Malleable coding with edit-distance cost

The MIT Faculty has made this article openly available. Please share how this access benefits you. Your story matters.

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>As Published</td>
<td><a href="http://dx.doi.org/10.1109/ISIT.2009.5205494">http://dx.doi.org/10.1109/ISIT.2009.5205494</a></td>
</tr>
<tr>
<td>Publisher</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>Version</td>
<td>Final published version</td>
</tr>
<tr>
<td>Accessed</td>
<td>Wed Jan 23 08:30:31 EST 2019</td>
</tr>
<tr>
<td>Citable Link</td>
<td><a href="http://hdl.handle.net/1721.1/60357">http://hdl.handle.net/1721.1/60357</a></td>
</tr>
<tr>
<td>Terms of Use</td>
<td>Article is made available in accordance with the publisher's policy and may be subject to US copyright law. Please refer to the publisher's site for terms of use.</td>
</tr>
<tr>
<td>Detailed Terms</td>
<td></td>
</tr>
</tbody>
</table>


t our interest is in average performance rather than worst-case performance of coding schemes.

Malleable coding with edit-distance cost is described in greater detail in [6], and we refer to this easily-accessible document for proofs of our results. A distinct formulation of malleable coding is studied in [7].

II. PROBLEM STATEMENT

Consider storage medium symbols drawn from the finite alphabet \( \mathcal{V} \). Unlike most source coding problems, the alphabet itself is relevant, not just the cardinality of sequences drawn from it; an abstract set of indices is not appropriate. It is natural to measure all rates in numbers of symbols from \( \mathcal{V} \).1

We require an edit distance [8] defined for \( \mathcal{V}^* \), the set of all finite sequences of elements of \( \mathcal{V} \). An example of an edit distance is the Levenshtein distance, which is constructed from insertion, deletion, and substitution operations.

Definition 1: An edit distance, \( d(\cdot, \cdot) \), is a function from \( \mathcal{V}^* \times \mathcal{V}^* \) to \([0, \infty)\), defined by a set of edit operations. The edit operations are a symmetric relation on \( \mathcal{V}^* \times \mathcal{V}^* \). The edit distance between \( a \in \mathcal{V}^* \) and \( b \in \mathcal{V}^* \) is 0 if \( a = b \) and is the minimum number of edit operations needed to transform \( a \) into \( b \) otherwise.

We define variable-length and block coding versions of our problem together, drawing distinctions only where necessary. Symbols are reused to conserve notation; context should make things clear. Let \( \{(X_i, Y_i)\}_{i=1}^{\infty} \) be a sequence of independent drawings of a pair of random variables \((X, Y)\), \( X \in \mathcal{X}, Y \in \mathcal{Y} \), where \( \mathcal{W} \) is a finite set and \( p_{X,Y}(x,y) = \Pr[X = x, Y = y] \). The joint distribution determines the marginals, \( p_X(x) \) and \( p_Y(y) \), as well as the modification channel, \( p_{Y|X}(y|x) \). If the joint distribution is such that the marginals are equal, the modification channel is said to perform stationary updating.

Variable-length Codes: A variable-length encoder and corresponding decoder with block length \( n \) are mappings \( f_E : \mathcal{W}^n \to \mathcal{V}^* \) and \( f_D : \mathcal{V}^* \to \mathcal{W}^m \). The encoder and decoder 1This is equivalent to using base-|\( \mathcal{V} \)| logarithms and all logarithms should be interpreted as such.
define a variable-length code; we further require the encoder-decoder pair to be instantaneous.

A (variable-length) encoder-decoder is applied as follows. Let \((A, B) = (f_E(X^n), f_E(Y^n))\), inducing random variables \(A\) and \(B\) that are drawn from the alphabet \(V^*\). Also let \((X^n, Y^n) = (f_D(A), f_D(B))\).

**Block Codes**: A block encoder for \(X\) with parameters \((n, K)\) is a mapping \(f_E^{(X)} : W^n \rightarrow V^{nK}\) and a block encoder for \(Y\) with parameters \((n, L)\) is a mapping \(f_E^{(Y)} : W^n \rightarrow V^{nL}\). Two encoders are specified for block coding to allow different levels of compression. Given these encoders, a common decoder with parameter \(n\) is \(f_D : V^n \rightarrow W^n\). The encoders and decoder define a block code. Since there is a common decoder, both codes should be in the same format.

A (block) encoder-decoder with parameters \((n, K, L)\) is applied as follows. Let \((A, B) = (f_E^{(X)}(X^n), f_E^{(Y)}(Y^n))\), inducing random variables \(A \in \mathbb{V}^{nK}\) and \(B \in \mathbb{V}^{nL}\). Also let \((\hat{X}_n^n, \hat{Y}_n^n) = (f_D(A), f_D(B))\).

For both variable-length and block coding, the error rate \(\Delta\) is defined as usual. Conventional performance criteria for the codes are the per-letter average lengths of codewords

\[ K = \frac{1}{n} E[|\ell(A)|] \quad \text{and} \quad L = \frac{1}{n} E[|\ell(B)|], \]

where \(|\ell(\cdot)|\) denotes the length of a sequence in \(V^*\). The final performance measure captures our concern with the cost of changing the coded representation. The malleability cost is the expected per-source-letter edit distance between the codes:

\[ M = \frac{1}{n} E[d(A, B)]. \]

**Definition 2**: Given a source \(p(X, Y)\) and an edit distance \(d\), a triple \((K_0, L_0, M_0)\) is said to be achievable for the variable-length coding problem if, for arbitrary \(\epsilon > 0\), there exists (for \(n\) sufficiently large) a variable-length code with error rate \(\Delta = 0\), average codeword lengths \(K \leq K_0 + \epsilon\), \(L \leq L_0 + \epsilon\), and malleability \(M \leq M_0 + \epsilon\).

**Definition 3**: Given a source \(p(X, Y)\) and an edit distance \(d\), a triple \((K_0, L_0, M_0)\) is said to be achievable for the block coding problem if, for arbitrary \(\epsilon > 0\), there exists (for \(n\) sufficiently large) a block code with error rate \(\Delta < \epsilon\), average codeword lengths \(K \leq K_0 + \epsilon\), \(L \leq L_0 + \epsilon\), and malleability \(M \leq M_0 + \epsilon\).

For the variable-length problem, the set of achievable rate-malleability triples is denoted \(\mathfrak{P}_V\); for the block version, the corresponding set is denoted \(\mathfrak{P}_B\). It follows from the definition that \(\mathfrak{P}_V\) and \(\mathfrak{P}_B\) are closed subsets of \(\mathbb{R}^3\) and have the property that if \((K_0, L_0, M_0) \in \mathfrak{P}\), then \((K_0 + \epsilon_1, L_0 + \epsilon_2, M_0 + \epsilon_3) \in \mathfrak{P}\) for any \(\epsilon_i \geq 0\), \(i = 1, 2, 3\). Consequently, \(\mathfrak{P}_V\) and \(\mathfrak{P}_B\) are completely defined by their lower boundaries, which too are closed.

Returning to Fig. 1, for given \(p(X, Y)\) the malleability constraint defines what is achievable in terms of \(p(A, B)\) with the additional constraints of lossless or near lossless maps between \(X^n\) and \(A\), and between \(Y^n\) and \(B\). An alternative formulation as a mapping between two metric spaces \(W^n\) and \(V^*\) is also possible.

III. EASILY ACHIEVED POINTS

To motivate the exposition, first consider four examples of how one might trade off between compression and malleability. This informal presentation is summarized in Fig. 2.

\[ a) \text{ No compression: Taking } A = X \text{ and } B = Y, \text{ it follows immediately that } K = 1 \text{ and } L = 1 \text{ and that the malleability cost is } M = E[d(X, Y)] \text{. If we take the edit distance to be the Hamming distance, then } M = \text{Pr}[X \neq Y] \hat{=} q \text{. Thus the triple } (K, L, M) = (1, 1, q) \text{ is achievable.} \]

\[ b) \text{ Fully compress } X_1^n \text{ and } Y_1^n \text{. One might naively apply an optimal source code. If the updating process } p_{Y|X} \text{ is stationary, then a common instantaneous code may be used to asymptotically achieve } K = H(X) \text{ and } L = H(Y); \text{ if not, then some rate loss is incurred [9]. It seems, however, that} \]

\[ c) \text{ Fully compress } X_1^n \text{ and an increment: One might optimally compress the update separately and append it to the representation of } X_1^n \text{. The new representation has length } n(H(X) + H(Y|X)) \geq nH(Y) \text{ bits. The extended Hamming malleability cost is } nH(Y|X) \text{ symbols.} \]

\[ d) \text{ Completely favor malleability over compression: Another coding scheme (due to R. G. Gallager) dramatically trades compression for malleability. The source } X_1^n \text{ is encoded with } 2^{nH(X)} \text{ symbols, using an indicator function to denote which typical sequence was observed. The same strategy is used to encode } Y_1^n, \text{ using } 2^{nH(Y)} \text{ symbols. Updating requires substituting only two symbols when } X_1^n \text{ and } Y_1^n \text{ are different.} \]

The coding schemes we develop will perform better than the schemes depicted in Fig. 2.

IV. CODING WITH GRAPH EMBEDDING

In this section, we develop a method of coding based on graph embedding and Gray codes. We then construct examples that show improved performance over naive schemes.

Before proceeding, consider some lower bounds for arbitrary sources \(p(X, Y)\). From the source coding theorems, \(K \geq H(X)\) and \(L \geq H(Y)\). Since distinct codewords must
have an edit distance of at least one, we can lower bound $M$ by assuming that minimal distance. Then edit distance is simply the probability of error for uncoded transmission. For $n = 1$, $M \geq \sum_{x \in W} \sum_{y \in W: y \neq x} P(x, y)$ and more generally,

$$M \geq \frac{1}{n} \sum_{x^n_1 \in W^n} \sum_{y^n_1 \in W^n: y^n_1 \neq x^n_1} p(x^n_1, y^n_1).$$

A weaker, simplified version of the bound is $M \geq \frac{1}{n}$, which is a worst-case measure.

Now we construct an example that simultaneously achieves the rate lower bounds and the malleability lower bound (1). Consider a memoryless, equiprobable source $p(x)$ with alphabet $W = \{k, x, y, z, g, j, b, y, c\}$, and thus $H(X) = 3$ bits. Consider the noisy typewriter update process with the binary reflected Gray code (see [12] and Fig. 3(c) for a description) to assign codewords through a given size, first considering size 3, as shown in Fig. 3(a).

Suppose that the edit distance is the Hamming distance. Now try to embed this adjacency graph into a hypercube of a given size, first considering size 3. The adjacency graph is exactly embeddable into the hypercube, as shown in Fig. 3(b). If it were not exactly embeddable, some of the low weight edges might have to be broken. After embedding into the hypercube, use the binary reflected Gray code (see [12] and Fig. 3(c) for a description) to assign codewords through correspondence.

Clearly the code is lossless so the error rate is $\Delta = 0$. Since all codewords are of length 3, clearly $K = L = 3$. To compute $M$, notice that any source symbol is perturbed to any one of its neighbors with probability 1/2. Further notice that the Hamming distance between neighbors in the hypercube is 1. Thus $M = 1/2$. This encoding scheme achieves the entropy bounds $H(X)$ and $H(Y)$. It also achieves the $n = 1$ lower bound for $M$ and is thus optimal for $n = 1$.

Since the embedding relation is true for $n = 1$, it is also true that $n$-fold Cartesian products of the adjacency graph are embeddable into $n$-fold Cartesian products of the hypercube. Such a scheme would achieve rates of $K = 3$ bits and $L = 3$ bits. It would also achieve $M$ of $\frac{1}{2} \Pr [X^n_1 \neq Y^n_1]$ since the Cartesian product of the adjacency graph exactly represents edit costs of 1. For each $n$, this matches the lower bound (1), and is thus optimal. Furthermore, asymptotically in $n$, the triple $(K, L, M) = (3, 3, 0)$ is achievable.

Observe that embeddability into a graph where graph distance corresponds to edit distance seems to be sufficient to guarantee good performance; we will explore this in detail in the sequel.

Similar constructions are possible for variable-length codes. When using such codes, the appropriate edit distance might be the Levenshtein distance, so a minimal change code-labeled Levenshtein distance graph rather than a Gray code-labeled hypercube would be used. When embedding in other graphs, codeword lengths must also be taken into account. If a common Huffman code for $p_X$ and for $p_Y$ is embeddable (with matched vertex labels) in the Levenshtein graph, then minimal $K$, $L$, and $M$ are simultaneously achievable.

V. GENERAL CHARACTERIZATIONS

Using the insights garnered from the example, detailed characterizations of the set of achievable rate–malleability triples are obtained. For variable-length coding, results are expressed in terms of the solution to an error-tolerant attributed subgraph isomorphism problem [4].

A. Error-Tolerant Attributed Subgraph Isomorphism

A vertex-attributed graph is a three-tuple $G = (V, E, \mu)$, where $V$ is the set of vertices, $E \subseteq V \times V$ is the set of edges, and $\mu : V \to \mathcal{V}^*$ is a function assigning labels to vertices. The set of labels is denoted $\mathcal{V}^*$.

Definition 4: Consider two vertex-attributed graphs $G = (V(G), E(G), \mu_G)$ and $H = (V(H), E(H), \mu_H)$. Then $G$ is said to be embeddable into $H$ if $H$ has a subgraph isomorphic to $G$. That is, there is an injective map $\phi : V(G) \to V(H)$ such that $\mu_G(v) = \mu_H(\phi(v))$ for all $v \in V(G)$ and that $(u, v) \in E(G)$ implies $(\phi(u), \phi(v)) \in E(H)$. This is denoted as $G \xrightarrow{\phi} H$.

Several graph editing operations may be defined, such as substituting a vertex label, deleting a vertex, deleting an edge, and inserting an edge. An edited graph is denoted through the operator $\mathcal{E}(\cdot)$ corresponding to the sequence of graph edit operations $E = (e_1, \ldots, e_k)$. There is a cost associated with each sequence of graph edit operations.

Definition 5: Given two graphs $G$ and $H$, an error-tolerant attributed subgraph isomorphism $\psi$ from $G$ to $H$ is the composition of two operations $\psi = (E, \phi) \in \mathcal{E}(G)$ that satisfies $E(G) \xrightarrow{\phi} H$.

Definition 6: The subgraph distance $\rho(G, H)$ is the cost of the minimum cost error-correcting attributed subgraph isomorphism $\psi$ from $G$ to $H$.

Note that in general, $\rho(G, H) \neq \rho(H, G)$.
B. Closeness Vitality

The subgraph isomorphism cost structure for malleable coding is based on a graph theoretic quantity closeness vitality [13]. An edge vitality index is the difference between some functional of a graph and that same functional of the graph with an edge removed.

Let $f_W(G)$ of a graph $G$ be the sum of the distances of all vertex pairs:

$$f_W(G) = \sum_{v \in V} \sum_{w \in V} d(v, w).$$

**Definition 7:** The closeness vitality $cv(G, r)$ of graph $G$ with respect to edge $r$ is: $cv(G, r) = f_W(G) - f_W(G/r)$.

C. $\mathfrak{P}_V$ Characterization

We are concerned with the error-tolerant embedding of an attributed, weighted source adjacency graph into the graph induced by a $V^*$-space edit distance. Edge deletion is the only graph editing operation that we need.

First consider the delay-free case, $n = 1$. A source $p(X, Y)$ and an edit distance $d(\cdot, \cdot)$ are given. Huffman coding provides the minimal redundancy instantaneous code and achieves expected performance $H(X) \leq K \leq H(X) + 1$. Similarly, a Huffman code for $Y$ yields $H(Y) \leq L \leq H(Y) + 1$. The rate loss for using an incorrect Huffman code is essentially a divergence quantity [9]. A source code may be thought of in terms of a random variable, here $Z$. For a given $Z$, there are several Huffman codes: those arising from different labelings of the code tree and also perhaps different trees [14]. Let us denote the set of all Huffman codes for $Z$ as $\mathcal{H}_Z$.

Since $K$ and $L$ are fixed by the choice of $Z$, all that remains is to determine the set of achievable $M$. Let $G$ be the graph induced by the edit distance $d(\cdot, \cdot)$, and $d_G$ its path metric. The graph $G$ is intrinsically labeled. Let $A$ be the weighted adjacency graph of the source $p(X, Y)$, with vertices $W$, edges $E(A) \subseteq W \times W$, and labels given by a Huffman code. That is, $A = (W, E(A), f_E)$ for some $f_E \in \mathcal{H}_Z$. There is a path semimetric, $d_A$, associated with the graph $A$.

The basic problem is to solve the error-tolerant subgraph isomorphism problem of embedding $A$ into $G$. In general for $n = 1$, the malleability cost under edit distance $d_G$ when using the source code $f_E$ is

$$M = \sum_{x \in W} \sum_{y \in W} p(x, y)d_G(f_E(x), f_E(y)).$$

The smallest malleability possible is when $A$ is a subgraph of $G$, and then

$$M_{\text{min}} = \sum_{x \in W} \sum_{y \in W} p(x, y)d_A(x, y) = \sum_{x \in W} \sum_{y \in W} p(x, y)d_G(f_E(x), f_E(y)) = E[f_W(A)] = \Pr[X \neq Y].$$

If edges in $A$ need to broken for embedding, $M$ increases. If an edge $\bar{e}$ is removed from the graph $A$, the resulting graph $A/\bar{e}$ induces its own path semimetric $d_{A/\bar{e}}$. The cost of removing edge $\bar{e}$ from the graph $A$ is:

$$\sum_{x, y \in W} p(x, y) \left[ d_{A/\bar{e}}(f_E(x), f_E(y)) - d_A(f_E(x), f_E(y)) \right],$$

which is the following function of the associated removal operation $e$:

$$C(e) = -E[cv(A, e)].$$

If $E$ is a sequence of edge removals, $\bar{E}$, then $C(E) = -E[cv(A, \bar{E})]$. Putting things together, $\mathfrak{P}_V$ contains any point

$$K = H(X) + D(p_X \| p_Z^1) + 1,$$

$$L = H(Y) + D(p_Y \| p_Z^2) + 1,$$

$$M = M_{\text{min}} + \min_{f_E \in \mathcal{H}_Z} \rho(A, G).$$

Increasing the block length beyond $n = 1$ may improve performance, which we show in the following.

**Theorem 1:** Consider a source $p(X, Y)$ with associated (unlabeled) weighted adjacency graph $A$ and an edit distance $d$ with associated graph $G$. For any $n$, let $\mathfrak{P}^{(aeh)}_V$ be the set of triples $(K, L, M)$ that are computed, by allowing an arbitrary choice of the memoryless random variable $p(Z^n)$, as follows:

$$K = H(X) + D(p_X \| p_Z^n) + \frac{1}{n},$$

$$L = H(Y) + D(p_Y \| p_Z^n) + \frac{1}{n},$$

$$M = \frac{1}{n} \Pr[X^n \neq Y^n] + \frac{1}{n} \min_{f_E \in \mathcal{H}_Z} \rho((W^n, E(A), f_E), G).$$

Then the set of triples $\mathfrak{P}^{(aeh)}_V$ is achievable instantaneously.

The theorem, proven in [6], states that error-tolerant subgraph isomorphism implies achievable malleability. The choice of the auxiliary random variable $Z$ is open to optimization. If minimal rates are desired, $p_Z$ must be on the geodesic connecting $p_X$ and $p_Y$. If $Z$ is not on the geodesic, then there is some rate loss, but perhaps also some malleability gains.

When $p_Y|X$ is a stationary update process, the simple lower bounds might be tight to this achievable region.

**Corollary 1:** Consider a source as given above in Theorem 1. If $p_Y|X$ is stationary, $p_X = p_Y$ is $|V|$-adic, and there is a Huffman-labeled $A$ for $p_X = p_Y$ that is an isometric subgraph of $G$, then the block length $n$ lower bound $(H(X), H(Y), \frac{1}{n} \Pr[X^n \neq Y^n])$ is tight to this achievable region for every $n$, and in particular to $(H(X), H(Y), 0)$ for large $n$.

D. $\mathfrak{P}_B$ Characterization

Now we turn our attention to the block-coding problem. For $\mathfrak{P}_B$, we use a joint typicality graph rather than the weighted adjacency graph used for $\mathfrak{P}_V$. Additionally we focus on binary block codes under Hamming edit distance, so we are concerned only with hypercubes rather than general edit distance graphs. We use standard typicality notations, definitions, and arguments from [15].

For the bivariate distribution $p_{X,Y}$, define a square matrix called the strong joint typicality matrix $A^{(X,Y)}_n$ as follows. There is one row (and column) for each sequence in $\mathcal{S}^{(X)}_n \cup \mathcal{S}^{(Y)}_n$.
The entry with row corresponding to \( x_1^n \) and column corresponding to \( y_1^n \) receives a one if \( (x_1^n, y_1^n) \) is strongly jointly typical and zero otherwise.

Let us temporarily restrict to stationary update: \( \mathcal{P} = \{ p(x, y) \mid p(x) = p(y) \} \). Asymptotically, \( A_{[XY]}^n \) will have approximately equal numbers of ones in all columns and in all rows. Think of \( A_{[XY]}^n \) as the adjacency matrix of a graph, where the vertices are sequences and edges connect sequence \( s \) rows. Think of \( \psi \) as approximately equal numbers of ones in all columns and in all rows. The graph will satisfy corresponding to \( \psi \) within \( \delta \) as

\[
(1 - \delta)2^{n(H(X) - \psi)} \leq |V(G^n)| \leq 2^{n(H(X) + \psi)},
\]

where \( \psi \to 0 \) as \( n \to \infty \) and \( \delta \to 0 \). The degree of each vertex, \( \deg_v \), will concentrate as

\[
2^{n(H(Y) - \nu)} \leq \deg_v \leq 2^{n(H(Y) + \nu)},
\]

where \( \nu \to 0 \) as \( n \to \infty \) and \( \delta \to 0 \). The basic topology of the strongly typical set is asymptotically a \( 2^{n(H(Y)/n)} \) regular graph on \( 2^{nH(Y)} \) vertices. Graph embedding ideas then yield a theorem on block coding achievability:

**Theorem 2:** For a source \( p(x, y) \in \mathcal{P} \) and the Hamming edit distance, a triple \( (K, K, M = M_{\min}) \) is achievable if \( G^n \to H_{nk} \), where \( H_{nk} \) is the hypercube of size \( nK \).

Using this result, we argue that a linear increase in malleability is at exponential cost in code length. A simple counting argument leads to a condition for embeddability.

**Theorem 3:** For a source \( p(x, y) \in \mathcal{P} \), if asymptotically \( G^n \to H_{nk} \) then

\[
nK \geq \max \left( nH(X), 2^{nH(Y/X)} \right).
\]

The space \( S^n \setminus S_{[X]}^n \setminus S_{[Y]}^n \) with the corresponding path metric, \( d_A \) induced by \( A_{[XY]}^n \) is a metric space. Hypercubes with their natural path metric, \( d_G \), are also metric spaces. Rather than requiring absolutely minimal \( nM \), it can be noted that \( M \) is asymptotically zero when the Lipschitz constant associated with the mapping between the source space and the representation space has nice properties in \( n \).

**Definition 8:** A mapping between metric spaces \( f : (S^n, d_A) \to (Y^{nK}, d_G) \) is called Lipschitz continuous if

\[
d_G^n(f(x_1), f(x_2)) \leq C d_A^n(x_1, x_2)
\]

for some constant \( C \) and for all \( x_1, x_2 \in S^n \). The smallest such \( C \) is the Lipschitz constant, \( \text{Lip}[f] \).

We can bound the malleability of a coding scheme that only represents sequences in \( S^n \) in terms of the Lipschitz constant.

**Theorem 4:** For a coding scheme \( f_E \) that only represents sequences in \( S^n = S_{[X]}^n \setminus S_{[Y]}^n \),

\[
M \leq \frac{\text{Lip}[f_E]}{n} \left( 1 + \delta \text{diam}(G^n) \right),
\]

where \( \text{diam}() \) is the graph diameter.

Results from theoretical computer science [6] and some source coding constructions [5] may provide further characterization of \( \text{Lip}[f_E] \).

VI. DISCUSSION AND CONCLUSIONS

We have formulated information theoretic problems motivated by costly writing on storage media. The problems exhibit a trade-off between compression efficiency and the costs incurred when updating using random access editing.

For the zero-error problem, we found that the subgraph distance between a source graph and a storage medium graph determines the rate–malleability relation. Since index assignment for joint source channel coding, signal constellation labeling, and this problem are similar, it is not surprising that Gray codes arise in each [12], [16]. All involve a transformation of objects of one kind into objects of a new kind so that the distances in the two spaces are approximately equal [8].

For block coding, we found that if minimal malleability costs are desired, then a rate penalty that is exponential in the conditional entropy of the update process must be paid. That is, unless the two versions of the source are very strongly correlated (conditional entropy logarithmic in block length), rate exponentially larger than entropy is needed. If we require malleability \( M = O(1/n) \), then rates \( K \) and \( L \) must be

\[
\Omega \left( \frac{1}{n^2} \right).
\]

ACKNOWLEDGMENTS


REFERENCES


