd Conference on Information-Theoretic Cryptography (ITC 2021). Leibniz International Proceedings in Informatics (LIPIcs), vol. 199, pp. 23:1–23:21. Schloss Dagstuhl–Leibniz-Zentrum für Informatik (2021). <https://doi.org/10.4230/LIPIcs.ITC.2021.23>
23. Ma, J., Yang, W., Luo, M., Li, N.: A study of probabilistic password models. In: 2014 IEEE Symposium on Security and Privacy. pp. 689–704. IEEE (2014). <https://doi.org/10.1109/SP.2014.50>
24. Melicher, W., Ur, B., Segreti, S.M., Komanduri, S., Bauer, L., Christin, N., Cranor, L.F.: Fast, lean, and accurate: Modeling password guessability using neural networks. In: USENIX Security Symposium. pp. 175–191. USENIX Association (2016)
25. Russell, A., Wang, H.: How to fool an unbounded adversary with a short key. In: Advances in Cryptology—EUROCRYPT 2002. Lecture Notes in Computer Science, vol. 2332, pp. 133–148. Springer (2002). https://doi.org/10.1007/3-540-46035-7_9
26. Schrijver, A.: Combinatorial Optimization: Polyhedra and Efficiency. Springer (2003)
27. Shannon, C.E.: Communication theory of secrecy systems. Bell System Technical Journal **28**(4), 656–715 (1949). <https://doi.org/10.1002/j.1538-7305.1949.tb00928.x>
28. Smith, G.: On the foundations of quantitative information flow. In: Foundations of Software Science and Computational Structures. Lecture Notes in Computer Science, vol. 5504, pp. 288–302. Springer (2009). https://doi.org/10.1007/978-3-642-00596-1_21
29. Wang, D., Cheng, H., Wang, P., Huang, X., Jian, G.: Zipf’s law in passwords. IEEE Transactions on Information Forensics and Security **12**(11), 2776–2791 (2017). <https://doi.org/10.1109/TIFS.2017.2721359>
30. Weir, M., Aggarwal, S., de Medeiros, B., Glodek, B.: Password cracking using probabilistic context-free grammars. In: 2009 IEEE Symposium on Security and Privacy. pp. 391–405. IEEE (2009)

A Exact Implementation and Complexity

A.1 Following a single path

Public cycle-selection policies. The base policy may depend on the current state and previous history, but its rule must be the same for every input edge. Earlier conditioned transitions can of course change the state supplied to that rule. The policy may itself be randomized. Conditional on any selected legal cycle, the probability of eventually including the input edge e remains X_e , because each subsequent update preserves this entry’s expectation. Conditioning on eventual inclusion therefore leaves the cycle-selection

probabilities unchanged; only the two endpoint probabilities need to be reweighted by (21). Different policies can induce different output distributions and actual TDSS values. Our guarantees hold for every legal policy, not because all policies give the same law.

Operation and memory bounds. Let E_0 and F_0 be the numbers of nonzero and fractional entries in the initial completed matrix. Then

$$F_0 \leq E_0 \leq \text{nnz}(Q) + n_m + n_k + 1.$$

Every rounding step makes at least one fractional entry integral, so there are at most F_0 steps. Rebuilding or scanning the fractional graph to find a cycle costs $O(E_0)$ per step and gives an $O(E_0^2)$ bound on the total number of arithmetic/graph operations.

A more efficient elementary implementation maintains doubly linked fractional adjacency lists at both endpoints of each edge, together with a list of nonisolated vertices. Starting at any nonisolated vertex, follow an edge that does not immediately return to the previous vertex. Such an edge exists because every encountered vertex has fractional degree at least two. Stop on the first repeated vertex and extract the resulting simple cycle. There are at most $n_m + n_k + 2$ vertices, so this takes $O(n_m + n_k)$ operations. Per-round timestamps record first visits without clearing all stored graph information. Computing the two step lengths and updating the cycle also takes $O(n_m + n_k)$ operations. Newly integral edges are removed from both adjacency lists in constant time each.

Initialization and all rounds therefore use $O(E_0 + F_0(n_m + n_k))$ arithmetic/graph operations and $O(E_0)$ numeric/index cells. The conditioning formula adds a constant number of arithmetic operations per round. This bound describes an efficient cycle policy, not every conceivable policy: deliberately searching for a globally optimized cycle may be more expensive. Encryption follows only one rounding path and never enumerates the tree of possible future matrices.

A.2 Ciphertext representation and decryption

An edge-list representation of R has $D \in \{\lfloor 1/\beta \rfloor, \lceil 1/\beta \rceil\}$ entries and uses

$$O(D(\log(n_m + 1) + \log(n_k + 1)))$$

bits. Scanning this list to find the entry for a given key takes $O(D)$ comparisons. A key-sorted list allows $O(\log(D + 1))$ comparisons, with the cost of constructing that order accounted for separately. Alternatively, a direct array over all n_k columns uses $O(n_k \log(n_m + 1))$ bits and allows one indexed lookup. Constant-time lookup in such an array does not remove the cost of receiving the ciphertext or manipulating its bit strings. Persistent ciphertext storage and encryption working memory are separate resources.

These bounds are polynomial in the explicit matrix dimensions. If n_m and n_k are exponential in the message and key bit lengths, they do not provide bit-length-polynomial encryption or short communication. The structured constructions and representation results elsewhere in the paper address different aspects of that limitation.

A.3 Rational probabilities and bit complexity

Suppose $Q_{ij} = a_{ij}/N$ and $\beta = u/v$ are rational. Every entry of the initial completed matrix is an integer multiple of $1/(Nu)$. Maximal feasible step lengths are distances of current entries from zero or one, so all future states remain on this same grid. State denominators therefore do not grow with the number of rounding steps.

Base endpoint probabilities and conditional probabilities are obtained from grid integers by a constant number of products and divisions. Their numerators and denominators consequently have polynomial bit length in the explicit rational input. The main parameter (38) also has polynomial bit length. If entries are given with different denominators, their product can serve as a common denominator: its *bit length*, not its numerical value, is at most the sum of the input denominator bit lengths.

An exact rational probability a/b can be sampled using fair random bits by drawing an integer uniformly from $\{0, \dots, b-1\}$ and comparing it with a . Sampling that integer by rejection from the smallest power-of-two range of size at least b takes fewer than two trials in expectation. Combining this with the bounded number of rounding steps proves expected polynomial bit complexity. Avoiding rejection of entire matrices does not preclude this elementary rejection sampler for a bounded integer. No finite-bit implementation claim is made for arbitrary unrepresented real probabilities.

A.4 Implementation error versus source-model error

Fix the true source Q . If an implemented channel $\tilde{W}$ and its ideal counterpart W satisfy

$$\kappa_Q = \sum_{i,j} Q_{ij} \text{TV}(\tilde{W}_{ij}, W_{ij}),$$

then each of MRS, KRS, and TDSS can increase by at most κ_Q . Indeed, the two joint input–ciphertext laws are at total variation distance κ_Q , so every fixed attack has success probabilities differing by at most this amount. Taking optimal attacks and predicates preserves the bound. Unlike source-model error, channel error leaves the message prior unchanged and needs no second marginal-error term.

If, at every common reachable history, the round- t transition laws differ by at most ζ_t in total variation, couple the executions until they first disagree. The probability of any disagreement is at most $\sum_t \zeta_t$, which also bounds their output-law distance. Small entrywise floating-point error alone does not establish this premise: the conditional transition divides by X_e , and rare edges can amplify numeric errors. If an approximate implementation can output a matrix omitting the real input edge, its correctness failure must also be stated separately; it is not eliminated by a security-distance bound.

B Recovery Slack and Instance-Specific Benchmarks

B.1 A Refined MRS Bound for Any Admissible Parameter

The dominant-row condition in Section 4 is sufficient for exact optimal MRS. A quantitative bound also describes what happens when that condition is not imposed. Assume

$p_1 = p_\star$ and fix an admissible $\beta \in [q_\star, L]$. Define

$$s = \lfloor p_1/\beta \rfloor, \quad t = p_1 - s\beta, \quad \Delta_\beta = \min \left\{ \beta - t, \sum_{i \neq 1} (p_i - s\beta)_+ \right\}. \quad (52)$$

Then the capacity construction satisfies

$$L \leq \text{Adv}^{\text{mr}} \leq \min\{1, p_1 + \Delta_\beta, \beta \lceil p_1/\beta \rceil\}. \quad (53)$$

To prove the middle bound, let d_i denote the real row counts under the base law μ . The first count is at least s , and all counts are at most $s + 1$. The maximum exceeds d_1 only when $d_1 = s$ and some other count equals $s + 1$; on this event the excess is exactly one. The first event has probability $1 - t/\beta$. For any other row, the probability of reaching $s + 1$ is $(p_i/\beta - s)_+$, so a union bound gives the alternative upper bound $\sum_{i \neq 1} (p_i/\beta - s)_+$. Multiplying the smaller bound by β and using $\beta \mathbb{E}_\mu d_1 = p_1$ proves the claim. The remaining upper bound follows directly from $d_i \leq \lceil p_1/\beta \rceil$.

The endpoint cases cause no exception. If $t = 0$, all other row counts are at most $s = d_1$, and the sum in (52) is zero. If $s = 0$, every row has capacity at most one; nonemptiness of the real output implies that its maximum row count is one. These are bounds under the base law, converted to actual recovery probabilities through the formulas of Section 4, not an identification of the base law with the ciphertext law.

B.2 An Exact Nonzero TDSS Calculation

Consider

$$Q = \frac{1}{100} \begin{pmatrix} 20 & 15 & 10 & 5 & 5 \\ 0 & 5 & 5 & 10 & 5 \\ 0 & 0 & 5 & 5 & 10 \end{pmatrix}. \quad (54)$$

The message marginal is $(11/20, 1/4, 1/5)$ and the key marginal is uniform on five values. Set $\beta = 1/5$. The row counts have expectations $(11/4, 5/4, 1)$, each count is either the floor or the ceiling of its expectation, and their sum is always five. Consequently their complete joint law is forced:

$$(d_1, d_2, d_3) = \begin{cases} (3, 1, 1), & \text{with probability } 3/4, \\ (2, 2, 1), & \text{with probability } 1/4. \end{cases} \quad (55)$$

Here $\beta \sum_i d_i = 1$, so the base law and the actual ciphertext law coincide. The first row always has a maximal count, giving $\text{Adv}^{\text{mr}} = 11/20$, while $\text{Adv}^{\text{kr}} = 1/5$.

For the predicate indicating the first message, the posterior probabilities are $3/5$ and $2/5$. Optimal prediction therefore succeeds with probability $3/5$ in either case. Its prior success probability is $11/20$, so its advantage is $1/20$. The second message has posterior probability either $1/5$ or $2/5$, always below $1/2$; always predicting that this message did not occur remains optimal. The third message has constant posterior probability $1/5$. Every nontrivial predicate on three messages is one of these singleton indicators or its complement. Thus

$$\text{Adv}^{\text{tdss}} = \frac{1}{20}. \quad (56)$$

In contrast, the fractional-row certificate in Theorem 3 equals $\sqrt{3}/20 \approx 0.08660$. This example distinguishes an exact advantage from a uniform analytic certificate, whose value here exceeds the actual advantage 0.05.

B.3 Unequal Component Weights Can Improve a Particular Source

The equal-weight capacity family does not exhaust the column-unique matrix decompositions. Let

$$Q = \frac{1}{40} \begin{pmatrix} 9 & 3 & 6 & 2 \\ 3 & 9 & 2 & 6 \end{pmatrix}, \quad p = (1/2, 1/2), \quad q = (3/10, 3/10, 1/5, 1/5). \quad (57)$$

Use two ciphertexts with decryption vectors $g = (1, 2, 1, 2)$ and $g' = (2, 1, 2, 1)$. Give them actual probabilities $3/4$ and $1/4$, and retain key posterior q under each ciphertext. Equivalently, their component entries are $A_g(i, j) = (3/4)q_j \mathbf{1}\{g(j) = i\}$ and $A_{g'}(i, j) = (1/4)q_j \mathbf{1}\{g'(j) = i\}$. These column-unique matrices sum to (57), so the matrix-to-channel conversion defines a correct encryption scheme. Under either ciphertext each message receives one key of mass $3/10$ and one of mass $1/5$. Therefore

$$\text{Adv}^{\text{mr}} = 1/2, \quad \text{Adv}^{\text{kr}} = 3/10, \quad C \perp M, \quad C \perp K. \quad (58)$$

At $\beta = 3/10$, however, the capacity construction has each real row count in $\{1, 2\}$ and the total count in $\{3, 4\}$. Its maximum row count is always two, so $\text{Adv}^{\text{mr}} = 3/5$. The sufficient parameter rule of Theorem 4 selects $\beta_* = 1/2$ to retain optimal MRS. Thus a more general unequal-weight decomposition can simultaneously attain the two prior recovery bounds where that rule does not.

One cannot in general eliminate unequal weights by splitting a real key into equal-weight virtual copies. If different copies of the same key decrypt the same ciphertext to different messages, the resulting object does not define a deterministic correct decoder for the real key. Any such transformation must preserve the column-uniqueness condition after the virtual copies are identified.

B.4 A Finite Linear Program for Small Instances

General column-unique decompositions also yield exact finite optimization benchmarks. Index decryption vectors by $g \in \{1, \dots, n_m\}^{n_k}$; a column with zero posterior mass can be assigned an arbitrary message without changing the experiment. Introduce $x_{g,j} = \Pr[C = g, K = k_j] \geq 0$. Consistency with the source is precisely

$$\sum_{g: g(j)=i} x_{g,j} = Q_{ij} \quad \text{for all } i, j. \quad (59)$$

Set $r_{g,i} = \sum_{j:g(j)=i} x_{g,j}$. To minimize KRS under an MRS budget A , introduce variables u_g, v_g and solve

$$\begin{aligned}
& \text{minimize} && \sum_g v_g \\
& \text{subject to} && (59), \quad x_{g,j} \geq 0, \\
& && u_g \geq r_{g,i} \quad \text{for all } g, i, \\
& && v_g \geq x_{g,j} \quad \text{for all } g, j, \quad \sum_g u_g \leq A.
\end{aligned} \tag{60}$$

The two maxima encoded by u_g and v_g are exactly the contributions to MRS and KRS. Conversely, any feasible solution specifies a correct channel through its column-unique components.

This characterization is not restricted to equal-weight constructions. Ciphertexts with the same decryption vector can be merged without increasing either recovery probability: for nonnegative vectors, the maximum of a sum is no larger than the sum of the maxima. Hence every correct scheme admits a representative in this finite optimization with no worse recovery guarantees. The price is its size: there are n_m^{nk} decryption vectors. The program is an exact benchmark for small sources, not a general efficient algorithm in the message and key bit lengths; the constructive complexity guarantees are discussed separately in Appendix A.

C Sharpness and Estimation in the Robustness Bounds

C.1 The factor two is attainable

Consider the design and true distributions

$$\widehat{Q} = \begin{pmatrix} 1/4 & 1/2 \\ 0 & 1/4 \end{pmatrix}, \quad Q_h = \begin{pmatrix} 1/4 - h & 1/2 \\ 0 & 1/4 + h \end{pmatrix}, \quad 0 \leq h \leq 1/4. \tag{61}$$

Fix a channel that encrypts both diagonal inputs to one diagonal-matching ciphertext and the upper-right input to a singleton ciphertext. The two ciphertexts have the evident correct decoding rules. Under $\widehat{Q}$, both the prior and the post-ciphertext optimal prediction success for the nonconstant message predicate are $3/4$. Thus $\text{Adv}_{\widehat{Q}}^{\text{tdss}} = 0$. Under Q_h , the latter becomes $3/4 + h$, whereas the prior baseline becomes $3/4 - h$. Hence $\text{Adv}_{Q_h}^{\text{tdss}} = 2h$, and both $\text{TV}(\widehat{Q}, Q_h)$ and the message-marginal distance equal h . Equality holds in (43).

This example also distinguishes zero TDSS from perfect message secrecy. Under $\widehat{Q}$, the ciphertexts have different message posteriors, but neither improves the optimal decision for the only nonconstant predicate, up to complementation. TDSS measures an improvement in decision success, not equality of every posterior with the prior.

C.2 Overestimation can still increase TDSS

Let

$$A_h = \begin{pmatrix} 1/4 & 1/2 + h \\ 0 & 1/4 \end{pmatrix}, \quad Q'_h = \begin{pmatrix} 1/4 - h & 1/2 + h \\ 0 & 1/4 \end{pmatrix}, \quad 0 < h \leq 1/4. \tag{62}$$

Use the same channel as above. We have $A_h \geq Q'_h$, so the underestimated mass u is zero. The normalized design source $A_h/(1+h)$ has TDSS zero: its post-ciphertext and prior prediction probabilities are both $(3/4 + h)/(1+h)$. Under the true source, these probabilities are $3/4 + h$ and $3/4$, respectively. Its TDSS is therefore h , exactly the row-overestimate term $v_M = h$. Removing v_M from (47) is invalid even when the design matrix only overestimates individual entries.

C.3 Obtaining an error certificate

The robustness theorem does not make accurate distribution estimation free. Suppose Q has s positive entries and $\widehat{Q}_N$ is its empirical distribution from N independent samples. For each cell e , $\mathbb{E}|(\widehat{Q}_N)_e - Q_e| \leq \sqrt{Q_e(1-Q_e)/N}$. Summing and using Cauchy–Schwarz gives

$$\mathbb{E} \text{TV}(Q, \widehat{Q}_N) \leq \frac{1}{2} \sqrt{s/N}. \quad (63)$$

A huge unstructured joint alphabet can therefore require many training samples. Moreover, the estimation penalty can dominate a small nominal TDSS certificate.

A conditional representation can yield a more interpretable bound. Write $Q_{ij} = q_j \mu_j(i)$ and $\widehat{Q}_{ij} = \widehat{q}_j \widehat{\mu}_j(i)$. Inserting $q_j \widehat{\mu}_j(i)$ and applying the triangle inequality yields

$$\text{TV}(Q, \widehat{Q}) \leq \text{TV}(q, \widehat{q}) + \sum_j q_j \text{TV}(\mu_j, \widehat{\mu}_j). \quad (64)$$

When the key marginals agree, this is an equality. Thus conditional model errors are weighted by the true key probabilities; a very rare key need not have the same absolute model accuracy as a common one.

If a training sample is independent of the challenge pair, the stability theorem applies after fixing that sample and the resulting channel. If the released training information depends on the challenge secrets, it is side information and the source and prediction baseline must instead be conditioned on it. Independence cannot be restored merely by referring to the released information as a model.

D Replayable Descriptions and Rational Compression Weights

D.1 A sufficient condition for safe public descriptions

Suppose a public descriptor c determines a matrix $R = G(c)$. A sufficient condition for preserving the original security is the existence of nonnegative, input-independent base weights $\rho(c)$ such that

$$\Pr[C = c \mid e] = \frac{\beta \rho(c) G(c)_e}{Q_e}, \quad \sum_{c: G(c)=R} \rho(c) = \mu(R). \quad (65)$$

Indeed, the joint mass of an input edge e and descriptor c is $\beta \rho(c) G(c)_e$. Given c , the input edge is uniform among the edges of $G(c)$, exactly as it is when given only that matrix in the original scheme. The descriptor therefore conveys no additional input

information once R is known. Summing over descriptors with the same matrix shows that MRS, KRS, and every fixed predicate's actual advantage are unchanged. This argument applies to capacity matrices as well as matchings.

D.2 Replaying endpoint choices, not revealing sampling coins

Fix a deterministic public rule for selecting and orienting a cycle at each state. Record whether each rounding step takes its positive or negative endpoint, producing a transcript $\tau = (b_1, \dots, b_h)$. Starting from the public initial matrix, a decoder can replay these updates without knowing the input edge or any conditional sampling probability. In the two-by-two example of Section 3, one bit distinguishes the diagonal and off-diagonal outputs; a larger instance simply requires more steps.

If $\rho(\tau)$ is the base probability of a transcript, telescoping the conditional transition ratios gives

$$\Pr[\tau \mid e] = \frac{\beta \rho(\tau) R(\tau)_e}{Q_e}. \quad (66)$$

Consequently the transcript satisfies (65). There are at most F_0 endpoint bits, where F_0 is the initial number of fractional entries; padding to F_0 bits is possible. Randomized cycle selection requires additional replay information. Although this can trade decryption time for representation space, F_0 may be $\Theta(n_m n_k)$, so a transcript need not be short.

The transcript is not the same as the raw uniform coins used to select its branches. The conditional branch threshold depends on the true input edge. For example, take the uniform 4×2 source with $\beta = 1/2$. Suppose the first cycle uses rows 1 and 2 and both real columns, with the positive direction increasing entries $(1, 1)$ and $(2, 2)$. Its base positive-branch probability is $1/2$. Conditioned on input $(1, 1)$, that probability is one; conditioned on the off-cycle input $(3, 2)$, it remains $1/2$.

Let W be the fresh uniform number used for this comparison. Consider a continuation with terminal matrix $R = \{(1, 1), (3, 2)\}$, which is reachable after the positive branch. If both R and the event $W \geq 1/2$ are revealed, the true edge cannot be $(3, 2)$: that input would have taken the negative branch, permanently setting $(1, 1)$ to zero. Input $(1, 1)$ is still compatible. Thus an additional coin value can distinguish edges that the matrix alone leaves equiprobable. Independence of a coin before encryption does not make its later disclosure harmless.

Similarly, publishing a seed that generates trial matrices together with the index of the first trial containing the input also reveals earlier failures. Those failures may rule out otherwise compatible inputs. This is an obligation to analyze such seed constructions, not an impossibility result for all of them.

D.3 Changing a distribution versus lossless compression

If the actual ciphertext distribution has N_R positive-probability matrices, an injective fixed-length description requires at least $\lceil \log_2 N_R \rceil$ bits. A prefix-free variable-length description has expected length at least $H_\nu(R)$, where $\nu(R) = \beta D(R) \mu(R)$ is the actual ciphertext distribution, not the base distribution. For example, all injections of n_k keys into n_m messages form a family of size $(n_m)_{n_k}$. Preserving that family losslessly

can require roughly $n_k \log_2 n_m$ bits, whereas Theorem 6 changes the family and its probabilities.

These representation bounds should not be confused with universal encryption bounds. Correctness implies $H(C) \geq H(M | K)$, since $H(M | K) = I(M; C | K) \leq H(C)$. If one positive-probability key has r possible messages, at least r ciphertext symbols are necessary. Neither observation forces the entire original decoding-matrix family to remain present.

There is also a probability-granularity issue: a deterministic expansion of a uniform fixed-length b -bit seed gives only event probabilities that are integer multiples of 2^{-b} , and cannot represent probability $1/3$ exactly. A short index sampled with nonuniform rational probabilities is not equivalent to a short uniform seed. Computational pseudorandomness cannot be substituted without changing the information-theoretic model.

D.4 A bounded-bit rational certificate

Exact preservation of original second moments can retain unnecessarily large denominators. Instead, retain only the inequalities used in the TDSS proof. Consider the finite set of zero-one matrices supported on $\text{supp}(Q)$ with at most one entry per column, row counts in $\{\ell_i, \ell_i + 1\}$, and total counts in $\{d, d + 1\}$, where $\ell_i = \lfloor p_i / \beta_* \rfloor$ and $d = \lfloor 1 / \beta_* \rfloor$. A row with zero fractional remainder is fixed at ℓ_i . Also impose the main-parameter supportwise condition: when $p_\star > q_\star$, a fixed row attaining $p_\star$ dominates all others; otherwise, all row counts are at most one. Set $B_i(R) = d_i(R) - \ell_i \in \{0, 1\}$ and $r_i = p_i / \beta_* - \ell_i$.

Require weights u_R to satisfy

$$\begin{aligned} u_R \geq 0, \quad \sum_R u_R = 1, \quad \sum_R u_R R_{ij} = Q_{ij} / \beta_*, \\ \sum_R u_R B_i(R) B_h(R) \leq r_i r_h \quad (i < h). \end{aligned} \quad (67)$$

The original rounding distribution proves feasibility. An extreme point of this finite feasible polytope has at most T_0 positive weights. Otherwise, on its positive support there would be a nonzero direction preserving every active constraint, and sufficiently small displacements in both directions would remain feasible, contradicting extremality. The edge constraints preserve the row means. The last constraints imply the required nonpositive pairwise covariances of the B_i . Together with the supportwise total-count constraint, they therefore give the same universal TDSS certificate, although not every original variance.

Suppose the initial expanded matrix lies on a $1/H$ grid. All right-hand sides in (67) have a common denominator dividing H^2 . Crucially, all coefficients other than those expressing nonnegativity are zero or one: the second-order features are $B_i B_h$, not the potentially larger $d_i d_h$. At an extreme point, the positive weights are determined by an invertible subsystem of order at most T_0 . Cramer's rule and the determinant bound for zero-one matrices give numerator and denominator bit lengths

$$O(\log(H + 1) + T_0 \log(T_0 + 1)). \quad (68)$$

Thus a finite public table with polynomial explicit-input bit length exists. The candidate family may still be enormous; this argument does not provide an efficient procedure for finding the table from a succinct source.

E Succinct Constructions for Structured Sources

The examples below obtain short ciphertexts and efficient operations by using additional source structure. They do not implement the full output distribution of arbitrary cycle rounding. Rather, they choose a succinct family whose first- and second-order properties can be analyzed directly.

E.1 Uniform independent sources and affine maps

Let $n_m > 1$ be a prime power. Messages are uniform in the finite field $\mathbb{F}_{n_m}$, and keys are uniform on $n_k \leq n_m$ distinct public field labels $\tau_1, \dots, \tau_{n_k}$, independently of the message. To encrypt (m, k_j) , choose a uniformly from $\mathbb{F}_{n_m} \setminus \{0\}$, set $b = m - a\tau_j$, and output (a, b) . Decryption returns $a\tau_j + b$.

The uniform base distribution on the $n_m(n_m - 1)$ affine maps includes each input edge with probability $1/n_m = Q_{mj}/(1/n_k)$, so encryption is exactly conditional sampling from that family. Each ciphertext has n_k distinct, equiprobable candidate messages and keys. Hence $\text{Adv}^{\text{mr}} = \text{Adv}^{\text{kr}} = 1/n_k$. For two distinct key labels their images are a uniform ordered pair of distinct field elements. For a predicate of prior probability θ , the corresponding candidate indicators have covariance $-\theta(1 - \theta)/(n_m - 1)$. Thus the posterior predicate probability Θ_C satisfies

$$\text{Var}(\Theta_C) = \theta(1 - \theta) \frac{n_m - n_k}{n_k(n_m - 1)}, \quad \text{Adv}^{\text{tdss}} \leq \frac{1}{2} \sqrt{\frac{n_m - n_k}{n_k(n_m - 1)}}. \quad (69)$$

For the second inequality, the function $x \mapsto \max\{x, 1 - x\}$ is 1-Lipschitz, so prediction advantage is at most $\mathbb{E}|\Theta_C - \theta| \leq \sqrt{\text{Var}(\Theta_C)}$. The candidates are sampled without replacement in pairs, not independently.

A ciphertext consists of two field elements. Encryption and decryption use a constant number of field operations, plus uniform sampling. Bit-length efficiency additionally requires efficient field arithmetic and computable key labels. For binary fields the length is exactly $2 \log_2 n_m$ bits. For example, $n_m = 2^{128}$ and $n_k = 2^{20}$ give 256-bit ciphertexts, recovery success 2^{-20} , and a TDSS certificate below 2^{-11} .

E.2 An explicitly shared label and independent conditional details

Let Z range over a subset of $\{0, 1\}^h$ with distribution w_z . Given $Z = z$, let A, J be independent uniform indices in $\{0, \dots, B - 1\}$, where $B = 2^b$, and sample a payload X according to $\mu_{z,A}$, independently of J . Define $M = (Z, A, X)$ and $K = (Z, J)$, so

$$Q((z, a, x), (z', j)) = \mathbf{1}\{z = z'\} \frac{w_z}{B^2} \mu_{z,a}(x). \quad (70)$$

The group label (z, a) is part of the message representation. Equal payloads in different groups must not be identified as the same message. Relabeling may reveal this structure but cannot create equal-mass groups or conditional independence. For instance, three atoms of masses 0.4, 0.4, 0.2 cannot be partitioned deterministically into two groups of mass 1/2.

This is stronger than assuming a latent common cause. Both the message and the key must determine the same label Z exactly. The shared-label part of this assumption arises naturally from two independent samples from a chosen population, each recorded together with its population label; the uniform-index and codec conditions remain additional requirements. Two noisy observations of an unrecorded common state do not generally have this property. In particular, if every message–key pair has positive probability, any such deterministic shared label must be constant. Common model architecture or semantic similarity does not by itself establish (70).

Choose $\ell \geq \max\{1, h + b\}$. For each group, assume a deterministic decoder $D_{z,a} : \mathbb{F}_{2^\ell} \rightarrow \mathcal{X}_{z,a}$ and randomized inverse encoder $E_{z,a}$ such that

$$D_{z,a}(E_{z,a}(x)) = x, \quad X \sim \mu_{z,a} \implies E_{z,a}(X) \sim \text{Unif}(\mathbb{F}_{2^\ell}). \quad (71)$$

Equivalently, each inverse image has size $2^\ell \mu_{z,a}(x)$ and the encoder samples uniformly from it. For an efficiency claim, these codecs must be implemented by a uniform short algorithm indexed by (z, a) , not by exponentially many hidden lookup tables. Let $e(z, a)$ be a public injection from group labels to field elements.

Encryption of (z, a, x) under (z, j) computes

$$\begin{aligned} U &\leftarrow E_{z,a}(x), & \rho &\leftarrow \text{Unif}(\mathbb{F}_{2^\ell}), & C &= (\rho, T, V). \\ T &= a - j \pmod{B}, & V &= U + \rho e(z, a), \end{aligned} \quad (72)$$

The value zero is included when sampling ρ . Given (z, j) , decryption sets $a = T + j \pmod{B}$ and returns $(z, a, D_{z,a}(V - \rho e(z, a)))$. The realized label Z is not transmitted; the genuine key provides it. Inputs outside the source support can use a separate fallback encoding.

Proposition 2. *For the source (70), the scheme is correct and has*

$$\text{Adv}^{\text{mr}} = \text{Adv}^{\text{kr}} = \frac{\max_z w_z}{B} = L(Q), \quad \text{Adv}^{\text{tdss}} \leq \frac{1}{2} \sqrt{\frac{\sum_z w_z^2}{B}} \leq \frac{1}{2} \sqrt{L(Q)}. \quad (73)$$

Its ciphertext length is $2\ell + b$ bits.

Proof. Correctness follows from (71). For each fixed key, T is uniform, and conditioned on T, ρ , the value V is uniform. The ciphertext is therefore uniform and independent of the key. Its key posterior remains w_z/B . Distinct keys decrypt into distinct groups: their z values differ, or their distinct j values yield distinct a values. Thus message and key recovery both have success $\max_z w_z/B$. Since each message probability is $w_z \mu_{z,a}(x)/B$, this is also $L(Q)$.

For a fixed predicate f , summing the key posterior and reindexing by a gives its posterior probability

$$\Theta_C = \sum_z \frac{w_z}{B} \sum_a f(z, a, D_{z,a}(V - \rho e(z, a))). \quad (74)$$

For two distinct group labels with field images g, g' , the map $(\rho, V) \mapsto (V - \rho g, V - \rho g')$ is invertible. Therefore the two resulting seeds are independent and uniform. Applying deterministic decoders and predicates preserves pairwise independence. Every indicator has variance at most $1/4$, whence $\text{Var}(\Theta_C) \leq \sum_z B(w_z/B)^2/4$. The same Lipschitz and variance argument as for (69) proves the TDSS bound. Finally $\sum_z w_z^2 \leq \max_z w_z$.

If w is nonuniform, the candidate probabilities are nonuniform. The scheme therefore fits the general weighted matching decomposition, but need not belong to the scalar-times-zero-one capacity subclass. Its algorithms do not need to evaluate w_z ; these probabilities enter the analysis.

E.3 A complete four-element example

Let Z, A, J be independent fair bits and let X be independent, with $\Pr[X = 0] = 3/4$. Encode $X = 0$ uniformly into 00, 01, 10, and encode $X = 1$ as 11. The seed is uniform on four values, and the decoder maps the first three values to zero and the last to one. Represent $\mathbb{F}_4$ by two bits, with addition given by XOR and multiplicative identity 01. This is field arithmetic, not integer arithmetic modulo four.

For message $1|0|0$ and key $1|1$, suppose this execution chooses $U = 01$ and $\rho = 01$. With $e(z, a) = za$, it produces $T = 1$, $V = 01 \oplus 10 = 11$, and ciphertext $(01, 1, 11)$. The true key recovers $a = 0$, then $U = 01$, then $X = 0$. The complete candidate list is

Key	Group index a	Recovered seed	Message
0 0	1	10	0 1 0
0 1	0	11	0 0 1
1 0	1	00	1 1 0
1 1	0	01	1 0 0

The parameter ρ must still be freshly uniform in the actual algorithm; this execution does not justify fixing it to 01. Here $\text{Adv}^{\text{mr}} = \text{Adv}^{\text{kr}} = 1/4$ and the TDSS certificate is $1/4$, not necessarily the actual advantage.

E.4 Exact codecs without large tables

Suppose that, given (z, a) , the payload consists of m independent bits, with bit i equal to one with probability $H_i(z, a)/2^t$. Assume the integers $0 \leq H_i(z, a) \leq 2^t$ are computed by a short uniform circuit. Set $\ell = \max\{1, mt, h + b\}$. Split the first mt seed bits into t -bit integers U_i and decode $x_i = \mathbf{1}\{U_i < H_i(z, a)\}$. To encode an admissible x , sample each U_i uniformly from the corresponding interval and choose any remaining seed bits independently and uniformly. This satisfies (71) exactly. It uses m circuit evaluations, bounded-integer sampling, and a constant number of field operations. With polynomial-size circuits and parameters, its expected bit complexity is polynomial in the description length; elementary field multiplication costs $O(\ell^2)$ bit operations.

Arbitrary real-valued conditional probabilities do not necessarily admit an exact finite uniform-seed codec. A neural probability model does not automatically provide either the needed inverse sampler or its complexity bound. Approximation requires separate statistical and implementation-error accounting.

E.5 The realized label is not free public information

Take a fair Z , uniform independent A, J in B positions, and no payload. Sending only $T = A - J \pmod{B}$ provides perfect message secrecy: T is uniform and independent of (Z, A) , while the key supplies Z, J for correct decryption. If Z is additionally transmitted, the balanced predicate “ $Z = 1$ ” is completely revealed and TDSS becomes $1/2$. A public rule for computing labels is different from publishing their realized values. When an application already gives the adversary Z , the security experiment must condition on it.

E.6 Sparse nonnegative mixtures and signed approximation errors

If $Q = \sum_z w_z Q_z$ and each component has a correct scheme, encryption of input (i, j) selects component z with probability $w_z(Q_z)_{ij}/Q_{ij}$ and outputs its labeled ciphertext (z, C_z) . Selecting z according to the prior w_z would be incorrect when the chosen component cannot produce the input. With visible labels, MRS and KRS are the weighted averages of component success probabilities. For a fixed predicate, write b_z for its component prior-only success probability and b for the global one. Then

$$\text{Adv}_Q^{\text{tdss}}(f) = \sum_z w_z \text{Adv}_{Q_z}^{\text{tdss}}(f) + \sum_z w_z b_z - b. \quad (75)$$

Each component advantage is evaluated for its component scheme, whereas the left-hand side is evaluated for the tagged mixture. The last term is nonnegative and measures information from the label alone. Two components concentrated on different fixed messages each have TDSS zero, but their equal mixture with a visible label has TDSS $1/2$.

For $Q = (1 - \alpha)P + \alpha S$, a general bound is

$$\text{Adv}_Q^{\text{tdss}} \leq (1 - \alpha) \text{Adv}_P^{\text{tdss}} + \alpha. \quad (76)$$

For any fixed predicate, the post-ciphertext success is at most $(1 - \alpha)(b_P + \text{Adv}_P^{\text{tdss}}) + \alpha$. The global prior can use the best constant guess under P , achieving at least $(1 - \alpha)b_P$. Subtraction and maximization prove the claim.

If S has s edges, explicit processing of that component is polynomial in the description length provided s , its weight bit lengths, the core scheme, support tests, and input-probability evaluation all satisfy the corresponding bounds. However, some sparse edge has source mass at least α/s , so $L(Q) \geq \alpha/s$. Constant probability mass on only polynomially many edges itself limits the recovery security achievable. For a signed residual $Q = \hat{Q} + \Delta$, use stability with $\text{TV}(Q, \hat{Q}) = \|\Delta\|_1/2$ instead; negative entries are not sampleable mixture components. If the message marginal is preserved, only one total-variation penalty is needed for TDSS.

F Recovery Tradeoffs and Worst-Case Limits

The simultaneous guarantees of Theorem 4 are worst-case guarantees over joint sources. They do not imply that one choice of the capacity parameter minimizes every security

measure for every source. This appendix supplies the proofs and examples behind the limitations summarized in Section 4: exact optimality for message recovery can force a loss in key-recovery security, and a general TDSS guarantee cannot improve on the square-root scale. Both statements concern all correct schemes, not only the rounding construction.

F.1 An Exact Two-Message Tradeoff

Consider the correlated source

$$Q = \begin{pmatrix} 1/3 & 1/6 & 0 \\ 0 & 1/6 & 1/3 \end{pmatrix}. \quad (77)$$

The messages are uniform on two values and the keys are uniform on three, so $L = 1/2$ and $q_\star = 1/3$. Suppose first that a correct scheme attains $\text{Adv}^{\text{kr}} = 1/3$. The largest key posterior is at least $1/3$ for every positive-probability ciphertext. Equality on average therefore forces each such posterior to be uniform on all three keys. Deterministic decryption assigns these keys to two messages. At least two keys must decrypt to the same message, giving $\text{Adv}^{\text{mr}} \geq 2/3$.

More generally, for any correct scheme with two messages and three keys, fix a ciphertext and partition its positive-posterior keys according to their decrypted messages. At least one of the two parts contains at most one key. The smaller message-posterior mass is consequently no greater than the largest key-posterior mass. The larger message-posterior mass is one minus the smaller mass, so averaging over ciphertexts gives

$$\text{Adv}^{\text{mr}} + \text{Adv}^{\text{kr}} \geq 1. \quad (78)$$

The construction of Section 3 attains both endpoints $(\text{Adv}^{\text{mr}}, \text{Adv}^{\text{kr}}) = (1/2, 1/2)$ and $(2/3, 1/3)$, using $\beta = 1/2$ and $\beta = 1/3$, respectively. To interpolate, choose between these two complete schemes using an input-independent Bernoulli selector with a chosen bias and include the choice as a public ciphertext tag. Conditional on the tag, the adversary faces the selected scheme; hence both recovery probabilities are the corresponding convex combinations. It follows that, under a budget $\text{Adv}^{\text{kr}} \leq b$ for $b \in [1/3, 1/2]$, the minimum possible MRS for (77) is exactly $1 - b$.

There is also a direct semantic cost. For a uniform binary message, the only non-constant predicates are the message indicator and its complement. Thus $\text{Adv}^{\text{tdss}} = \text{Adv}^{\text{mr}} - 1/2$. Along this optimal recovery frontier, reducing KRS necessarily increases the actual TDSS advantage, not merely a bound used in its analysis.

F.2 Proof of Theorem 2 and the Divisibility Effect

The uniform-marginal recovery cost stated in Section 4 follows from a counting argument that does not require independence.

Proof. For every positive-probability ciphertext, the largest message posterior is at least $1/n_m$. If its average is $1/n_m$, each message posterior must therefore equal $1/n_m$ for every such ciphertext. Put $s = \lfloor n_k/n_m \rfloor$. Since deterministic decryption partitions at most n_k

positive-posterior keys among the n_m messages, some message is assigned at most s keys. Their posterior masses sum to $1/n_m$, so at least one key has posterior mass at least $1/(n_ms)$. This pointwise statement implies the same lower bound on KRS.

For achievability, use the capacity construction with $\beta = 1/(n_ms)$. The inequalities $1/n_k \leq \beta \leq 1/n_m$ make this an admissible parameter. Every real row has the integer expected count $p_i/\beta = s$; its rounded count is therefore exactly s . The total real count is n_ms , and every ciphertext has message posterior $1/n_m$ on each message. Consequently, $\text{Adv}^{\text{mr}} = 1/n_m$ and the ciphertext is independent of the message. The key-recovery formula from Section 4 gives $\text{Adv}^{\text{kr}} = \beta$. No step of this argument assumes independence of the source variables.

The divisibility effect in this theorem has a direct interpretation. Every message must receive exactly s candidate keys in the achieving construction, so each ciphertext retains n_ms keys in total. If n_m divides n_k , all keys remain candidates and the prior key-recovery bound $1/n_k$ is attained. Otherwise, some keys are excluded by each ciphertext, and the remaining n_ms keys have equal posterior mass $1/(n_ms)$. Although the message posterior is unchanged, the key posterior need not be. Perfect message secrecy therefore does not imply perfect key secrecy.

F.3 A Posterior-Support Lower Bound for TDSS

The next obstruction is different: a ciphertext and n_k possible keys can specify at most n_k candidate messages. If the prior is uniform on a much larger space, some fixed predicate must benefit from this restricted posterior support.

Theorem 7 (TDSS lower bound from posterior support). *Suppose the message is uniform on n_m values and the key has at most $n_k \leq n_m$ possible values. Every correct scheme with deterministic decryption satisfies*

$$\text{Adv}^{\text{tdss}} \geq \frac{1}{2} [a_{n_k} - a_{n_m}]_+, \quad a_r := \frac{1}{r} \mathbb{E} \left| \sum_{i=1}^r \sigma_i \right|, \quad (79)$$

where the σ_i are independent uniform signs in $\{-1, 1\}$ and $[x]_+ = \max\{x, 0\}$.

Proof. Write π_c for the message posterior under ciphertext c . Correctness implies $|\text{supp}(\pi_c)| \leq n_k$: each possible key decrypts to at most one message. For a probability vector v , define $\kappa(v) = \mathbb{E}_{\sigma} |\sum_i \sigma_i v_i|$. If v has at most n_k nonzero coordinates, pad it with zeros to length n_k . The average of all coordinate permutations of this vector is the uniform vector on n_k coordinates. Convexity of absolute value and permutation invariance of κ therefore imply

$$\kappa(v) \geq \kappa((1/n_k, \dots, 1/n_k)) = a_{n_k}. \quad (80)$$

To find a hard predicate, independently assign a sign σ_i to every message and let $f_{\sigma}(m_i) = (1 + \sigma_i)/2$. For a fixed sign vector, optimal posterior prediction succeeds

with probability $1/2 + (1/2)\mathbb{E}_C|\sum_i \sigma_i \pi_C(i)|$. The prior success probability is $1/2 + (1/2)|\sum_i \sigma_i/n_m|$. Averaging their difference over the signs yields

$$\mathbb{E}_\sigma \text{Adv}^{\text{tdss}}(f_\sigma) = \frac{1}{2} (\mathbb{E}_C \kappa(\pi_C) - a_{n_m}) \geq \frac{1}{2} (a_{n_k} - a_{n_m}). \quad (81)$$

At least one fixed predicate attains this average, and every predicate advantage is non-negative. This proves the claim. The random signs are only a proof device: the resulting predicate is fixed before the challenge ciphertext is observed.

This order of quantifiers is essential. Small posterior support can make a ciphertext-dependent set highly informative, but such a set is not an admissible TDSS challenge predicate. Averaging over random signs instead produces one predicate, depending only on the public scheme and source, whose gain is large on average over the actual ciphertexts. The argument therefore proves a lower bound for the stated prediction game, without replacing it by a stronger posterior-distance requirement.

The constants in this bound have the exact expression

$$a_r = \frac{\binom{2h}{h}}{2^{2h}}, \quad h = \lfloor r/2 \rfloor, \quad a_r \sim \sqrt{\frac{2}{\pi r}}. \quad (82)$$

For completeness, summing over the number of positive signs and using $(2j-r)\binom{r}{j} = r[\binom{r-1}{j-1} - \binom{r-1}{j}]$ gives the exact identity by telescoping. The asymptotic expression follows from the standard central-binomial-coefficient estimate.

For uniform independent sources with $n_k \rightarrow \infty$ and $n_m/n_k \rightarrow \infty$, we have $L = q_\star = 1/n_k$ and $a_{n_m} = o(a_{n_k})$. Theorem 7 and (82) therefore imply the worst-case lower bound (37). This conclusion does not give a positive square-root lower bound for every source. Indeed, $a_{2h} = a_{2h+1}$, so (79) can be zero even when $n_m > n_k$.

F.4 What Cycle Selection Can Still Improve

Theorem 3 certifies every permitted public cycle-selection rule. It does not assert that these rules induce the same output law. For a fixed predicate, edge expectations determine the mean of the signed row-count sum, but its expected absolute value depends on the joint distribution of the counts. Different rules can therefore have different actual TDSS advantages while sharing the same certificate.

A useful benchmark makes the distinction precise. Consider a uniform independent source with $n_m \geq n_k$ and the equal-weight matching construction at $\beta = L = 1/n_k$. The posterior is uniform on its n_k candidate messages. If, in addition, the candidate set is uniform among all n_k -subsets, then for a predicate holding on a fraction θ of messages its posterior probability H is a sample average without replacement. Hence

$$\text{Var}(H) = \theta(1-\theta) \frac{n_m - n_k}{n_k(n_m - 1)}, \quad \text{Adv}^{\text{tdss}} \leq \frac{1}{2} \sqrt{\frac{n_m - n_k}{n_k(n_m - 1)}} \quad (n_m > 1). \quad (83)$$

The bound follows because $\mathbb{E}H = \theta$ and the function $x \mapsto \max\{x, 1-x\}$ is 1-Lipschitz, so the prediction gain is at most $\mathbb{E}|H - \theta| \leq \sqrt{\text{Var}(H)}$. Both uniform subset selection

and a uniform posterior within that subset are needed for this argument; uniform support alone does not suffice.

When $n_m \gg n_k$, this benchmark changes the constant at the square-root scale. When $n_m = n_k + 1$, it is $1/(2n_k)$, and when $n_m = n_k$ it is zero. Such improvements in other regimes can follow from source structure or a sharper analysis, and cannot automatically be attributed to a cycle heuristic. Conversely, the worst-case lower bound does not reduce every possible instance-specific improvement to a constant factor.