Simulating Suboptimal Steganographic Embedding

Christy Kin-Cleaves
Christy.Kin-Cleaves@cs.ox.ac.uk
Department of Computer Science, University of Oxford
Oxford, United Kingdom

Andrew D. Ker
adk@cs.ox.ac.uk
Department of Computer Science, University of Oxford
Oxford, United Kingdom

ABSTRACT

Researchers who wish to benchmark the detectability of stegano-graphic distortion functions typically simulate stego objects. However, the difference (coding loss) between simulated stego objects, and real stego objects is significant, and dependent on multiple factors. In this paper, we first identify some factors affecting the coding loss, then propose a method to estimate and correct for coding loss by sampling a few covers and messages. This allows us to simulate suboptimally-coded stego objects which are more accurate representations of real stego objects. We test our results against real embeddings, and naive PLS simulation, showing our simulated stego objects are closer to real embeddings in terms of both distortion and detectability. This is the case even when only a single image and message as used to estimate the loss.

KEYWORDS

Steganography, steganalysis, adaptive embedding, syndrome trellis codes, simulated embedding

1 INTRODUCTION

There are two ways to test a steganographic algorithm that minimises an additive cost function: embed messages using a Syndrome Trellis Code (STC) [14], or simulate an optimal code that meets the rate-distortion bound [12]. In a survey of several top venues for steganography and steganalysis research, we found significantly more work where researchers had simulated [1, 3–5, 9, 15–17, 22, 24–28, 30, 35–43, 47], as opposed to creating real stego objects [10, 11, 18–20, 44, 45]. There are two reasons that may have motivated them. First, the STC algorithm is complex to implement and slow to run on very large data sets. Second, separation of concerns suggests testing the cost function in isolation from the coding.

However, it is well known that the state-of-art STC method does *not* meet the rate-distortion bound. As we shall see, this coding loss is variable, depending on parameters of the STC and the

cost function itself. When authors simulate embedding, instead of performing genuine coding, the simulations have less distortion than genuine stego objects. Furthermore, this may advantage one distortion function over another.

In this paper we propose a simple method to estimate the coding loss from a small number of genuine encodings – in fact, we shall see that one is sufficient – and correct the simulation so that the realised distortion is close to that of STCs. It could be applicable to other coding methods as well. The relative error in the simulated distortion is reduced by an order of magnitude. Thus, for a very small overhead, researchers can simulate accurate levels of distortion and detectability. We only consider the binary case throughout this work, but our method could be applied to non-binary embedding.

1.1 Notation

We shall denote vectors as lowercase boldface x, the ith element of x denoted x_i, and the size of the set as $|x|$.

2 BACKGROUND

Stego objects can be simulated as follows. We briefly summarise the discussion in[13]. For a cover $x = \{x_1, \ldots, x_n\}$ a distortion function:

$$D(x, y) = \sum_{i=0}^{n} c_i \left[x_i \neq y_i \right] \quad | \quad c_i \in \mathbb{R}, \tag{1}$$

returns the sum of the additive change costs $c = \{c_1, \ldots, c_n\}$, for a stego object $y = \{y_1, \ldots, y_n\}$; changes are binary and independent. Suppose that we communicate a message with entropy m bits. The rate-distortion bound arises from the solution to the Payload-Limited Sender (PLS) optimisation problem:

$$\underset{p_1,\ldots,p_n}{\text{Minimise}} \ d = \sum_{i=1}^{n} p_i c_i \text{ subject to } m \leq \sum_{i=1}^{n} H(p_i), \tag{2}$$

where H is the binary entropy function, and then p_i is the optimal probability of changing x_i. Throughout this paper, we denote the minimum objective d by $PLS(c, m)$, where $PLS :$ **covers** $\times \mathbb{R} \rightarrow \mathbb{R}$. The same probabilities are the solution to the dual Distortion-Limited Sender (DLS) problem

$$\underset{p_1,\ldots,p_n}{\text{Maximise}} \ m = \sum_{i=1}^{n} H(p_i) \text{ subject to } \sum_{i=1}^{n} p_i c_i \leq d, \tag{3}$$

where d a distortion budget; in this case the aim is to communicate the maximum entropy m. We denote this maximum objective by $DLS(c, d)$, where $DLS :$ **covers** $\times \mathbb{R} \rightarrow \mathbb{R}$.

The solution to (2) and (3) is $p_i = 1/(1 + e^{\lambda c_i})$, for some λ such that the payload- or distortion-constraint is equality. Suitable values of λ can be found by conducting a binary search. Stego objects can be simulated by making each change c_i with probability p_i.
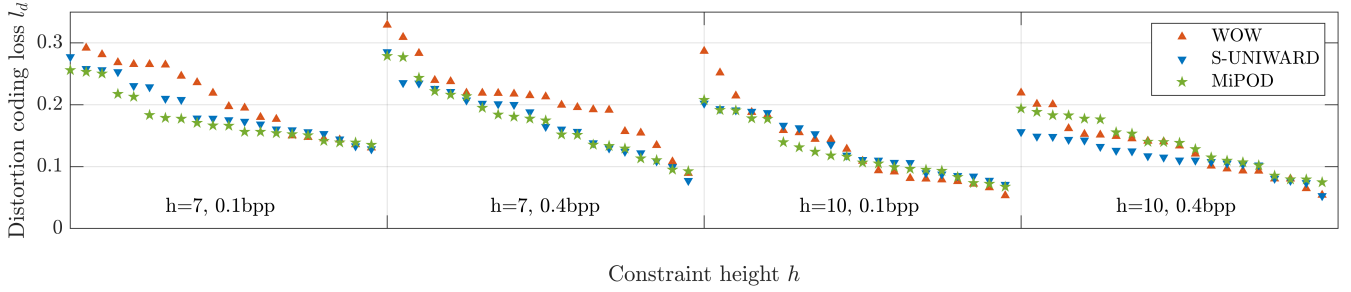
Figure 1: Comparison of per-cover distortion coding loss (l_d) for WOW, S-UNIWARD, and Mi-POD.

Genuine stego objects, on the other hand, can be created using the STC algorithm, which is based on the Viterbi algorithm [14]. The trellis has a parameter h, the *constraint height*, and its realised distortion approaches the rate-distortion bound asymptotically as $h \to \infty$; however, the time complexity is $O(2^h)$: in practice, $h \in (6, 10)$ provides an acceptable trade-off [14]. In this paper we use the notation $STC^h(c, m)$ for the distortion realised by an STC of constraint height h encoding message m into cover with costs c.

We might hope that the distortion realized by STCs is close to the bound, i.e.

$$STC^h(c, m) \approx PLS(c, |m|). \tag{4}$$

Indeed, for some m the STC distortion can even be lower (for example a cover which already happens to contain message m). But we typically witness *coding loss*: in the case of PLS, the STC distortion exceeds the bound; in the case of DLS, the STC can only embed a message shorter than m. Whilst we commonly see steganographic coding loss expressed as the *entropy coding loss* l_h [12, 14, 19]

$$l_h = 1 - \frac{m}{DLS(x, STC^h(x, m))}, \tag{5}$$

for the purpose of indicating loss on a distortion level, we also will use a slight variant, *distortion coding loss* l_d:

$$l_d = \frac{STC^h(x, m)}{PLS(x, m)} - 1. \tag{6}$$

In practice, the coding loss is not only dictated by the constraint height h but is also a product the embedding rate ($\alpha = m/n$) and the distortion function D. We illustrate this in Fig. 1, showing coding loss (here measured by l_d) for 25 images from BOSSBase [2] ($n = 262144$), with payload rates $\alpha = \{0.1, 0.4\}$, constraint heights $h \in \{7, 10\}$, and commonly-used spatial-domain steganographic distortion functions: two that derive costs from filters WOW [21], S-UNIWARD [23], and one based on a model MiPOD [34]. It is evident that h, α, and D, all affect coding loss, which also depends on the cover.

The existence of coding loss is well understood, and some literature has compared STC embedding with simulated embedding [21, 23, 29, 31, 32, 46]. However, most researchers continue to use simulated stego objects: as well as neglecting the coding loss that must occur in practice, this could advantage one distortion function more than another.

3 PROPOSED METHOD

Our method aims to *increase* the entropy used in simulated embedding, to reflect the coding loss of STCs. We estimate the coding loss by embedding n_m random messages in each of n_c randomly-chosen covers.

Rather than estimating l_h directly, we estimate $\tilde{l} = 1/(1 - l_h)$ as the average over covers and messages. For a given distortion function, constraint height h, and embedding rate α, let the n_c covers be $\{x_1, \ldots, x_{n_c}\}$. Over a set of n_m messages we get the average STC cost as:

$$\frac{1}{n_m} \sum_{j=1}^{n_m} STC^h(x, m_j), \tag{7}$$

where $|m_j| = \alpha|x|$. Using the average STC cost as a distortion budget, we can find the covers maximum entropy:

$$DLS\left(x, \frac{1}{n_m} \sum_{j=1}^{n_m} STC^h(x, m_j)\right). \tag{8}$$

Finally, averaging over the set of n_c covers, we can express $\tilde{l}$ as the average per cover ratio between maximum entropy and original message length, normalised per cover length:

$$\tilde{l} = \frac{1}{n_c} \sum_{i=1}^{n_c} \frac{DLS\left(x_i, \frac{1}{n_m} \sum_{j=1}^{n_m} STC^h(x_i, m_j)\right)}{\alpha|x_i|}. \tag{9}$$

Having estimated $\tilde{l}$, we can use it to correct the entropy in simulated embedding, solving (2) for $m = \tilde{l}\alpha|x_i|$ rather than $m = \alpha|x_i|$. Our expectation is that

$$STC^h(c, m) \approx PLS(c, \tilde{l}|m|), \tag{10}$$

for any cover c from the same source, and any message m. Hence, $\tilde{l}|m|$ will be called the *corrected entropy*, and objects simulated using it will be termed *corrected stego objects*.

4 EXPERIMENTAL RESULTS

We can evaluate whether our method is successful by testing whether equation (10) holds more closely than (4). Because, over a set of images and messages, distortion will vary across orders of magnitude, average or RMS error is an inappropriate benchmark dominated by a few stego objects with large distortion. Instead, we use a Root

Mean Square Relative Error (RMSRE),

$$RMSRE = \sqrt{\frac{1}{N} \sum_i \left(\frac{\tilde{d}_i - d_i}{\tilde{d}_i}\right)^2}, \qquad (11)$$

where d_i is the true STC cost, $\tilde{d}_i$ is the simulated (uncorrected or corrected) cost, and N is the number of samples.

This will be measured across a set of covers, either the 10,000 standard steganography cover images BOSSBase [2] (all images the same size, $n = 262144$), or 10,000 images from the ALASKA [7] data set (image sizes vary from $n = 262144$ to $n = 1048576$).

4.1 Determining n_m and n_c

Table 1: Variance σ^2 of the corrected entropy $\tilde{l}\alpha|x_i|$, for values of n_m, and n_c, using S-UNIWARD, $h = 7$, $\alpha = 0.1$.

$n_m \setminus n_c$	1	2	4	8	16	32
1	45 270	21 439	7716	6655	2322	1851
2	44 543	22 329	10 653	5506	3213	1466
4	46 152	21 213	10 217	6256	2814	1451
8	34 695	25 433	11 884	5162	1891	1203
16	40 921	20 078	13 772	5796	2916	1419

We first explore the parameters n_m and n_c. To determine which has more effect on the stability of the results, we measured the variance of $\tilde{l}\alpha|\vec{c}|$ for $\alpha = 0.1$. We tested $n_m \in \{1, 2, 4, 8, 16\}$, and $n_c \in \{1, 2, 4, 8, 16, 32\}$ against an STC with $h = 7$, repeating 100 times the selection of covers from BOSSBase. In Table 1 we see that n_m has an almost negligible effect, so that we might as well set $n_m = 1$. On the other hand, the variance of the corrected entropy is approximately inversely proportional to n_c. This would be predicted by standard sampling theory [33, §7.3].

Having settled on $n_m = 1$, we can determine the lowest suitable value of n_c, as the user will want to perform the smallest sufficient number of STC embeddings. We expected that very small values of n_c would incorrectly estimate the entropy correction, or be subject to large deviations depending on the covers that were sampled.

Using S-UNIWARD, the BOSSBase cover-set, and an STC ($h = 7$), we found the average STC distortion (true cost) for 16 randomly generated messages for each of the embedding rates $\alpha = \{0.1, 0.2, \ldots, 0.5\}$. Using (2) we found the uncorrected simulated distortion for the original payload rates. For each value of $n_c \in \{1, 2, 4, 8, 16\}$ we found the cover-subset coding loss factor approximation using (9) for $n_m = 1$, then the per-cover corrected simulated distortion using (10), calculating the RMSRE for each combination of parameters. This experiment was performed 100 times for different random selections of n_c covers.

In Table 2 we show the RMSRE (above), minimum and maximum (below) RMSRE over all simulations for various values of n_c. We achieve a RMSRE that is at least an order of magnitude lower than uncorrected PLS embedding. Higher values of n_c have only a small effect on the average outcome, but to our surprise, it is also the case that even $n_c = 1$ – a single sample of cover and message – did not cause large deviations in the accuracy of the simulated distortion. Even in the worst case (out of 100) selection of a single cover, in most cases, an order of magnitude reduction in RMSRE is observed.

Table 2: RMSRE of per cover distortion values (simulated versus true cost) for uncorrected, and corrected, S-UNIWARD, $h = 7$.

α	Uncorrected	$n_c = 1$	$n_c = 2$	$n_c = 4$	$n_c = 8$	$n_c = 16$
			RMSRE $\times 10^{-2}$			
0.1	16.89	1.36	1.20	1.06	1.02	1.00
		(0.95, 2.53)	(0.95, 2.32)	(0.95, 1.48)	(0.95, 1.24)	(0.95, 1.20)
0.2	18.28	0.97	0.83	0.78	0.73	0.72
		(0.69, 1.68)	(0.69, 1.64)	(0.69, 1.21)	(0.69, 0.91)	(0.69, 0.85)
0.3	14.25	0.55	0.51	0.44	0.42	0.41
		(0.40, 0.98)	(0.40, 1.00)	(0.40, 0.70)	(0.40, 0.56)	(0.40, 0.46)
0.4	14.13	0.18	0.14	0.12	0.11	0.10
		(0.09, 0.55)	(0.09, 0.32)	(0.09, 0.25)	(0.09, 0.21)	(0.09, 0.14)
0.5	7.06	0.42	0.38	0.34	0.32	0.32
		(0.03, 0.81)	(0.03, 0.66)	(0.03, 0.50)	(0.03, 0.42)	(0.04, 0.36)

4.2 Cover distortion accuracy

Using the values $n_m = 1$ and $n_c = 1$, in Fig. 2 and Fig. 3 we show the per-cover-message: uncorrected, corrected (averaged over 100 simulations), and corrected (worst-case maximum RMRSE simulation) distortion values plotted against the true per-cover-message average distortion values using $h = 7$, and $h = 10$ respectively. It is clear that our method predicts accurate distortion values which negate the coding loss in the estimation. We emphasise that for each combination of distortion function, constraint height, and embedding rate we performed only a single true STC embedding to estimate the corrected entropy $\tilde{l}\alpha|\vec{c}|$, which was then used to simulate our adjusted stego objects: a small overhead to find an accurate estimation of the true cost.

To summarise the results, in Fig. 4 we compare our corrected distortion method to the uncorrected PLS method, using 10,000 covers from BOSSBase, and again with 10,000 more heterogeneous images from ALASKA. We chose ALASKA as our second cover set to show that our method can be used on different sized covers. For a set of distortion functions, constraint heights, and embedding rates we show the deviation from the true cost using RMSRE. Our method reduces, typically by an order of magnitude or more, the relative error in distortion compared with a plain PLS method, across all combinations of distortion functions, constraint heights, embedding rates and cover sets.

4.3 Detectability

So far we have only tested whether our correction method achieves the STC-realized *distortion*; now we test briefly whether it does the same for *detectability*. We benchmarked our corrected method ($\tilde{l}$ values as per Table 3), the uncorrected PLS method, and real STC stego objects, embedding in 10,000 BOSSBase covers at embedding rates $\alpha = \{0.1, 0.3, 0.5\}$. For STC embedding we used $h = \{7, 10\}$, and for the correction we used a single STC realization: $n_c = 1$ and $n_m = 1$. For the detector, we use the Spatial Rich Model with Single Quantization ($SRMQ1$) [16] feature sets with 12,753 dimensions, with the Low-complexity Linear Classifier ($LCLSMR$) [8], using half the cover and stego features for training, and the rest for testing and repeating under 10 splits of training and testing data. We measure
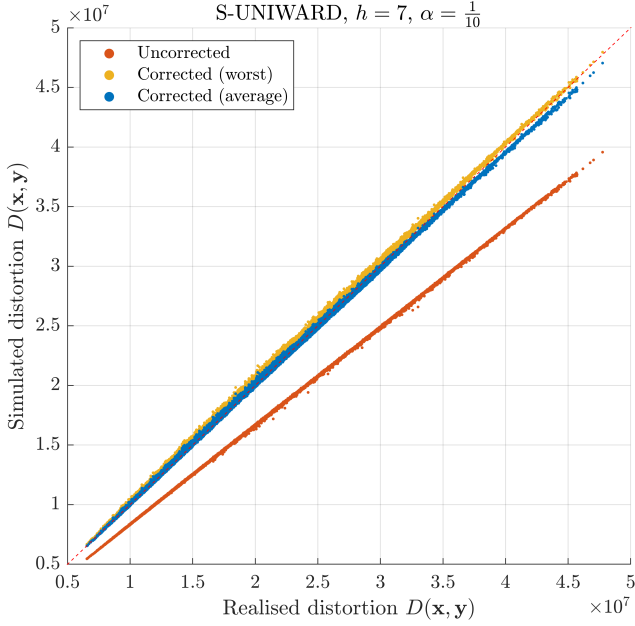
Figure 2: Corrected, and uncorrected simulated distortion versus realised distortion values. S-UNIWARD distortion function, STC $h = 7$, $n_c = 1$, $n_m = 1$.
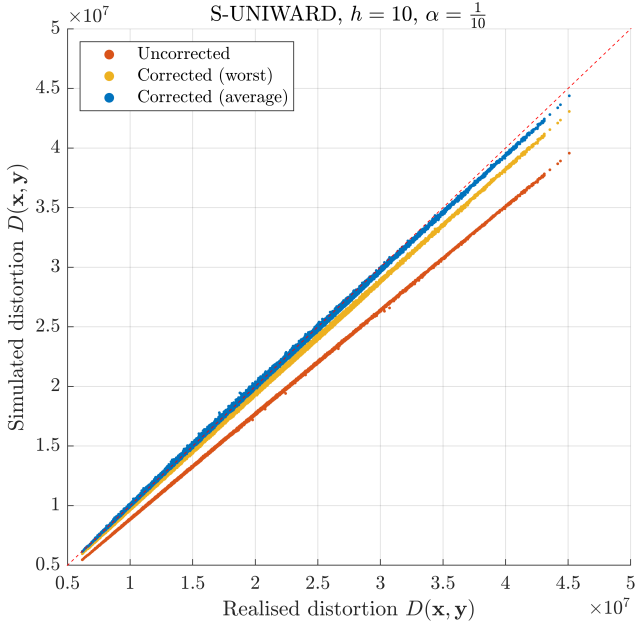

Figure 3: Corrected, and uncorrected simulated distortion versus realised distortion values. S-UNIWARD distortion function, STC $h = 10$, $n_c = 1$, $n_m = 1$.

detector error by the minimal total probability of error under equal Bayesian priors, P_E.

Table 3: Values of $\tilde{l}$ used for Section 4.3, $n_c = 1$, $n_m = 1$.

Cover set	h	α	S-UNIWARD	WOW	MiPOD
BOSSBase	7	0.1	1.144	1.131	1.147
		0.3	1.112	1.101	1.110
		0.5	1.071	1.067	1.074
	10	0.1	1.101	1.085	1.090
		0.3	1.070	1.066	1.075
		0.5	1.044	1.041	1.045
ALASKA	7	0.1	1.159	1.148	1.153
		0.3	1.112	1.111	1.104
		0.5	1.074	1.075	1.070
	10	0.1	1.106	1.104	1.110
		0.3	1.075	1.068	1.073
		0.5	1.047	1.045	1.045

Table 4: Average detection error $\tilde{P}_E$, as well as standard deviation amongst folds, for corrected ($n_c = 1$, $n_m = 1$), and uncorrected simulated, compared to STC embedding, S-UNIWARD. All numbers are expressed as percentage points of error.

Method	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$
Uncorrected PLS	39.82% (± 0.08)	23.34% (± 0.11)	12.49% (± 0.10)
STC $h = 7$	38.64% (± 0.07)	21.37% (± 0.06)	11.04% (± 0.09)
Corrected $h = 7$	38.55% (± 0.12)	21.30% (± 0.12)	11.03% (± 0.08)
STC $h = 10$	38.99% (± 0.08)	21.99% (± 0.08)	11.50% (± 0.13)
Corrected $h = 10$	38.96% (± 0.08)	21.85% (± 0.11)	11.62% (± 0.10)

In Table 4 we show the error rates, as well as the standard deviation amongst the 10 training/testing splits. As expected, under the larger constraint height $h = 10$ the detectability of the STC is slightly closer to a pure PLS simulation, but a significant gap still remains: the simulation was under-distorting the stego objects compared with genuine STC-embedded stego objects. Although the absolute difference between error rates of 23.34% and 21.99% (taking, for example, the case $\alpha = 0.3$) does not seem large, it is statistically significant with $p < 10^{-6}$ if we treat the folds as a random sample. Moreover, it is not uncommon for detection differences of around 1% to be treated as significant when incrementally-better detectors are published [6].

But when we use our proposed method to increase the PLS entropy to match the STC distortion, PLS embedding gives an error rate of 21.85% that is practically identical to that of real stego objects, and the difference is either not- or marginally-significant ($p > 0.05$ or $p > 0.01$, depending on α and h). Thus our method is simulating stego objects with detectability that matches that of STC-embedded objects, as intended.

5 CONCLUDING REMARKS

We have shown that our method can create simulated stego objects which are closer, in terms of distortion and detectability, to actual stego objects than those created by traditional PLS embedding. Our method can efficiently sample the coding loss of real stego objects and derive an adjusted estimate of the entropy which must be introduced to the simulated stego object, to match the real world stego objects. Even as few as one real stego object is sufficient
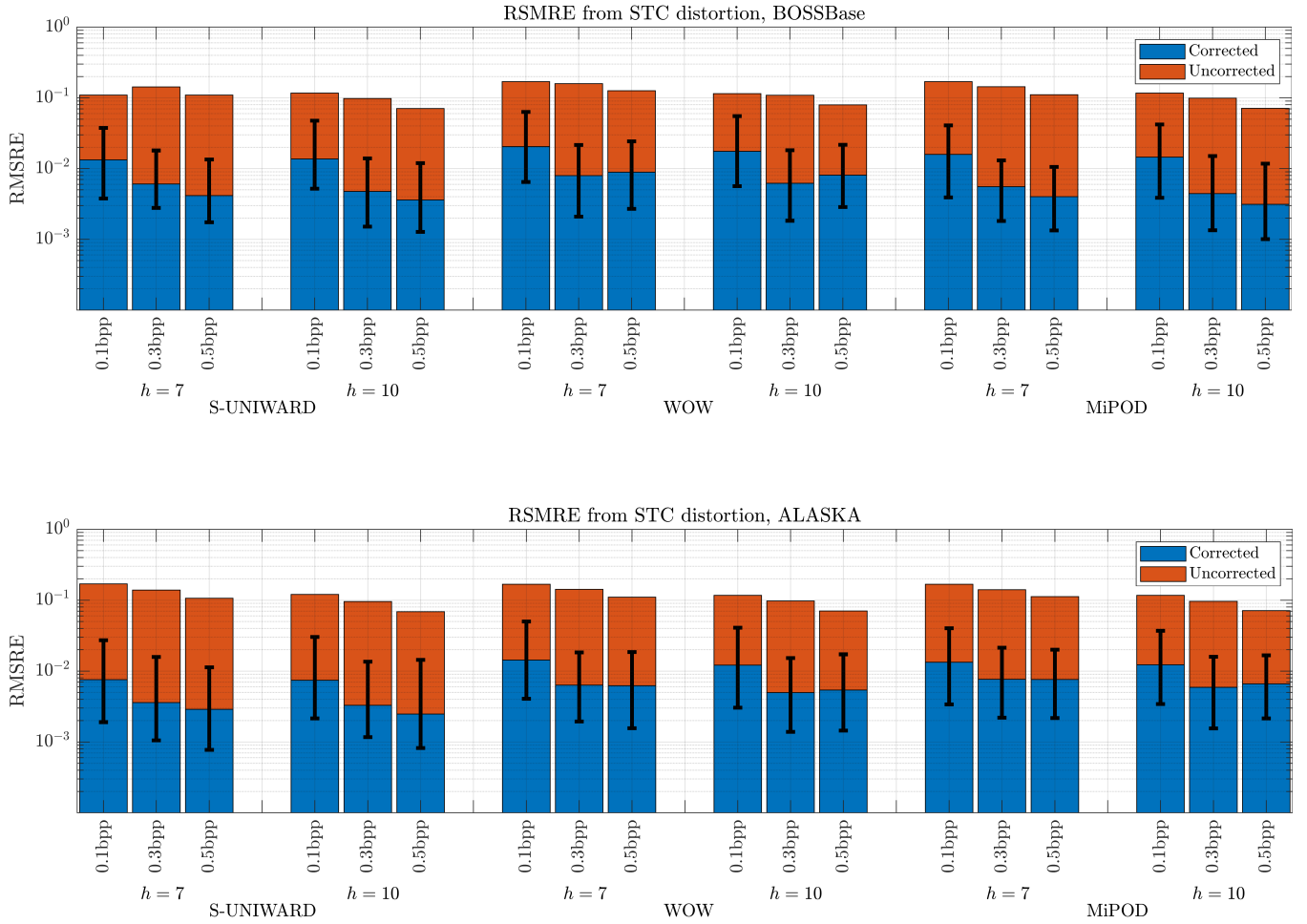
Figure 4: Comparison of uncorrected, and corrected RMSRE from achieved distortion, for BOSSBase (top) and ALASKA (bottom), $n_c = 1$, $n_m = 1$. Error bars indicate minimum and maximum RMSRE values obtained over all simulations.

to find an accurate entropy correction. The result is a low-cost sampling technique which significantly minimises the divergence between simulated and real-world stego objects, allowing the benefits in terms of speed and simplicity of simulating, with a closer representation to actual coding.

Our method also allows the researcher to simulate how their distortion function may behave under different conditions (constraint heights, cover sources), not just embedding rates. Further work may apply similar corrections for aspects not covered in this paper; non-additive distortion functions, JPEG steganography, and q-ary embedding. A method of analysis not included in this paper, but would yield interesting results, is the divergence of the per-element change probabilities to real embedding.

ACKNOWLEDGMENTS

REFERENCES

[1] P. Bas. 2016. Steganography via Cover-Source Switching. In *2016 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, Abu Dhabi, United Arab Emirates, 1–6. https://doi.org/10.1109/WIFS.2016.7823905

[2] P. Bas, T. Filler, and T. Pevný. 2011. 'Break Our Steganographic System': The Ins and Outs of Organizing BOSS. *Information Hiding* (2011), 59–70. https://doi.org/10.1007/978-3-642-24178-9_5

[3] B. Chen, W. Luo, and P. Zheng. 2019. Enhancing Steganography via Stego Post-Processing by Reducing Image Residual Difference. In *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '19)*. ACM, New York, NY, USA, 63–68. https://doi.org/10.1145/3335203.3335716

[4] K. Chen, W. Zhang, H. Zhou, N. Yu, and G. Feng. 2016. Defining Cost Functions for Adaptive JPEG Steganography at the Microscale. In *2016 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 1–6. https://doi.org/10.1109/WIFS.2016.7823900

[5] K. Chen, H. Zhou, W. Zhou, W. Zhang, and N. Yu. 2019. Defining Cost Functions for Adaptive JPEG Steganography at the Microscale. *IEEE Transactions on Information Forensics and Security* 14, 4 (April 2019), 1052–1066. https://doi.org/10.1109/TIFS.2018.2869353

[6] M. Chen, V. Sedighi, M. Boroumand, and J. Fridrich. 2017. JPEG-Phase-Aware Convolutional Neural Network for Steganalysis of JPEG Images. In *Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '17)*. Association for Computing Machinery, New York, NY, USA, 75–84. https://doi.org/10.1145/3082031.3083248

[7] R. Cogranne, Q. Giboulot, and P. Bas. 2019. The ALASKA Steganalysis Challenge: A First Step Towards Steganalysis. In *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '19)*. ACM, New York, NY, USA, 125–137. https://doi.org/10.1145/3335203.3335726

[8] R. Cogranne, V. Sedighi, J. Fridrich, and T. Pevný. 2015. Is Ensemble Classifier Needed for Steganalysis in High-Dimensional Feature Spaces?. In *2015 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 1–6. https://doi.org/10.1109/WIFS.2015.7368597

[9] T. Denemark and J. Fridrich. 2015. Side-Informed Steganography with Additive Distortion. In *2015 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 1–6. https://doi.org/10.1109/WIFS.2015.7368589

[10] T. Denemark and J. Fridrich. 2017. Steganography With Multiple JPEG Images of the Same Scene. *IEEE Transactions on Information Forensics and Security* 12, 10 (Oct 2017), 2308–2319. https://doi.org/10.1109/TIFS.2017.2705625

[11] B. Feng, W. Lu, and W. Sun. 2015. Secure Binary Image Steganography Based on Minimizing the Distortion on the Texture. *IEEE Transactions on Information Forensics and Security* 10, 2 (Feb 2015), 243–255. https://doi.org/10.1109/TIFS.2014.2368364

[12] T. Filler and J. Fridrich. 2010. Gibbs Construction in Steganography. *IEEE Transactions on Information Forensics and Security* 5, 4 (2010), 705–720.

[13] T. Filler and J. Fridrich. 2011. Design of Adaptive Steganographic Schemes for Digital Images. In *Media Watermarking, Security, and Forensics III*, Vol. 7880. International Society for Optics and Photonics, 78800F.

[14] T. Filler, J. Judas, and J. Fridrich. 2011. Minimizing Additive Distortion in Steganography Using Syndrome-Trellis Codes. *IEEE Transactions on Information Forensics and Security* 6, 3 (2011), 920–935. https://doi.org/10.1109/tifs.2011.2134094

[15] J. Fridrich. 2013. Effect of Cover Quantization on Steganographic Fisher Information. *IEEE Transactions on Information Forensics and Security* 8, 2 (Feb 2013), 361–373. https://doi.org/10.1109/TIFS.2012.2235832

[16] J. Fridrich and J. Kodovsky. 2012. Rich Models for Steganalysis of Digital Images. *IEEE Transactions on Information Forensics and Security* 7, 3 (June 2012), 868–882. https://doi.org/10.1109/TIFS.2012.2190402

[17] M. Goljan, J. Fridrich, and R. Cogranne. 2014. Rich Model for Steganalysis of Color Images. In *2014 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 185–190. https://doi.org/10.1109/WIFS.2014.7084325

[18] L. Guo, J. Ni, and Y. Q. Shi. 2012. An Efficient JPEG Steganographic Scheme using Uniform Embedding. In *2012 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 169–174. https://doi.org/10.1109/WIFS.2012.6412644

[19] L. Guo, J. Ni, and Y. Q. Shi. 2014. Uniform Embedding for Efficient JPEG Steganography. *IEEE Transactions on Information Forensics and Security* 9, 5 (May 2014), 814–825. https://doi.org/10.1109/TIFS.2014.2312817

[20] L. Guo, J. Ni, W. Su, C. Tang, and Y. Shi. 2015. Using Statistical Image Model for JPEG Steganography: Uniform Embedding Revisited. *IEEE Transactions on Information Forensics and Security* 10, 12 (Dec 2015), 2669–2680. https://doi.org/10.1109/TIFS.2015.2473815

[21] V. Holub and J. Fridrich. 2012. Designing Steganographic Distortion using Directional Filters. In *2012 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 234–239. https://doi.org/10.1109/WIFS.2012.6412655

[22] V. Holub and J. Fridrich. 2013. Random Projections of Residuals for Digital Image Steganalysis. *IEEE Transactions on Information Forensics and Security* 8, 12 (Dec 2013), 1996–2006. https://doi.org/10.1109/TIFS.2013.2286682

[23] V. Holub, J. Fridrich, and T. Denemark. 2014. Universal Distortion Function for Steganography in an Arbitrary Domain. *EURASIP Journal on Information Security* 2014, 1 (2014). https://doi.org/10.1186/1687-417x-2014-1

[24] D. Hu, H. Xu, Z. Ma, S. Zheng, and B. Li. 2018. A Spatial Image Steganography Method Based on Nonnegative Matrix Factorization. *IEEE Signal Processing Letters* 25, 9 (Sep. 2018), 1364–1368. https://doi.org/10.1109/LSP.2018.2856630

[25] X. Hu, J. Ni, and Y. Shi. 2018. Efficient JPEG Steganography Using Domain Transformation of Embedding Entropy. *IEEE Signal Processing Letters* 25, 6 (June 2018), 773–777. https://doi.org/10.1109/LSP.2018.2818674

[26] S. Islam, M. R. Modi, and P. Gupta. 2014. Edge-Based Image Steganography. *EURASIP Journal on Information Security* 2014, 1 (27 Apr 2014), 8. https://doi.org/10.1186/1687-417X-2014-8

[27] J. Kodovsky and J. Fridrich. 2014. Effect of Image Downsampling on Steganographic Security. *IEEE Transactions on Information Forensics and Security* 9, 5 (May 2014), 752–762. https://doi.org/10.1109/TIFS.2014.2309054

[28] B. Li, S. Tan, M. Wang, and J. Huang. 2014. Investigation on Cost Assignment in Spatial Image Steganography. *IEEE Transactions on Information Forensics and Security* 9, 8 (Aug 2014), 1264–1277. https://doi.org/10.1109/TIFS.2014.2326954

[29] B. Li, M. Wang, X. Li, S. Tan, and J. Huang. 2015. A Strategy of Clustering Modification Directions in Spatial Image Steganography. *IEEE Transactions on Information Forensics and Security* 10, 9 (Sep. 2015), 1905–1917. https://doi.org/10.1109/TIFS.2015.2434600

[30] W. Li, W. Zhang, K. Chen, W. Zhou, and N. Yu. 2018. Defining Joint Distortion for JPEG Steganography. In *Proceedings of the 6th ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '18)*. ACM, New York, NY, USA, 5–16. https://doi.org/10.1145/3206004.3206008

[31] T Pevný, T. Filler, and P Bas. 2010. Using High-Dimensional Image Models to Perform Highly Undetectable Steganography. In *Information Hiding*, Rainer Böhme, Philip W. L. Fong, and Reihaneh Safavi-Naini (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 161–177.

[32] T. Pevný and A. Ker. 2018. Exploring Non-Additive Distortion in Steganography. In *Proceedings of the 6th ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '18)*. ACM, New York, NY, USA, 109–114. https://doi.org/10.1145/3206004.3206015

[33] J. A. Rice. 1995. *Mathematical Statistics and Data Analysis* (2nd ed. ed.). Duxbury Press, Belmont, Calif.

[34] V. Sedighi, R. Cogranne, and J. Fridrich. 2016. Content-adaptive Steganography by Minimizing Statistical Detectability. *IEEE Transactions on Information Forensics and Security* 11, 2 (Feb 2016), 221–234. https://doi.org/10.1109/TIFS.2015.2486744

[35] V. Sedighi and J. Fridrich. 2015. Effect of Imprecise Knowledge of the Selection Channel on Steganalysis. In *Proceedings of the 3rd ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '15)*. ACM, New York, NY, USA, 33–42. https://doi.org/10.1145/2756601.2756621

[36] T. Taburet, P. Bas, J. Fridrich, and W. Sawaya. 2019. Computing Dependencies Between DCT Coefficients for Natural Steganography in JPEG Domain. In *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '19)*. ACM, New York, NY, USA, 57–62. https://doi.org/10.1145/3335203.3335715

[37] W. Tang, B. Li, W. Luo, and J. Huang. 2016. Clustering Steganographic Modification Directions for Color Components. *IEEE Signal Processing Letters* 23, 2 (Feb 2016), 197–201. https://doi.org/10.1109/LSP.2015.2504583

[38] W. Tang, B. Li, S. Tan, M. Barni, and J. Huang. 2019. CNN-Based Adversarial Embedding for Image Steganography. *IEEE Transactions on Information Forensics and Security* 14, 8 (Aug 2019), 2074–2087. https://doi.org/10.1109/TIFS.2019.2891237

[39] W. Tang, H. Li, W. Luo, and J. Huang. 2016. Adaptive Steganalysis Based on Embedding Probabilities of Pixels. *IEEE Transactions on Information Forensics and Security* 11, 4 (April 2016), 734–745. https://doi.org/10.1109/TIFS.2015.2507159

[40] W. Tang, S. Tan, B. Li, and J. Huang. 2017. Automatic Steganographic Distortion Learning Using a Generative Adversarial Network. *IEEE Signal Processing Letters* 24, 10 (Oct 2017), 1547–1551. https://doi.org/10.1109/LSP.2017.2745572

[41] Z. Wang, X. Zhang, and Z. Yin. 2018. Joint Cover-Selection and Payload-Allocation by Steganographic Distortion Optimization. *IEEE Signal Processing Letters* 25, 10 (Oct 2018), 1530–1534. https://doi.org/10.1109/LSP.2018.2865888

[42] J. Yang, D. Ruan, X. Kang, and Y. Shi. 2019. Towards Automatic Embedding Cost Learning for JPEG Steganography. In *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '19)*. ACM, New York, NY, USA, 37–46. https://doi.org/10.1145/3335203.3335713

[43] J. Ye, J. Ni, and Y. Yi. 2017. Deep Learning Hierarchical Representations for Image Steganalysis. *IEEE Transactions on Information Forensics and Security* 12, 11 (Nov 2017), 2545–2557. https://doi.org/10.1109/TIFS.2017.2710946

[44] W. Zhang, J. Liu, X. Wang, and N. Yu. 2010. Generalization and Analysis of the Paper Folding Method for Steganography. *IEEE Transactions on Information Forensics and Security* 5, 4 (Dec 2010), 694–704. https://doi.org/10.1109/TIFS.2010.2065804

[45] Z. Zhao, Q. Guan, H. Zhang, and X. Zhao. 2019. Improving the Robustness of Adaptive Steganographic Algorithms Based on Transport Channel Matching. *IEEE Transactions on Information Forensics and Security* 14, 7 (July 2019), 1843–1856. https://doi.org/10.1109/TIFS.2018.2885438

[46] Z. Zhao, Q. Guan, and X. Zhao. 2016. Constructing Near-Optimal Double-Layered Syndrome-Trellis Codes for Spatial Steganography. In *Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '16)*. ACM, New York, NY, USA, 139–148. https://doi.org/10.1145/2909827.2930802

[47] W. Zhou, W. Zhang, and N. Yu. 2017. A New Rule for Cost Reassignment in Adaptive Steganography. *IEEE Transactions on Information Forensics and Security* 12, 11 (Nov 2017), 2654–2667. https://doi.org/10.1109/TIFS.2017.2718480