Moving Steganography and Steganalysis from the Laboratory into the Real World
Andrew Ker, Patrick Bas, Rainer Böhme, Rémi Cogranne, Scott Craver, Tomáš Filler, Jessica Fridrich, Tomas Pevny

To cite this version:
Andrew Ker, Patrick Bas, Rainer Böhme, Rémi Cogranne, Scott Craver, et al.. Moving Steganography and Steganalysis from the Laboratory into the Real World. ACM IH-MMSEC 2013, Jun 2013, Montpellier, France. pp.ACM 978-1-4503-2081-8/13/06, 2013. <hal-00836407>

HAL Id: hal-00836407
https://hal.archives-ouvertes.fr/hal-00836407
Submitted on 20 Jun 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
ABSTRACT

There has been an explosion of academic literature on steganography and steganalysis in the past two decades. With a few exceptions, such papers address abstractions of the hiding and detection problems, which arguably have become disconnected from the real world. Most published results, including by the authors of this paper, apply “in laboratory conditions” and some are heavily hedged by assumptions and caveats; significant challenges remain unsolved in order to implement good steganography and steganalysis in practice. This position paper sets out some of the important questions which have been left unanswered, as well as highlighting some that have already been addressed successfully, for steganography and steganalysis to be used in the real world.

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

Keywords
Steganography; Steganalysis; Security Models; Minimal Distortion; Optimal Detection; Game Theory

1. INTRODUCTION

Steganography is now a fairly standard concept in computer science. One occasionally reads, in mainstream media, of criminals hiding information in digital media ([1, 4], see [3] for other links) and, recently, of malware using it to conceal communications with command and control servers [5]. In the 1990s, the possibility of digital steganography served as an argument in debates about regulating cryptography, and it allegedly convinced some European governments to liberalize the use of cryptography [31]. We also read of the desire for certain privacy-enhancing technologies to use steganography to evade censorship [67]. If steganography becomes commonly used, so should steganalysis, though the concept is not as well recognized in nonspecialist circles.

However, where details of real-world use of steganography are known, it is apparent that they bear little resemblance to techniques described in modern literature. Indeed, they often suffer from flaws known to researchers for more than a decade. How has practice become so disconnected from research? The situation is even more stark in steganalysis, where most researchers would agree that their detectors work well only in laboratory conditions: unlike steganography, even if practitioners wanted and were technically able to implement state-of-the-art detectors, their accuracy would be uneven and unreliable.

The starting point for scientific research is to make a model of the problem. The real world is a messy place, and the model is an abstraction which removes ambiguities, sets certain parameters, and makes the problem amenable to mathematical analysis or empirical study. In this paper we contend that knowledge is the most important component in a model of the steganography and steganalysis problems. Does the steganographer have perfect knowledge about their source of covers? Does the steganalyist know the embedding method used by the steganographer? There are many questions of this type, often left implicit in early research.
By considering different levels of knowledge, we identify a number of models of the steganography and steganalysis problems. Some of them have been well-studied but, naturally enough, it is usually the simplest models which have received the most attention. Simple models may (or may not) provide robust theoretical results giving lower or upper bounds, and they increase our understanding of the fundamental problems, but they are tied to the laboratory. In this paper we identify the models which bring both steganography and steganalysis nearer to the real world. In many cases the scientific community has barely scratched their surface, and we highlight open problems which are, in the view of the authors, important to address in future research.

At the present time, steganography and steganalysis research divides into two cover types: digital media (primarily compressed and uncompressed images, but also video and audio) and network traffic (timing channels and the content of web traffic). The authors of this paper have their interest mainly in the former, and we contend that steganography and steganalysis is significantly more sophisticated in this domain than in network channels. Although network-based steganography is perhaps closer to real-world implementation, we will argue that the field needs to learn lessons from digital media steganography.

Many of the principles in this paper apply to any type of cover, but we shall be motivated by some general properties of digital media: the complexity of the cover and the lack of perfect models, the relative ease of (visual) imperceptibility, and large capacity per object. When, in examples, we refer to spatial domain we mean uncompressed images, and DCT or transform domain refers to JPEG-compressed images, both grayscale unless otherwise mentioned.

The paper has a simple structure. In section 2 we discuss current solutions, and open problems, relevant to applying steganography in the real world. In section 3 we do the same for steganalysis.

The Steganography Problem

We briefly recapitulate the steganography problem, refining Simmons’ original Prisoners’ Problem [92] to the contemporary definition of steganography against a passive warden.

A sender, often called Alice but who will throughout the paper be known as the steganographer, wishes to send a covert communication or payload to a recipient. She possesses a source of covers drawn from a larger set of possible communications, and there exists a channel for the communications (for most purposes we may as well suppose that the communication is unidirectional). The channel is monitored by an adversary, also known as an attacker or Warden but for the purposes of this paper called the steganalyst, who wishes to determine whether payload is present or not.

One solution is to use a channel that the adversary is not aware of. This is how traditional steganography has reportedly been practiced since ancient times, and most likely prevails in the Internet age [46]. Examples include tools that hide information in metadata structures, at the end of files where standard parsers ignore it [103], or modifying network packet headers such as TCP time stamps [37]. (See [74] for a systematic discussion.)

However, this approach is not satisfactory because it relies on the adversary’s ignorance, a form of “security through obscurity”. In Simmons’ formulation, inspired by conservative assumptions typical in cryptology, the steganalyst is granted wide knowledge: the contents of the channel is perfectly observable by both parties, writable by the steganographer, and (for the “passive Warden” case which dominates this paper) read-only by the steganalyst. To enable undetectability, we must assume that cover messages run through the channel irrespective of whether hidden communication takes place or not, but this is something that we will need to make more precise later. The intended recipient of the covert payload is distinguished from the steganalyst by sharing a secret key with the steganographer (how such a key might be shared will be covered in section 2.5).

As we shall see later, this model is still imprecise: the Warden’s aims, the parties’ knowledge about the cover source, and even their knowledge about each other’s knowledge, all create different versions of the steganography and steganalysis problems.

We fix some notation used throughout the paper. Cover objects generated by Alice’s source will be denoted by \(X\), broken down where necessary into \(n\) elements (e.g. pixels in the spatial domain pixels, or DCT coefficients in the transform domain) \(X_1, \ldots, X_n\). The objects emitted by the steganographer – which may be unchanged covers or payload-carrying stego objects – will be denoted \(Y\), or sometimes \(Y_\beta\) where \(\beta\) denotes the size of the payload relative to the size of the cover (the exact scaling factor will be irrelevant). Thus \(Y_0\) denotes a cover object emitted by the steganographer.

In parts of the paper we will assume a probability distribution for cover and stego objects (even though, as we argue in section 2.1, this distribution is unknowable precisely): the distribution of \(Y_\beta\) will be denoted \(P_\beta\), or if the distribution depends on other parameters \(\Theta\) then \(P_\beta^{\Theta}\). Thus \(P_0\) is the distribution of cover objects from the steganographer’s source.

2. STEGANOGRAPHY

Steganographic embedding in a single grayscale image could be implemented in the real world, with a high degree of undetectability against contemporary steganalysis, if practitioners were to use today’s state of art. In this section we begin by outlining that state of art, and highlighting the open problems for its further improvement. However, the same cannot be said of creating a steganographic channel in a stream of multiple objects — which is, after all, the essential aim for systems supporting censorship resistance — nor for robust key exchange, and our discussion is mainly of open problems barely treated by the literature.

We begin, in section 2.1, with some results which live purely in the laboratory. They apply to the security model in which the steganographer understands her cover source perfectly, or has exponential amounts of time to wait for a perfect cover. In section 2.2 we move closer to the real world, describing methods which help a steganographer to be less detectable when embedding a given payload. They require, however, the steganographer to know a tractably-optimizable distortion function, which is really a property of her enemy. Such research was far from the real world until recently, and is moving to practical applicability at the present time. But it does not tell the steganographer whether her size of payload is likely to be detectable; some purely theoretical research is discussed in section 2.3, which gives rules of thumb for how payload should scale as properties of the cover vary, but it remains an open problem to determine an appropriate payload for a given cover.
In section 2.4 we modify the original steganography model to better account for the repeated nature of communications: if the steganographer wants to create a covert channel, as opposed to a one-shot covert communication, new considerations arise. There are many open research problems in this area. Section 2.5 addresses the key exchange between the steganographer and her participant. The problem is well-understood with a passive warden opponent, but in the presence of an active warden it may even be impossible.

Section 2.6 briefly surveys other ways in which weaknesses may arise in practice, having been omitted from the model, and section 2.7 discusses whether the steganographer can encourage real-world situations favourable to her.

2.1 The laboratory: perfect steganography

One can safely say that perfectly secure steganography is now well understood. It requires that the distribution of stego objects be identical to that of cover objects.

In a model where the covers are sequences (usually of fixed length) of symbols from a fixed alphabet, the steganographer fully understands the cover source if they know the distribution of the symbols, including any conditional dependence between them. In such a case, perfect steganography is a coding problem and the capacity or rate (the number of bits per cover symbol) of perfectly secure steganography is bounded by the entropy of the cover distribution. Constructions for such coding have been proposed, including the cases of a distortion-limited sender (the sender is limited in how much the cover can be modified) and even a power-limited active Warden (the Warden can inject a distortion of limited power), for i.i.d. and Markov sources [101].

However, such a model of covers is necessarily artificial. The distinction between artificial and empirical cover sources has been proposed in [14] and is pivotal to the study of steganography in digital media. Artificial sources prescribe a probability distribution from which cover objects are drawn, whereas empirical sources take this distribution as given somewhere outside the steganographic system, which we could call reality. The steganographer can sample an empirical distribution, thereby obtaining projections of parts of reality; she can estimate salient features to devise, calibrate, and test models of reality; but she arguably can never fully know it. The perfect security of the preceding constructions rests on perfect knowledge of the cover source, and any violation of this assumption breaks the security proof. In practical situations, it is difficult to guarantee such an assumption. In other words, secure steganography exists for artificial sources, but we can never be sure if the artificial source exists in practice. More figuratively, artificial channels sit in the corner of the laboratory farthest away from the real world. But they can still be useful as starting points for new theories or as benchmarks.

Perfect steganography is still possible, albeit at higher cost, with empirical cover sources. If (1) secure cryptographic one-way functions exist, (2) the steganalyst is at most equally limited in her knowledge about the cover source as the steganographer, and (3) the cover source can be efficiently sampled, then perfect steganography is possible (the rejection sampler), but embedding requires an exponential number of samples in the message length [14, Ch. 3]. Some authors work around the inconvenient embedding complexity by tightening the third assumption and requiring that sampling is efficient conditional to any possible history of transmitted cover objects [41, 85, 44], which is arguably as strong as solving the original steganography problem.

2.2 Optimal embedding

If the steganographer has to use imperfect steganography, which does not preserve exactly the distribution of objects, how should she embed to be less detectable? Designing steganography for empirical cover sources is challenging, but there has been great progress in recent years. The steganographer must find a proxy for detectability, which we call distortion. Then message embedding is formulated as source coding with a fidelity constraint [91] – the sender hides her message while minimizing an embedding distortion [58, 79, 39]. As well as providing a framework for good embedding, this permits one to compute the largest payload embeddable below a given embedding distortion, and thus evaluate the efficiency of a specific implementation (coding method).

There are two challenges here: to design a good distortion function, and to find a method for encoding the message to minimize the distortion. We consider the latter problem first.

Early steganographic methods were severely limited by their ability to minimize distortion tractably. The most popular idea was to embed the payload while minimizing the number of changes caused (matrix embedding [21]). Counting the embedding changes, however, implicitly assumes that each change contributes equally to detectability, which does not coincide with experimental experience.

The idea of adaptive embedding, where each cover element is assigned a different embedding cost, dates to the early days of digital steganography [31]. A breakthrough technique was to use syndrome-trellis codes (STCs) [29], which solve certain versions of the adaptive embedding problem. The designer defines an additive distortion between the cover and stego objects in the form

$$D(X, Y) = \sum_i \rho_i(X, Y_i),$$

where $\rho_i \geq 0$ is a local distortion measure that is zero if $Y_i = X_i$, and then embeds her message using STCs, which minimize distortion between cover and stego objects for a given payload.

STCs only directly solve the embedding problem for distortion functions that are additive in the above sense, or where an additive approximation is suitable. Recently, suboptimal coding schemes able to minimize non-additive distortion functions were proposed, thereby modelling interactions among embedding changes, using the Gibbs construction. This can be used to implement embedding with an arbitrary distortion that can be written as a sum of locally supported potentials [27]. Unfortunately, such schemes can only reach the rate-distortion bound for additive distortion measures. Moving to wider classes of distortion function, along with provably optimal and practical coding algorithms, is an area of current research.

Open Problem 1 Design efficient coding schemes for non-additive distortion functions.

How, then, to define the distortion function? For the steganographer, the distortion function is a property of her enemy, the steganalyst. If she were to know what steganalysis she is up against then it would be tempting to use the same feature representation as her opponent, defining $D(X, Y) = ||f(X) - f(Y)||$, where $f$ is the feature extrac-
Design of heuristic distortion functions is currently a highly active research direction. It seems that the key is to assign high costs to changes to areas of a cover which are “predictable” from other parts of the stego object or other information available to the steganalist. For example, one may use local variance to compute pixel costs in spatial domain images [97]. The embedding algorithm HUGO [79] uses an additive approximation of a weighted norm between cover and stego features in the SPAM feature space [78], with high weights assigned to well-populated feature bins and low weights to sparsely populated bins that correspond to more complex content. An alternative distortion function called WOW (Wavelet Obtained Weights) [40] uses a bank of directional high-pass filters to assign high distortion where the content is predictable in at least one direction. It has been shown to resist steganalysis using rich models [35]. A further development is published in these proceedings.

One can expected that future research will turn to computer vision literature, where image models based on Markov Random Fields [102, 87, 94] are commonly trained and then utilized in various Bayesian inference problems.

In the domain of grayscale JPEG images, by far the most successful paradigm is to minimize the distortion w.r.t. the raw, uncompressed cover image, if available [58, 86, 100, 43]. In fact, this “side-informed embedding” can be applied whenever the sender possesses a higher-quality “precover” that was quantized to obtain the cover. Currently, the most secure embedding method for JPEG images that does not use any side information is the heuristically-built Uniform Embedding Distortion [39] that substantially improved the previous state of the art: the nS5 algorithm [36].

Open Problem 3 Distortion functions which take account of side information.

We conclude by highlighting the scale of research advances seen in embedding into grayscale (compressed or uncompressed) images. The earliest aims to reduce distortion attempted to correct macroscopic properties (e.g., an image histogram) by compensating embedding changes with additional correction changes, but in doing so made themselves more detectable, not less. We have progressed through a painful period where distortion minimization could not tractably be performed, to the most recent adaptive methods. However, we know of no literature addressing the parallel problems:

Open Problem 4 Distortion functions for colour images and video, which take account of correlations in these media.

Network steganography has received substantial attention from the information theory community through the analysis of covert timing channels [6, 98], which uses delays between network packets to embed the payload. However, the implementations are usually naive, using no distortion with respect to delays of normal data [16, 12]. The design of the embedding schemes focuses mainly on robustness with respect to the network itself, because network steganography is an active steganography problem. To the knowledge of the authors, the only work that considers a statistical distortion between normal and stego traffic is provided in [9].

2.3 Scaling laws

In this section we discuss some theory which has relevance to real-world considerations. These results rest on some information theory: the data processing theorem for Kullback-Leibler (KL) divergence [69]. We are interested in KL divergence between cover objects and stego objects, which we will denote $D_{KL}(P_0 || P_\beta)$. Cachin [17] described how an upper bound on this KL divergence implies an upper bound on the performance of any detector; we do not repeat the argument here. What matters is that we can analyze KL divergence, for a range of artificial models of covers and embedding, and obtain interesting conclusions.

As long as the family of distributions $P_\theta$ satisfies certain smoothness assumptions, for fixed cover parameters $\theta$ the Taylor expansion to the right of $\beta = 0$ is

$$D_{KL}(P_0 || P_\beta) \sim \frac{n}{2} \beta^2 I^\theta(0),$$

where $n$ is the size of the objects and $I^\theta(0)$ is the so-called Fisher information. This can be interpreted in the following manner: in order to keep the same level of statistical detectability as the cover length $n$ grows, the sender must adjust the embedding rate so that $n^{3/2}$ remains constant. This means that the total payload, which is $n^{3/2}$, must be proportional to $\sqrt{n}$. This is known as the square root law of imperfect steganography. Its effects were observed experimentally long before it was formally discovered first within the context of batch steganography [50], experimentally confirmed [57], and finally derived for sources with memory [30], where the reader should look for a precise formulation.

The law also tells us that the proper measure of secure payload is the constant of proportionality, $I^\theta(0)$, the Fisher information. The larger $I^\theta(0)$, the smaller the secure payload that can be embedded and vice versa. When practitioners design their steganographic schemes for empirical covers, one can say that they are trying to minimize $I^\theta(0)$, and it would be of immense value if the Fisher information could be determined for practical embedding methods. But it depends heavily on the cover source, and particularly on the likelihood of rare covers, which by definition is difficult.
to estimate empirically, and there has as yet been limited
progress in this area, benchmarking [26] and optimizing [53]
simple embedding only in restrictive artificial cover models.

**Open Problem 5** Robust empirical estimate of steganographic Fisher information.

What is remarkable about the square root law is that, al-
though both asymptotic and proved only for artificial sources,
it is robust and manifests in real life. This is despite the
fact that practitioners detect steganography using empirical
classifiers which are unlikely to approach the bound given
by KL divergence, and the fact that empirical sources do
not match artificial models. Beware, though, that it tells us
how the secure payload scales when changing the number
of cover elements, without changing their statistical proper-
ties — e.g. when cropping homogeneous images or creating
a panorama by simple composition — but not when a cover
is resized, because resizing changes the statistical properties
of the cover pixels by weakening (if downscaling without an-
tialiasing) or strengthening (if using a resampling kernel)
their dependencies.

We can still say something about resized images, if we
accept a Markov chain cover model. When nearest neigh-
bour resizing is used, one can compute numerically $I^0(0)$
as a function of the resizing factor (which should be thought of
as part of $\theta$) [64]. This allows the steganographer to adjust
her payload size with rescaling of the cover, and the theory
aligns robustly with experimental results.

**Open Problem 6** Derivation of Fisher information for
other rescaling algorithms, and richer cover models.

Finally, one can ask about the impact of quantization.
This is relevant as practically all digital media are obtained
by processing and quantizing the output of some analogue
sensor, and a JPEG image is obtained from a raw image
by quantizing the real-valued output of a transform. For
example, how much larger payload can one embed in 10-
bit grayscale images than in 8-bit? (Provided that both bit
depths are equally plausible on the channel.) How much
more data can be hidden in a JPEG with quality factor
98 than quality factor 75? We can derive (in an appropriate
limit) $I^0(0) \sim \Delta^s$, where $\Delta > 0$ is the quantization step and
$s$ is the quantization scaling exponent that can be calculated
from the embedding operation and the smoothness of the
unquantized distribution [32]. In general, the smoother the
unquantized distribution, the larger $s$ is and the smaller the
Fisher information (larger secure payload). The exponent $s$
is also larger for embedding operations that have a smooth-
ing effect. Because the KL divergence is an error exponent,
quantization has a profound effect on security. The experi-
ments in [32] indicate that even simple LSB matching may
be practically undetectable in 10–12 bit grayscale images.
However, unlike the scaling predicted by the square root law,
since the result for quantization depends strongly on the
distribution of the unquantized image, it cannot quan-
titatively explain real life experiments.

2.4 Multiple objects

Simmons’ 1983 paper used the term “subliminal channel”,
but the steganography we have been describing is not fully
a channel: it focused on embedding a certain length payload
in one cover object. For a channel, there must be infinitely
many stego objects (perhaps mixed with infinitely many in-
ocent cover objects) transmitted by the steganographer.

How do we adapt steganographic methods for embedding in
one object to embedding in many? How should one allocate
payload between multiple objects? There has been very littlere search on this important problem, which is particularly
relevant to hiding in network channels, where communica-
tion is naturally repeated.

In some versions of the model, this is fundamentally no dif-
ferent from the simple steganography problem in one object.
Take the case, for example, where the steganographer has
a fixed number of covers, and decides how to allocate pay-
load amongst them (the batch steganography problem posed
in [48]). Treating the collection as a single large object is
possible if the full message and all covers are instantly avail-
able and go through the same channel (e.g., stay on the same
disk as a steganographic file system). In principle, this re-
duces the problem to what has been said above. It is worth
pointing out that local statistical properties are more likely
to change between covers than between symbols within one
cover. However, almost all empirical practical cover sources
are heterogeneous (non-stationary): samplers and distortion
functions have to deal with this fact anyway. And knowing
the boundaries between cover objects is just another kind of
side information.

The situation is more complicated in the presence of real-
time constraints, such as requirements to embed and com-
municate before the full message is known or before all cov-
ers are drawn. This happens, for example, when tunnelling
bilateral protocols through steganographic channels. Few
publications have addressed the stream steganography prob-
lem (in analogy to stream ciphers) [31, 52]. One interesting
result is known for payload allocation in infinite streams with
imperfect embedding (and applies only to an artificial setup
where distortion is exactly square in the amount of payload
per object): the higher the rate that payload is sent early,
the lower the eventual asymptotic square root rate [52].

A further generalization is to replace the “channel” by a
“network” communications model, where the steganographer
serves multiple channels, each governed by specific cover
source conventions, and with realtime constraints emerging
from related communications. Assuming a global passive
stenanalyst who can relate evidence from all communica-
tions, this becomes a very hard instance of a steganogra-
phy problem, and one that seems relevant for censorship-
resistant multiparty communication or to tunnel covert col-
laboration [10].

**Open Problem 7** Theoretical approaches and practical
implementations for embedding in multiple objects in the
presence of realtime constraints.

2.5 Key exchange

A curious problem in a steganographic environment is that
of key exchange. If a reliable steganographic system exists,
can parties use that channel to communicate, without first
sharing a secret key? In the cryptographic world, Alice and
Bob use a public-key cryptosystem to effect a secret key
exchange, and then communicate with a symmetric cipher;
one would assume that some similar exchange would enable
communication with a symmetric stegosystem. However,
a steganographic channel is fundamentally different from a
traditional communications channel, due to its extra con-
straint of undetectability. This constraint also limits our
ability to transmit datagrams for key establishment.
Key exchange has been addressed with several protocols and, paradoxically, negative results. The first protocol for key exchange under a passive warden [7] was later augmented to survive an active warden [8]. Here Alice and Bob use a public embedding key to transmit traditional key exchange datagrams: a public encryption key, and then a session key encrypted with that public key. These datagrams are visible to the warden, but they are designed to resemble channel noise so that the warden cannot tell if the channel is in use. This requires a complete lack of observable structure in the keys.

To prevent an active warden from altering the datagrams, the public embedding key is made temporarily private: first a datagram is sent with a secret embedding key, and then this key is publicly broadcast after the stego object passes the warden. In [22] it was argued that a key broadcast is not allowed in a steganographic setting, but that a key could be encoded as semantic content of a cover.

This may seem to settle the problem, but recent results argue that these protocols, and perhaps any such protocols, are practically impossible because the datagrams are sensitive to even a single bit error. If an active warden can inflict a few errors, we have a problem due to a fundamental difference between steganographic and traditional communication channels: we cannot use traditional error correction, because its presence is observable structure that betrays the existence of a message. In [71], it was shown that this fragility cannot be fixed in general: most strings are a few surgical errors away from a failed transmission; this allows key exchange to be derailed with an asymptotically vanishing error rate. It is not clear who will have the upper hand in practice: an ever-vigilant warden can indefinitely postpone key exchange with little error, but a brief opportunity to transmit some uncorrupted datagrams results in successful key transmission, whereupon the warden loses.

A final problem in steganographic key exchange is the state of ignorance of sender and receiver, and the massive computational burden this implies. Because key datagrams must resemble channel noise, nobody can tell if or when they are being transmitted; by the constraints of the problem, neither Alice nor the warden can tell if Bob is participating in a protocol, or innocently transmitting empty covers. This is solved by brute force: Bob assumes that the channel noise of every image is a public key, and sends a reply. Alice makes similar assumptions, both repeatedly attempting to generate a shared key until they produce one that works.

Open Problem 8 Is this monstrous amount of computation necessary, or is there a protocol with more efficient guesswork to allow Alice and Bob to converge on a key?

2.6 Basic security principles

Finally, even when a steganographic method is secure, its security can be broken if there is information leakage of the secret key, or of the steganography software. We recall some basic principles that should be followed by the steganographer, in order to avoid security pitfalls.

- Her embedding key must be long enough to avoid exhaustion attacks [34], and any pseudorandom numbers generated from it must be strong.
- Whenever she wants to embed a payload in several images, she must avoid using the same embedding locations for each. Otherwise the steganalist can use noise residuals to estimate the embedding locations, reducing the entropy of the secret key [51]. One way to force the locations to vary is to add a robust hash of the cover to the seed.
- She must act identically to any casual user of the communication channel, which implies hiding also the use of steganographic software, and deleting temporary cover and stego objects. An actor that performs cover selection by emitting only contents that are known to be difficult to analyze (such as textured images) can seem suspicious in itself.

Open Problem 9 How to perform cover selection, if at all? How to detect cover selection?

- She has to beware of the pre- and post-processing operations that can be associated with embedding. Double compression can be easily detected [80] and forensic details, such as the ordering of different parts of a JPEG file, can expose the processing path [38].
- She should benchmark her embedding appropriately. In the case of digital images for example, it is not because the software produces imperceptible embedding that the payload is undetectable. Image quality metrics such as the PSNR and psychovisual metrics are of little interest in steganography.
- Her device capturing the cover should be trusted, and contents generated from this device should also stay hidden. Covers must not be re-used.

Several general principles should be kept in mind when designing a secure system. These include:

- The Kerckhoffs Principle, that a system should remain secure under the assumption that the adversary knows the system, although interpretations for steganography differ in whether this includes knowledge of the cover source or not.
- The Usability Principle (also due to Kerckhoffs), that a system should be easy for a layperson to use correctly. For example, steganographic software should enforce a square root law rather than expecting an end user to apply it.
- The Law of Leaky Abstractions [93], which requires us to be aware of, for example, statistical models of cover sources, assumptions about the adversary, or the abstraction of steganography as a generic communication channel. Even if we have provable security within the model, reality may deviate from the model in a way that causes a security weakness.
- The fact that steganographic channels are not communications channels in the traditional sense, and their limitations are different. Challenges of capacity, fidelity, and key exchange must be examined anew.

Open Problem 10 Are there abstractions that hold for steganography? Are its building blocks securely compos-able?

2.7 Engineering the real world for steganography

If we perfectly understood our cover sources, secure steganography would reduce to a coding problem. Engineering secure steganography for the real world is so difficult precisely because it requires us to understand the real world as well as our artificial models. If there is a consensus that the real world needs secure steganography, a completely different approach could be to engineer the real world so that parts of it match the assumptions needed for security proofs. This implies changing the conventions, via protocols and norms, towards more randomness in everyday communications, so that more artificial channels knowingly exist in the real world. For example, random nonces in certain protocols, or synthetic pseudorandom textures in video-games (if
implemented with trustworthy randomness) already provide opportunities for steganographic channels. Adding more of these increases the secure capacity ([23] proposes a concrete system). But this approach creates new challenges, many outside the domain of typical engineering, such as the social coordination problem of giving up bandwidth across the board to protect others’ communication relations, or the difficulty of verifying the quality of randomness.

Open Problem 11 Technical and societal aspects of inducing randomness in communications to simplify steganography.

3. STEGANALYSIS

Approaches to the steganalysis problem depend heavily on the security model, and particularly on the steganalist’s knowledge about the cover source and the behaviour of his opponent. The most studied models are quite far from real-world application, and (unlike steganography) most researchers would agree that state of the art steganalysis could not yet be used effectively in the real world.

Laboratory conditions apply in section 3.1, where we assume that the steganalist has perfect knowledge of (1) the cover source, (2) the embedding algorithm used by the steganographer, and (3) which object they should examine. This is as unrealistic as the parallel conditions in section 2.1, but the laboratory work provides a conservative attack framework (which in images might include size, camera settings, colour space, and so on).

When the embedding rate $\beta$ and all cover parameters $\theta$ are known, the steganalysis problem is to choose between the following hypotheses: $H_0 = \{ Y \sim P^0_\theta \}$ vs $H_1 = \{ Y \sim P^1_\theta \}$. These are two simple hypotheses, for which the Neyman-Pearson Lemma [70, Th. 3.2.1] provides a simple way to design an optimal test, the Likelihood Ratio Test (LRT):

$$\delta^{\text{LRT}} = \left\{ \begin{array}{ll} H_0 & \text{if } \Lambda(Y) = \frac{P^0_\theta(Y)}{P^1_\theta(Y)} < \tau \\ H_1 & \text{if } \Lambda(Y) = \frac{P^0_\theta(Y)}{P^1_\theta(Y)} \geq \tau, \end{array} \right. \quad (3)$$

with $\Lambda$ the likelihood Ratio (LR) and $\tau$ a decision threshold.

The LRT is optimal in the following sense: among all the tests which guarantee a maximum false-alarm probability $\alpha \in (0,1)$ the LRT maximizes the correct detection probability. This is not the only possible measure of optimality, which we return to in section 3.6.

Accepting, for a moment, the optimal detection framework, we can deduce some interesting “laboratory” results. Assume that pixels from a digital image are i.i.d.: then the statistical distribution $P^\theta$ of an image is its histogram. If cover samples follow a Gaussian distribution $X_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$, it has been shown [107] that the LR for the LSB replacement scheme can be written: $\Lambda(Y) \propto \sum_i (y_i - \bar{y}_i)(y_i - \mu_i)/\sigma_i^2$, where $k = k + (-1)^k$ is the integer $k$ with flipped LSB. This LR is similar to the well-known Weighted Stego-image statistic [33, 54] and justifies it post hoc as an optimal hypothesis test. Similarly, the LR for the LSB matching scheme can be written [18]: $\Lambda(Y) \propto \sum_i ((y_i - \mu_i)^2 - \frac{1}{k})/\sigma_i^2$. This shows that optimal detection of LSB matching is essentially based on pixel variance. Particularly since LSB matching has the effect of masking the true cover variance, this explains it has proved a tougher nut to crack than LSB replacement.

However, the assumption that pixels can be modelled as i.i.d. random variables is unrealistic. Similarly, the model of statistically independent pixels following a Gaussian distribution (with different expectation and variance) is of limited interest in the real world.

The description of the steganalysis problem in the framework of hypothesis testing theory emphasizes the practical difficulties. First, it seems highly unlikely that the embedding rate $\beta$ would be known to a steganalyst, unless they already know that steganography is being used. And when $\beta$ is unknown the design of an optimal statistical test becomes much harder because the alternative hypothesis $H_1$ is composite: it gathers different hypotheses, for each of which a different most powerful test exists.

There are two approaches to overcome this difficulty: design a test which is locally optimal around a target embedding rate [19, 107] (again these tests rely on a statistical model of pixels); or design a test which is universally optimal for any embedding rate [18] (unfortunately their optimality assumptions are seldom met outside “the laboratory”).

Open Problem 12 Theoretically well-founded, and practically applicable, detection of payload of unknown length.

Second, it is also unrealistic to assume that the vector parameter $\theta$, which defines the statistical distribution of the whole inspected medium, is perfectly known. In practice,
these parameters are unknown and would have to be estimated using a model. Here one could employ the Generalized Likelihood Ratio Test (GLRT), which estimates unknown parameters in the LRT by the method of maximum likelihood. Unfortunately, maximum likelihood estimators again depend on a particular models of covers, and furthermore the GLRT is not usually optimal.

Although models of digital media are not entirely convincing, a few have been used for steganalysis, e.g. [20], as well as models of camera post-acquisition processing such as demosaicking and colour correction [95]. Much is unexplored.

**Open Problem 13** Apply models from the digital imaging community, which do not require independence of pixels, to the optimal detection framework.

However, it is sobering to observe that a well-developed detector based on testing theory and Laplacian model of DCT coefficients [106] performs poorly in practice compared to the rather simple WS detector adapted to the JPEG domain [13]. As we have repeatedly stated, digital media steganography is a particularly difficult domain in which to understand the covers.

### 3.2 Binary classification

Absent a model of covers, currently the best image steganalyzers are built using feature-based steganalysis and machine learning. They rest on the assumption that the steganalyst has some samples from the steganographer’s cover source, so that its statistical properties can be learned, and also that they can create or otherwise obtain stego objects from these covers (for example by knowing the exact embedding algorithm). Typically, one starts by representing the media using a feature of a much smaller dimensionality, usually designed by hand using heuristic arguments. Then, a training database is created from the cover and stego examples, and a binary classifier is trained to distinguish the two classes.

Machine-learning steganalysis is fundamentally different from statistical signal processing approaches because one does not need to estimate the distribution of cover and stego images. Instead, this problem is replaced with a much simpler one: merely to distinguish the two classes. Thus, one can build classifiers that use high-dimensional features even with a limited number of training examples. When trained on the correct cover source, feature-based steganalysis usually achieves significantly better detection accuracy than analytically derived detectors (with the exception of LSB replacement).

There are two components to this approach: the features, and the classification algorithm.

Image steganalysis features have been well-studied in the literature. In the spatial domain, one usually starts by computing *noise residuals*, by creating and then subtracting an estimate of each cover pixel using its neighbours. The pixel predictors are usually built from linear filters, such as local polynomial models or 2-dimensional neighbourhoods, and can incorporate nonlinearity using the operations of maximum and minimum. The residuals improve the SNR (stego signal to image content). Typically, residuals are truncated and quantized into $2^T + 1$ bins, and the final feature vector is the joint probability mass function (co-occurrence) or conditional probability distribution (transition matrix) of $D$ neighbouring quantized residuals [78]. The dimensionality of this feature vector is $(2^T + 1)^D$, which quickly grows especially with the co-occurrence order $D$, though it can somewhat be reduced by exploiting symmetry.

In the JPEG domain, one can think of the DCT coefficients already as residuals and form co-occurrences directly from their quantized values. Since there exist dependencies among neighboring DCT coefficients both within a single $8 \times 8$ block as well as across blocks, one usually builds features as two-dimensional intra-block and inter-block co-occurrences [60]. It is also possible to build the co-occurrences only for specific pairs of DCT modes [62]. A comprehensive list of source code for feature vectors for raw and compressed images, along with references, is available at [2]. The current state of art in feature sets are unions of co-occurrences of different filter residuals, so-called *rich models*. They tend to be high-dimensional (e.g., $30000$ or more) but they also tend to exhibit the highest detection accuracy [35, 63].

We note that, in parallel to the steganography situation, steganalysis literature is mostly specialized to grayscale images: there exists only a little literature on steganalysis in video, e.g. [15, 47], and for various kinds of network traffic analysis [16, 104, 12]. The latter methods only use basic statistics such as the variance of inter-packet delays or quantiles of differences between arrival times. There is scope to transfer lessons from grayscale image steganalysis to these domains.

**Open Problem 14** Design features for colour images and video, which take account of correlations in these media, and rich features for network steganalysis.

Another problem specific to steganalysis of network traffic is the difficulty of acquiring large and diverse data sets.

The second component, the machine learning tool, is a very important part. When the training sets and feature spaces are small, the tool of choice is the support vector machine (SVM) [88] with Gaussian kernel, and this was predominant in the literature to 2011. But with growing feature dimensionality, one also needs larger training sets, and it becomes computationally unfeasible to search for hyperparameters. Thus, recently, simpler classifiers have become more popular. An example is the ensemble classifier [66], a collection of weak linear base learners trained on random subspaces of the feature space and on bootstrap samples of the training set. The ensemble reaches its decision by combining the decisions of individual base learners. (In contrast, decision trees are not suitable for steganalysis, because among the features there is none that is strong alone.) When trying to move the tools from the laboratory to the real world, one likely needs to further expand the training set, which may necessitate *online learning* such as the simple perceptron and its variants [72]. There has been little research in this direction. Online learning also requires fast extraction of features, which is in tension with the trend towards using many different convolution filters.

Although highly refined, the paradigm of training a binary classifier has some limitations. First, it is essentially a binary problem, which presupposes that the steganalyst knows exactly the embedding method and payload size used by their attacker. Dealing with unknown payload sizes has been approached in two ways: quantitative steganalysis (see section 3.7), or effectively using a uniform prior by creating the stego training set with random payload lengths [77]. An unknown embedding method is more difficult and changes to the problem to either a multi-class classification (com-
putationally expensive [76]) or one-class anomaly detection (section 3.4).

A more serious weakness is that the classifier is only as good as its training data. Although it is possible, in the real world, that the steganalyst has access to the steganographer’s cover source (e.g., he arrests her and seizes her camera), it seems an unlikely situation. Thus the steganographer must train the classifier on some other source. This leads to cover source mismatch, and the resulting classifier suffers from decreased accuracy. The extent of this decrease depends on the features and the classifier, in a way not yet fully understood. It is fallacious to try to train on a large heterogeneous data set as somehow “representative” of mixed sources, because it guarantees a mismatch and may still be an unrepresentative mixture.

Machine learning literature refers to the problem of domain adaptation, which could perhaps be applied to this challenge.

**Open Problem 15** Attenuate the problems of cover source mismatch.

A final issue in moving machine-learning steganalysis to the real world is the measure of detection accuracy. Popular measures such as $\min \frac{1}{2}(P_{FP} + P_{FN})$ correspond to the minimal Bayes risk under equally likely cover and stego images, which is doubtless in practice. Indeed, one might expect that real-world steganography is relatively rarely observed, so real-world steganalysis should be required to have very low false positive rates, yet steganalysis with very low false positive rates has hardly been studied. Even having a reliable false positive rate would be a good start, and there has been some research designing detectors with constant false-alarm rate (CFAR) [68], but it relies on artificial cover models and is also vulnerable to cover source mismatch. It should be noted that establishing classification error probabilities remains unsolved in general [90].

### 3.3 Adaptive classification

Suppose that, for different cover parameters $\theta$, we have trained different specialized binary classifiers. One possibility is to select the optimal classifier for each observed stego object. This approach has been used to tackle images which have double JPEG compression, and those with different JPEG quality factors (in the absence of quantization-blind features, such images have to be considered as coming from completely different sources) [76]. A similar approach specializing detectors to different covers has been pursued in [42].

This is a special case of fusion, where multiple classifiers have their answers combined in some weighted fashion. It presupposes that the cover parameters $\theta$ can reliably be estimated from the observed stego image, and that training data was available for all reasonable combinations of parameters. It is also very expensive in terms of training. In machine learning this architecture is known as a mixture of experts [105].

**Open Problem 16** Apply other fusion techniques to steganalysis.

### 3.4 Universal steganalysis

It is not always realistic to assume that the embedder knows anything about the embedding algorithm used by the steganographer. Universal steganalysis focuses on such a scenario, assuming that the steganalyst can draw empirically from the cover source but is otherwise ignorant. Despite being almost neglected by the community, such a problem is important for deployment of steganalysis in the real world.

Universal steganalysis considers the following hypothesis test: $H_0 = \{ Y \sim P_0^\theta \} \vs H_1 = \{ Y \sim P_\theta \}$. We can distinguish two cases: either the cover source is entirely known to the detector ($\theta$ is known and $H_0$ is simple), or not (both hypotheses are composite). The first version of the problem is unrealistic in the real world, for the reasons we previously cited. The second shows that detector design is about modeling a cover source, and practical approaches resort to modeling the distribution of cover images in a space determined by steganographic features. In comparison with the binary hypothesis testing scenario of section 3.2, this problem is much more difficult, because learning a probability distribution is unavoidably more difficult than learning a classifier [96]. We must expect that universal steganalyzers have inferior performance to targeted binary classifiers. In fact it is not straightforward to benchmark universal steganalysis, because there is no well-defined alternative hypothesis class from which to test for false negatives.

Universal steganalysis can be divided into two types: supervised and unsupervised. The former uses samples from the cover-source to create the cover model, e.g., by using one-class support vector machines [88] designed to solve the above hypothesis test under a false positive constraint. This approach has been investigated in [82, 73]. Obviously, the accuracy of supervised steganalysis is limited if the training data is not perfectly representative of the steganographer’s cover source and, if mismatched, the accuracy might be as bad as random guessing.

Unsupervised universal steganalysis tries to circumvent the problem of model mismatch by postponing building a cover model until the classification phase. It analyses multiple images at once, assuming that most of them are covers, and is therefore a form of outlier detection. To our knowledge there is no literature dealing with this scenario in steganalysis, though there are works dealing with it on the level of actors, treated in section 3.5.

**Open Problem 17** Unsupervised universal steganalysis.

The accuracy of universal steganalysis is to a large extent determined by the steganographic features, and features suitable for binary classification are not necessarily right for universal steganalysis. The features should be sensitive to changes caused by embedding, yet insensitive to variations between covers (including perhaps unnatural but non-steganographic processing techniques). Particularly in the case of unsupervised learning, the latter condition requires them to have low dimension, because unsupervised learning cannot learn to ignore irrelevant noise. A small number of features also facilitates training of supervised detectors, as it decreases the required number of samples to learn the probability distribution. An unstudied problem is therefore:

**Open Problem 18** Design of features suitable for universal steganalysis.

### 3.5 Pooled and multi-actor steganalysis

So far, the security models have assumed that the steganalyst has one object to classify, or if they have many then they know exactly which one to look at. This is highly un-
realistic and if steganalysis is to move to the real world it
will have to address the problem of *pooled steganalysis* [48]:
combining evidence from multiple objects to say whether
they collectively contain payload. It is in opposition to the
steganographic channel of section 2.4.

Although posed in 2006, there has been little success in
attacking this problem. One might say that it is no different
to binary steganalysis: simply train a classifier on multiple
images. But there are many practical problems to overcome:
should the feature set be the sum total of features from indi-
vidual images (if so, this loses information), or concatenated
(in which case how does one impose symmetry under permu-
tation)? To our knowledge, there has been no such detector
proposed in the literature, except for simple examples studied
when the problem was first posed [48, 49].

A related problem which, to the best of our knowledge, has
never been studied is *sequential detection*. When inspecting
VOIP traffic, for instance, it would be interesting to perform
online detection. The theoretically optimal detection is more
complex because time-to-decision also has to be taken into
account. The statistical framework of sequential hypothesis
tests should be applicable [99].

**Open Problem 19** Any detector for multiple objects, or
based on sequential hypothesis tests.

We can widen the steganalysis model still further, to a
realistic scenario relevant to network monitoring, if the steg-
analyst does not know even which user to examine. In this
situation the steganalyst intercepts many objects each from
many actors (e.g. social network users); their problem is to
determine which actor(s), if any, are using steganography in
some or all of their images.

This is the most challenging version of steganalysis, but
recent work [56, 55] has shown that the size of the problem
can be turned to the steganalist’s advantage: by calibrating
the behaviour of actors (as measured through steganalysis
features) by the behaviour of the majority, steganographers
can potentially be determined in an unsupervised and uni-
versal way. It amounts to an anomaly detection where the
unit is the actor, not the individual object. This can be
related to unsupervised intrusion detection systems [24].

This is a new direction in steganalysis and we say no more
about it here, but highlight the danger of false accusations:

**Open Problem 20** Can steganographers be distinguished
from unusual (non-stego) cover sources, by a detector which
remains universal?

### 3.6 Game theoretic approaches

The pooled steganalysis problem exposes an essentially
game-theoretic situation. When a (batch) steganographer
hides all their payload in one object, a certain type of de-
tector is optimal; when they spread their payload in many
objects, a different detector is optimal. These statements
can be proved in artificial models and observed in practice.
Indeed, the same can be said of *single* images: if the embed-
der always hides in noisy areas, the detector can focus their
attention there, and vice versa. A parallel situation most
likely exists in non-media covers.

Game theory offers an interesting perspective from which
to study steganography. If both steganographer and steg-
analyst know the cover source and are computationally uncon-
strained, the steganographer can embed perfectly; with a
shorter key if the steganalyst is computationally bounded.

If the steganographer is computationally bounded, but not
the steganalyst, the best she can do is to minimize the KL
divergence, subject to her constraints. Another way to frame
this is that she plays a minimax strategy against the best-
possible detector [45].

This may not add a lot of insight in the lab. But once
we step out into the real world, where knowledge of the
cover source is incomplete and computational constraints
defy finding globally optimal distortion functions or detec-
tors, then game theory becomes very useful. It offers a
wealth of solution concepts for situations where no maximin
or minimax strategies exist. A popular one is the notion of a
Nash equilibrium. It essentially says that among two sets of
strategies, one for the steganographer (choice of embedding
operation, distortion function, parameters etc.) and one for
the steganalyst (feature space, detector, parameters such as
local weights, etc.), there exist combinations where no player
can improve his or her outcome unilaterally. Although ex-
ploration of game theory for steganography has just begun,
and we are aware of only four independent approaches [25,
49, 75, 89], it seems to be a promising framework which
allows us to justify certain design choices, such as payload
distribution in batch steganography or distortion functions
in adaptive steganography. This is a welcome step to replace
heuristics with (some) rigor in the messy scenarios of limited
knowledge and computational power, as we find them in the
real world.

However, game theory for steganography is in its infancy,
and there are substantial obstacles to be overcome, such as:

**Open Problem 21** Find equilibria for practical covers, and
transfer insights of game-theoretic solutions from current toy
models to the real world.

### 3.7 Forensic steganalysis

Finally, what does the steganalyst do after detecting hid-
den data in an object? The next steps might be called *foren-
ic* steganalysis, and only a few aspects have been studied
in the literature.

If the aim of the steganalyst is to find targets for further
surveillance, or to confirm the existence of already-suspected
covert communication, circumstantial evidence such as sta-
tistical steganalysis is probably sufficient in itself. But for
law enforcement it is probably necessary to demonstrate the
content of a message by extracting it, in which case the first
step is to determine the embedding algorithm. This prob-
lem, largely neglected, has been studied in [81] for JPEG
images. The detection of different algorithms based on sta-
tistical properties will not be perfect, as methods with simi-
lar distortion functions and embedding changes are likely to
be confused, but this has not been studied for recent adap-
tive embedding methods.

**Open Problem 22** Can statistical steganalysis recognize
different adaptive embedding algorithms?

Some identify a specific implementation by a signature,
effectively relying on implementation mistakes [11, 103], but
this is unsatisfactory in general.

Once the embedding method is known, the next step is a
brute-force search for the embedding key. Very little re-
search has been done in this area, though two complemen-
tary approaches have been identified: using headers to verify
the correctness of a key [84], and comparing statistics along
potential embedding paths [34] in which the correct key deviates from the rest.

**Open Problem 23** Is there a statistical approach to key brute-forcing, for adaptive steganography?

Additionally, forensic steganalysis includes estimation of the length of the hidden message (quantitative steganalysis). This knowledge is useful to prevent “plausible deniability”, where the steganographer hides two messages, one of which is not incriminating and can be disclosed if forced. Such a scheme is uncovered if the total embedded payload can be estimated. Quantitative steganalysis is a regression problem parallel to binary classification, and the state of the art applies regression techniques to existing steganalysis features [83, 59].

4. CONCLUSIONS

Over the last ten years, ad-hoc solutions to steganography and steganalysis problems have evolved into more refined techniques. There has been a disparity in the rate of progress: grayscale images have received most of the attention, which should be transferred to colour images, video, other digital media, and non-media covers such as network traffic. Such transfer would bring both steganography and steganalysis closer to real-world implementation.

For steganography, we have stressed the distortion-minimization paradigm, which only became practical with recent developments in coding. There is no good reason not to use such a technique: there are efficiencies from the coding, and if there is a fear that current distortion functions might make detection paradoxically easier, one can use this feedback to redesign the distortion function, and continue the cycle of development. We expect further advances in coding to widen the applicability of such techniques.

For steganalysis, the binary classification case is well-developed, but there is a need to develop techniques that work with unknown algorithms, multiple objects, and multiple actors. Even the theoretical framework which we have highlighted, that of KL divergence as a fundamental measure of security, has yet to be adapted to these domains.

Acknowledgments

The work of A. Ker and T. Pevný is supported by European Office of Aerospace Research and Development under the research grant numbers FA8655-11-3035 and FA8655-13-1-3020, respectively. The work of S. Craver and J. Fridrich is supported by Air Force Office of Scientific Research under the research grant numbers FA9550-12-1-0124 and FA9550-09-1-0666, respectively. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of EOARD, AFOSR, or the U.S. Government.

The work of R. Cogranne is funded by Troyes University of Technology (UTT) strategic program COLUMBO. The work of T. Pevný is also supported by the Grant Agency of Czech Republic under the project P103/12/P514.

5. REFERENCES


[18] R. Cogranne and F. Retraint. An asymptotically uniformly most powerful test for LSB matching


A. D. Ker. Batch steganography and pooled steganalysis. In Information Hiding, 8th


