Submitted to ICASSP 2001.

UNWRAPPING PHASE IMAGES BY PROPAGATING PROBABILITIES ACROSS GRAPHS

Ralf Koetter[1], *Brendan J. Frey*[2], *Nemanja Petrovic*[1], *David C. Munson, Jr.*[1]

1. Electrical and Computer Engineering
University of Illinois at Urbana-Champaign

2. Department of Computer Science
University of Waterloo

ABSTRACT

Phase images are derived from source images by applying a modulus operation to each pixel value. *Phase unwrapping* is the problem of inferring the original, unwrapped values from the wrapped values, using prior knowledge about the smoothness of the image. One approach to solving this problem is to infer the gradient vector field of the unwrapped image and then integrate the gradient field. The gradient in a particular direction at a pixel is equal to the observed pixel difference plus an unknown integer number of shifts. We introduce a technique for inferring these shifts using the low-complexity *probability propagation* algorithm, applied in a graphical model that prefers shifts that match the phase image and that constrains the shifts to satisfy the properties of a gradient field. We present results for a phase image from the region of the Sandia National Laboratories.

1. INTRODUCTION

It is a pleasure to discover a problem that is considered fundamental in a community, that has been shown in some form to be NP-hard, that intuitively seems as though it should be solvable, and that remains unsolved. To us, the problem of inferring unwrapped images from phase-wrapped images has this character. Phase unwrapping is a fundamental problem in magnetic resonance imaging [1] and interferometric synthetic aperture radar (SAR) [2]. Fig. 1 shows a 1-dimensional phase signal and a 2-dimensional phase image obtained from synthetic aperture radar (SAR) measurements at Sandia National Laboratories, New Mexico.

Phase unwrapping is a well-formed problem only if assumptions are made about what types of unwrapped signals are preferable. (Otherwise, the original, wrapped image is itself a valid unwrapping that matches the observed data perfectly.) These assumptions usually take the form of an incarnation of "neighboring unwrapped values are more likely to be close together than further apart". For the 1-dimensional signal shown above, it is apparent that we can work our way from left to right, unwrapping measurements as we go. Keeping track of some previous measurements allows us to backtrack and make corrections in ambiguous areas. In fact, if the prior assumption about the smoothness

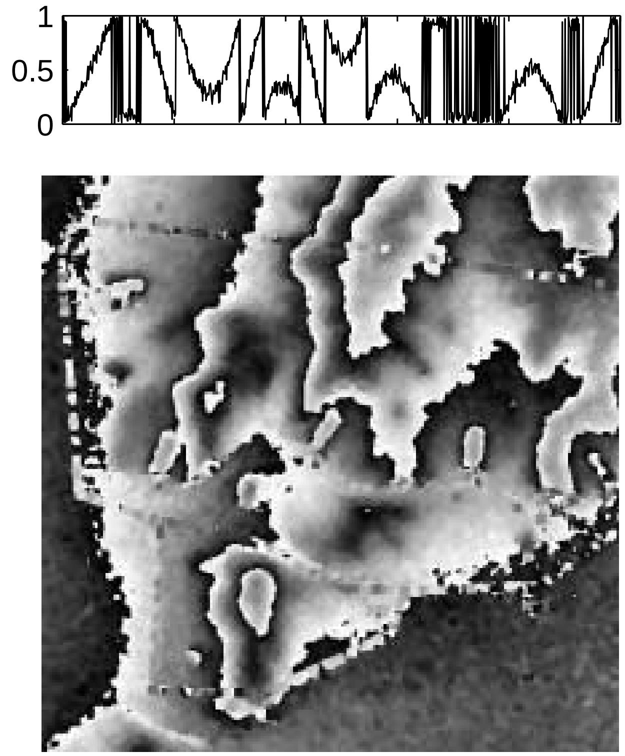

Fig. 1. A phase-wrapped 1-dimensional signal and a phase-wrapped SAR elevation map from Sandia National Laboratories, New Mexico. Pixel values close to 0 are painted white, whereas pixel values close to 1 (the wavelength) are painted black.

of the unwrapped signal bounds the sizes of jumps in the signal, dynamic programming offers an efficient, exact solution to the 1-dimensional phase unwrapping problem.

In contrast, "the final solution" to phase unwrapping in 2-dimensional topologies has not yet been found. Between any two points in the image (see above), there is a large number of paths and after the image is unwrapped, each of these paths must satisfy the prior assumptions. In fact, if phase unwrapping is cast as a "minimum L^0 norm problem" in integer programming, it turns out to be NP-hard [3].

Approaches to solving the phase unwrapping problem

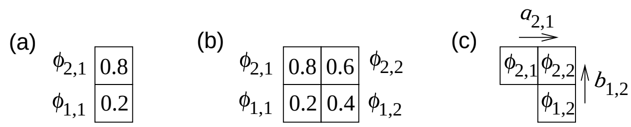

Fig. 2. Phase measurements in small image patches. From (a) it appears that a shift occurred between points $1,1$ and $2,1$. From (b), it appears that a shift probably did *not* occur between points $1,1$ and $2,1$. (c) The phase at 2,2 can be predicted from the phases at 2,1 and 1,2, plus the shifts $a_{2,1}$ and $b_{1,2}$.

include least squares estimates (these are *not* MMSE estimates) [4, 2, 5, 1], integer programming methods [6, 3] and branch cut techniques [7].

Our approach is motivated by our recent work on the probability propagation (sum-product) algorithm for iteratively decoding error-correcting codes, such as "turbocodes" [8]. Until recently, optimal decoding on Gaussian channels was thought to be intractable. However, it turns out that probability propagation in a graphical model describing the code solves the problem for practical purposes. In this work, we show how probability propagation in a graphical model can be used to unwrap phase images.

2. FACTOR GRAPH MODEL

A sensible goal in solving the phase unwrapping problem is to infer the gradient vector field of the unwrapped image, using the wrapped image as input. Then, the gradient field can be integrated to obtain the unwrapped image. The difference between two neighboring pixels in the unwrapped image is equal to the difference between the two pixels in the wrapped image, minus an unknown integer. So, we can view phase unwrapping as the problem of finding these integers, which we will call *shifts*, subject to the constraint that the sum of the shifts around every loop (taking into account the direction of the shifts) must be zero. We refer to this constraint as the *zero curl constraint*.

Let $\phi_{i,j} \in [0,1)$ be the phase value at i,j. (We assume that measurements are taken *modulus* 1 – *i.e.*, the wavelength is 1.) Let $a_{i,j} \in \mathcal{I}$ be the unknown shift between points i,j and $i,j+1$. So, the difference in the unwrapped values at pixels $i,j+1$ and i,j is $\phi_{i,j+1} - \phi_{i,j} - a_{i,j}$. Similarly, let $b_{i,j} \in \mathcal{I}$ be the unknown shift between points at i,j and $i+1,j$.

Consider the two patches of image shown in Fig. 2. From Fig. 2a, the difference in the unwrapped values at $1,1$ and $2,1$ is $0.8 - 0.2 - a_{1,1}$. Assuming the values are more likely to be closer together than further apart, we decide that $a_{1,1} = 1$, so that $0.8 - 0.2 - a_{1,1}$ is as close to 0 as possible.

We can make these local decisions for every neighboring pair of points in a large image, but the resulting set of shifts will not satisfy the constraint of summing to zero around every loop. If we make local decisions for the patch in Fig. 2b, then we decide that $a_{1,1} = 1$, $b_{1,2} = 0$, $a_{2,1} = 0$ and $b_{1,1} = 0$. The sum of these shifts around a counter-clockwise loop is $a_{1,1} + b_{1,2} - a_{2,1} - b_{1,1} = 1$, giving a *curl violation*. We can fix this curl violation by changing one or more of the shifts, at the cost of not keeping the unwrapped pixel differences as close to zero as possible.

To choose the form of the above cost, we develop a probability model of the shifts and the observed phases. We choose a probability model on the shifts that is uniform over all configurations that satisfy the zero curl constraint:

$$p(a,b) \propto \prod_{i,j} \delta(a_{i,j} + b_{i,j+1} - a_{i,j+1} - b_{i,j}), \qquad (1)$$

where $\delta()$ evaluates to 1 if its integer argument is 0 and evaluates to 0 otherwise. In fact, $p(a,b)$ should take into account a preference for smooth surfaces, where the a's and the b's tend to be small. To incorporate this knowledge and also to avoid an algorithm that requires searching over all integers, we restrict the values of the a's and b's to be in $\{-1,0,1\}$.

The density of the observed phase measurements can be formulated recursively. As shown in Fig. 2c, the phase at 2,2 can be predicted from the phases at 2,1 and 1,2, plus the shifts $a_{2,1}$ and $b_{1,2}$. The prediction from 2,1 is $\phi_{2,1} + a_{2,1}$, while the prediction from 1,2 is $\phi_{1,2} + b_{1,2}$. The average prediction is $(\phi_{2,1} + a_{2,1} + \phi_{1,2} + b_{1,2})/2$. Assuming a Gaussian likelihood we obtain the general form

$$p(\phi|a,b) \propto \prod_{i,j} \Big(\exp[-(\phi_{i,j+1} - \phi_{i,j} - a_{i,j})^2/2\sigma^2] \cdot \exp[-(\phi_{i+1,j} - \phi_{i,j} - b_{i,j})^2/2\sigma^2] \Big) \qquad (2)$$

The joint distribution $p(a,b,\phi) = p(a,b)p(\phi|a,b)$ can be represented using a factor graph [9], as shown in Fig. 3a. Each white disc corresponds to an unobserved shift (a's and b's) and is associated with a likelihood term from (2), while each black disc corresponds to a zero curl constraint and is associated with a term in the prior (1).

3. PROBABILITY PROPAGATION — THE SUM-PRODUCT ALGORITHM

Using the above model, phase unwrapping consists of making inferences about the a's and b's in the probability model. For example, the marginal probability that $a_{i,j}$ has the value k, given an observed image ϕ, is

$$p(a_{i,j} = k|\phi) \propto \sum_{a,b: a_{i,j}=k} p(\phi|a,b)p(a,b). \qquad (3)$$

If there are N pixels, the above sum has roughly 3^N terms (recall that the a's and b's can be -1, 0 or 1) and so exact inference is intractable.

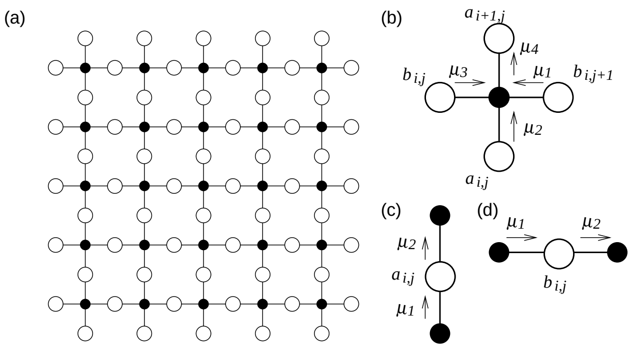

Fig. 3. (a) Factor graph, with white discs representing shifts and black discs representing zero curl constraints. Probability propagation (the sum-product algorithm) consists of computing 3-vectors (μ's) that are passed between vertices. (b) shows that constraint-to-shift messages are computed from incoming shift-to-constraints messages. (c) and (d) show that shift-to-constraint messages are computed from incoming constraint-to-shift messages.

Recently, it has been shown that the probability propagation message-passing algorithm (sum-product algorithm) can produce excellent results for inference in graphical models like the one shown above [8]. It is well-known that probability propagation is exact in graphs that are trees [10], but it has been discovered only recently that it can produce excellent results in graphs with many cycles.

Probability propagation (the sum-product algorithm) consists of computing 3-vectors (μ's) that are passed between vertices. The elements of the 3-vectors correspond to the allowed values of the shift variables, -1, 0 and 1. Each of these 3-vectors can be thought of as a probability distribution over the 3 possible values that the shift variable can take on.

As shown in Fig. 3b, constraint-to-shift messages are computed by combining incoming shift-to-constraints messages with the zero curl constraint. If μ_{1l}, μ_{2k} and μ_{3n}, $k, l, n \in \{-1, 0, 1\}$ are the incoming messages, then the outgoing message μ_{4m}, $m \in \{-1, 0, 1\}$ is computed from

$$\mu_{4m} = \sum_k \sum_l \sum_n \delta(k + l - m - n) \mu_{2k} \mu_{1l} \mu_{3n}. \qquad (4)$$

The 3-vector is usually normalized after this computation.

As shown in Fig. 3c and d, shift-to-constraint messages are computed by combining incoming constraint-to-shift messages with the likelihood for the shift. For shifts in the horizontal direction (Fig. 3c), if μ_{1n}, $n \in \{-1, 0, 1\}$ is the incoming message, the outgoing message μ_{2k}, $k \in \{-1, 0, 1\}$ is computed from

$$\mu_{2k} = \mu_{1k} \exp[-(\phi_{i,j+1} - \phi_{i,j} - k)^2 / 2\sigma^2]. \qquad (5)$$

Similarly, for shifts in the vertical direction (Fig. 3d), if μ_{1n}, $n \in \{-1, 0, 1\}$ is the incoming message, the outgoing message μ_{2k}, $k \in \{-1, 0, 1\}$ is computed from

$$\mu_{2k} = \mu_{1k} \exp[-(\phi_{i+1,j} - \phi_{i,j} - k)^2 / 2\sigma^2]. \qquad (6)$$

Again, these 3-vectors are usually normalized after they are computed.

Given a phase image, probability vectors are passed across the graph in an iterative fashion. Various message-passing schedules are possible, but in our experiments, messages were passed in parallel. That is, at each step, a new value for *every* message is computed from the values of the neighboring messages in the previous step.

This is an inefficient message-passing schedule, so we are currently running experiments with a "forward-backward-up-down"-type schedule, in which messages are passed across the network to the right, then to the left, then up and then down. This schedule appears to speed up convergence by an order of magnitude.

4. EXPERIMENTAL RESULTS

To unwrap the phase image shown in Fig. 1, we first estimated the parameter σ^2 be averaging the squared differences between neighboring pixels in the wrapped image. The value of σ^2 determined in this way is 0.0185.

If decisions for the a's and b's are made locally as described in the introduction, 281 zero curl constraints are violated. After 180 iterations of probability propagation, this number is reduced to 28. These remaining violations correspond to areas of significant shading/overlay or areas with abrupt change in the terrain height. For the 512×512 Sandia phase image, the time required for each iteration of the probability propagation algorithm is on the order of 1 minute on a 600MHz Pentium machine.

We reconstructed the unwrapped image using a weighted least-squares error approach [11]. Normally, this method uses the values of the integers determined locally. Instead of using the locally determined values, we thresholded the estimates of the posterior marginals, $p(a_{i,j}|\phi)$ produced by the probability propagation algorithm. We did not use any additional information about the interferogram – other methods use extra information to apply a mask to cover the areas of significant noise/overlay.

The final surface was reconstructed using 20 iterations of the weighted least-squares algorithm. Fig. 4 shows the reconstructed surface obtained from our method. Compared to the standard least squares estimate (not shown), this reconstruction contains significantly more detail.

5. CONCLUSIONS

We introduced an algorithm that operates in a graphical model that incorporates the zero curl constraints and data likeli-

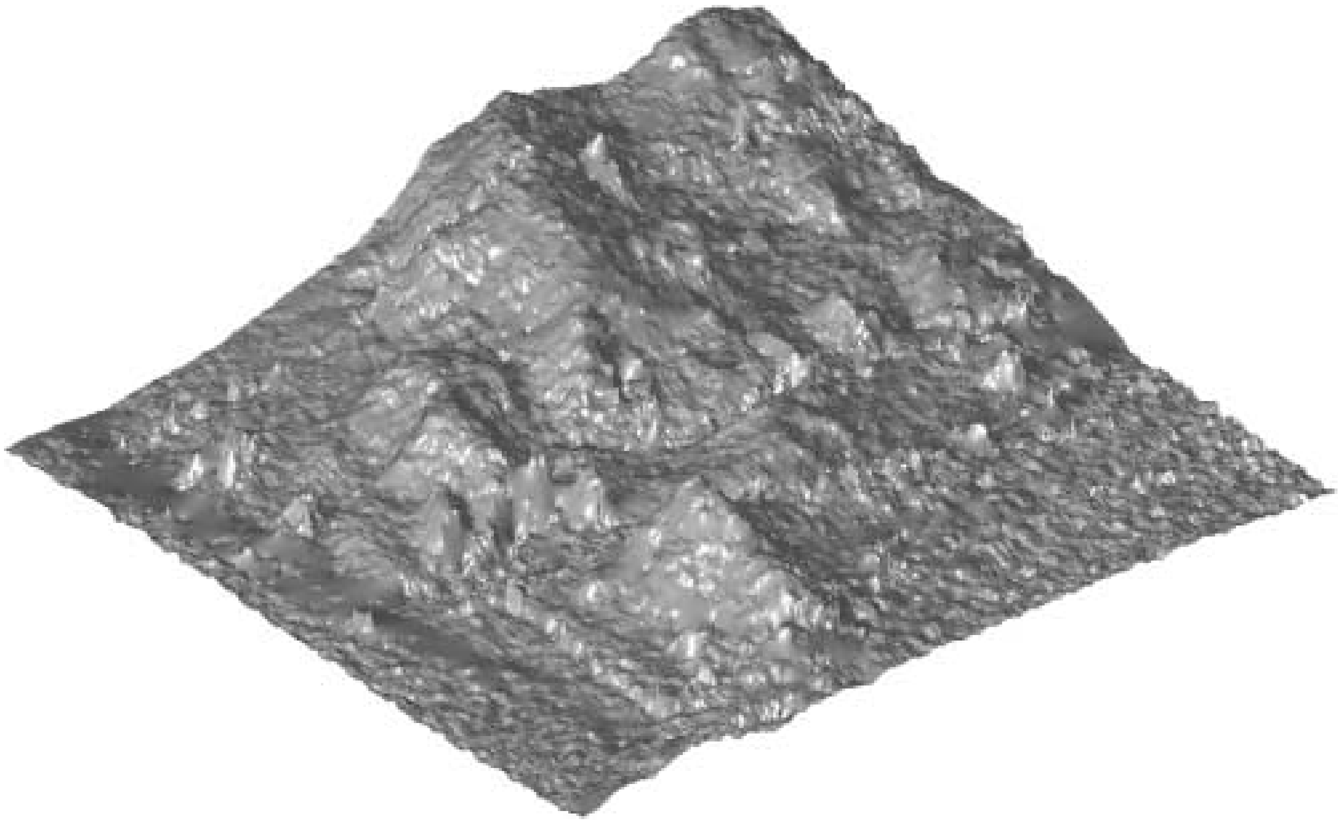

Fig. 4. Reconstructed output of the probability propagation algorithm for the SAR phase image shown in Fig. 1 from Sandia National Laboratories, New Mexico.

hood for phase unwrapping. We found that probability propagation is an efficient algorithm for inferring the relative shift between neighboring pixels at each point in the image.

The algorithm presented here limits the number of shifts between neighboring pixels to -1, 0 or 1. We are currently developing a graphical model that uses Geman and Geman line processes in order to model 1-dimensional discontinuities (*e.g.*, cliffs) in the unwrapped image.

6. REFERENCES

[1] Z.-P. Liang and P. C. Lauterbur, *Principles of Magnetic Resonance Imaging : A Signal Processing Perspective*, IEEE Press, New York NY., 2000.

[2] C. V. Jakowatz, Jr., D. E. Wahl, P. H. Eichel, and P. A. Thompson, *Spotlight-mode Synthetic Aperture Radar: A Signal Processing Approach*, Kluwer Academic Publishers, Boston MA., 1996.

[3] C. W. Chen and H. A. Zebker, "Network approaches to two-dimensional phase unwrapping: intractability and two new algorithms," *Journal of the Optical Society of America A*, vol. 17, no. 3, pp. 401–414, 2000.

[4] S. M.-H. Song, S. Napel, N. J. Pelc, and G. H. Glover, "Phase unwrapping of MR phase images using Poisson's equation," *IEEE Transactions on Image Processing*, vol. 4, no. 5, pp. 667–676, May 1995.

[5] M. Costantini, A. Farina, and F. Zirilli, "A fast phase unwrapping algorithm for SAR interferometry," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37, no. 1, pp. 452–460, January 1999.

[6] M. Costantini, "A phase unwrapping method based on network programming," in *Proceedings of the Fring '96 Workshop*. 1996, ESA SP-406.

[7] R. M. Goldstein, H. A. Zebker, and C. L. Werner, "Satellite radar interferometry: Two-dimensional phase unwrapping," *Radio Science*, vol. 23, no. 4, pp. 713–720, 1988.

[8] B. J. Frey, *Graphical Models for Machine Learning and Digital Communication*, MIT Press, Cambridge MA., 1998, See `http://www.cs.utoronto.ca/∼frey`.

[9] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," To appear in the *IEEE Transactions on Information Theory, Special Issue on Codes on Graphs and Iterative Algorithms*, available at `http://www.cs.utoronto.ca/∼frey`, January 2001.

[10] J. Pearl, *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufmann, San Mateo CA., 1988.

[11] D. C. Ghiglia and M. D. Pritt, *Two-Dimensional Phase Unwrapping. Theory, Algorithms and Software*, John Wiley & Sons, 1998.