

MAXIMIZING ENTROPY OF IMAGE MODELS FOR 2-D CONSTRAINED CODING

S. Forchhammer, M. Danieli, N. Burini, M. Zamarin, and A. Ukhanova

DTU Fotonik, Technical University of Denmark,
B. 343, DK-2800 Lyngby, DENMARK, sofo@fotonik.dtu.dk

ABSTRACT

This paper considers estimating and maximizing the entropy of two-dimensional (2-D) fields with application to 2-D constrained coding. We consider Markov random fields (MRF), which have a non-causal description, and the special case of Pickard random fields (PRF). The PRF are 2-D causal finite context models, which define stationary probability distributions on finite rectangles and thus allow for calculation of the entropy. We consider two binary constraints and revisit the hard square constraint given by forbidding neighboring 1s and provide novel results for the constraint that no uniform 2×2 squares contains all 0s or all 1s. The maximum values of the entropy for the constraints are estimated and binary PRF satisfying the constraint are characterized and optimized w.r.t. the entropy. The maximum binary PRF entropy is 0.839 bits/symbol for the no uniform squares constraint. The entropy of the Markov random field defined by the 2-D constraint is estimated to be (upper bounded by) 0.8570 bits/symbol using the iterative technique of Belief Propagation on 2×2 finite lattices. Based on combinatorial bounding techniques the maximum entropy for the constraint was determined to be 0.848.

1. INTRODUCTION

We consider problems related to determining the entropy of image models with focus on constrained images. Assume we want to store data in the form of bits in a 2-D grid (we can for simplicity start by assuming that the grid is an $M \times N$ rectangle), subject to a constraint related to the physical storage media. The hard-square constraint defined as a binary image with no two 4-neighboring ones has been studied as a simple basic 2-D constraint [1]. We shall revisit this constraint as well as analyzing another binary constraint.

Example 1. No 2×2 uniform blocks. Consider the 2-D constraint, that no 2×2 block contains all 0s or all 1s, i.e. no matter how we select a 2×2 square within the rectangular grid, it will not contain either all zeros or all ones.

It is interesting to compute the combinatorial entropy, or capacity, for this constraint

$$C = \lim_{M, N \rightarrow \infty} F(M, N)/(MN), \quad (1)$$

where $F(M, N)$ is the number of admissible configurations in our $M \times N$ rectangle subject to the constraint [2,

p. 122]. The quantity tells the *capacity* in terms of how much information we can store per grid symbol. While computing the combinatorial entropy can be a daunting task, because it entails counting the number of configurations and the problem scales exponentially with M and N , there is another approach that can be taken, namely a probabilistic one, which hopefully also can provide guidance towards 2-D coding techniques. In order to evaluate these models, the problems of determining the entropy of fields satisfying the constraint and maximizing this entropy are addressed.

The maximum entropy gives the capacity of the 2-D constrained field, which can be used for 2-D constrained coding for storage applications. The maximum entropy of the PRF provides a causal coding model of 2-D bit-stuffing coding, but provides a slightly lower entropy than the capacity of the constraint. A maximum entropy PRF may also be seen as a (causal) approximation of an MRF.

We will focus on the class of binary Pickard random fields and determine PRF for the two constraints above and optimize the entropy. These results are compared with other techniques for estimating the (maximum) entropy including simulation of an MRF. The comparisons can shed some light on the performance of each of the techniques.

2. PROBABILISTIC MODELS

In general we can aim at finding a probability distribution for the symbols on our grid. Let X denote a random variable describing the field. One goal is to maximize the entropy $H(X)$, which may approximate C for large rectangles. The entropy can be expressed by means of the chain rule:

$$H(X) = \sum_{j=1}^{MN} H(x_j | x_1, x_2, \dots, x_{j-1}), \quad (2)$$

where we assign a one-dimensional index j to each symbol by reading the elements in the grid e.g. in a row-by-row fashion. This does not comply well with the non-causal interactions in images as e.g. expressed by Markov random fields. We shall return to these at the end of the section but start with the special case of Pickard Random Fields, which are causal and in some cases can provide a model for a 2-D constraint.

2.1. Pickard random fields

There exists one class of nontrivial stationary Markov fields investigated by Pickard [3][4]. The so called Pickard random fields (PRFs) have some desirable properties as causality, allowing to simulate a random field sequentially, and the property that the symbols in rows and columns are outcomes of irreducible Markov chains over a finite alphabet.

A PRF is stationary and its joint probability can be determined by a measure on a 2×2 pixel-square that must satisfy several conditions. We shall focus on the unilateral PRF [4] on finite alphabets, especially binary alphabets. Following this, the probability of each individual column or row is described by a Markov chain with transition probabilities deduced from the measure on the square. The Markov chain for rows and columns may be different (rows from left to right and columns bottom up; the Markov chain may be different when reversed). To define the process of constructing PRF, let us first discuss the independence conditions.

2.1.1. Independence conditions

To simplify notation, we consider a 2×2 square and we introduce the following notation:

$$\begin{bmatrix} X_{ij} & X_{i,j+1} \\ X_{i+1,j} & X_{i+1,j+1} \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

where X_{ij} are random variables on the 2-D grid.

A state in the two-row Markov chain is defined by a pair of symbols, and the probability distribution of this pair, $X_{i,j}, X_{i+1,j}$, is found from the Markov chain describing a single column, where i and $i+1$ denote the upper and the lower row, respectively. The conditional probability of the next symbol in the upper row, $P(X_{i,j+1}|X_{i,j})$, also follows from the single-row Markov chain, but with the important assumption that it is independent of $X_{i+1,j}$, $P(X_{i,j+1}|X_{i,j}, X_{i+1,j}) = P(X_{i,j+1}|X_{i,j})$, which implies

$$P(ABC) = P(C)P(A|C)P(B|A). \quad (3)$$

With this assumption, it is possible to continue to find the conditional probabilities of the following symbols in the upper row, and the row simply becomes a Markov chain given by the distribution (AB) . There are two ways of assuring that the lower row is described by the same Markov chain. One is the condition $P(X_{i+1,j+1}|X_{i,j}, X_{i+1,j}) = P(X_{i+1,j+1}|X_{i+1,j})$, which is symmetric to the first condition and lets the lower row be continued forward in the same way. The alternative is $P(X_{i+1,j}|X_{i,j+1}, X_{i+1,j+1}) = P(X_{i+1,j}|X_{i+1,j+1})$, which allows the lower row to be continued in the reverse direction using the reversed Markov chain. In this case

$$P(BCD) = P(C)P(D|C)P(B|D). \quad (4)$$

We will focus on the second condition (4), but PRFs based on the first condition may be treated in the same way.

2.1.2. PRF for given Markov chains

As in [2], we start with the two Markov chains with matrices R and S for rows and columns, respectively. The Markov chains and the independence conditions (3-4) lead to the distributions (ABC) and (BCD) as described above. For consistency with the joint measure $(ABCD)$ the two distributions must have identical marginals on the pair BC . Since the transition from C to B can be achieved in two different ways (RS or SR) and since the transition probabilities from C to B must be unique, the products RS and SR must be equal,

$$RS = SR. \quad (5)$$

For Markov chains satisfying this, the last requirement is to determine $P(AD|bc)$ consistent with $P(A|bc)$ and $(D|bc)$, i.e. determine the joint conditional probability consistent with the two marginal conditional probabilities. For given distributions (ABC) and (BCD) , this gives

$$\sum_d P(A, D = d|bc) = P(A|bc) \quad (6)$$

$$\sum_a P(A = a, D|bc) = P(D|bc). \quad (7)$$

Solving these for maximum entropy $H(AD|bc)$, for each bc , leads to the maximum entropy PRF, $H(D|ABC)$, for the given Markov chains S and R , as $H(AD|BC) = H(A|BC) + H(D|ABC)$ and $H(A|BC)$ is fixed when the independence requirement is satisfied for given Markov chains. $P(D|ABC)$ may thereafter be derived from solution to (6), $P(AD|BC)$. Due to the independence property (3), the dominating terms of the entropy (2) may be expressed as

$$H(D|ABC) = - \sum_{abcd} P(abcd) \log P(d|abc). \quad (8)$$

Asymptotically, the entropy of the PRF will converge to $H(D|ABC)$ as the terms of the interior will dominate over those of the Markov chain boundaries [2]. For values of bc for which all combinations are valid, the parameters achieving maximum entropy are simply given by $P(D|ABC) = P(D|BC)$, which may be expressed by $P(d|bc) = P(bcd)/P(bc)$ obtained from (4).

In [5] it was pointed out that the max entropy solution could be obtained by iterative scaling. Here for the binary PRF examples explicit solutions are calculated.

For a binary PRF, the requirement that the two Markov chains have identical stationary probabilities is not only a necessary but also a sufficient condition for the matrices to commute (5). The solutions for $P(AD|bc)$ are also straightforward. Values of bc for which all combinations of ad are valid are treated as above. If one or more combinations of ad for the given bc are forbidden, there are no (additional) free variables. If there is a solution the conditional probabilities are given directly by (5). Below we apply this to two binary 2×2 constraints.

2.1.3. PRF for the hard square constraint

In a binary random field, the hard square constraint forbids 4-neighboring 1s; in other words, 1s must be surrounded by 0s. Such a field can be constructed as a PRF through two Markov chains. Normally the two chains represent rows and columns of the field, but it is possible to use alternative arrangements, for instance rows (left to right) and diagonals (bottom-left to top-right). For this purpose, the 2×2 square introduced above is modified by shifting the bottom row to the left by one place:

$$\begin{bmatrix} - & A & B \\ C & D & - \end{bmatrix}$$

Below we consider the conditions that must be met in order to construct the field correctly.

The two Markov chains must comply with the constraint. In a row, each 1 must be followed by a 0, while there are no restrictions for zeros. Diagonals do not have such restrictions because the constraint only regards 4-neighborhoods. The transition matrices R , for lines, and S , for diagonals, are:

$$R = \begin{bmatrix} p & 1-p \\ 1 & 0 \end{bmatrix} \quad S = \begin{bmatrix} q & 1-q \\ 1-r & r \end{bmatrix}, \quad (9)$$

where p is the probability of a zero following a zero along a row, while q is the probability of a zero following a zero and r the probability of a one following a one in a diagonal.

The stationary distribution of both chains has to be the same. Following the definition of stationary distribution, $p_r^* \times R = p_r^*$, the result for R is

$$p_r^* = \begin{bmatrix} \frac{1}{2-p} & \frac{1-p}{2-p} \end{bmatrix}. \quad (10)$$

Satisfying the requirement that the stationary probabilities must be identical leads to the updated values of R and S :

$$R = \begin{bmatrix} p & 1-p \\ 1 & 0 \end{bmatrix} \quad S = \begin{bmatrix} q & 1-q \\ \frac{1-q}{1-p} & \frac{q-p}{1-p} \end{bmatrix}. \quad (11)$$

As remarked this means that the matrices commute. This leaves two free variables, e.g. p and q . Now r may be expressed by p and q as $r = \frac{q-p}{1-p}$. The probability values in S lead to the restriction in the parameter range that $0 \leq p \leq q \leq 1$.

Through the use of p_r^* , the Markov chains, and the PRF independence, we calculate $P(ABC)$ and $P(BCD)$; configurations that are not allowed (in this case, neighboring ones) must have probability zero, while the probabilities of other configurations must be consistent with the values of $P(BC)$. For all four combinations of bc there is at least one value of $P(ad|bc)$ which is 0, therefore all values of $P(AD|BC)$ are derived directly from the Markov chains R and S and expressed by p and q .

The same concept can be applied to calculate the joint probability $P(ABCD)$. The joint probabilities $P(ABCD)$ and $P(ABC)$ can finally be used to determine $P(D|ABC)$.

For the particular case of the hard square constraint, the resulting probabilities of $P(D|ABC)$ require that $q \geq \frac{1}{1+p}$.

In PRF, entropy is dominated by $H(D|ABC)$. In the horizontal/diagonal scheme, the only ABC configurations that do not coerce D are 000 and 010. Then, according to the definition of conditional entropy (8), the expression of $H(D|ABC)$ is

$$\begin{aligned} H(D|ABC) &= P(A_0B_0C_0)H(D|A_0B_0C_0) \quad (12) \\ &+ P(A_0B_1C_0)H(D|A_0B_1C_0) \\ &= \frac{pq}{2-p}H_b\left(\frac{1-q}{pq}\right) + \frac{q(1-p)}{2-p}H_b\left(\frac{q-p}{q(1-p)}\right), \end{aligned}$$

where H_b is the binary entropy function and short notation as A_0 for $A = 0$ is used.

2.1.4. PRF for no uniform 2×2 squares

For the no uniform 2×2 squares (nus) constraint, the basic elements of the PRF are selected to cover the square and the constraint is enforced by selecting $P(0000) = P(1111) = 0$. Due to the symmetry, we select $S = R$, thus the stationary probabilities are identical and (5) satisfied. Further, due to symmetry of the symbols, we select $P(0|0) = P(1|1) = p$ for S and R , which implies the stationary distribution $P(0) = P(1) = 1/2$.

Due to the symmetry, we only consider $bc = 00$ and $bc = 01$ and obtain the solutions for $bc = 11$ and $bc = 10$ by exchanging 0s and 1s. Given $bc = 00$, $ad = 00$ is forbidden. Thus $P(AD|bc = 00)$ is expressed directly by the marginals, $P(A|bc = 00)$ and $P(D|bc = 00)$. For $P(AD|bc = 01)$, there are no forbidden configurations and we have one free variable, which could be $P(D = 0|A = B = 0, C = 1)$. The choice yielding maximum entropy, $P(D = 0|A = B = 0, C = 1) = 1/2$ may be obtained by selecting the joint probabilities $P(AD|bc = 01)$ as the product of the marginals (6). The probabilities of each of the four combinations of a, b and c , for which $b \neq c$ is $p(1-p)/2$. For $b = c$, there is only a contribution to the entropy if $a \neq b$, which occurs with probability $(1-p)^2/2$ and in this case the conditional probability is $p^2/(1-p)^2$. Thus, for given $p \leq 1/2$, the maximum entropy for the PRF satisfying nus is

$$H(D|ABC) = 2p(1-p) + (1-p)^2H_b\left(\frac{p^2}{(1-p)^2}\right). \quad (13)$$

Restraining from maximizing the entropy, the parameters could be chosen to obtain a bi-lateral PRF [3] and thereby more symmetry. In this case, the entropy is

$$\begin{aligned} H(D|ABC) &= 2p(1-p)H_b\left(\frac{p}{(1-p)}\right) \quad (14) \\ &+ (1-p)^2H_b\left(\frac{p^2}{(1-p)^2}\right). \end{aligned}$$

2.2. Belief Propagation

In this section we will consider MRF and a quite different approach. We present *Belief Propagation* (BP), an algorithm that performs marginalization of probability distributions. We will only give a simple introduction to

the topic. For further exploration we direct the interested reader to the abundant relevant literature, for instance the excellent tutorial by Loeliger [6].

When a joint probability function defined over a set of random variables \mathbf{x} can be factorized as:

$$p(\mathbf{x}) = \frac{1}{Z} \prod_a f_a(\mathbf{x}_a), \quad (15)$$

where Z represents a normalization constant and \mathbf{x}_a is a subset of the set of variables \mathbf{x} , we can apply *Belief Propagation* (also known as *sum-product algorithm*) in order to find the marginal probabilities defined over each single variable, $p_i(x_i)$. The algorithm works over a factor graph, namely a graph whose nodes represent either *variables* or *factors* in the factorization of (15). Connections in the graph only occur between nodes of different type, and an edge appears between factor f_a and variable x_i if \mathbf{x}_a contains x_i .

Belief Propagation consists in exchanging *beliefs* (or probabilistic information) between neighboring nodes. The process can be seen as *message passing* from variable nodes to factor nodes and vice-versa. We will not get into the details of how this process works, we just mention that when a variable x_i sends a message to factor f_a , this message is the probability mass function for x_i (and therefore a set of probabilities with the same number of elements as the cardinality of the alphabet for x_i) obtained by multiplying the probabilities coming from all neighboring factor nodes except for f_a , whereas a message from factor f_a to x_i is the marginal distribution over x_i computed by marginalizing f_a after weighing it according to the beliefs related to all the variables in \mathbf{x}_a but x_i .

The algorithm is guaranteed to converge to the actual marginal distribution if the graph is a tree, meaning that every pair of nodes only has one path connecting them. In this case the algorithm stops when all the variable nodes have received messages from all the neighboring factor nodes, and the marginals are computed as products of these beliefs:

$$p_i(x_i) \propto \prod_{a \in n(i)} m_{a \rightarrow i}(x_i), \quad (16)$$

where $n(i)$ are the factor nodes neighboring variable node i (in other words the factors in which x_i appear as a variable), $m_{a \rightarrow i}$ is the message sent from factor a to variable i , representing the belief the factor node has about the probabilities of x_i , and the proportionality sign signals that the product must be normalized to make p_i a consistent probability mass function.

If the graph has loops the algorithm can still converge after some iterations, but there is no guarantee that the value to which it converges is the actual marginal distribution.

2.2.1. Computing marginals with BP

We can upper bound the conditional entropy by removing conditioning variables in (2).

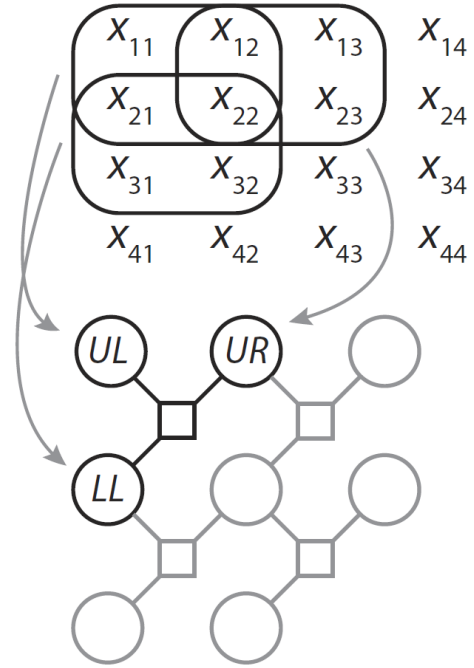


Figure 1. The factor graph structure that models our grid

We want to find an upper bound on the symbol entropy of the field by computing conditional probabilities $H(D|ABC)$ for squares located within a grid with relatively large side N and far from the edges, where the entropy is higher due to the fact that at the edges there are fewer neighboring symbols that exert an influence on the admissible configurations. To do this the joint probability $P(ABCD)$ is needed. An approximation of this value can be found by applying Belief Propagation on a factor graph describing our grid.

In order to take advantage of the simplicity of BP to perform marginalization and compute the conditional entropy $H(D|ABC)$ we first need to model our $N \times N$ grid with a factor graph. At first sight it might seem reasonable to associate a variable in the factor graph to each bit in the square, and then create factors defined over 4 neighboring bits in order to enforce the constraint by assigning null probability to forbidden configurations. If we run the BP algorithm on this graph, what we would get is the marginal probability over the single nodes, hence the bits in the grid, but we already know that 0s and 1s are equally likely (symmetry is necessary to guarantee maximum entropy), and we would fail to capture the probabilistic dependence between bits in a 2×2 square. If we want to compute $P(ABCD)$ we need to define a node for each 2×2 square, as shown in Fig. 1. Consider a 3×3 square. Four 2×2 squares can be identified within it. We decided to connect three of these by means of a factor node, namely the upper left (UL), the upper right (UR) and the lower left (LL) squares. The probability distribution in the factor node will serve to two purposes:

1. To enforce the consistency constraint that the second column of UL coincides with the first column

of UR and that the second row of UL coincides with the first row of UR;

2. To initialize the algorithm by assigning equal probabilities to all admissible configurations.

The second purpose is to make sure that the system has maximum entropy. As the BP algorithm is run over the graph, the final probabilities (that can be read at each iteration by multiplying the beliefs at the factor nodes) will deviate from a fair distribution in order to favor configurations in the squares that leave more freedom when defining neighboring squares. For instance, if we are to choose a UL square and we compare

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \text{ and } \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

we readily see that the square on the right leaves all possibilities open when it comes to completing UR and LL, whereas the one on the left prevents completing them with ones, because this would lead to a forbidden configuration. A stationary distribution $P(ABCD)$ that maximizes the conditional probability $H(D|ABC)$ under the given constraints will therefore assign a higher probability to the configuration $ABCD = 0110$ than to the configuration $ABCD = 0111$. Intuitively, the iterations of BP push the marginals for the squares away from a situation where all admissible configurations have equal probabilities. At the same time, it is proven that BP approximates a maximum-entropic distribution in accordance with the "probabilistic" evidence provided by the factors in the factor graph, which motivates our use of BP to find the distribution with maximum entropy [7].

3. BOUNDING THE CAPACITY

In order to get tight capacity bounds, we applied the combinatorial bounds introduced by Calkin and Wilf [2]. Consider an M by n rectangle and the possible configurations of the first column (with index i) and the second column (with index j). A transition matrix, \mathbf{T} , is defined by the admissible transitions from i to j , such that $t_{ij} = 1$ if the transition is admissible and $t_{ij} = 0$ otherwise. For variable n , we refer to the rectangles as a band. A lower and an upper bound was derived for the hard square constraint, but these bounds actually apply to all constraints with symmetric transition matrix [2] and thus also to the nus constraint. A lower bound, C_{low} , is given by

$$C_{low} = h(2m+2) - h(2m+1), \quad (17)$$

where $h(M) = \log \Lambda_M$ and Λ_M is the largest eigenvalue of the transition matrix T_M of a band of height M .

For the upper bound, the transition matrix is modified by also checking the constraint on a periodic extension of the vectors. This construction is referred to as a cylinder. This leads to the upper bound

$$C_{up} = \frac{\log \Lambda'_{2m}}{2m}, \quad (18)$$

where Λ'_{2m} is the largest eigenvalue of the modified transition matrix T'_{2m} for the cylinder with vectors of $2m$ elements.

4. SIMULATIONS AND RESULTS

In this section we present simulation and numeric results for the entropy of the two constraints.

4.1. Hard-square

For the hard-square, maximizing the entropy of the PRF (12) over p and q gave a maximum value of $H_{max} = 0.5872772$ bits/symbol obtained with $p = 0.7090136$ and $q = 0.8112441$. This coincides with the result obtained in [1], where the model was established for the specific constraint using properties of the constraint, whereas the model here is derived as a PRF. For comparison, the Calkin and Wilf bounds (17,18) gave [2]

$$0.587891161 < C < 0.587891164.$$

4.2. No uniform 2×2 blocks constraint

For the nus constraint both PRF and MRF techniques were applied.

4.2.1. PRF entropy for nus

For the nus, maximizing the entropy of the PRF (13) over p gave a maximum value of $H_{max} = 0.839$ bits/symbol for $p = 0.3875$. For the bilateral PRF (14), we get $H_{max} = 0.826$ bits/symbol. The Calkin and Wilf bounds (17,18) gives $H(16) - H(15) = 0.8482624 \leq C \leq 0.8482626$ for $2m = 16$ in (18).

4.2.2. Belief propagation

In order to give a better picture of how the simulations have been set up, we would like to describe the nature of the factors connecting triplets of nodes. The factors are probability functions

$$f(x_{UL}, x_{UR}, x_{LL}). \quad (19)$$

They are therefore functions of three variables, each of these variables describing a 2×2 binary square. There are 16 configurations for such a square, thus the alphabet of each variable has cardinality 16. We associate a number i in the set $0, \dots, 15$ to each configuration, where i corresponds to the decimal representation of the binary number $ABCD$. For example, here are the associations for the squares shown earlier:

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} = 7, \quad \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = 6.$$

The factor itself is created by associating 1 to each possible configuration, for instance $f(3, 6, 12) = 1$ (notice that the squares have compatible edges and satisfy the constraint) and then by normalizing the function so that

$$\sum_{x_{UL}} \sum_{x_{UR}} \sum_{x_{LL}} f(x_{UL}, x_{UR}, x_{LL}) = 1. \quad (20)$$

The value $H(D|ABC)$ that we compute by means of the procedure outlined above is an approximation (for one reason) to an upper bound (for two reasons). It is an approximation because when BP is run on graphs with cycles we are not guaranteed to converge to the correct value for the marginal distributions; to an upper bound on one hand because by only considering A , B and C as conditioning variables we are ignoring the influence of farther neighboring bits on D , on the other hand because we can only model *finite*, $N \times N$ squares, but N should tend to infinity if we want to get a value for $H(D|ABC)$ that is not conditioned by the edges of the grid. We can mitigate this problem by taking larger squares.

Once the BP algorithm has been running for a sufficient number of times, so that further iterations do not change the value of the marginals, what we get is the joint probability $P(ABCD)$ for all $2^4 = 16$ values of $ABCD$ for all the squares in the grid. In order to compute $P(D|ABC)$, which is the fundamental ingredient to compute $H(D|ABC)$, we need to apply Bayes rule:

$$P(D|ABC) = \frac{P(ABCD)}{\sum_D P(ABCD)}. \quad (21)$$

What we do is taking $P(ABCD)$ from the square that is furthest from the edges, namely the one in the center of the grid. We have verified that already for grids with N as small as 16 the value of $H(D|ABC)$ has converged to 7 decimal figures, to the conjectured upper bound $H(D|ABC) \approx 0.8569835$. Although the approximation does not represent by any means a tight bound for the entropy, the results seem to be in line with those achieved by means of the Calkin and Wilf bounds.

5. DISCUSSION

The most accurate results for the capacity of the constraints were obtained by the Calkin and Wilf bounds through the computation of the maximum eigenvalues of the transition matrices. But this does not define an image model or give insights into how efficient 2-D coding techniques could be obtained. Furthermore, the requirement of a symmetric transition matrix, limits the applicability as this is not the case for constraints defined on larger squares. The PRF provides image models and causal descriptions, which could be the basis of simple bit-stuffing coding, but while the PRF is close, it does not achieve capacity. For constraints defined on larger squares, the PRF was extended to a block PRF in [5]. MRF are max-entropic image models, but they too do not render a coding scheme and furthermore determining the value of the entropy is an unsolved issue, which we in the examples considered can illuminate by the results of the other techniques. As the different approaches each have their strength and weaknesses, the approach chosen should reflect the objective of the work.

6. REFERENCES

- [1] R. M. Roth, P. H. Siegel, and J. K. Wolf, "Efficient coding schemes for the hard-square constraint,"

IEEE Trans. Inform. Theory, vol. 47, 1166–1176, Mar. 2001.

- [2] J. Justesen and S. Forchhammer, *Two-dimensional information theory and coding*, Cambridge: Cambridge University Press, 2010.
- [3] D. K. Pickard, "A curious binary lattice process," *J. Appl. Probab.*, vol. 14, pp. 717-731, 1977.
- [4] D. K. Pickard, "Unilateral Markov fields," *Advances in applied probability*, vol. 12, pp. 655-671, 1980.
- [5] S. Forchhammer and J. Justesen, "Block Pickard models for two-dimensional constraints," *IEEE Trans. Inform. Theory*, vol. 55, 4626–4634, Oct. 2009.
- [6] H. A. Loeliger, "An introduction to factor graphs," *IEEE Signal Proc. Mag.*, pp. 28-41, Jan. 2004.
- [7] J. S. Yedidia, W. T. Freeman and Y. Weiss, "Constructing free energy approximations and generalized belief propagation algorithms," *IEEE Trans. Inform. Theory*, vol. 51, pp. 2282–2312, 2004.