Phase Unwrapping for Large SAR Interferograms: Statistical Segmentation and Generalized Network Models

Curtis W. Chen and Howard A. Zebker, *Fellow, IEEE*

***Abstract*—Two-dimensional (2-D) phase unwrapping is a key step in the analysis of interferometric synthetic aperture radar (InSAR) data. While challenging even in the best of circumstances, this problem poses unique difficulties when the dimensions of the interferometric input data exceed the limits of one's computational capabilities. In order to deal with such cases, we propose a technique for applying the statistical-cost, network-flow phase-unwrapping algorithm (SNAPHU) of Chen and Zebker to large datasets. Specifically, we introduce a methodology whereby a large interferogram is partitioned into a set of several smaller tiles that are unwrapped individually and then divided further into independent, irregularly shaped reliable regions. These regions are subsequently assembled into a full unwrapped solution, with the phase offsets between regions computed in a secondary optimization problem whose objective is to maximize the *a posteriori* probability of the final solution. As this secondary problem assumes the same statistical models as employed in the initial tile-unwrapping stage, the technique results in a solution that approximates the solution that would have been obtained had the full-size interferogram been unwrapped as a single piece. The secondary problem is framed in terms of network-flow ideas, allowing the use of an existing nonlinear solver. Applying the algorithm to a large topographic interferogram acquired over central Alaska, we find that the technique is less prone to unwrapping artifacts than more simple tiling approaches.**

***Index Terms*—Network optimization, statistical estimation, synthetic aperture radar interferometry, two-dimensional phase unwrapping.**

I. Introduction

In the context of interferometric synthetic aperture radar (InSAR) applications, two-dimensional (2-D) phase unwrapping is the process of estimating unambiguous phase values from observed phase data known only modulo 2π rad. As this is a key step in the analysis of InSAR data [1], [2], many phase-unwrapping algorithms have been proposed in recent years [3]–[5]. Some of these algorithms have shown promise in handling the problem's intrinsic difficulties, but new difficulties arise when the dimensions of the interferometric input dataset exceed the limits imposed by computer memory constraints or throughput requirements. In this paper, we propose a technique for applying Chen and Zebker's [5] statistical-cost, network-flow phase-unwrapping algorithm (SNAPHU) to large datasets. We use the term "large" to describe interferograms approaching or exceeding the limits of typical computer resources available for unwrapping them.

An obvious strategy for circumventing these resource limitations is to adopt an approach in which a large interferogram is treated as a mosaic of smaller interferograms or "tiles." The tiles can be unwrapped in sequence to reduce the required memory, or they can be unwrapped in parallel to reduce the required execution time [4], [6]. Unfortunately, because of boundary effects (see below), smaller interferograms are often more difficult to unwrap correctly than larger interferograms. Moreover, the task of combining individually unwrapped tiles is not straightforward. Consequently, a mosaic of tile solutions may be more error prone than a solution obtained by unwrapping the entire interferogram as a single piece. To avoid such errors as much as possible, we propose here a tiling scheme that extends the maximum *a posteriori* (MAP) probability estimation framework assumed by SNAPHU [5]. That is, based on previously derived models for the problem statistics, we develop a method of assembling separate unwrapped tiles into a maximally probable full-size solution. We use a generalized network model for solving the posed optimization problem. The proposed scheme, thus, draws upon the same theoretical foundations as the phase-unwrapping algorithm used in its early stages for unwrapping the individual tiles.

All popular unwrapping algorithms are built around the assumption that the complex interferogram's spatial sampling rate is, in most places, sufficient to avoid aliasing. That is, throughout most of the scene, unwrapped neighboring-pixel phase gradients are assumed to equal their wrapped-phase counterparts, where the latter are, by definition, less than one-half cycle (π rad) in magnitude. If this assumption were true everywhere, phase unwrapping would be trivial; the wrapped gradients could simply be integrated from pixel to pixel along an arbitrary path through the array to obtain the unwrapped solution. (Note that in keeping with the literature on phase unwrapping, we use the term "gradient" to denote a scalar phase difference between neighboring pixels.) For most interferograms of interest, however, many unwrapped gradients do exceed one-half cycle. Such gradients are called discontinuities, and whether due to interferometric decorrelation [7], [8] or true variations in the underlying phase field, discontinuities make the unwrapped solution dependent upon the choice of wrapped-gradient integration paths. Thus, the task

Manuscript received October 23, 2001; revised May 14, 2002. This work was supported by the National Aeronautics and Space Administration.

C. W. Chen was with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305-9515 USA. He is now with the Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109-8099 USA.

H. A. Zebker is with the Department of Electrical Engineering and the Department of Geophysics, Stanford University, Stanford, CA 94305-9515 USA.

Publisher Item Identifier 10.1109/TGRS.2002.802453.

of a phase-unwrapping algorithm is equivalent to determining how these discontinuities are to be arranged.

Goldstein *et al.* [9] pointed out that, for any conservative unwrapped phase field, the sum of unwrapped phase gradients around a closed, directed loop of 2×2 pixels must be zero. Because corresponding unwrapped and wrapped gradients are unequal where there are discontinuities, however, the sums of wrapped gradients around such loops may give nonzero results called residues. Residues, thus, indicate the presence of local inconsistencies with the assumption that the data are adequately sampled and discontinuities necessarily run between residues of opposite signs. Therefore, Goldstein *et al.* [9] proposed a phase-unwrapping algorithm in which discontinuities are assumed to lie on approximate shortest paths between positive and negative residues. A unique, self-consistent unwrapped solution results from the selection of a wrapped-gradient integration path that avoids these discontinuities.

Costantini [4] proposed an alternate method for locating phase discontinuities by formulating similar ideas in terms of network theory. In the network model (see Fig. 1), each square of 2×2 pixels is represented by a node, with nodes of positive and negative residues assigned single units of surplus and demand of some imagined commodity. Flow of this commodity is allowed on arcs that connect neighboring nodes, and constraints are defined such that the net flow out of each node must equal the node's net surplus. Thus, when these constraints are met, flow originates at surplus nodes, passes through intermediate nodes as necessary, and terminates at demand nodes; such a flow is said to be feasible. Since arcs in the network model correspond to phase differences between pixels, the direction and magnitude of flow on an arc physically represent the sign and total difference, in cycles, between the unwrapped and wrapped phase gradients associated with that arc. In other words, the amount of flow on an arc is equivalent to the number of extra cycles that must be added to the wrapped gradient in order to obtain the unwrapped gradient. Consequently, all feasible-flow solutions for a given network correspond uniquely to unwrapped solutions for the wrapped input data defining the network [10]. The well-developed ideas and fast optimization routines of network theory may, hence, be applied to phase unwrapping [4], [5], [10], [11].

The network model, itself, does not directly prescribe how the final unwrapped solution is to be chosen from among the many possible solutions, however. For this purpose, a constrained optimization approach has often been adopted. In such an approach, a preselected objective function maps possible unwrapped solutions to scalar costs, and a solver routine is used to find a minimum-cost solution. Many popular phase-unwrapping algorithms have objective functions that can be described by the minimum L^p-norm framework suggested in [12]. Such objective functions are least when the unwrapped and wrapped gradients agree as much as possible, though as described above, unwrapped and wrapped gradients usually cannot be equal everywhere. A parameter p, thus, determines how differences between the two are penalized. The Goldstein *et al.* [9] and Costantini [4] algorithms are L^0 and L^1 algorithms, respectively. Least squares algorithms [3], [13], which have also been popular, are L^2 algorithms.

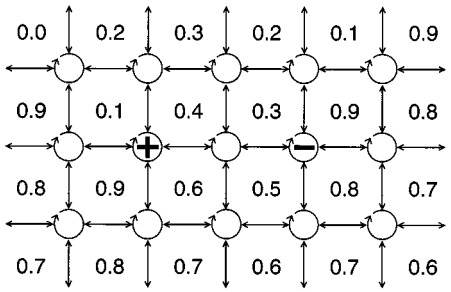

Fig. 1. An example network equivalent of the phase-unwrapping problem. The numbers represent phase values (in cycles) of individual interferogram pixels. Each 2×2 clockwise loop integral of wrapped phase gradients is a node in the network, and positive and negative residues result in supply and demand nodes. Neighboring nodes are connected by arcs or possible flow paths.

Other objective functions are also possible, though. Departing from the simple geometrical shapes of minimum L^p-norm objective functions, Chen and Zebker [5] proposed an objective function that casts phase unwrapping as a statistical estimation problem. This objective function takes the form

$$\text{minimize} \left\{ \sum_k g_k \left(\Delta\phi_k, \Delta\psi_k \right) \right\} \tag{1}$$

where $\Delta\phi$ and $\Delta\psi$ are individual unwrapped and wrapped gradients, respectively. Each gradient, or each arc in the network model, is assigned an independent arbitrarily shaped cost function $g(\cdot)$ which maps the arc's flow $(\Delta\phi - \Delta\psi)/2\pi$ to a scalar cost. The summation with index k includes all rowwise and columnwise gradients, so the total cost of the objective function is the sum of all the arc costs. Employing a MAP estimation framework, Chen and Zebker [5] defined arc cost functions such that minimizing the total cost approximately maximizes the probability of the unwrapped solution given the observable input data. These cost functions are formed from the negative logarithms of the unwrapped-gradient probability density functions (PDFs):

$$g_k \left(\Delta\phi_k, \Delta\psi_k \right) = -\log \left(f \left(\Delta\phi_k | \Delta\psi_k, I, \rho \right) \right). \tag{2}$$

Here, $f(\cdot)$ represents the conditional PDF of a particular unwrapped gradient $\Delta\phi$ given the observed wrapped gradient $\Delta\psi$, image intensity I, and interferometric correlation ρ [7]. The cost functions $g(\cdot)$, hence, depend on the local values of the observable quantities and vary in shape throughout the interferogram. The actual shapes of these functions, thus, follow from statistical models that were derived specifically, with phase unwrapping in mind, for different InSAR applications [5].

Unfortunately, the complex shapes of these statistical cost functions pose phase unwrapping as a difficult nonconvex optimization problem. However, the problem can be solved approximately with use of nonlinear network-flow techniques [5], [10]. The SNAPHU solver implements an iterative algorithm for making improvements to an initial feasible-flow solution. At each iteration, the solver computes the incremental costs of sending a discrete amount of additional flow on each arc in either direction. The solver then augments flow on closed, directed cycles that have net negative incremental costs. Thus,

the solver steps from one feasible solution to another, decreasing the total cost.

The posed optimization problem is *NP*-hard [10], though, and an exact (globally minimal) solution generally cannot be computed efficiently. That is, according to computational complexity theory, the problem is intractable in the absence of simplifying assumptions [14]. Hence, for the sake of practicality, several approximations are incorporated into both the solver and the statistical models used by SNAPHU. Nevertheless, the algorithm has yielded promisingly accurate results on real InSAR data [5].

In the following sections, we present a technique for applying SNAPHU to large interferograms. Extending previous work [5], [6], we use statistical models to partition the data into small, individually unwrapped, arbitrarily shaped regions that are believed to be free of relative local unwrapping errors. These regions are then assembled into a full solution through the use of a topologically irregular network model that again accounts for the problem statistics. We examine the performance of the technique by applying it to a long strip of topographic InSAR data acquired over the Alaskan interior.

II. Basic Tiling Strategies

As noted above, a simple strategy for dealing with a large interferogram is to divide the interferogram into smaller tiles. For interferograms with long physical discontinuities, however, unwrapping accuracy often degrades with the degree of interferogram subdivision. Consider a topographic interferogram, such as the one illustrated schematically in Fig. 2, where slant range is assumed to increase toward the right. The heavy black line represents a phase discontinuity associated with the decorrelated front face of a mountain in layover. The rest of the interferogram, including the shaded region, is well correlated. If the interferogram is unwrapped as a single piece, all parts of the interferogram can be connected by reliable integration paths that do not cross the discontinuity. On the other hand, if the scene is partitioned into two smaller tiles as shown, the discontinuity isolates the shaded area from the rest of the left tile, and an unwrapping error likely results. Such tiling artifacts are even more apt to occur in interferograms containing longer or more sinuous discontinuities. Thus, the adoption of a tiling strategy requires great care. In this section, we examine a relatively simple tiling strategy, and in so doing, we review the basic issues raised by interferogram subdivision.

Breaking up an interferogram into separate tiles is easy; putting those tiles back together is the difficult step. It has been noted, however, that the problem of mosaicing individually unwrapped tiles is similar to the original (single-piece) phase-unwrapping problem in that integer-cycle offsets must be added to each tile in order to align the phase values of adjacent tiles [6], [15]. That is, the edges of adjacent tiles must be brought to within one-half cycle of one another throughout most of the array of tiles. Tiles in the tile-reassembly problem are, hence, analogous to pixels in the original phase-unwrapping problem. For brevity of notation, we refer here to the pixel-level unwrapping problem and the subsequent tile-reassembly problem as the primary and secondary problems, respectively.

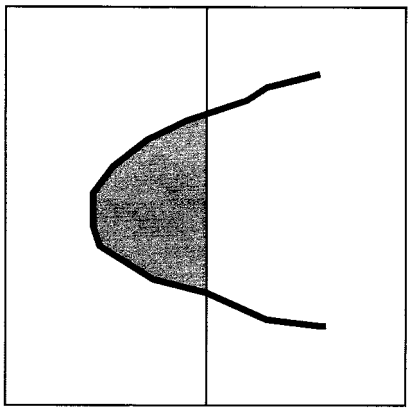

Fig. 2. Representation of an unwrapping artifact due to tiling. The heavy black line represents a phase discontinuity following the decorrelated front face of a mountain in layover. Slant range is assumed to increase toward the right. If the interferogram is split into left and right halves as shown, the shaded area is isolated from the rest of the left half by the decorrelated region.

For the tile-reassembly (secondary) problem, the degree of alignment between adjacent tiles can be quantified in various ways. We use the following procedure. For each pair of pixels straddling the tile boundary, we compute the unwrapped phase difference and round the result to the nearest integer cycle. We then find the most common value of the rounded pixel differences and call this value the "tile difference" between the two tiles. In virtually all real-world situations characterized by reasonable coherence, the true tile difference between neighboring tiles is zero as long as the tiles have been unwrapped correctly. This is because most pixel differences along the tile boundary are by assumption less than one-half cycle, even if some are not.

Thus, if all individual tiles are unwrapped correctly, there must exist an arrangement of tile offsets such that all tile differences are zero. These offsets can be found by aligning adjacent tiles following some path from tile to tile, just as the integration of phase gradients from pixel to pixel results in a self-consistent primary solution for an interferogram in which there are no residues.

If many of the pixels along a tile boundary are unwrapped in error, however, the affected tile difference may no longer be zero in the desired secondary solution (see Fig. 3). Tile reassembly then becomes a problem similar to phase unwrapping in the presence of residues: tiles must be arranged so that tile differences are zero *most* of the time—but tile differences affected by primary unwrapping errors may need to be nonzero for self-consistency in the secondary solution.

Realizing this, Carballo [6] showed that the secondary problem of arranging phase offsets between tiles can be described by a network model nearly identical to the network model used for the primary problem. The secondary network for the tile set of Fig. 3 is shown in Fig. 4. Nodes exist at the tile corners, and with arcs connecting neighboring nodes, there is a tile difference for each arc in the network. Therefore, as in the primary network, flow on a secondary arc represents the additional number of cycles that should be included in the corresponding tile difference. Residues arise in Fig. 4 because of the unwrapping error in the primary solution for tile B. That is, the sums of tile differences around the closed loops

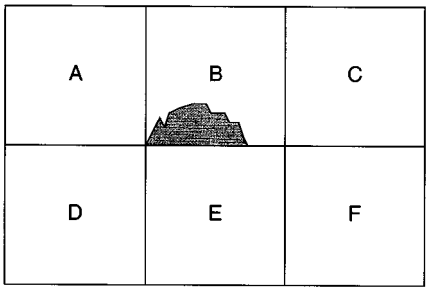

Fig. 3. Effects of an unwrapping error on tile differences. The shaded area in tile B represents an unwrapping error; it is offset from the rest of the tile by an incorrect integer number of cycles. The tile difference (i.e., phase difference) between tiles B and E must be nonzero to account for this error if the tile differences between the rest of the tiles are assumed to be zero.

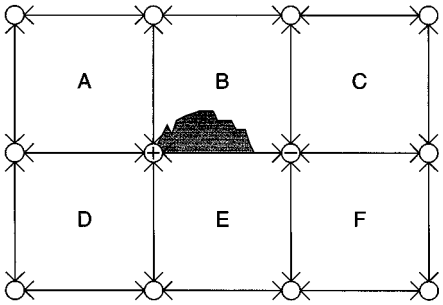

Fig. 4. Secondary network for the tile set of Fig. 3. Nodes exist at tile corners and arcs between nodes correspond to tile differences. Residues exist because the unwrapping error in tile B affects the tile difference between tiles B and E.

{A,B,E,D,A} and {B,C,F,E,B} are nonzero. Equivalently, if the primary tile solutions are arranged with arbitrary integer-cycle offsets and the secondary arc flows are calculated from the resulting unwrapped tile differences, then there is nonzero net secondary flow out of the nodes associated with residues. Cost functions for the secondary arcs are discussed below.

With the network defined in this way, network-flow solvers (see [4], [5], and [10]) can be used to arrange the tile offsets according to some objective function for the secondary problem. This simple tiling strategy may be acceptable for interferograms whose tiles are easy to unwrap individually, but for interferograms characterized by long phase discontinuities, tiling artifacts may become much more problematic. This is because relative unwrapping errors that exist in the primary solutions of individual tiles will also exist as errors in any final tile-reassembled solution; no arrangement of tile offsets can correct for unwrapping errors internal to individual tiles. The likelihood of tiling artifacts can be lessened by specifying larger tiles, but doing so also mitigates the computational gains motivating the technique in the first place. The same applies to the use of widely overlapping tiles.

III. Reliable Regions and Tile Segmentation

To avoid the artifacts associated with the simple tiling approach described in Section II, Carballo [6] insightfully proposed the following technique. After the separate tiles are unwrapped, each tile is partitioned into independent, arbitrarily shaped regions that are believed to be free of relative internal unwrapping errors. The secondary network problem is then carried out between these reliable regions rather than between whole tiles. Thus, with the offsets of individual regions independently adjustable, major unwrapping errors within primary tile solutions can be corrected as the secondary problem is solved. Hensley [16] independently developed similar ideas in a "bootstrapping" technique that relates disconnected regions of unwrapped data between InSAR processing patches.

For defining reliable regions, Carballo [6] used a scheme in which multiple primary unwrapped solutions from different interferogram partitionings are compared with one another and to a coherence map. The use of redundant primary solutions might impose a significant additional computational burden, however. In this section, we propose an alternative method for defining reliable regions. Our method is very efficient and is justified theoretically by the statistical-cost framework described in [5]. We discuss techniques for assembling these regions in Section IV.

The reliability of a region can be quantified as the probability that the primary unwrapped gradients within the region are estimated correctly. Fittingly, the statistical cost functions of [5] are, by design, measures of these very probabilities. For example, if a primary arc has high incremental costs given the current primary unwrapped solution, there is a low probability that flow should be added to or removed from the arc, so there is a low probability that the corresponding phase gradient is estimated in error. Thus, after applying SNAPHU to the individual tiles, we form reliable regions by employing a statistical-cost region-growing approach. Region-growing techniques have also been used in several other contexts for phase unwrapping [6], [9], [17], [18].

Our approach is described as follows. For each primary phase gradient, we first compute the incremental statistical costs for positive and negative unit flow increments given the primary tile solution. That is, for flow increments $\delta = +1$ and $\delta = -1$, we compute incremental costs $\Delta g^{(+)}$ and $\Delta g^{(-)}$ for each arc through

$$\Delta g = g(\chi_0 + \delta) - g(\chi_0) \tag{3}$$

where χ_0 is the arc flow in the primary solution. Next, because lower incremental costs (those closer to $-\infty$) imply higher probabilities of errors, we define a scalar arc cost c_s as the lesser of the arc's two incremental costs Δg.

$$c_s = \min\left(\Delta g^{(+)}, \Delta g^{(-)}\right). \tag{4}$$

We then smooth these scalar costs using convolutional methods. Having defined criteria for how regions will be grown, we now select a seed pixel as the beginning of a region, and we successively add adjacent nonregion pixels to the region if they can be reached without crossing phase gradients for which c_s falls below a certain threshold. When no more pixels can be added to the region, a new seed is chosen from the set of unvisited pixels, and another region is grown. This process continues until all pixels in the tile have been assigned to regions. Note that these regions are defined by groups of pixels in the interferometric data, not by nodes in the network model (see Fig. 1). Seed pixels may be selected somewhat arbitrarily, since the region boundaries are determined by the costs c_s.

As a final step, we enforce a lower limit on the size of each region in order to keep the number of different regions and, hence, the size of the secondary problem, manageable. Specifically, we identify regions containing fewer than a preset number of pixels and we merge these regions with their closest neighbors. Thus, where reliable regions might otherwise be very small (e.g., decorrelated areas), regions are clustered together across gradients with the highest possible incremental costs. Phase-unwrapping errors might accumulate within merged regions, but such areas often provide little useful information in any event. Our intention is to prevent such errors from spreading into more reliable areas. An alternative approach is to mask out unreliable areas by discarding any pixels that belong to regions smaller than a preselected size.

IV. Region Reassembly and the Secondary Network

Once reliable regions have been defined within the individually unwrapped tiles, our task is to solve the secondary problem of arranging the phase offsets between different regions. Since regions are arbitrarily shaped, however, the implied secondary network no longer has the same regular, grid-like topological structure as the primary network. Nevertheless, a network model can still be used for the secondary problem.

Carballo [6] suggested that the secondary network be constructed from a Delaunay triangulation of control points representing the different regions. This scheme has several disadvantages, however. Because secondary nodes in this scheme correspond to triangles, these nodes are all of degree three or less. The secondary network must consequently contain extraneous nodes and arcs which might degrade computational efficiency and obscure the relationship between the secondary network and the region-assembly problem. The computation of the Delaunay triangulation itself also introduces additional complexity [19], [20]. Moreover, if the region offsets are defined in reference only to single-pixel control points, the scheme may not be robust to small, localized errors within regions. We now propose a new secondary network model that avoids such shortcomings.

Recall that in the primary network model, arcs correspond to phase differences between pixels. We now assume analogously that secondary arcs correspond to phase differences between regions, so we include one secondary arc in the network for each boundary between two regions. We then place secondary nodes at tile corners and at other locations where more than two regions touch. Thus, nodes occur where flow paths split, as illustrated in Fig. 5. That is, the placement of secondary nodes is determined by the arrangement of the secondary arcs. Since the phase offset between any two neighboring regions is described by a unique secondary arc, any possible arrangement of offsets can be represented by a feasible flow.

We construct the secondary network from the region data using an algorithm similar to a tree search. That is, beginning from a primary node known to correspond to a secondary node (e.g., a node at a tile corner), we trace out secondary arcs by following sequences of primary arcs that have different regions on either side of them or that lie on the edges of the interferogram. A secondary arc terminates when it enters a primary node that touches more than two regions. At these locations, we create secondary nodes if such nodes do not yet exist. We then continue, tracing unvisited secondary arcs that depart from these secondary nodes, until all secondary arcs have been followed.

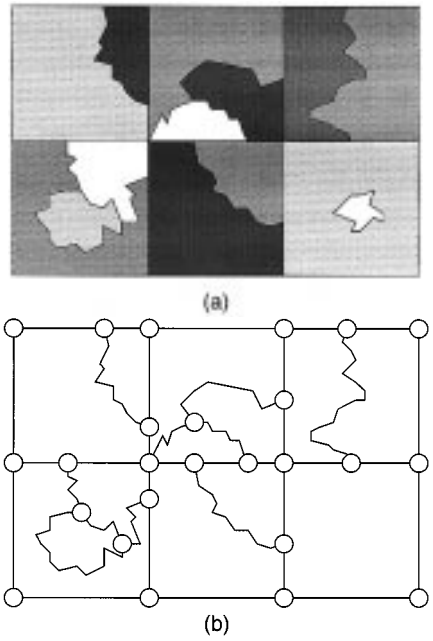

Fig. 5. Example region-based secondary network. (a) Region-set. (b) Corresponding secondary network. Shaded areas in (a) represent reliable regions, indicated with arbitrary gray levels, for the set of six square tiles shown. Circles and lines in (b) represent secondary nodes and arcs. Arrowheads on the arcs are omitted for clarity, but all arcs are assumed to be bidirectional, as elsewhere in this paper. Arcs correspond to boundaries between regions, and nodes occur where arcs split. The region in the center of the lower-right tile is not represented in the network, since this region cannot be reached from any tile edge.

Note that the secondary network does not contain nodes or arcs for regions that are unaffected by the inclusion of adjacent tiles (e.g., the small, central region in the lower-right tile of Fig. 5). The phase offsets of such regions are set relative to their surroundings in the primary problem; these offsets should not change in the secondary problem.

The feasibility of a secondary flow solution depends in principle on the distribution of surplus and demand on the secondary nodes. We need not explicitly compute residues for the secondary problem, though. Instead, since the unwrapped primary solutions provide estimates of the relative offsets between regions within the same tile, we simply initialize the offsets between whole tiles and find feasible flow values for the secondary arcs. That is, we choose an initial, temporary set of bulk tile offsets by aligning neighboring tiles across the top row then down each column, defining tile differences as above. Such an initialization is analogous to a trivial path-integration unwrapping method for the primary problem. With relative offsets thus implied for all regions, we can directly compute initial values for the secondary arc flows as follows. Based on the unwrapped primary pixel values and the initial relative region offsets, we compute the flow for each primary arc that is part of a secondary arc. We then initialize the secondary flows, setting them equal to the most common values of the primary flows along those secondary arcs. If desired, residues may be computed at this point

by summing the net flow out of each secondary node. Doing so is unnecessary, though; with the use of nonlinear network-flow solvers [5], [10], we can make iterative improvements to the secondary flow initialization while maintaining solution feasibility.

In order to make improvements, however, we must define costs for the secondary arcs. Here, we again depart from the scheme proposed in [6]. Instead, we use an approach in which secondary costs draw physical meaning and theoretical justification from the statistical cost functions used in the primary problem. Since secondary arcs are simply sequences of primary arcs, we define the incremental cost of a secondary arc to be the sum of the incremental costs of the primary arcs traced by the secondary arc, given the current primary flows. In other words, adding δ units of flow to a secondary arc is equivalent to adding δ units of flow (in the appropriate direction) to every primary arc traced by the secondary arc; therefore, we equate the incremental cost of the former to that of the latter. Expressed mathematically,

$$g_2(\chi_2+\delta)-g_2(\chi_2)=\sum_k\left(g_{1k}(\chi_{1k}+\delta)-g_{1k}(\chi_{1k})\right) \quad (5)$$

where $g_1(\cdot)$ and $g_2(\cdot)$ are the cost functions for the primary and secondary arcs; χ_1 and χ_2 are the current primary and secondary arc flows; and the summation with index k is taken over all primary arcs comprising the secondary arc (see Fig. 6).

We handle the edges of the secondary network by assigning zero cost to all secondary arcs on the scene edge. All nodes on the scene edge, thus, form a single effective "ground" or "superconducting-edge" node that is assigned whatever charge necessary to make the network neutral. The interferogram boundary has been treated equivalently in several other primary phase-unwrapping algorithms [4], [9], [10].

With the secondary network thus defined, we use the nonlinear network-flow solver of [5] to find a feasible secondary flow that approximately maximizes the *a posteriori* probability of the region-offset solution. Note that this solver accepts arbitrary network topologies, so the secondary network's lack of grid-like regularity poses no extra difficulty in solving the secondary optimization problem. Once the secondary flow solution is found, we adjust the primary flows to reflect the new secondary flows, and we integrate the resulting primary unwrapped gradients over the full interferogram.

The result we obtain is an approximation to the solution that would have been obtained had the interferogram been unwrapped as a single piece. That is, our tiling approach produces a solution that approximately minimizes the MAP objective function defined for the *full-size* interferogram. To see this, imagine that the interferogram is unwrapped as a single piece, starting from an initialization that consists of an arbitrary mosaic of the primary tile solutions. Improvements to this full-size initialization will be necessary because of tiling artifacts, but many locally correct areas will not need adjustment; flow will be placed only on a small subset of the arcs in the scene—it is these arcs whose connectivity and costs are represented by our secondary network. The other, more reliable parts of the scene need not be examined further. Hence, by using our statistical cost functions to identify reliable regions that are unlikely to need improvement, we can ignore most of the variables of the full-size problem. We can represent the remaining variables through the secondary network constructions described above. Thus, the tiling methodology described above is not just an *ad hoc* collection of procedures, it is a cohesive extension of the MAP optimization framework.

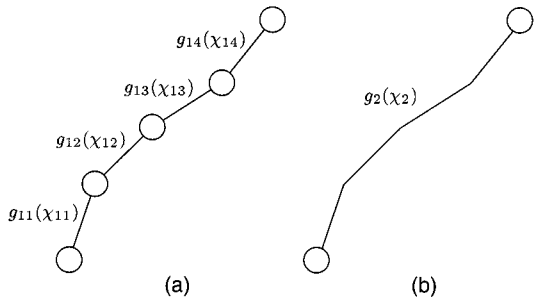

Fig. 6. The cost of adding flow to a sequence of primary arcs (a) is equal to the cost of adding the same flow to the corresponding secondary arc (b).

V. Tile Size Considerations

While the approach described above is meant to approximate the unwrapping of a large interferogram as a single piece, the true single-piece approach may still be more accurate. Though expensive—perhaps even prohibitive—computationally, the single-piece method avoids uncertainties related to how reliable regions are defined. That is, the region-growing method described here relies only on local arc costs, so some regions might enclose paths that require flow updates. Because such updates would not be possible in the secondary problem, the tiling approach would contain errors that might be avoided by an iterative algorithm whose scope is the full interferogram. Consequently, there may remain something of a tradeoff between unwrapping accuracy and computational efficiency or capability.

As a result, tile size is an important consideration in the practical application of our approach. Clearly, memory limitations impose strict upper bounds on tile sizes. Moreover, since most phase-unwrapping algorithms have execution times that are worse than linear in the size of their inputs, the total execution time should decrease with tile size as long as overhead associated with region reassembly remains small. For interferograms that are easy to unwrap, there may be no noticeable penalty in accuracy if the scene is divided into many small tiles. All other things being equal, however, we might expect the likelihood of errors to increase as tile sizes decrease. It might, therefore, be prudent to keep the tiles significantly larger than the expected characteristic lengths of phase discontinuities in the input data. Such lengths may be related, for example, to the characteristic physical sizes of terrain features in a topographic interferogram. Moreover, if phase discontinuities are known to favor a particular orientation (e.g., they tend to run in azimuth more so than range), tile aspect ratios can be adjusted accordingly.

VI. Application to Experimental Data

We now examine our algorithm's performance on a large (23240×4800 pixel) topographic interferogram from the Alaska SAR Facility's Alaska Digital Elevation Model Project. This

project entails the use of ERS-1/ERS-2 tandem data for mapping the state's topography, and because Alaska covers a large geographical area with a great deal of rugged terrain, the project poses a difficult challenge for phase unwrapping. The data, thus, provide a thorough test of the algorithm's feasibility for such applications.

To evaluate algorithm performance, we compare unwrapped results with an unambiguous simulated unwrapped phase field generated from a reference digital elevation model (DEM). However, the DEM is relatively coarse and not always accurate, and noticeable disparities exist between the interferometric data and the reference data. Nevertheless, the DEM is adequate for the identification of most significant phase-unwrapping errors, as phase-unwrapping artifacts are manifest as integer-cycle jumps. Other possible error sources include surface displacement [2], atmospheric effects [21], noise in the interferogram, and artifacts from transforming and registering the DEM to radar coordinates. Interferometric height measurements may also be slightly different than those derived from other sources, since the observed radar echoes may have scattering centers that lie above or below the ground surface. These effects are probably most significant in areas that are covered by vegetation canopies or characterized by significant microwave ground penetration (e.g., icy surfaces).

Because the images shown in this section are quite large, we have greatly reduced their resolutions for reproduction purposes. Note, however, that all processing was done on the large, high-resolution data files. Note also that the images are shown in radar coordinates, so they appear rotated and slightly distorted with respect to their true geographical orientations.

The interferogram of Fig. 7(a) is formed from data acquired during ERS-1/ERS-2 orbits 22 210 and 2537. The wrapped phase is indicated by the color, and the interferogram magnitude is indicated by the grayscale brightness. As this scene covers an area approximately 460 km long and 100 km wide, the interferogram is characterized by a variety of terrain types, from well-correlated flatlands to poorly correlated mountains in layover. Owing to its large size and perpendicular baseline of approximately -136 m, however, phase fringes in some areas are so finely spaced that they are not visible at the resolutions used for reproduction. The interferometric correlation is shown in color in Fig. 7(b) with the interferogram magnitude again shown in grayscale brightness.

The interferogram was unwrapped as a 32×7 array of tiles, with neighboring tiles overlapping by 20 pixels. These parameters were chosen because they result in reasonably sized tiles (about 750×700) given the considerations of Section V. SNAPHU was applied to each tile, and the proposed statistical-cost, region-based secondary technique was used to partition and reassemble the primary tile solutions. With the selected region-formation cost threshold and a minimum region size of 200 pixels, the number of regions per tile is typically on the order of 50. Results after secondary optimization are shown in Fig. 7(c). Color in this image represents relative unwrapped phase error with respect to the reference DEM, while the grayscale brightness once again depicts the interferogram magnitude. A number of slowly varying differences between the unwrapped solution and the DEM-derived reference phase are evident in the error image, but the lack of integer-cycle jumps in most areas suggests that these differences are due to the sources described above rather than to phase unwrapping errors. Most of the interferogram is correctly unwrapped, and there are no large-scale errors, though some localized errors are apparent in the rough terrain of the Alaska Range near the middle of the image. A few relatively small tiling artifacts are also evident upon close inspection of this area; they appear as straight, abrupt, integer-cycle jumps at tile boundaries. Overall, however, our region-assembly approach performs well, given the difficulty of the interferogram. Note that the area near the upper left corner of Fig. 7(c) is masked out because reference data there were missing. The algorithm always produces a complete solution.

The above solution is clearly superior to the simple tile-based solution shown in Fig. 7(d). To obtain the latter, the primary unwrapped tile solutions from SNAPHU were simply aligned across the top tile row and down each column as described in Section II. Note that the tiles appear to be aligned across the bottom row and up each column in Fig. 7(d) because the images are shown flipped vertically for display purposes; this orientation more closely matches natural ground coordinates. Tiling artifacts are quite apparent in the upper part of Fig. 7(d) (i.e., the lower part of the data array when it was unwrapped) because errors within the primary tile solutions in the central mountainous area cause tile misalignment. Consequently, columns of tiles in the upper part of Fig. 7(d) are offset relative to one another even though many of the tiles are correctly unwrapped internally. Smaller errors within individual tiles are visible as well. Both types of errors are avoided, for the most part, in the region-reassembled solution of Fig. 7(c).

Fig. 8 shows (enlarged) the indicated parts of the interferogram, coherence map, and error images of Fig. 7. Note that the boundaries of this 2000×1760 area are unrelated to tile edges. Though the topography in this part of the scene is not the most severe of the full-size interferograms, the data here are still characterized by layover and decorrelation. Hence, phase unwrapping for such an area is quite difficult, so some of the errors present in Fig. 8(c) and (d) are not altogether unexpected. The errors of Fig. 8(d) are not confined to the mountainous area, however. That is, in the simple tiling approach, relative unwrapping errors propagate into areas of more gentle terrain. Local errors near tile boundaries lead to additional artifacts as well. The solution obtained after secondary optimization [Fig. 8(c)] is clearly superior.

Fig. 9 gives an example of the relationship between unwrapped solutions from the single-piece approach and the region-based tiling approach. This figure shows the relative errors obtained when the part of the interferogram depicted in Fig. 8 is unwrapped *ab initio* as a single piece [Fig. 9(a)] and as a 4×4 array of tiles [Fig. 9(b)]. As expected, there are fewer errors in the single-piece solution. Both solutions contain edge artifacts near the upper right and lower left parts of the scene, but the tile-based solution also contains a larger error affecting the lower right of the image. Still, errors in Fig. 9(b) are confined mainly to the more rugged areas and do not spill over into the upper part of the image as in Figs. 7(d) and 8(d). Note

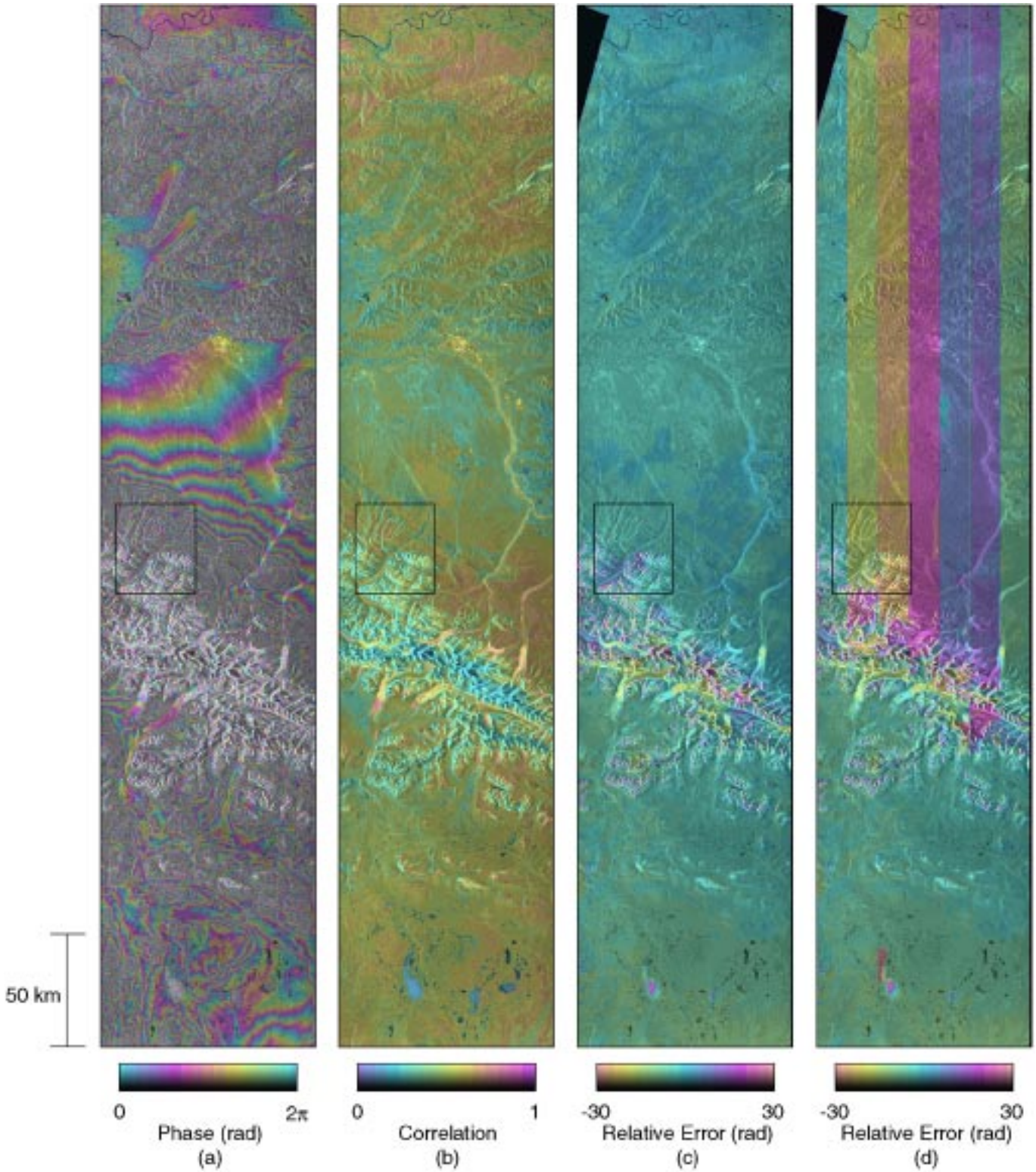

Fig. 7. Test data and results. (a) Wrapped phase. (b) Coherence estimate. (c) Relative unwrapped phase error from the proposed technique. (d) Relative unwrapped phase error from a simple tile-based approach. Parts of the error images are masked out because reference data there were missing. All panels depict the interferogram magnitude in grayscale brightness. The boxed areas are shown enlarged in Fig. 8.

that although the single-piece solution of Fig. 9(a) was obtained without tiling, we would not necessarily expect it to contain fewer errors than the tile-based solution of Fig. 8(c) because the latter, being a subset of a larger image, benefits from the contextual information of surrounding areas (see Fig. 7).

The data shown here were unwrapped with parallel use of the four 700 MHz Intel Pentium III processors of a Dell PowerEdge 6300 server. For the results of Fig. 7(c), the aggregate processor time required was 4:40:08, while the elapsed real time was 1:16:14. During the primary unwrapping stage, each processor

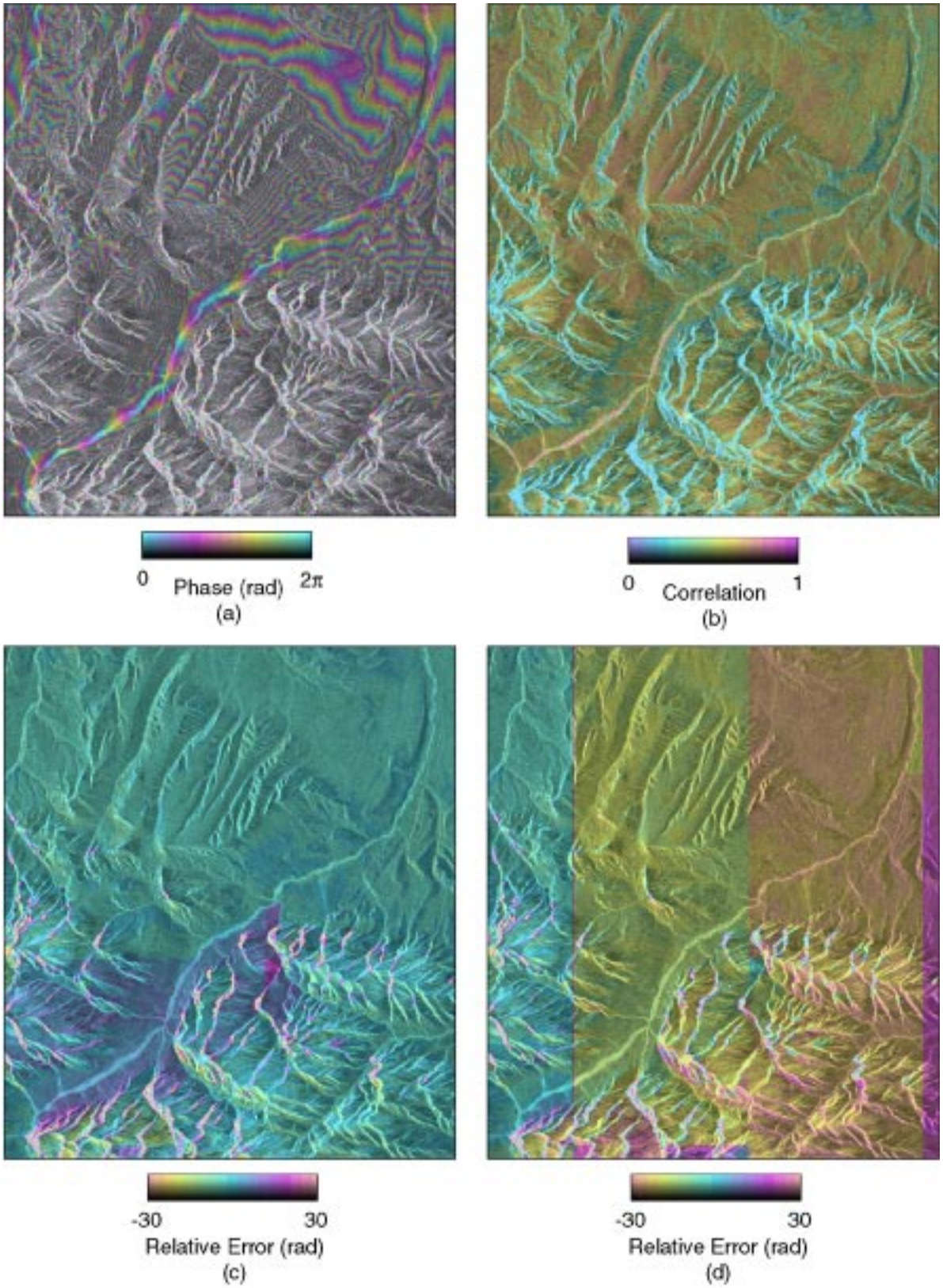

Fig. 8. Enlargements of the boxed areas in Fig. 7. (a) Interferogram. (b) Coherence map. (c) Error image for proposed technique. (d) Error image for simple tiling approach.

required approximately 52 MB of memory. The time and memory requirements for the solution of Fig. 7(d) were similar to those for Fig. 7(c) because region formation and secondary network initialization and optimization are very fast compared to the primary tile-unwrapping step. For the results of Fig. 9(a), the required memory and the single-processor execution time

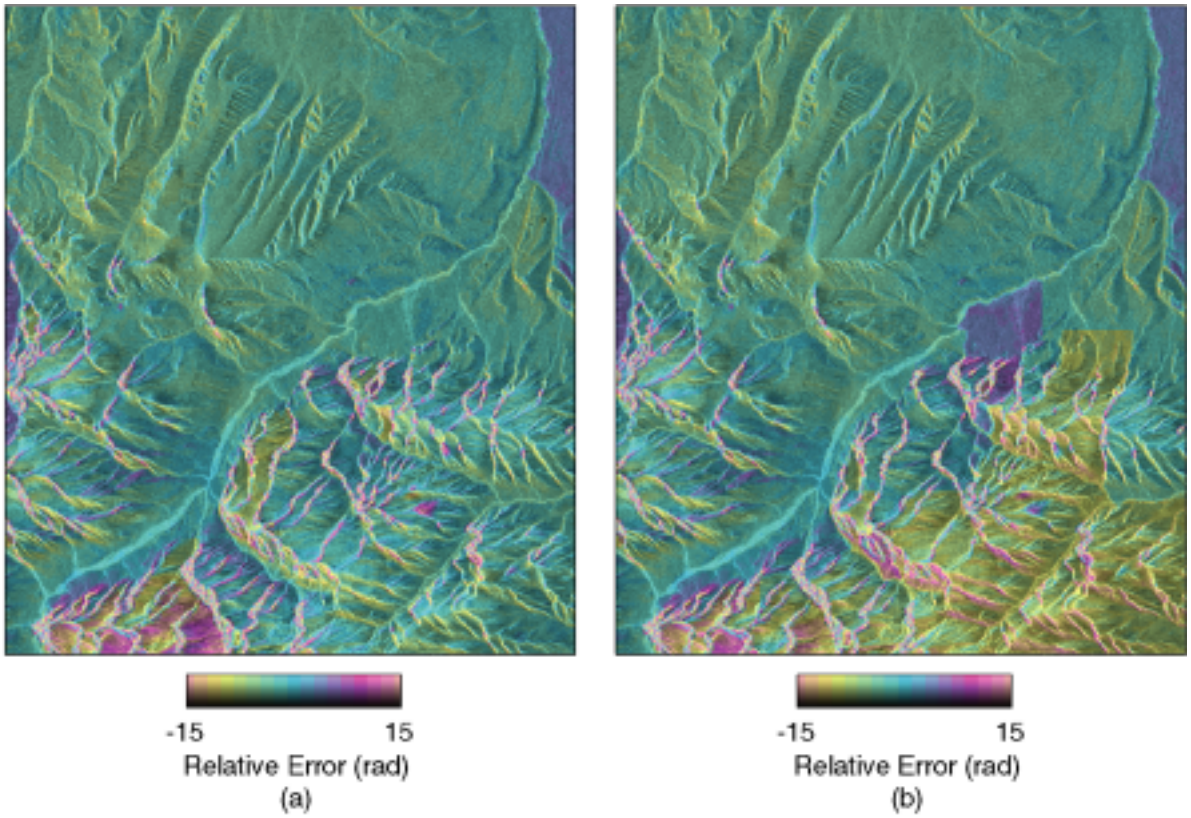

Fig. 9. Unwrapped results for the interferogram piece shown in Fig. 8. (a) Interferogram unwrapped as a single piece. (b) Interferogram unwrapped as a 4×4 array of tiles.

were 300 MB and 0:22:00. The required memory per processor, the aggregate processor time, and the elapsed real time for the results of Fig. 9(b) were 25 MB, 0:06:16, and 0:02:05.

Note that in our implementation of the algorithm, region growth is performed in parallel directly after the primary tile-unwrapping step. For each tile, the primary cost, solution, and region data are written to disk rather than stored in memory; once all the tiles have been unwrapped, the data are read from disk one by one as the secondary network is formed, so the total memory required is always on the order of the tile size, not the size of the full interferogram. We estimate that if the full-size interferogram of Fig. 7 were unwrapped as a single piece, the required memory would be more than 15 GB for a 64-bit implementation of SNAPHU.

VII. Conclusions

Phase unwrapping is a difficult problem that is made even more difficult by the constraints of limited computational resources. Because phase unwrapping is unavoidable in many InSAR applications, however, we have proposed a tiling technique for applying SNAPHU to large datasets. This tiling technique extends the MAP framework of the primary phase-unwrapping problem to the secondary problem of defining and assembling individually unwrapped regions of the interferogram in order to form a full unwrapped solution. The MAP framework, thus, provides a theoretical foundation for the task of interferogram subdivision. In implementation, the algorithm performs well on the interferometric data examined here. More empirical tests may be needed for determining ideal tile dimensions, however. The algorithm may also benefit from new heuristics for more accurately defining reliable regions. Finally, as different unwrapping techniques are often required for different situations, we note that our secondary network model is general enough that it can be used not only within the MAP context, but with other optimization frameworks and solvers as well.

Acknowledgment

The interferometric data and reference DEM shown here were provided by the Alaska SAR Facility. Many helpful comments were provided by G. L. Tyler and by the anonymous reviewers.

References

[1] H. Zebker and R. Goldstein, "Topographic mapping from interferometric SAR observations," *J. Geophys. Res.*, vol. 91, pp. 4993–4999, 1986.

[2] A. K. Gabriel, R. M. Goldstein, and H. A. Zebker, "Mapping small elevation changes over large areas: Differential radar interferometry," *J. Geophys. Res.*, vol. 94, pp. 9183–9191, 1989.

[3] D. C. Ghiglia and M. D. Pritt, *Two-Dimensional Phase Unwrapping: Theory, Algorithms and Software*. New York: Wiley, 1998.

[4] M. Costantini, "A novel phase unwrapping method based on network programming," *IEEE Trans. Geosci. Remote Sensing*, vol. 36, pp. 813–821, May 1998.

[5] C. W. Chen and H. A. Zebker, "Two-dimensional phase unwrapping with use of statistical models for cost functions in nonlinear optimization," *J. Opt. Soc. Amer. A*, vol. 18, pp. 338–351, 2001.

[6] G. Carballo, "Statistically-based multiresolution network flow phase unwrapping for SAR interferometry," Ph.D. dissertation, Royal Inst. Technol., Stockholm, Sweden, 2000.

[7] H. A. Zebker and J. Villasenor, "Decorrelation in interferometric radar echoes," *IEEE Trans. Geosci. Remote Sensing*, vol. 30, pp. 950–959, Sept. 1992.
[8] J.-S. Lee, K. W. Hoppel, S. A. Mango, and A. R. Miller, "Intensity and phase statistics of multilook polarimetric and interferometric SAR imagery," *IEEE Trans. Geosci. Remote Sensing*, vol. 32, pp. 1017–1028, Sept. 1994.
[9] R. M. Goldstein, H. A. Zebker, and C. L. Werner, "Satellite radar interferometry: Two-dimensional-dimensional phase unwrapping," *Radio Sci.*, vol. 23, pp. 713–720, 1988.
[10] C. W. Chen and H. A. Zebker, "Network approaches to two-dimensional phase unwrapping: Intractability and two new algorithms," *J. Opt. Soc. Amer. A*, vol. 17, pp. 401–414, 2000.
[11] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
[12] D. C. Ghiglia and L. A. Romero, "Minimum L^p-norm two-dimensional phase unwrapping," *J. Opt. Soc. Amer. A*, vol. 13, pp. 1999–2013, 1996.
[13] ——, "Robust two-dimensional weighted and unweighted phase unwrapping that uses fast transforms and iterative methods," *J. Opt. Soc. Amer. A*, vol. 11, pp. 107–117, 1994.
[14] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco, CA: Freeman, 1979.
[15] A. Baldi, "Two-dimensional phase unwrapping by quad-tree decomposition," *Appl. Optics*, vol. 40, pp. 1187–1194, 2001.
[16] S. Hensley, private communication, 2001.
[17] W. Xu and I. Cumming, "A region-growing algorithm for InSAR phase unwrapping," *IEEE Trans. Geosci. Remote Sensing*, vol. 37, pp. 124–134, Jan. 1999.
[18] G. Fornaro and E. Sansosti, "A two-dimensional region growing least squares phase unwrapping algorithm for interferometric SAR processing," *IEEE Trans. Geosci. Remote Sensing*, vol. 37, pp. 2215–2226, Sept. 1999.
[19] L. Guibas and J. Stolfi, "Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams," *ACM Trans. Graphics*, vol. 4, pp. 74–123, 1985.
[20] S. Fortune, "A sweepline algorithm for Voronoi diagrams," *Algorithmica*, vol. 2, pp. 153–174, 1987.
[21] H. A. Zebker, P. A. Rosen, and S. Hensley, "Atmospheric effects in interferometric synthetic aperture radar surface deformation and topography maps," *J. Geophys. Res.*, vol. 102, pp. 7547–7563, 1997.

Curtis W. Chen received the A.B. degree in political science and the B.S., M.S., and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA in 1996, 1996, 1999, and 2001, respectively.

Since 2001, he has been with the Jet Propulsion Laboratory, Pasadena, CA. His research interests include SAR data processing and interferometry, 2-D phase unwrapping, network optimization, and algorithmic approaches to signals analysis.

Howard A. Zebker (M'87–SM'89–F'99) received the B.S. degree from the California Institute of Technology, Pasadena, in 1976, the M.S. degree from the University of California, Los Angeles, in 1979, and the Ph.D. degree from Stanford University, Stanford, CA, in 1984.

He holds a joint appointment in the Geophysics and Electrical Engineering Departments at Stanford and studies earth processes from the viewpoint of spaceborne instruments. His group is involved in basic research ranging from crustal deformation to global environmental problems, as evidenced by the flow and distribution of ice in the polar regions. The group is also developing new observational technologies such as radar interferometry. He is involved in the definition and scientific applications of new spaceborne imaging systems, especially those containing imaging radar systems.