The Square Root Law Does Not Require a Linear Key

Andrew D. Ker
Oxford University Computing Laboratory
Parks Road
Oxford OX1 3QD, UK
adk@comlab.ox.ac.uk

ABSTRACT

Square root laws are theorems about imperfect steganography, embedding which fails to preserve all statistical properties of covers. They show that, in various situations, capacity of covers grows only with the square root of the available cover size. In a paper given at this conference last year [14], we showed an important caveat: when the sender's and recipient's shared embedding key determines the embedding path, its length must be at least linear in the size of the hidden payload to avoid their enemy exhausting over all possible sets of locations. It was left open to show that a linear key is sufficient.

There is no necessity, however, for the recipient to know exactly which locations were changed during the embedding process. In this paper we remove that condition, allowing the embedder to combine more than one cover location to convey one bit of payload. As long as the embedder lives beneath the classic square root law bound, we can do more than prove the sufficiency of a linear key: we can even show that asymptotically perfect steganographic security is possible with no key at all. Furthermore, by computing Steganographic Fisher Information, we can show that the keyless embedding tends to perfect security at least as fast as the "ideal" embedding, which requires an unfeasibly large key to spread payload uniformly at random over the cover. Finally, we show asymptotic perfect security of a simple matrix embedding, which allows payload capacity of order $\sqrt{n}\log n$ to be achieved.

Categories and Subject Descriptors

D.2.11 [**Software Engineering**]: Software Architectures—*information hiding*; H.1.1 [**Models and Principles**]: Systems and Information Theory—*information theory*

General Terms

Security, Algorithms

MM&Sec'10, September 9–10, 2010, Roma, Italy.

Keywords

Steganographic Capacity, Square Root Law, Steganography, Steganalysis

1. INTRODUCTION

The *square root law* is a collection of mathematical results about *imperfect* steganography, which is when the embedding fails to preserve exactly all statistical properties of covers. They apply in different mathematical models, with the common theme that the capacity of covers grows only with the square root of the available cover size. Such theorems can be found, for example, in [10, 12] in the case of multiple independent covers and [5] in the case of single covers under the assumption that the source is a Markov chain satisfying certain conditions.

In a paper published at this conference last year [14], we showed an important limitation to the simplest square root law for single covers, whose elements consist of independent and identically distributed (i.i.d.) elements, whereby the sender and recipient are required to share a large enough secret key to make it impossible for their enemy to guess the embedding locations used: it was shown that a secret key of size at least linear in the hidden payload is required, otherwise detection is asymptotically certain no matter how slowly the payload size grows with the cover size. We dubbed the result, and the paper,"the square root law requires a linear key." It was conjectured, but not proved, that a secret key linear in the payload is sufficient; at that stage, all that could be proved was that a key of order $m \log m$ bits, where m indicates the payload size, is sufficient.

This paper changes the hypotheses very slightly, allowing the embedder to combine more than one cover location to convey one bit of payload. This means that the recipient, and hence the enemy, need not know the exact locations which were changed. In this situation we can go even further than proving the sufficiency of a linear key, in fact showing that *no key* is required for steganographic security as long as the embedder lives below the square root law bound. (A cryptographic secret key would still be needed to keep the content of the payload from the enemy.) This parallels a result by Ryabko [16], which applies in the very different context of *perfect* steganography. The results in [16] demonstrate that no stego key is needed even when embedding an asymptotically maximum capacity payload, though the case of perfect steganography is completely different and the capacities are linear in the cover size.

The paper proceeds as follows. We outline our notation (Subsect. 1.1) and prove a useful convergence result from

pure probability theory (Subsect. 1.2). In Sect. 2 we recapitulate the classic square root law for i.i.d. covers, along with its caveat about key length, as described in [14]. We examine options for locating the payload without informing the recipient (or enemy) exactly where it can be found, and prove new results about an old embedding scheme for which no steganographic key is required at all, in Sect. 3. The rate of convergence to perfect security is related to Steganographic Fisher Information [13, 4], and we examine this in Sect. 4: we are able to prove that the keyless embedding is equally secure as uniform embedding requiring unfeasibly large keys, so that no security is lost. The keyless embedding also adapts naturally to a more efficient encoding, allowing payload size superlinear in the number of embedding changes via matrix embedding [8], and this is examined in Sect. 5. Finally, Sect. 6 closes the paper with a discussion of the new results' significance, and possible extensions.

1.1 Notational Conventions

We will use uppercase Roman letters for random variables, and sets; lowercase and Greek letters are for realisations of random variables, constants, and functions. Uppercase calligraphic letters will be used for probability distributions (and sometimes for sets) and uppercase boldface for matrices.

$X \oplus Y$ indicates the symmetric difference (exclusive-or union) of sets X and Y, and $\mathcal{P}(X)$ the powerset (set of all subsets) of X. Expectation and variance of a random variable are written $\mathrm{E}[X]$ and $\mathrm{Var}[X]$; when we want to emphasise that X has distribution $\mathcal{P}$, we use a subscript: $\mathrm{E}_{X\sim\mathcal{P}}[X]$. $\mathrm{I}(A)$ is an indicator random variable, which takes value 1 if A is true and 0 otherwise. Note that $\mathrm{E}\big[X\,\mathrm{I}(A)\big] = \mathrm{E}[X|A]\Pr(A)$.

Vectors will be written $(x_1,\ldots,x_n)$, or equivalently x_1^n; these will be *column* vectors for the purpose of matrix multiplication. The notation $\psi(n)\sim\phi(n)$ indicates that ψ and ϕ are asymptotically equal: $\psi(n)/\phi(n)\to 1$ as $n\to\infty$. Logs are to natural base unless otherwise indicated.

We will use Knuth's notation [9] for the falling Pochhammer symbol $n^{\underline{k}} = n(n-1)\cdots(n-k+1)$, so that the binomial coefficients are $\binom{n}{k} = n^{\underline{k}}/k!$. Note that, for $0<x<y$,

$$\left(\frac{x-k+1}{y-k+1}\right)^k \;<\; \frac{x^{\underline{k}}}{y^{\underline{k}}} \;<\; \left(\frac{x}{y}\right)^k. \tag{1}$$

1.2 Convergence Lemma

Our results will often involve the asymptotic expectations of certain sequences of random variables, and we begin with some relevant lemmas from pure probability theory. First, a useful bound for the logarithm function:

LEMMA 1. *If $0<a<1<b$ and $a\le x\le b$ then*

$$x-1-\tfrac{1}{2a}(x-1)^2 \le \log x \le x-1-\tfrac{1}{2b}(x-1)^2.$$

PROOF. For the lower bound, differentiate the difference $\log x-(x-1)+\frac{1}{2a}(x-1)^2$; it has a local maximum at a and a local minimum, with value zero, at 1, hence it is positive for $x\ge a$. The upper bound is symmetrical. □

Next, some conditions for convergence of expectations of logarithms of random variables, and their asymptotic limits. This is a tricky because the logarithm is not *uniformly* continuous on the positive reals.

LEMMA 2. *Let Y_n be a sequence of positive random variables. Consider the following conditions:*

(C1) $\mathrm{E}[Y_n]=1$ *for all n;*

(C2) $\mathrm{Var}[Y_n]\neq 0$ *for all n;*

(C3) for any $\epsilon>0$, as $n\to\infty$

$$\frac{\mathrm{E}\big[(Y_n-1)^2\,\mathrm{I}(|Y_n-1|\ge\epsilon)\big]}{\mathrm{Var}[Y_n]}\to 0;$$

(C4) there exists a lower bound $0<l<1$ such that $l\le Y_n$, for all n.

Under (C1)–(C3),

$$\limsup\frac{\mathrm{E}[\log Y_n]}{\mathrm{Var}[Y_n]} \le -\frac{1}{2}. \tag{2}$$

If also (C4), $\liminf\frac{\mathrm{E}[\log Y_n]}{\mathrm{Var}[Y_n]}\ge -\frac{1}{2}$ *and hence*

$$\mathrm{E}[\log Y_n]\sim -\tfrac{1}{2}\mathrm{Var}[Y_n]. \tag{3}$$

PROOF. (C2) is needed to make the quotients well-defined. Take any $\epsilon>0$. First, consider

$$Z_n=\frac{\log(Y_n)-(Y_n-1)+\frac{1}{2(1+\epsilon)}(Y_n-1)^2}{\mathrm{Var}[Y_n]}.$$

Note that $Z_n\le 0$ if $Y_n\le 1+\epsilon$, by Lemma 1, and regardless of Y_n we have

$$Z_n\le\frac{1}{2(1+\epsilon)}\frac{(Y_n-1)^2}{\mathrm{Var}[Y_n]},$$

because of the inequality $\log x\le x-1$. Therefore

$$\begin{aligned}
&\frac{\mathrm{E}[\log Y_n]}{\mathrm{Var}[Y_n]}+\frac{1}{2(1+\epsilon)}\\
&\quad=\ \mathrm{E}[Z_n]\\
&\quad=\ \mathrm{E}\big[Z_n\,\mathrm{I}(|Y_n-1|\le\epsilon)\big]+\mathrm{E}\big[Z_n\,\mathrm{I}(|Y_n-1|>\epsilon)\big]\\
&\quad\le\ 0+\frac{1}{2(1+\epsilon)}\frac{\mathrm{E}\big[(Y_n-1)^2\,\mathrm{I}(|Y_n-1|>\epsilon)\big]}{\mathrm{Var}[Y_n]}\\
&\quad\to\ 0.
\end{aligned}$$

We used (C1) at the first equality, and (C3) for the final limit. Therefore, for sufficiently large n,

$$\frac{\mathrm{E}[\log Y_n]}{\mathrm{Var}[Y_n]}\le -\frac{1}{2(1+\epsilon)}+\epsilon,$$

proving (2).

Second, consider

$$Z_n=\frac{\log(Y_n)-(Y_n-1)+\frac{1}{2(1-\epsilon)}(Y_n-1)^2}{\mathrm{Var}[Y_n]}.$$

This time, $Z_n\ge 0$ if $Y_n\ge 1-\epsilon$. And for other values of Y_n, now using (C4),

$$Z_n\ge\left(\frac{1}{2(1-\epsilon)}-\frac{1}{2l}\right)\frac{(Y_n-1)^2}{\mathrm{Var}[Y_n]}.$$

Therefore the same calculations give

$$\begin{aligned}
&\frac{\mathrm{E}[\log Y_n]}{\mathrm{Var}[Y_n]} + \frac{1}{2(1-\epsilon)} \\
&\quad = \mathrm{E}[Z_n] \\
&\quad = \mathrm{E}\big[Z_n \,\mathrm{I}(|Y_n - 1| \le \epsilon)\big] + \mathrm{E}\big[Z_n \,\mathrm{I}(|Y_n - 1| > \epsilon)\big] \\
&\quad \ge 0 + \left(\frac{1}{2(1-\epsilon)} - \frac{1}{2l}\right) \frac{\mathrm{E}\big[(Y_n-1)^2 \,\mathrm{I}(|Y_n - 1| > \epsilon)\big]}{\mathrm{Var}[Y_n]} \\
&\quad \to 0.
\end{aligned}$$

For sufficiently large n,

$$\frac{\mathrm{E}[\log Y_n]}{\mathrm{Var}[Y_n]} \ge -\frac{1}{2(1-\epsilon)} - \epsilon,$$

proving (3). □

(C3) ensures that the contribution of the tails of Y_n to its variance is not too large. It is related to the so-called Lindeberg condition, which implies a generalised central limit theorem (CLT) [3, VIII.4, Th. 3]. In the case of the CLT, Lindeberg's condition is implied by an easier-to-verify condition (Lyapounov's condition [3, VIII.10.17]) on some of the moments of the random variables; in our case (C3) is implied by an analogous condition, also easier to check in practice:

LEMMA 3. *As long as* $\mathrm{E}[Y_n] = 1$*, the condition*

$$\frac{\mathrm{E}\big[(Y_n-1)^4\big]}{\mathrm{Var}[Y_n]} \to 0.$$

is sufficient for (C3).

PROOF. We break down $\mathrm{E}\big[(Y_n-1)^2 \,\mathrm{I}(|Y_n-1| \ge \epsilon)\big]$ into slices of width ϵ:

$$\begin{aligned}
&\mathrm{E}\big[(Y_n-1)^2 \,\mathrm{I}(|Y_n-1| \ge \epsilon)\big] \\
&\quad = \sum_{k=1}^{\infty} \mathrm{E}\big[(Y_n-1)^2 \,\mathrm{I}\big(k\epsilon < |Y_n-1| \le (k+1)\epsilon\big)\big] \\
&\quad \le \sum_{k=1}^{\infty} \big((k+1)\epsilon\big)^2 \Pr(|Y_n-1| \ge k\epsilon) \\
&\quad \overset{(a)}{\le} \sum_{k=1}^{\infty} (k+1)^2 \epsilon^2 \frac{\mathrm{E}\big[(Y_n-1)^4\big]}{k^4\epsilon^4} \\
&\quad = \mathrm{E}\big[(Y_n-1)^4\big] \frac{1}{\epsilon^2} \sum_{k=1}^{\infty} \frac{(k+1)^2}{k^4},
\end{aligned}$$

where (a) is Markov's inequality. The sum is convergent, and this implies the required result. □

2. SQUARE ROOT LAWS

Square root laws are generally asymptotic results, showing that the probability of correct detection rises to one if payload increases asymptotically faster than $\sqrt{n}$, where n is the cover size, and that it falls to zero if payload increases asymptotically slower. Hence the "rate" $\sqrt{n}$ is the critical point, and we say that capacity follows a square root law. We begin by recapitulating one result from [14] (with some slight re-wording), to set the scene for our novel work.

THEOREM 1. *Suppose that the cover consists of n pixels $(X_1, \ldots, X_n)$ each drawn from a finite alphabet $\mathcal{X}$, independent and identically distributed each with mass function $p(x)$. Suppose that a payload of size m causes exactly m pixels to be replaced with those of the same alphabet, but distributed with mass function $q(x)$, and that this pixel selection is made uniformly from all $\binom{n}{m}$ possibilities. Finally, suppose that $p(x) \ne 0$ and $q(x) \ne 0$ for all $x \in \mathcal{X}$, and there exists $y \in \mathcal{X}$ such that $p(y) \ne q(y)$.*

(i) If $m/\sqrt{n} \to \infty$ then, for sufficiently large n, covers and stego objects can be distinguished with arbitrarily low error rate.

(ii) If $m/\sqrt{n} \to 0$ then, for sufficiently large n, any detector must have arbitrarily high error rate.

We have called the cover elements "pixels", but they could be equally another representation of the cover such as transform domain coefficients. On embedding, some of the pixels are altered. It will not matter, for our purposes, exactly what embedding function is used, as long as the change at each embedding location is independent of the others (this covers many practical embedding schemes). We will say more about the embedding operation later.

The proofs of (i) and (ii) are quite different. For (i), we merely need to construct a detector with the stated performance; it turns out that simply counting the number of pixels with value y is sufficient, which can be proved using tail inequalities. For (ii), we compute the Kullback-Leibler (KL) divergence between the distribution of covers and stego objects, and show that it tends to zero under the condition $m/\sqrt{n} \to 0$. This can be difficult, especially in the case of Theorem 1 where the embedding causes a weak dependence between the pixels of the stego image. (Some square root laws also contain a case (iii), when $m/\sqrt{n} \to r$, a positive constant called the *root rate*. We will not pursue such cases in this work, but our analysis in Sect. 4 is related to it.)

We highlight two of the preconditions for Theorem 1. First, the cover has been modelled as i.i.d. elements on a finite alphabet, and that will be the model used in the rest of this paper too. A finite alphabet is very reasonable, but i.i.d. elements make a poor model for the practice of steganography in digital media. However, this model illustrates the square root law without the complicated analysis required in [5]. We are confident that the same results will hold for richer cover models, such as the Markov chains analysed in [5], though new abstractions will be probably be required before such a proof can be constructed.

Second, we assumed that the m embedding locations which convey the payload (presumably of m bits, though other bases could be used), were uniformly selected from all possibilities. We will call this *ideal embedding*, because there is no information leaked to the enemy about the likelihood of any particular locations being used for embedding. However, it is far from ideal in practice, because the sender and recipient need to agree on m locations chosen from n possibilities, so they need to distinguish $n!/(n-m+1)!$ different possible embedding paths, which itself would need $m \log m$ bits of information in the case when $m \sim \sqrt{n}$: their secret embedding key has to be longer than the hidden message! This is unsatisfactory, so in [14] we examined the case when the number of embedding paths is drawn from a smaller set of K possibilities, with the following result:

THEOREM 2. *Suppose that the cover consists of n pixels $(X_1, \ldots, X_n)$, independent and identically distributed each with mass function $p(x)$. Suppose a payload of size m which will cause exactly m pixels to be replaced with mass function $q(x)$. Suppose that the sender, recipient, and attacker share knowledge of a set K of secret keys, each of which generates a path of length m determining the payload locations, but only the sender and recipient know which key is used. Finally, suppose that there exists y such that $p(y) \neq q(y)$.*

If $(\log |K|)/m \to 0$, as $m \to \infty$, and $m \to \infty$ as $n \to \infty$, then, for sufficiently large n, covers and stego objects can be distinguished with arbitrarily low error rate.

Since the sender and recipient need $\log |K|$ bits of secret information to determine the embedding path, this shows that a secret key sublinear in the message size will lead to asymptotically perfect detection, no matter how slowly the payload size itself grows. It is proved by constructing such a detector, which simply exhausts over all possible embedding paths and looks for an outlier in the number of y pixels observed.

Note that Theorem 2 is analogous to part (i) of Theorem 1: it shows that capacity cannot grow beyond a certain rate (in this case, it cannot grow at all). The analogue of (ii), showing that a linear embedding key is sufficient to ensure asymptotic perfect security (assuming the payload size keeps below the square root law bound) was not proved in [14] and was the motivation for the work reported in this paper.

In this paper we will concentrate on analogues of part (ii) of Theorem 1, showing that various embedding schemes *are* asymptotically secure. The analogues of part (i) also hold, but will not be our focus. Note that the results in [14] make the assumption that payload is proportional to the number of changes. It is by relaxing that assumption, using embedding schemes for which it does not hold, we are able to prove asymptotic perfect security, even without an embedding key at all.

So before we continue to novel results, we will need to make a slight change to the terminology. There is an important difference between embedding locations *used* to convey payload, and those *changed* in the embedding of payload. In the case of simple bit replacement, for example, only half of those locations used will need to be changed, on average, because the others already contained the correct bit; in the presence of source coding at the embedder, such as matrix embedding [8], it is entirely possible that many locations are used, conveying many bits of payload, but only a small number have to be changed.

We will therefore alter our meaning of q, in this paper, to be the mass function of pixels *changed* by embedding. It makes no difference to the validity of the square root laws, but it affects the more subtle analysis of asymptotic rates we will undertake in Sect. 4.

We will also be more concrete about some possible embedding functions. Let us assume that the sender makes public some parity function $P : \mathcal{X} \to \{0, 1\}$. This could simply be the least significant bit of a pixel value, or something more complicated which varies according to position. In the embedding schemes we will devise, the parity function will either be used to extract one bit of payload from a single location, or to make up codewords from which bits can be extracted. If the embedder needs to change the parity value of a location we will assume that they can always do so, and it is the result of these modifications which produces pixels distributed according to q.

It will turn out that properties of $q(X)/p(X)$, where X is a random variable representing a single cover symbol (i.e. has mass function $p(x)$), are important to the asymptotic performance of embedding. Note that

$$\mathrm{E}\left[\frac{q(X)}{p(X)}\right] = \sum_{x \in \mathcal{X}} p(x) \frac{q(x)}{p(x)} = 1$$

and, because the domain of X is finite, all the moments of $q(X)/p(X)$ are finite. We will define constants for the central moments of this random variable:

$$\mu_k = \mathrm{E}\left[\left(\frac{q(X)}{p(X)} - 1\right)^k\right] = \sum_{x \in \mathcal{X}} p(x) \left(\frac{q(x)}{p(x)} - 1\right)^k .$$

3. STEGANOGRAPHY WITH NO KEY

We now explore ways in which the steganographer can communicate with their recipient without the latter knowing the precise location of the embedding changes. Communication using such a *non-shared selection channel* has already been studied, and we consider whether the literature is helpful. Then we outline a very simple embedding scheme, which spreads one bit of payload across up to $\sqrt{n}$ locations.

3.1 Wet Paper Codes

Wet paper codes are an example of embedding using side information. The usual setup is that certain embedding locations are "wet" and cannot be changed by the embedder, but the recipient does not know which were the wet locations when they receive the message [7]. This is a stronger requirement than needed for our situation, which is only that the recipient (and hence enemy) not know the location of the payload, but it is useful to examine whether practical wet paper codes allow us to prove a square root law in the presence of a small embedding key.

The simple construction in [7] is to use a pseudorandom binary matrix $\mathbf{D}$, of size $m \times n$, to communicate m bits of information in n locations: $\mathbf{D}$ can be shared with the recipient either by making it public or by using the pseudorandom number generator seed which created $\mathbf{D}$ as the secret embedding key. Create a column vector c_1^n with elements $c_i = P(x_i)$, the parity values of the elements of the cover, and write the payload as a vector of bits p_1^m; the stego object is created by flipping some of the parity values of the cover so that the vector of its parity values, s_1^n, satisfies the simultaneous linear equations

$$\mathbf{D} s_1^n = p_1^m . \tag{4}$$

Thus the embedding relies on solving (4) for s_1^n, over binary arithmetic, subject to the condition that at wet locations we force $s_i = c_i$. For suitably chosen pseudorandom matrices $\mathbf{D}$, and fewer than $n - m$ wet pixels, it is shown that this is possible with high probability. Extraction of the hidden payload is simply computing the product (4) for the given stego object, and we can see that the recipient never finds out which locations had actually been altered.

We need only designate about $n - m$ pixels as wet to ensure that there are no more than m embedding changes, and therefore the changes will not grow at an asymptotically higher rate than for simple bit replacement; we already

know that fewer than $O(\sqrt{n})$ changes are asymptotically undetectable, so it appears that we have recovered a simple square root law, with at most a small secret embedding key to determine $\mathbf{D}$.

However, how do we know that solutions to (4) do not bias the likely embedding change locations? The enemy must be assumed to know the embedding procedure, and even if $\mathbf{D}$ was generated by a small secret key, the enemy could perhaps exhaust over different possible $\mathbf{D}$ matrices. Can we prove that they cannot then predict the likely locations of some of the changes? A lot depends on the algorithm used to solve (4), and Böhme has already shown that there can be weaknesses which are potentially exploitable by steganalysis [2]. This makes it difficult to prove a proper square root law about standard random matrix wet paper codes, and we know of no result about any other wet paper code construction which proves that the enemy gains no information about the location of changes. Instead, we will turn to what amounts to the trivial wet paper code, with $\mathbf{D}$ a matrix with blocks of 1s on the diagonal and zero elsewhere, for which a proof is relatively easy to construct.

3.2 A Simple Solution

Let us consider a very simple form of embedding, which spreads one bit of payload amongst many locations, yet allows the recipient to recover the hidden message without knowing exactly which bits have been altered. To our knowledge this was first described in [1].

Algorithm 1. Divide a cover of size n, $(x_1, \ldots, x_n)$, into m groups of $\lfloor n/m \rfloor$ pixels each, with any remainder pixels left unused. In each group G, compute the sum of the parities: $\sum_{x_i \in G} P(x_i) \pmod 2$; if this matches the next payload bit, do nothing, otherwise choose one of the pixels from the group, uniformly at random, and alter it to flip its parity value.

The recipient recovers the embedded payload by computing $\sum_{x_i \in G} P(x_i) \pmod 2$ in each group. They need to know the parity function, and the division of the cover into groups, but neither of these need be kept secret from the enemy.

Now we show that this algorithm is asymptotically perfectly secure as long as $m/\sqrt{n} \to 0$. Note that we *cannot* use the original square root law proof, because the enemy has extra information about the location of the changes if they know the groups: they know that exactly zero or one changes have been made in each group. With extra analysis, however, we can prove that this additional information does not alter the asymptotic perfect security of the embedding.

THEOREM 3. *Suppose that the cover consists of n pixels $(X_1, \ldots, X_n)$ each drawn from a finite alphabet $\mathcal{X}$, independent and identically distributed each with mass function $p(x)$, and that Algorithm 1 is applied to embed a payload of size m. Suppose that altered pixels have mass function $q(x)$. Finally, suppose that $p(x) \neq 0$ and $q(x) \neq 0$ for all $x \in \mathcal{X}$.*

If $m/\sqrt{n} \to 0$ then, for sufficiently large n, any detector must have arbitrarily high error rate, even if they have knowledge of the groups and the parity function.

PROOF. Consider one group of $k = \lfloor n/m \rfloor$ pixels which, for sake of simplicity, we shall call $(X_1, \ldots, X_k)$. With no embedding, the pixels are i.i.d. with distribution $p(x)$; call this joint distribution $\mathcal{P}$. With embedding, there is probability $1/2$ that they have the same distribution, and, for each $j = 1 \ldots k$, probability $1/2k$ that pixel j has been altered and now has distribution $q(x)$; call this joint distribution $\mathcal{Q}$.

Now define $R_i = q(X_i)/p(X_i)$. Note that R_i are independent and identically distributed. In Sect. 2 we showed that, with expectations taken over $X_1^k \sim \mathcal{P}$, $\mathrm{E}[R_i] = 1$ and $\mathrm{Var}[R_i] = \mu_2$, some finite constant.

Now we compute the KL divergence

$$\begin{aligned} D_1 &= D_{\mathrm{KL}}(\mathcal{P} \parallel \mathcal{Q}) \\ &= -\mathrm{E}\left[\log\left(\frac{\mathcal{Q}(X_1^k)}{\mathcal{P}(X_1^k)}\right)\right] \\ &= -\mathrm{E}\left[\log\left(\frac{\frac{1}{2}\prod_i p(X_i) + \frac{1}{2k}\sum_j q(X_j)\prod_{i\neq j} p(X_i)}{\prod_i p(X_i)}\right)\right] \\ &= -\mathrm{E}\left[\log\left(\frac{1}{2} + \frac{1}{2k}\sum_{j=1}^{k} R_j\right)\right]. \end{aligned}$$

Write $Y_k = \frac{1}{2} + \frac{1}{2k}\sum_j R_j$. We wish to apply Lemma 2 to determine the asymptotics of D_1.

We can check (C1),

$$\mathrm{E}[Y_k] = \frac{1}{2} + \frac{1}{2k}\sum_{j=1}^{k} \mathrm{E}[R_j] = 1,$$

and compute the variance

$$\mathrm{Var}[Y_k] = \frac{1}{4k^2}\sum_{j=1}^{k} \mathrm{Var}[R_j] = \frac{\mu_2}{4k}.$$

As long as p and q are not identical mass functions (in which case $D_1 = 0$ and there is nothing to prove), $\mu_2 > 0$ and so $\mathrm{Var}[Y_k] > 0$, verifying (C2).

To verify (C3) we use Lemma 3:

$$\begin{aligned} &\frac{\mathrm{E}\left[(Y_k - 1)^4\right]}{\mathrm{Var}[Y_k]} \\ &= \frac{\mathrm{E}\left[\left(\frac{1}{2k}\sum_j (R_j - 1)\right)^4\right]}{\mathrm{Var}[Y_k]} \\ &= \frac{\frac{1}{16k^4}\sum_j \mathrm{E}\left[(R_j-1)^4\right] + \frac{3}{16k^4}\sum_{i\neq j}\mathrm{Var}[R_i]\mathrm{Var}[R_j]}{\frac{1}{4k}\mu_2} \\ &= \frac{\mu_4}{4k^2\mu_2} + \frac{3(k-1)\mu_2}{4k^2} \quad \to \quad 0. \end{aligned}$$

For (C4), note that R_i is positive so Y_k is bounded below by $1/2$. Thus we apply Lemma 2 with the result that

$$D_1 \sim \frac{1}{2}\mathrm{Var}[Y_n] = \frac{\mu_2}{8k}. \tag{5}$$

Finally, consider the KL divergence between entire cover and stego objects: possibly apart from some unchanged remainder pixels (which contribute nothing) the total KL divergence is a sum of m independent groups, which is

$$D = mD_1 \sim m\frac{\mu_2}{8k} \sim \frac{\mu_2}{8}\frac{m^2}{n}$$

which tends to zero, given the hypothesis that $m/\sqrt{n} \to 0$. As with the classical square root law, the performance of any detector must tend to purely random as $n \to \infty$. □

We have demonstrated that Algorithm 1 has asymptotic perfect security, in the sense that sufficiently large covers

give arbitrarily small probability of detection; this is because the payload is spread more thinly in larger covers. But, since the entire procedure should be considered public (e.g. because of Kerckhoffs' Principle), there is nothing to stop the attacker reading the hidden message by using the same procedure as the recipient. This emphasises the difference between *steganographic* security, where the covers are not altered sufficiently for the changes to be detectable, and *cryptographic* security which prevents the enemy from understanding the payload. The same applies to the perfect steganography result in [16], where the embedding distortion is perfectly undetectable but the enemy can still read the payload if it is not encrypted.

In practice, then, the sender and recipient would indeed share a secret key, and use it to encrypt the payload prior to embedding. The key need only be long enough to prevent against exhaustion by the enemy. It is important that the encryption produces high entropy output, which cannot be distinguished from random noise, so that the enemy does not recognise cyphertext when they see it. However, that is beyond the scope of this paper.

Note that the converse to Theorem 3, that $m/\sqrt{n} \to \infty$ implies asymptotic perfect detectability, also holds. It follows immediately from the classic square root law because the detector need not use the additional information (zero or one changes per group) to construct such a detector.

4. STEGANOGRAPHIC FISHER INFORMATION

Algorithm 1 shares the asymptotic perfect secrecy property of ideal uniformly-spread embedding, but we might still believe that it is less secure: perhaps the KL divergences tend to zero at different rates? This question is related to Steganographic Fisher Information (SFI), which is studied in [13] and [4]. Those papers refer to embedding in particular cover models, but the general concept applies more widely. Given covers of size n and payloads of size m, when

$$D_{\mathrm{KL}}(\mathcal{P} \parallel \mathcal{Q}) \sim \frac{I}{2}\frac{m^2}{n},$$

in the limit as $m/\sqrt{n} \to 0$, then I is called the Steganographic Fisher Information for the embedding. Lower SFI corresponds to lower KL divergence, less evidence, and hence more secure embedding. The connection between Fisher Information and asymptotic steganographic security was first noted in [11], which also explains the reason for the appearance of the squared ratio m^2/n.

In this section, we show that Algorithm 1 is at least as secure as ideal uniformly-spread embedding in the sense that its SFI is no greater. The challenge is in the analysis of ideal embedding. First, we consider how to model more complex embedding schemes than the zero-or-one changes of Algorithm 1.

Given k locations, we can number them $1 \dots k$. We can then model the effect of embedding as a probability distribution on $S_k = \mathcal{P}(\{1, \dots, k\})$ giving the likelihood of changes at each location. For example, the embedding of Algorithm 1 makes changes C with $\Pr(C{=}\emptyset) = 1/2$ and $\Pr(C{=}\{i\}) = 1/2k$ for each i. The distribution of stego objects, if unchanged locations have mass function $p(x)$ and changed locations have mass function $q(x)$, is

$$\begin{aligned}\mathcal{Q}(X_1^k) &= \sum_{C' \in S_k} \Pr(C = C') \prod_{c \in C} q(X_c) \prod_{c \notin C} p(X_c) \\ &= \Big(\prod_{i=1}^{k} p(X_i)\Big) \mathrm{E}_{C \sim \mathcal{C}}\Big[\prod_{c \in C} \frac{q(X_c)}{p(X_c)}\Big]\end{aligned}$$

where $\mathcal{C}$ is the distribution of change locations. The log of the second term is what we will need to deal with when computing the KL divergence between cover and stego groups. When more than one change is possible it is not a sum of independent components, so its analysis is more difficult than in Theorem 3. We now prove a useful lemma about its variance.

LEMMA 4. *Suppose a group of pixels of size k, with embedding changes located independently of cover content with distribution $\mathcal{C}$. Let $\mathcal{P}$ represent the probability distribution for which the X_i are i.i.d. with mass function $p(x)$. Then*

$$\mathrm{Var}_{X \sim \mathcal{P}}\left[\mathrm{E}_{C \sim \mathcal{C}}\left[\prod_{c \in C} \frac{q(X_c)}{p(X_c)}\right]\right] = \mathrm{E}_{\substack{C_1 \sim \mathcal{C} \\ C_2 \sim \mathcal{C}}}\left[(\mu_2 + 1)^{|C_1 \cap C_2|}\right] - 1.$$

PROOF. For $C' \in S_k$ write $R(C') = \prod_{c \in C'} \frac{q(X_c)}{p(X_c)}$. Then, for any C',

$$\mathrm{E}_{X \sim \mathcal{P}}\big[R(C')\big] = \prod_{c \in C'} \mathrm{E}\left[\frac{q(X_c)}{p(X_c)}\right] = 1,$$

so

$$\mathrm{E}_{X \sim \mathcal{P}}\Big[\mathrm{E}_{C \sim \mathcal{C}}\big[R(C)\big]\Big] = \mathrm{E}_{C \sim \mathcal{C}}\Big[\mathrm{E}_{X \sim \mathcal{P}}\big[R(C)\big]\Big] = 1. \quad (6)$$

Therefore

$$\begin{aligned}&\mathrm{Var}_{X \sim \mathcal{P}}\left[\mathrm{E}_{C \sim \mathcal{C}}\left[\prod_{c \in C} \frac{q(X_c)}{p(X_c)}\right]\right] \\ &\quad= \mathrm{E}_{X \sim \mathcal{P}}\left[\left(\sum_{C'} \Pr(C{=}C')R(C)\right)^2\right] - 1 \\ &\quad= \sum_{C_1}\sum_{C_2} \Pr(C{=}C_1)\Pr(C{=}C_2)\mathrm{E}\big[R(C_1)R(C_2)\big] - 1 \\ &\quad= \mathrm{E}_{\substack{C_1 \sim \mathcal{C} \\ C_2 \sim \mathcal{C}}}\Big[\mathrm{E}_{X \sim \mathcal{P}}\big[R(C_1)R(C_2)\big]\Big] - 1.\end{aligned}$$

And for any $C_1, C_2 \in S_k$,

$$\begin{aligned}\mathrm{E}\big[R(C_1)R(C_2)\big] &= \prod_{c \in C_1 \oplus C_2} \mathrm{E}\left[\frac{q(X_c)}{p(X_c)}\right] \prod_{c \in C_1 \cap C_2} \mathrm{E}\left[\left(\frac{q(X_c)}{p(X_c)}\right)^2\right] \\ &= \prod_{c \in C_1 \cap C_2} \mu_2 + 1 \\ &= (\mu_2 + 1)^{|C_1 \cap C_2|},\end{aligned}$$

which gives the result. □

Note that the embedding need not actually be a process independent of the cover: if the distribution of the output of the parity function P is uniform bits, and so is the payload, then most source coding schemes will satisfy this property. Next, some pure probability-theoretic results about the binomial and hypergeometric distributions:

LEMMA 5. *(i) If B has a binomial distribution with parameters n and p (n independent trials each with probability p of being a "success" counted in B) then, for any a,*

$$\mathrm{E}[a^B] = (1 - p + ap)^n.$$

(ii) If H has a hypergeometric distribution with parameters n, m, and m (m drawn, without replacement, from n objects of which m are "successes" counted in H) and $m/\sqrt{n} \to 0$ then, for any $a > 1$,

$$\mathrm{E}\big[a^H\big] - 1 \sim (a-1)\frac{m^2}{n}.$$

PROOF. (i) is just the probability generating function and it follows immediately from the binomial theorem.

For (ii), consider the mass function of the hypergeometric distribution:

$$\Pr(H = h) = \tbinom{m}{h}\tbinom{n-m}{m-h}/\tbinom{n}{m}.$$

On one hand,

$$\begin{aligned}
\sum_{h=0}^{m} a^h \Pr(H=h) &= \sum_{h=0}^{m} a^h \binom{m}{h} \frac{(n-m)^{\underline{m-h}}\, m^{\underline{h}}}{n^{\underline{m}}} \\
&= \sum_{h=0}^{m} a^h \binom{m}{h} \frac{(n-m)^{\underline{m}}}{n^{\underline{m}}} \frac{m^{\underline{h}}}{(n-2m+h)^{\underline{h}}} \\
&\overset{(a)}{\leq} \left(1 - \frac{m}{n}\right)^m \sum_{h=0}^{m} \binom{m}{h} \left(\frac{am}{n-2m}\right)^h \\
&\overset{(b)}{=} \left(1 - \frac{m}{n}\right)^m \left(1 + \frac{am}{n-2m}\right)^m \\
&\overset{(c)}{=} 1 + (a-1)\frac{m^2}{n} + O\left(\left(\frac{m^2}{n}\right)^2\right)
\end{aligned}$$

where (a) comes from (1), (b) is the binomial theorem, and (c) is a binomial expansion. On the other,

$$\begin{aligned}
\sum_{h=0}^{m} a^h \Pr(H=h) &\geq \Pr(H=0) + a\Pr(H=1) \\
&= \frac{(n-m)^{\underline{m}}}{n^{\underline{m}}} + am^2 \frac{(n-m)^{\underline{m-1}}}{n^{\underline{m}}} \\
&= \frac{(n-m)^{\underline{m}}}{n^{\underline{m}}} \left(1 + am^2 \frac{1}{n-2m+1}\right) \\
&\overset{(a)}{\geq} \left(1 - \frac{m}{n-m}\right)^m \left(1 + a\frac{m^2}{n}\right) \\
&\overset{(b)}{=} 1 + (a-1)\frac{m^2}{n} + O\left(\left(\frac{m^2}{n}\right)^2\right)
\end{aligned}$$

where (a) comes from (1) and (b) is a binomial expansion. These inequalities combine, along with the assumption that $m/\sqrt{n} \to 0$, to give the required result. □

We can now put everything together and prove that Algorithm 1 has (at least) the same asymptotic security of ideal uniformly-spread embedding, despite the former needing no embedding key at all while the latter needs one of size $O(m \log m)$.

THEOREM 4. *The Steganographic Fisher Information for ideal embedding, with paths chosen uniformly from all possibilities, is at least as large as that provided by Algorithm 1, even if the enemy knows the groups used by the latter.*

PROOF. Equation (5) tells us that the SFI of Algorithm 1 is $\frac{\mu_2}{4}$. We now compute the SFI for uniformly-spread embedding. We proceed as in Theorem 3, letting $\mathcal{P}$ denote the joint distribution of a cover object $(X_1, \ldots, X_n)$ and $\mathcal{Q}$ denote the corresponding stego object. Write $\mathcal{C}$ for the distribution of cover changes. Recall that m out of n locations are chosen to convey payload, uniformly from all possibilities, and of those chosen each is changed, independently of the others, with probability 1/2. We begin by following Theorem 3:

$$\begin{aligned}
D &= D_{\mathrm{KL}}(\mathcal{P} \parallel \mathcal{Q}) \\
&= -\mathrm{E}\left[\log\left(\frac{\mathcal{Q}(X_1^n)}{\mathcal{P}(X_1^n)}\right)\right] \\
&= -\mathrm{E}\left[\log\left(\mathrm{E}_{C\sim\mathcal{C}}\left[\prod_{c\in C} \frac{q(X_c)}{p(X_c)}\right]\right)\right]
\end{aligned}$$

We wish to apply Lemma 2 to

$$Y_n = \mathrm{E}_{C\sim\mathcal{C}}\left[\prod_{c\in C} \frac{q(X_c)}{p(X_c)}\right].$$

(C1) is verified at (6), and (C2) follows because p and q are not identical. To compute the variance we use Lemma 4. Let H be the number of payload locations *used* by both of the independent random embeddings C_1 and C_2: 3/4 of these will involve *changes* under only one, or neither, embedding. More precisely, conditional on H, $B = |C_1 \cap C_2|$ is binomial with parameters H and $\frac{1}{4}$. Therefore, conditioning on H, we use Lemma 4 and parts (i) and (ii) of Lemma 5 in succession to compute

$$\begin{aligned}
\mathrm{Var}_{X\sim\mathcal{P}}[Y_n] &= \mathrm{E}_B\big[(\mu_2+1)^B\big] - 1 \\
&= \mathrm{E}_H\Big[\mathrm{E}_B\big[(\mu_2+1)^B \mid H\big]\Big] - 1 \\
&= \mathrm{E}_H\Big[\big(\tfrac{3}{4} + \tfrac{1}{4}(\mu_2+1)\big)^H\Big] - 1 \\
&\sim \big(\tfrac{3}{4} + \tfrac{1}{4}(\mu_2+1) - 1\big)\frac{m^2}{n} \\
&= \frac{\mu_2}{4}\frac{m^2}{n}
\end{aligned}$$

It remains to verify (C3): the best proof found by the author is extremely long and we include a sketch in Appendix. A. Then we can apply the first part of Lemma 2, telling us that asymptotically,

$$D \geq \frac{1}{2}\frac{\mu_2}{4}\frac{m^2}{n}.$$

□

In fact, it is possible to show that the full conclusion of Lemma 2 holds, and that the SFI for the ideal uniformly-spread embedding is *exactly* the same as that of Algorithm 1, but we cannot get there using the tools we have outlined in this paper: condition (C4) does not hold here. In any case the conclusion is unimportant compared with that already proven: Algorithm 1 is at least as secure as ideal uniformly-spread embedding, as well as having the advantage of needing no embedding key at all.

The techniques of this and the previous section may provide a template for analysis of other embedding operations. We will study one other in the following section.

5. SOURCE CODING

The technology of Sect. 4 can also be applied to more efficient source coding. A common example of efficiency-boosting coding is *matrix embedding* [8], which allows relatively fewer embedding changes to be made, when embedding below-maximal payloads. In previous square root laws, such source coding has either been excluded, or the capacity result stated in terms of embedding changes instead of payload. It has often been stated, for example in [15], that asymptotically maximally efficient source coding turns the capacity law $\sqrt{n}$ into $\sqrt{n}\log n$, but that was to ignore any dependencies introduced by the matrix embedding process (or to assume that it was kept secret from the enemy, but that presents the same key size problems as with the choice of embedding locations). We now confirm that the structure of a simple embedding scheme does not give the enemy any additional advantage.

Like the scheme used in Sect. 3, the embedding method is probably the simplest, first published in [17]. It uses syndromes of binary Hamming codes: let $\mathbf{H}_p$ be the parity check matrix for a $[2^p-1, 2^p-1-p]$ binary Hamming code, namely the binary $p \times 2^p - 1$ matrix whose columns are the binary representations of numbers $1, \ldots, 2^p - 1$.

Algorithm 2. Let the cover size be n and the payload size be m; compute the largest integer p such that

$$\left\lceil \frac{m}{p} \right\rceil \le \frac{n}{2^p - 1}.$$

Temporarily we will write $q = 2^p - 1$ as a convenient shorthand. Divide the cover, $(x_1, \ldots, x_n)$, into $\lfloor n/q \rfloor$ groups of q pixels each, with any remainder pixels left unused. Form the payload as a sequence of bits $(b_1, \ldots, b_m)$ into $\lceil m/p \rceil$ groups of p bits each, padding if necessary.

In each cover pixel group $G = (x_1, \ldots, x_q)$, form the vector of parities $c_1^q = (P(x_1), \ldots, P(x_q))$, and denote the corresponding group of payload bits a_1^p. Compute the column vector over binary arithmetic

$$y_1^p = a_1^p - \mathbf{H}_p c_1^q;$$

if y_1^p is a zero vector, alter none of the pixels in the group; if y_1^p is the vector which is the binary expansion of k then alter the parity of pixel x_k.

The recipient recovers the embedded payload by dividing the stego object pixels into the same groups and converting them to parity, then for each group of parities s_1^q computes

$$r_1^p = \mathbf{H}_p s_1^q.$$

Correctness of the algorithm follows because the columns of $\mathbf{H}$ enumerate the binary numbers, so $\mathbf{H}(s_1^q - c_1^q) = y_1^p$. Then $r_1^p = \mathbf{H}(s_1^q - c_1^q) + \mathbf{H}c_1^q = y_1^p + \mathbf{H}c_1^q = a_1^p$. Algorithm 2 embeds p payload bits into each group of $q = 2^p - 1$ cover locations, altering at most one location per group. As is well known, this is more efficient than the average of 1/2 changes per embedded bit of simple replacement. We now show the equivalent of a square root law for this situation.

THEOREM 5. *Suppose that the cover consists of n pixels $(X_1, \ldots, X_n)$ each drawn from a finite alphabet $\mathcal{X}$, independent and identically distributed each with mass function $p(x)$, and that Algorithm 2 is applied to embed payload of size m. Suppose that altered pixels have mass function $q(x)$. Finally, suppose that $p(x) \neq 0$ and $q(x) \neq 0$ for all $x \in \mathcal{X}$.*

If

$$\frac{m}{\sqrt{n}\log n} \to 0 \tag{7}$$

then, for sufficiently large n, any detector must have arbitrarily high error rate, even if they have knowledge of the groups and the parity function.

PROOF. We proceed as in Theorem 3, first considering a single group of $2^p - 1$ pixels. There is probability of 2^{-p} of no change, or of changing any one of the group. Let $\mathcal{P}$, $\mathcal{Q}$, and R_i be defined analogous to Theorem 3. By the same argument, the KL divergence of one group is

$$\begin{aligned} D_1 &= D_{\mathrm{KL}}(\mathcal{P} \parallel \mathcal{Q}) \\ &= -\mathrm{E}\left[\log\left(\frac{\mathcal{Q}(X_1^k)}{\mathcal{P}(X_1^k)}\right)\right] \\ &= -\mathrm{E}\left[\log\left(2^{-p}\left(1 + \textstyle\sum_{j=1}^{2^p-1} R_j\right)\right)\right]. \end{aligned}$$

Write $Y_p = 2^{-p}\big(1+\sum_j R_j\big)$; as in Theorem 3 we easily verify $\mathrm{E}[Y_p] = 1$ and compute both

$$\mathrm{Var}[Y_p] = \frac{2^p - 1}{4^p}\mu_2,$$

giving (C2), and

$$\mathrm{E}\big[(Y_p - 1)^4\big] = \frac{2^p - 1}{16^p}\mu_4 + \frac{3(2^p - 1)(2^p - 2)}{16^p}\mu_2^2,$$

which satisfies the condition of Lemma 3 so that we can deduce (C3). (C4) follows because $\mathcal{X}$ is finite and q nonzero, so $R_j = q(X_j)/p(X_j)$ is bounded below away from zero. Therefore Lemma 2 gives

$$D_1 \sim \frac{1}{2}\mathrm{Var}[Y_p] \sim \frac{\mu_2}{2}\frac{2^p - 1}{4^p}.$$

Then consider the KL divergence between entire cover and stego objects: it is a sum arising from $\lceil m/p \rceil$ independent groups,

$$D = \left\lceil \frac{m}{p} \right\rceil D_1 \sim \left\lceil \frac{m}{p} \right\rceil \frac{\mu_2}{2}\frac{2^p - 1}{4^p} \sim \frac{\mu_2}{2}\frac{m}{p\,2^p}.$$

It remains to show that $D \to 0$ given (7); the analysis is a bit fiddly.

Recall that, by definition of p,

$$\left\lceil \frac{m}{p+1} \right\rceil (2^{p+1} - 1) > n. \tag{8}$$

We first show that, for sufficiently large n,

$$p \ge \tfrac{1}{4}\log_2 n. \tag{9}$$

Suppose not, then there exists a sufficiently large n with $m+1 \le \sqrt{n}\log n$ (by (7)) and $2\log_2 n \le n^{1/4}$ (by elementary analysis), and we obtain the contradiction with (8),

$$\left\lceil \tfrac{m}{p+1} \right\rceil (2^{p+1} - 1) < 2(m+1)2^p < 2(m+1)n^{\frac{1}{4}} \le 2n^{\frac{3}{4}}\log n \le n.$$

Then consider the cases $m \geq p$ and $m < p$ separately:

$$\begin{aligned}\frac{m^2}{n(\log_2 n)^2} &\overset{(8)}{>} \frac{m^2}{\left(\frac{m}{p+1}+1\right)(2^{p+1}-1)(\log_2 n)^2} \\ &> \frac{m^2 p}{(m+p)2^{p+1}(\log_2 n)^2} \\ &\overset{(a)}{\geq} \frac{mp}{4\,2^p(\log_2 n)^2} \\ &\overset{(9)}{\geq} \frac{m}{64\,p\,2^p}\end{aligned}$$

with (a) holding as long as $m \geq p$, or

$$\frac{m}{p\,2^p} < \frac{1}{2^p} \leq \frac{1}{n^{1/4}}$$

otherwise. Thus

$$D \sim \frac{\mu_2}{2}\frac{m}{p\,2^p} < \frac{\mu_2}{2}\max\left(\frac{64\,m^2}{n(\log_2 n)^2}, \frac{1}{n^{1/4}}\right) \to 0.$$

□

The converse, analogous to Theorem 1(i), is also true. It states that $m/\sqrt{n}\log n \to \infty$ leads to asymptotically perfect detection. This holds not only for the simple Hamming code matrix embedding used here, but for any embedding scheme. It uses the well-known rate distortion bound and is not difficult, but it is outside the scope of this work.

It is interesting that the oldest matrix embedding idea still attains the asymptotically best embedding "rate" $\sqrt{n}\log n$. One is then motivated to ask: what is the equivalent of SFI in this case? The natural equivalent is to look for a constant I satisfying

$$D_{\mathrm{KL}}(\mathcal{P} \parallel \mathcal{Q}) \sim \frac{I}{2}\frac{m^2}{n\log n},$$

where here $\mathcal{P}$ denotes the distribution of entire covers of size n, and $\mathcal{Q}$ of stego objects with payload size m. However, there is a difficulty. For the scheme described in Algorithm 2, the quotient

$$D_{\mathrm{KL}}(\mathcal{P} \parallel \mathcal{Q})\frac{n\log n}{m^2} \tag{10}$$

does not converge! It has a limit inferior which is half its limit superior, and oscillates more and more slowly between these asymptotes. This is because Algorithm 2 relies on the discrete Hamming code family, and for unfavourable fractions m/n a lot of cover is used inefficiently. Speculatively, we would suggest that *Equivalent Steganographic Fisher Information* (ESFI), the equivalent of SFI for embedding when source coding is used, should be defined by the limit inferior of (10). This could be argued because combinations of codes can usually be concatenated to achieve a favourable average, but there is certainly more work required to justify this properly.

6. DISCUSSION

Despite the (admittedly tongue-in-cheek) title, the results in this paper do not contradict those in [14]. For the situation described there, in which each payload bit or word is placed at a single embedding location, a linear key is indeed necessary to avoid key exhaustion attacks by the enemy. This is an essential weakness of embedding schemes which require the recipient to know the exact embedding locations. What we have shown here is that the issue can be circumvented entirely, by spreading the possible payload locations more widely in larger covers: no key is needed, yet asymptotically perfect steganographic security is achieved. In a sense, this is a sort of "public key" steganography, which is asymptotically perfectly secure[1].

We also followed a similar line for matrix embedding, verifying that the capacity of imperfect steganography is of order $\sqrt{n}\log n^2$. There are some complications with the simple Hamming code we used, because its best efficiencies are only gained for rare combinations of cover and payload size.

Algorithms 1 and 2 have one thing in common: they are the simplest examples of their type, respectively a trivial wet paper code and the simplest nontrivial matrix embedding. It was necessary to consider simple examples because, when proving asymptotic security, it is important to take account of the structure caused by the embedding procedure which introduces dependencies into the embedding. In these cases the structure is the fact that no pixel group can ever have more than one change. It required some quite intricate analysis to bound the KL divergence, and it would have been even more difficult to examine more complicated wet paper codes or matrix embedding schemes.

In fact, there is arguably no need to examine more general wet paper codes: Theorem 4 shows not only that the embedding in Algorithm 1 achieves the asymptotically optimal embedding rate of $\sqrt{n}$, but also that its constant multiple is equal to that of "ideal" uniform embedding. Wet paper codes certainly have useful applications when the wet pixels cannot be altered, but they cannot give any useful advantage to the "public key" problem we have addressed in this work.

On the other hand, there may well be value in considering better matrix embedding schemes than Algorithm 2, as it is known that greater embedding efficiency (bits embedded per location changed) is possible. We performed some preliminary work in this direction, with surprising initial results. One of the leading source coding schemes is given by the construction in [18], dubbed ZZW after its authors, which converts one matrix embedding scheme into another. The construction is (in a sense described in [6]) optimal when it is applied to the trivial embedding of one bit per location. It involves, like Algorithm 2, breaking the cover image into groups and using a Hamming code within each group; it also involves a wet paper code so that the changes in each block can signal further information (see [18] for the details). The second stage makes its analysis difficult, because the different blocks are no longer independent.

However, if we pretend that the different blocks are indeed independent we can apply a similar analysis to that in Theorem 5. Like Algorithm 2, it achieves a $\sqrt{n}\log n$ capacity rate. What is interesting is that its Equivalent Steganographic Fisher Information, if we use the speculative definition $2\liminf D_{\mathrm{KL}}(\mathcal{P} \parallel \mathcal{Q})n\log n/m^2$, appears to

[1]We proved that capacity below $\sqrt{n}$ is asymptotically perfectly secure; that capacity above $\sqrt{n}$ is asymptotically perfectly detectable comes for free from the classical square root law.

[2]We proved that capacity below $\sqrt{n}\log n$ is asymptotically perfectly secure; it indeed true that above this rate leads to asymptotic perfect detection, but that lies beyond the scope of this work.

be the same as for Algorithm 2. Note that the ZZW scheme may be *less* secure than this, because we have ignored some of the correlation structure the embedding causes, but it could not be any more secure when that is taken into account. Therefore, it seems, the plain Hamming code is at least as secure as this ZZW construction, in the same sense that Algorithm 1 is at least as secure as ideal uniform embedding. This would be a curious result, because the ZZW construction is more efficient in terms of bits conveyed per change. Perhaps this can be reconciled by the comment in [6], where it is noted that both Hamming and ZZW families "parallel" the rate distortion bound, when embedding efficiency against embedding rate is charted in a particular way. We might conjecture that all schemes which have the same property are equally secure as regards their asymptotic KL divergence ESFI. The advantage of ZZW lies in its application to small, finite, cases.

There is some natural further work arising. The results should be extended to more practical cover models, particularly the Markov model considered in [5]; however, the analysis is likely to be very difficult and new abstractions will be required. It may be reassuring, to those who have to use them, to extend the analysis of Algorithm 1 to random wet paper codes: it seems likely that the method for solving (4) will have to be carefully chosen, perhaps even to the point of finding *all* solutions and choosing uniformly between them, otherwise attacks such as in [2] might be applied.

Finally, we have sketched a possible extension of the concept of Steganographic Fisher Information, which is closely related to the maximum possible *root rate* r when $m \sim r\sqrt{n}$ [4, 13], to embedding which allows for source coding. Such Equivalent Steganographic Fisher Information would be related to the maximum possible r when $m \sim r\sqrt{n}\log n$, but the discrete nature of codes creates a discontinuity making even the definition of ESFI rather difficult. Perhaps it would be best to take the approach of [13] (amongst many other publications on theoretical capacity) and keep the number of permissible changes entirely separate from the choice of embedding code.

7. ACKNOWLEDGMENTS

The author is a Royal Society University Research Fellow.

8. REFERENCES

[1] R. Anderson and F. Petitcolas. On the limits of steganography. *IEEE J. Selected Areas in Communications, Special Issue on Copyright and Privacy Protection*, 16(4):474–481, 1998.

[2] R. Böhme. Wet paper codes for public key steganography? Unpublished rump session talk at 7th Information Hiding Workshop, Barcelona, Spain, 2005. Available at `http://www.inf.tu-dresden.de/~rb21/publications/Boehme2005_IHW_RumpSession.pdf`.

[3] W. Feller. *An Introduction to Probability Theory and Its Applications, Volume II*. Wiley, 1966.

[4] T. Filler and J. Fridrich. Fisher Information determines capacity of ϵ-secure steganography. In *Proc. 11th Information Hiding Workshop*, volume 5806 of *Springer LNCS*, pages 31–47, 2009.

[5] T. Filler, A. Ker, and J. Fridrich. The square root law of steganographic capacity for Markov covers. In *Media Forensics and Security XI*, volume 7254 of *Proc. SPIE*, pages 0801–0811, 2009.

[6] J. Fridrich. Asymptotic behavior of the ZZW embedding construction. *IEEE Trans. Information Forensics and Security*, 4(1):151–153, 2009.

[7] J. Fridrich, M. Goljan, P. Lisoněk, and D. Soukal. Writing on wet paper. *IEEE Trans. Sig. Proc., Special Issue on Media Security*, 53:3923–3935, 2005.

[8] J. Fridrich, P. Lisoněk, and D. Soukal. On steganographic embedding efficiency. In *Proc. 10th Information Hiding Workshop*, volume 4437 of *Springer LNCS*, pages 60–71, 2008.

[9] R. Graham, D. Knuth, and O. Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley, 1994.

[10] A. Ker. A capacity result for batch steganography. *IEEE Signal Processing Letters*, 14(8):525–528, 2007.

[11] A. Ker. The ultimate steganalysis benchmark? In *Proc. 9th ACM Workshop on Multimedia and Security*, pages 141–148, 2007.

[12] A. Ker. Steganographic strategies for a square distortion function. In *Security, Forensics, Steganography and Watermarking of Multimedia Contents X*, volume 6819 of *Proc. SPIE*, pages 0401–0413, 2008.

[13] A. Ker. Estimating Steganographic Fisher Information in real images. In *Proc. 11th Information Hiding Workshop*, volume 5806 of *Springer LNCS*, pages 73–88, 2009.

[14] A. Ker. The Square Root Law requires a linear key. In *Proc. 11th ACM Workshop on Multimedia and Security*, pages 85–92, 2009.

[15] A. Ker, T. Pevný, J. Kodovský, and J. Fridrich. The square root law of steganographic capacity. In *Proc. 10th ACM Workshop on Multimedia and Security*, pages 107–116, 2008.

[16] B. Ryabko and D. Ryabko. Asymptotically optimal perfect steganographic systems. *Problems of Information Transmission*, 45(2):184–190, 2009.

[17] A. Westfeld. F5 – a steganographic algorithm: High capacity despite better steganalysis. In *Proc. 4th Information Hiding Workshop*, volume 2137 of *Springer LNCS*, pages 289–302, 2001.

[18] W. Zhang, Z. Zhang, and S. Wang. Maximizing steganographic embedding efficiency by combining Hamming codes and wet paper codes. In *Proc. 10th Information Hiding Workshop*, volume 4437 of *Springer LNCS*, pages 282–296, 2008.

APPENDIX

A. (C3) FOR UNIFORM EMBEDDING

We have

$$Y_n = \mathrm{E}_{C\sim\mathcal{C}}\left[\prod_{c\in C}\frac{q(X_c)}{p(X_c)}\right]$$

and wish to apply Lemma 3 to deduce (C3). Hence we need to show that

$$\frac{\mathrm{E}\left[(Y_n-1)^4\right]}{\mathrm{Var}[Y_n]} \to 0.$$

Only a sketch proof is included.

First, use the expansion

$$\mathrm{E}\big[(Y_n-1)^4\big] = \mathrm{E}[Y_n^4-1] - 4\mathrm{E}[Y_n^3-1] + 6\mathrm{E}[Y_n^2-1]. \quad (11)$$

Then Lemma 4 can be extended to cover the higher powers of Y_n, transforming (11) into

$$\begin{aligned}
&\mathrm{E}_{\substack{C_1\sim\mathcal{C}\\C_2\sim\mathcal{C}\\C_3\sim\mathcal{C}\\C_4\sim\mathcal{C}}}\Big[\kappa^{|4\text{ of }(C_1\ldots C_4)|}\lambda^{|3\text{ of }(C_1\ldots C_4)|}(\mu_2+1)^{|2\text{ of }(C_1\ldots C_4)|}\Big]\\
&-4\,\mathrm{E}_{\substack{C_1\sim\mathcal{C}\\C_2\sim\mathcal{C}\\C_3\sim\mathcal{C}}}\Big[\lambda^{|3\text{ of }(C_1,C_2,C_3)|}(\mu_2+1)^{|2\text{ of }(C_1,C_2,C_3)|}\Big] \qquad (12)\\
&+6\,\mathrm{E}_{\substack{C_1\sim\mathcal{C}\\C_2\sim\mathcal{C}}}\Big[(\mu_2+1)^{|2\text{ of }(C_1,C_2)|}\Big]
\end{aligned}$$

where "i of $(C_1,\ldots,C_n)$" indicates the set whose elements are those in precisely i of $C_1,\ldots,C_n$; κ and λ are positive constants whose value need not concern us here.

We must then consider overlaps between three or four independent sets of embedding locations, under ideal uniform embedding. Lemma 5(b) tells us that for H, a hypergeometric random variable which represents two or more coincidences between two independent sets of embedding locations, the cases $H \geq 2$ contribute only negligibly to $\mathrm{E}[a^H]$. In the same way it can be shown that, if J represents the number of triple or quadruple coincidences in three or four independent sets of embedding changes, $J \geq 1$ is negligible and hence the terms involving powers of κ and λ are negligibly different from 1.

Thus, writing ϵ_1 and ϵ_2 rather loosely for negligible terms, (12) becomes

$$\begin{aligned}
&(1+\epsilon_1)\,\mathrm{E}_{\substack{C_1\sim\mathcal{C}\\C_2\sim\mathcal{C}\\C_3\sim\mathcal{C}\\C_4\sim\mathcal{C}}}\Big[(\mu_2+1)^{|2\text{ of }(C_1,\ldots,C_4)|}\Big]\\
&-4(1+\epsilon_2)\,\mathrm{E}_{\substack{C_1\sim\mathcal{C}\\C_2\sim\mathcal{C}\\C_3\sim\mathcal{C}}}\Big[(\mu_2+1)^{|2\text{ of }(C_1,C_2,C_3)|}\Big] \qquad (13)\\
&+6\,\mathrm{E}_{\substack{C_1\sim\mathcal{C}\\C_2\sim\mathcal{C}}}\Big[(\mu_2+1)^{|2\text{ of }(C_1,C_2)|}\Big]
\end{aligned}$$

Now we can use the symmetry of $C_1,\ldots,C_4$ to reduce (13) to

$$\begin{aligned}
&\big((1+\epsilon_1).6 - 4(1+\epsilon_2).3 + 6\big)\mathrm{E}_{\substack{C_1\sim\mathcal{C}\\C_2\sim\mathcal{C}}}\Big[(\mu_2+1)^{|C_1\cap C_2|}\Big]\\
&= (6\epsilon_1 - 12\epsilon_2)\mathrm{Var}[Y_n].
\end{aligned}$$

Hence

$$\frac{\mathrm{E}\big[(Y_n-1)^4\big]}{\mathrm{Var}[Y_n]}$$

is negligible as $n\to\infty$.