

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/224384384>

A Particle Filter Approach for InSAR Phase Filtering and Unwrapping

Article in IEEE Transactions on Geoscience and Remote Sensing · May 2009

DOI: 10.1109/TGRS.2008.2008095 · Source: IEEE Xplore

CITATIONS

53

READS

242

3 authors, including:



T. Martinez-Marin
University of Alicante

37 PUBLICATIONS 355 CITATIONS

[SEE PROFILE](#)



Juan Manuel Lopez-Sanchez
University of Alicante

180 PUBLICATIONS 2,530 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Servicios Operacionales de Observación de la Tierra Basados en un Enfoque Dinámico para Datos Radar Multidimensionales (TIN2014-55413-C2-2-P) [View project](#)



SINCOHMAP: Exploitation of Sentinel-1 Interferometric Coherence for Land Cover and Vegetation Mapping [View project](#)

A Particle Filter Approach for InSAR Phase Filtering and Unwrapping

Juan J. Martinez-Espla, Tomás Martinez-Marin, and Juan M. Lopez-Sanchez, *Senior Member, IEEE*

Abstract—This paper presents a new phase-unwrapping (PU) algorithm for SAR interferometry that makes use of a particle filter (PF) to perform simultaneously noise filtering and PU. The formulation of this technique provides independence from noise statistics and is not constrained by the nonlinearity of the problem. In addition, an enhanced variant of this method combining a PF with artificial-intelligence search strategies and an omnidirectional local phase estimator, based on the mode of the power spectral density, is also presented. Results obtained with synthetic and real data show a significant improvement with respect to other conventional unwrapping algorithms in some situations.

Index Terms—Artificial intelligence (AI), extended Kalman filter (EKF), grid filter, particle filter (PF), path following, phase unwrapping (PU), SAR interferometry, state space.

I. INTRODUCTION

ONE OF THE key processing steps in all SAR interferometry applications is phase unwrapping (PU). A lot of literature works concerning different techniques to solve the PU problem have been published during the last years. We refer to the book by Ghiglia and Pritt [1] for an excellent overview. Traditionally, there have been two general types of conventional PU methods. The algorithms of the first group, generally named path-following or region-growing algorithms, isolate and/or mask problematic zones containing residues (we refer to [1] for the definition of residues) and unwrap the interferogram by avoiding these zones. The network cost flow algorithm by Costantini *et al.* [2] can be regarded as an extension of the classical branch-cut methods contained in this first group. The techniques of the second group provide a global solution which minimizes a cost function over the whole interferogram. Independently from this traditional classification, some techniques make use of a prefiltering stage before starting the unwrapping procedure with the filtered phase (for instance, [3]–[8]). Consequences of all these strategies are the following: Phase information in noisy pixels is not recovered by the path-following algorithms; noisy pixels distort the solution in global approaches, thus affecting the noise-free areas; and the information contained in noisy pixels is lost if a prefiltering stage is carried out.

Manuscript received April 2, 2008; revised August 12, 2008. First published February 10, 2009; current version published March 27, 2009. This work was supported in part by the Spanish Ministry of Education and Science (MEC) under Projects TEC2005-06863-C02-02 and TEC2008-06764-C02-02, and in part by the Generalitat Valenciana under Project ACOMP07/087.

The authors are with the Department of Physics, System Engineering and Signal Theory, University of Alicante, 03080 Alicante, Spain (e-mail: jjme1@alu.ua.es; tomas@dfists.ua.es; juanma-lopez@ieee.org).

Digital Object Identifier 10.1109/TGRS.2008.2008095

During the last years, a complementary group of approaches has appeared, known as “multichannel” techniques, which require multiple acquisitions (not only two images) for being applied. See, for instance, solutions presented in [9]–[13]. Multichannel techniques have the advantage to eliminate or at least mitigate the ambiguity problem related to the wrapping operator. Note that the approach proposed in this paper is devised for conventional “single-channel” interferometry, i.e., for a single interferogram obtained with only one pair of images.

In a Bayesian framework, different PU techniques are analyzed in [14], and a stochastic nonlinear filtering approach is also presented in [15]. However, Wishart nature of the noise present in the model observation is not considered, and results involving real data are not shown.

From our point of view, the key idea is to recover as much information as possible from the interferogram, including noisy pixels. Therefore, an ideal method would consider every pixel to be simultaneously unwrapped and filtered.

There exists an algorithm that shares the same objective and was already published some years ago in [16]–[20]. This approach, recently revisited in [21], combines a local phase estimator with an extended Kalman filter (EKF) in an elegant fashion to simultaneously achieve noise reduction and PU. Unfortunately, that solution is founded on the following two assumptions. First, both evolution model and measurement model are available and correspond to linear functions. Second, the noise affecting both evolution and measurement stages is Gaussian. When the constraint of Gaussian noise holds but the models are not linear functions, the EKF has been used in many cases as a good solution that approximates the nonlinearities by local linearizations. The main problem arises when the noise present in the interferogram is not Gaussian. The EKF cannot guarantee a successful PU, as we will show later with some examples. Note that a detailed tutorial about different methods to broach nonlinear/non-Gaussian problems can be consulted in [22].

In a previous work, we proposed to substitute the EKF by a grid-based filter (GbF), which is not subject to any linear or Gaussian constraints, and a good performance was demonstrated with both simulated [23] and real data [24].

In this paper, we introduce a particle-filter phase-unwrapping (PFPU) algorithm. We will show, with some examples from synthetic and real data, that this solution also performs as the GbF, but with the great advantage of a lower computational cost. The proposed method integrates a particle filter (PF) and a local phase estimator to simultaneously unwrap and filter the interferometric phase. In addition, this approach can be combined with path-following strategies to enhance their

performance with the embedded noise filtering. We present first the algorithm without any conditioned scheme for choosing the unwrapping path, showing its good performance when dealing with residues. In particular, the basic algorithm introduced in this paper runs the interferogram row by row, passing through zones of low coherence and containing residues, unwrapping and filtering the phase at the same time.

There exist also scenes that include zones of high density of residues or low coherence, which would be preferable not to unwrap until more favorable areas have already been unwrapped. Consequently, a better approach would incorporate some strategy to unwrap the best pixels first and the worst ones at the end. To broach this kind of situations, we have also developed a second improved version of the PFPU algorithm, the so-called 2PFPU, by combining a PF with an enhanced omnidirectional variant of the local phase estimator and a path-following method, based on artificial-intelligence (AI) search strategies. We refer to the book by Nilsson [25] for an excellent overview on AI search strategies.

This paper is organized as follows. Section II introduces the basic formulation of the PU problem. The PF formulation is presented in Section III. The application of PFPU and its improved version (2PFPU) is presented in Section IV. Then, Section V illustrates the performance of the new methods with some examples from both a synthetic data set extracted from [1] and real interferograms. A comparison with some conventional algorithms, namely, Goldstein branch-cut, Flynn's, and preconditioned conjugate gradient (PCG) algorithms [1], the cited EKF approach [16]–[21] and a GbF [23], [24] solution, is also discussed. Finally, conclusions are drawn in Section VI.

II. PU BASIC FORMULATION

Assuming a 1-D notation, the complex coherence between two SAR images γ is, at pixel k

$$\gamma(k) = a(k) \cdot \exp[j \cdot \tilde{\varphi}(k)] \quad (1)$$

where $a(k)$ is the observed interferometric coherence (between 0 and 1) and $\tilde{\varphi}(k)$ is the modulo 2π mapped interferometric phase, also called the wrapped phase. This phase mapping can be expressed by

$$\tilde{\varphi}(k) = [\varphi(k) + \tilde{e}_\varphi(k)]_{|2\pi} = \varphi(k) + \tilde{e}_\varphi(k) \pm n \cdot 2\pi \in (-\pi, \pi] \quad (2)$$

where $\varphi(k)$ is the true unambiguous absolute phase at pixel k and $\tilde{e}_\varphi(k)$ is the mapped phase error at pixel k .

The final objective is to obtain the unwrapped or absolute phase $\varphi(k)$ from the noisy 2π mapped phases $\tilde{\varphi}(k)$ contained in the interferogram. At this step, we will be interested in the calculus of the phase difference from one pixel to the next

$$\tilde{\Delta}_\varphi(k) = [\tilde{\varphi}(k+1) - \tilde{\varphi}(k)]_{|2\pi} \quad (3)$$

that yields

$$\tilde{\Delta}_\varphi(k) = [\delta_\varphi(k) + [\tilde{e}_\varphi(k+1) - \tilde{e}_\varphi(k)]_{|2\pi}]_{|2\pi} \quad (4)$$

where $\delta_\varphi(k)$ is the true discrete phase derivative and its modulus is always supposed to be smaller than π (i.e., there is no aliasing).

Finally, the unwrapped phase φ can be obtained through the following recursive expression:

$$\varphi(k+1) = \varphi(k) + \tilde{\Delta}_\varphi(k). \quad (5)$$

III. PARTICLE FILTER

A. Introduction

As mentioned earlier, most of the traditional PU methods suffer from the lack of filtering capacity. Kalman-filter-based methods provide filtering capacity, but they are subject to linear and Gaussian constraints. The EKF is suitable to address the linear restriction, provided that the model presents weak nonlinearities which can be linearized in a proper manner. Instead, if the model has discontinuous nonlinearities, the EKF should not be employed [26].

In the case of PU, the observation model is nonlinear with non-Gaussian statistics. Moreover, if the unwrapped phase is chosen as state variable (i.e., we employ a 1-D EKF), then the observation model is nonlinear and discontinuous (wrapped phase). In this case, the application of the EKF is not straightforward. The correct application of the EKF requires the 2-D formulation proposed in [16]–[21], where the observation model contains real and imaginary parts of the complex interferogram [21]. Thus, the model remains nonlinear but continuous. A more complete introduction to the use of this type of approaches for PU can be consulted in [21].

In contrast to EKF, the application of a PF solution is more straightforward, since it exhibits the advantage of using only one dimension per pixel (directly the phase), hence bringing down complexity to its simplest case, and it is not subject to any linear or Gaussian constraints.

B. PF Formulation

Hereinafter, to be coherent with the notation of state-space methods, the unwrapped phase at pixel k , $\varphi(k)$, will be referred to as the corresponding state x_k at pixel k .

The suboptimal solution of a PF [22] can be adopted, since the continuous state space can be divided into N cells or states, $\{x_k^i; i = 1, \dots, N\}$, which correspond to all possible values of the phase. The resulting grid must be sufficiently dense to get a good approximation to the continuous state space.

Conceptually, the PF is simply a solution that represents the posterior probability function (*pdf*) $p(x_k|z_{1:k})$, by a distribution of N_s particles, defined hereinafter, extracted from the state space. The objective is to calculate estimates of the state x_k based on the set of all available observations $z_{1:k} = \{z_i, i = 1, \dots, k\}$ up to pixel k .

The particles are defined as the set of states whose weights are the highest for each step. These particles, or selected group of samples, and their weights become the starting point for the next step of the algorithm. The basic stages at each step of the algorithm are represented in Fig. 1.

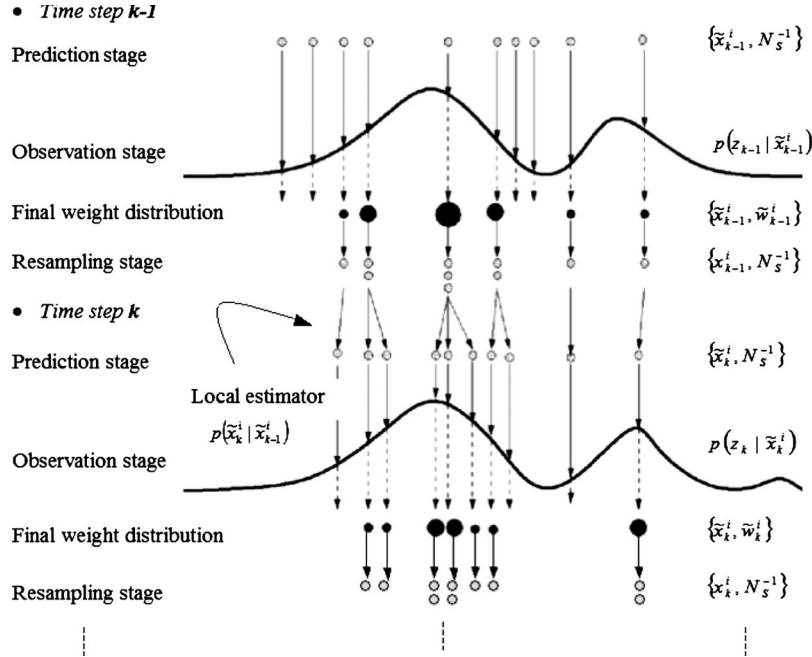


Fig. 1. Illustration of the stages of the PF algorithm for a pixel.

Considering the Markovian assumption, the pdf $p(x_k | z_{1:k})$ can be obtained recursively from the pdf $p(x_{k-1} | z_{1:k-1})$ calculated at previous pixel $k-1$. It is supposed that the initial pdf $p(x_0 | z_0) \equiv p(x_0)$ is known. If it is not the case, the samples can be initially distributed uniformly over the state space. Next, at the present pixel k , the N_s samples are updated according to the action taken and the current observation z_k .

At first sight, this sequential problem could be divided into two stages: *prediction* and *update*. However, as it will be explained in the following, a third stage, named *resampling*, needs to be added. To broach prediction and update stages, it is assumed that evolution model and observation model exist and are known.

The prediction stage is solved by applying the evolution model $p(x_k | x_{k-1})$ to each one of the N_s particles, generating a new set of samples. These samples represent the prediction of the state variable x_k without bearing in mind the observation at pixel k .

To consider the current observation z_k , the update stage is performed. The weights associated with each sample, $w_k^i; i = 1, \dots, N_s$, are obtained according to the observation model $p(z_k | x_k)$.

As it was introduced before, a third stage named *resampling* needs to be added in order to improve the performance of the filter. Let $\{x_{0:k}^i, w_k^i\}_{i=1}^{N_s}$ denote a random measure to characterize the posterior pdf $p(x_{0:k} | z_{1:k})$, where $\{x_{0:k}^i; i = 1, \dots, N_s\}$ is a set of support points with associated weights $\{w_k^i; i = 1, \dots, N_s\}$, and $x_{0:k} = \{x_j; j = 0, \dots, k\}$ is the set of all states up to pixel k . The weights are normalized, $\sum_i w_k^i = 1$. Then, the posterior pdf at pixel k can be approximated as

$$p(x_{0:k} | z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(x_{0:k} - x_{0:k}^i) \quad (6)$$

where $\delta(\cdot)$ represents the Dirac delta.

Let $x_i, i = 1, \dots, N_s$ be samples easily generated from a function $q(\cdot)$ called *importance density*. The weights are chosen by means of the *importance resampling* principle, which relies on the following. Suppose that $p(x) \propto \pi(x)$ is a probability density from which it is not easy to draw samples but for which $\pi(x)$ can be evaluated (as well as $p(x)$ up to proportionality). In this context, a weighted approximation to $p(\cdot)$ is

$$p(x) \approx \sum_{i=1}^{N_s} w^i \delta(x - x^i) \quad (7)$$

where

$$w_k^i \propto \frac{\pi(x^i)}{q(x^i)} \quad (8)$$

represents the normalized weight of the i th particle.

If the samples $x_{0:k}^i$ are drawn by using an importance density, the weights are then defined by

$$w_k^i \propto \frac{p(x_{0:k}^i | z_{1:k})}{q(x_{0:k}^i | z_{1:k})}. \quad (9)$$

As presented in [22], if the importance density is chosen to factorize such that

$$q(x_{0:k} | z_{1:k}) = q(x_k | x_{0:k-1}, z_{1:k}) q(x_{0:k-1} | z_{1:k-1}) \quad (10)$$

then samples $x_{0:k}^i \sim q(x_{0:k} | z_{1:k})$ can be obtained by augmenting each of the existing samples $x_{0:k-1}^i \sim q(x_{0:k-1} | z_{1:k-1})$ with the new state $x_k^i \sim q(x_k | x_{0:k-1}, z_{1:k})$.

Concerning $p(x_{0:k} | z_{1:k})$, after a few operations [22], it can be shown that

$$p(x_{0:k} | z_{1:k}) \propto p(z_k | x_k) p(x_k | x_{k-1}) p(x_{0:k-1} | z_{1:k-1}). \quad (11)$$

Therefore, by substituting (10) and (11) into (9)

$$\begin{aligned} w_k^i &\propto \frac{p(z_k|x_k^i) p(x_k^i|x_{k-1}^i) p(x_{0:k-1}^i|z_{1:k-1})}{q(x_k^i|x_{0:k-1}^i, z_{1:k}) q(x_{0:k-1}^i|z_{1:k-1})} \\ &= w_{k-1}^i \frac{p(z_k|x_k^i) p(x_k^i|x_{k-1}^i)}{q(x_k^i|x_{0:k-1}^i, z_{1:k})}. \end{aligned} \quad (12)$$

Furthermore, if we consider that $q(x_k|x_{0:k-1}, z_{1:k}) = q(x_k|x_{k-1}, z_k)$, then only x_k^i and the current observation z_k need to be stored. The weight is then

$$w_k^i \propto w_{k-1}^i \frac{p(z_k|x_k^i) p(x_k^i|x_{k-1}^i)}{q(x_k^i|x_{k-1}^i, z_k)}. \quad (13)$$

The posterior density of probability is given by

$$p(x_k|z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(x_k - x_k^i) \quad (14)$$

where the weights are defined in (13). It can be shown that the larger the N_s , the more (14) approaches the true *pdf* $p(x_k|z_{1:k})$.

Finally, it is convenient to choose the importance density to be

$$q(x_k|x_{k-1}^i, z_k) = p(x_k|x_{k-1}^i). \quad (15)$$

This selection simplifies the calculus of the weights as follows:

$$w_k^i \propto w_{k-1}^i p(z_k|x_k^i). \quad (16)$$

The graphical evolution of the samples at each stage of a standard PF algorithm is also shown in Fig. 1.

It is assumed that the samples estimate only one parameter, distributed over the horizontal axis. The samples are represented by circles centered on the value of the parameter that they represent. The area of the circles represents the weight of the corresponding sample. In this example, the algorithm starts at pixel $k-1$ with an unweighted distribution $\{\tilde{x}_{k-1}^i, N_s^{-1}\}$. It provides an approximation of $p(x_{k-1}|z_{1:k-2})$. For each particle, its associated weight is computed by using the information provided by the observation model $p(z_{k-1}|\tilde{x}_{k-1}^i)$ at pixel $k-1$. It results in the weighted measure $\{\tilde{x}_{k-1}^i, \tilde{w}_{k-1}^i\}$, which yields an approximation of $p(x_{k-1}|z_{1:k-1})$. Finally, the resampling step, which will be explained hereinafter, selects only the “fittest” particles to obtain the unweighted measure $\{x_{k-1}^i, N_s^{-1}\}$, which is still an approximation of $p(x_{k-1}|z_{1:k-1})$. The next step of the algorithm starts with the evolution stage, and information from the local phase estimator $p(\tilde{x}_k^i|\tilde{x}_{k-1}^i)$ is added at this moment. The evolution stage introduces a prediction and its uncertainty, yielding the measure $\{\tilde{x}_k^i, N_s^{-1}\}$, which is an approximation of $p(x_k|z_{1:k-1})$. The sequence of the algorithm is the same for the rest of subsequent pixels.

An *a priori* drawback of PF methods could be the constraint of a finite state space, but this is not the case in the interferometric PU problem. The PF solution that we propose here makes

```

 $\{x_k^i, w_k^i\}_{i=1}^{N_s} = PF \left[ \{x_{k-1}^i, w_{k-1}^i\}_{i=1}^{N_s}, z_k \right]$ 
FOR  $i = 1, \dots, N_s$ :
  - Draw  $x_k^i \sim q(x_k|x_{k-1}^i, z_k)$ 
  - Assign the particle a weight,  $w_k^i$ , according to (13)
END FOR

- Calculate total weight  $t = SUM \left[ \{w_k^i\}_{i=1}^{N_s} \right]$ 

FOR  $i = 1, \dots, N_s$ :
  - Normalize  $w_k^i = w_k^i / t$ 
END FOR

```

Fig. 2. Pseudocode of a generic PF.

use of a sliding window W_k^s to cover the complete state space. This window is 2π wide and is centered in the previous value for the unwrapped phase, given by PF_{mx} , introduced in the following. It is based on the assumption that the phase difference between two contiguous pixels must be contained in the interval $(-\pi, +\pi]$. In this way, this *sliding state space* allows us to afford the interferometric PU problem independently of the width of the complete state space. Note that computational cost increases linearly with N_s .

A common and critical problem affecting PFs is the degeneracy phenomenon. It means that after a few steps of the algorithm, all but one particle will have negligible weight. This problem implies that a considerable computational work will be dedicated to updating particles whose contribution to the approximation of the posterior (*pdf*) $p(x_k|z_{1:k})$ is nearly zero. Therefore, degeneracy is an undesirable issue that should be avoided. The first idea to remove this effect could be the brute force, i.e., the use of a very large N_s , which is normally infeasible and inefficient. However, there exist some other solutions, like resampling or good choice of the importance density [22]. Resampling will be the one to be used with the PF presented in this paper due to its simplicity and excellent results. More details of the resampling are provided in next section. The pseudocode of a generic PF is shown in Fig. 2. Note that the sliding window update and resampling stages have been omitted and will be introduced as part of the PU algorithm.

C. Resampling

The pseudocode of the resampling algorithm is shown in Fig. 3. This method allows the reduction of the effects of degeneracy. The basic idea of resampling is to drop particles that have small weights and to concentrate on those with large weights. A new set of samples is generated by resampling the set of samples and taking out, with replacement, N_s samples from the current set, proportionally to their weights. In this new set, for instance, the samples with the lowest probabilities will disappear. Next, the weights associated with the samples are scaled in order to represent the probability associated with each sample. The resulting set of samples is in fact an i.i.d. sample from the discrete posterior probability function $p(x_k|z_{1:k})$. Therefore, the weights can now be reset to $w_k^i = 1/N_s$.

```

 $\left[ \{x_k^j, w_k^j\}_{j=1}^{N_s} \right] = \text{Resample} \left[ \{x_k^i, w_k^i\}_{i=1}^{N_s} \right]$ 
- Initialize the CWS*:  $c_1 = 0$ 
FOR  $i = 2, \dots, N_s$ 
  - Build CWS:  $c_i = c_{i-1} + w_k^i$ 
END FOR
- Start at the bottom of CWS:  $i = l$ 
- Draw a starting point:  $u_l \sim U[0, 1/N_s]$ 
FOR  $j = l, \dots, N_s$ 
  - Move along the CWS:  $u_j = u_l + (j-l) * (1/N_s)$ 
  WHILE  $u_j > c_i$ 
     $i = i + 1$ 
  END WHILE
  - Assign sample:  $x_k^j = x_k^i$ 
  - Assign weight:  $w_k^j = 1/N_s$ 
END FOR
* Cumulative Weight Sum (CWS) vector

```

Fig. 3. Pseudocode of the resampling algorithm.

IV. PARTICLE FILTER PHASE UNWRAPPING

A. Introduction

The unidimensional PU problem will be addressed first for a better process understanding. In this case, the probabilities $p(x_k^i | x_{k-1}^i)$ and $p(z_k | x_k^i)$ in (13) are defined by the evolution and observation models at pixel k , given by

$$x_k^i = x_{k-1}^i + \Delta \hat{\varphi}_{k-1} + n_{k-1} \quad (17)$$

$$z_k = x_k^i + v_k \quad (18)$$

where $\Delta \hat{\varphi}_{k-1}$ represents a local phase-slope estimate, while n_k and v_k are supposed to be approximate models for the noise present in both evolution and observation stages, respectively.

Several techniques can be used to obtain an estimation of the phase slope. For instance, the mode of the power spectral density was used in [16] and [17] due to its supposed insensitiveness and robustness to superimposed white noise, whereas a phase-average-based method was used in [27], and a matrix pencil approach was proposed in [28]. All of them are perfectly applicable. The first one has been selected in this case for the sake of comparison.

It is well known that the estimated covariance matrix for correlated SAR images is statistically described by the complex Wishart distribution, with the distribution of the interferometric phase being a marginal density distribution of the Wishart one [29], [30]. Therefore, noise v_k has been modeled as a discrete marginal Wishart distribution. It is important to note that this modelization is not strictly necessary, since this approach can work with any distribution or probability density function, known or not, thus providing independence from noise statistics. When the statistics is known *a priori*, the algorithm can use its analytical expression, thus making the computations easier. When the statistics is not available, the algorithm can calculate it.

Concerning n_k , it models the uncertainty introduced by the local phase estimator (based on the power spectral density

- Initial weight distribution: $w_{o,i}^i$, $i=1, \dots, N_s$. An initial state x_o is selected.

FOR $k = l, \dots, T$

- Phase slope estimations $\Delta \hat{\varphi}_H$ and $\Delta \hat{\varphi}_V$

$$\left[\{x_k^i, w_k^i\}_{i=1}^{N_s} \right] = PF \left[\{x_{k-1}^i, w_{k-1}^i\}_{i=1}^{N_s}, z_k \right]$$

- Unwrapped phase:

- Cell whose weight is maximum: $x_k = x_k^i |_{\max(w_k^i)}$

- Sliding window update: \bar{W}_k

- Resample particles using algorithm *Resampling*:

$$* \left[\{x_k^j, w_k^j\}_{j=1}^{N_s} \right] = \text{Resample} \left[\{x_k^i, w_k^i\}_{i=1}^{N_s} \right]$$

END FOR

Fig. 4. Pseudocode of the proposed 2-D PU algorithm using a PF (PFPU).

in this paper). It has been modeled as a Gaussian discrete distribution for the sake of comparison with [16]–[21].

B. PFPU Algorithm

To apply the PFPU to 2-D interferograms, any prediction estimate will be calculated, depending on two neighbors. Hence, the stage concerning the evolution model (17) needs to be modified as follows:

$$x_k^i \sim \frac{1}{2} \cdot [p_H(x_k^i | x_{k-1}^i) + p_V(x_k^i | x_{k-m}^i)] \quad (19)$$

where $p_H(x_k | x_{k-1}^i)$ is the evolution model in the horizontal direction at pixel $k-1$, given by

$$x_k^i = x_{k-1}^i + \Delta \hat{\varphi}_{k-1}^H + n_{k-1}^H \quad (20)$$

where x_{k-1}^i refers to the unwrapped phase at pixel $k-1$, $\Delta \hat{\varphi}_{k-1}^H$ refers to the horizontal local phase estimation at pixel $k-1$, and n_{k-1}^H refers to the Gaussian noise sample at pixel $k-1$ that models the uncertainty introduced by the local phase estimator. Note that pixel $k-1$ is the previously unwrapped one, placed in the same row and adjacent to the current pixel k .

On the other hand, $p_V(x_k | x_{k-m}^i)$ is the evolution model in the vertical direction, at pixel $k-m$, given by

$$x_k^i = x_{k-m}^i + \Delta \hat{\varphi}_{k-m}^V + r_{k-m}^V \quad (21)$$

where x_{k-m}^i refers to the unwrapped phase at pixel $k-m$, $\Delta \hat{\varphi}_{k-m}^V$ refers to the vertical local phase estimation at pixel $k-m$, and r_{k-m}^V refers to the Gaussian noise sample at pixel $k-m$ that models the uncertainty introduced by the local phase estimator. Note that pixel $k-m$ is the previously unwrapped one, placed in the same column and adjacent to the current pixel k .

The pseudocode of the 2-D PFPU algorithm is shown in Fig. 4.

C. 2PFPU Algorithm

As it will be shown later with some examples, the row-by-row PFPU solution presented in the previous section (see Fig. 5)

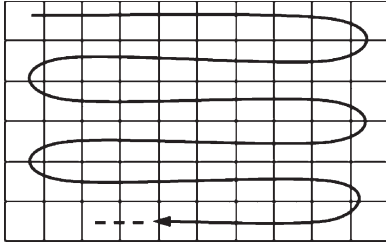


Fig. 5. Path of the row-by-row PFP.

performs generally better than conventional solutions when working with sparse distributions of residues, showing a good recovery from errors induced by them.

However, in many occasions, zones containing high density of residues or low coherence have to be processed and unwrapped. In this kind of situations, some added strategy to know the best path to follow for unwrapping would be much appreciated. As outlined in Section I, the PFP algorithm can be combined with path-following approaches to enhance the performance of these well-known techniques with the noise filtering incorporated by the PF.

The so-called 2PFP solution presented here combines a PF with a path-following method, making use of AI search strategies based on the use of cost functions and an enhanced omnidirectional variant of the local phase estimator. Both enhanced features are described in the following.

AI search strategies are efficient and provide the optimal solution. In this paper, the optimal solution will be the path that maximizes its associated cost function. It works as follows: At each step, the pixel to be unwrapped will be the one with the highest associated cost function. The cost function will be computed as the weighted summation of the coherence of pixel k and the number of unwrapped neighbors. To provide the solution with at least the same reliability as the PF method presented in Section IV-B, two or more unwrapped neighbors should be available. The cost function f_c for pixel k is then expressed as

$$f_c = c_1 \cdot COH + c_2 \cdot UNW \quad (22)$$

where c_1 and c_2 are adjustable weights, COH represents the coherence of pixel k , and UNW denotes the number of unwrapped neighbors. Note that cost function is updated per pixel and per step. Coherence for each pixel has been estimated with a 5×5 -pixel window.

The local phase estimator presented so far, based on the mode of the power spectral density, provides slope estimations in horizontal and vertical directions. This method has been improved in this paper to provide slope estimations also in oblique directions. To do so, two types of 5×5 windows around the pixel under study have been used simultaneously. These two windows are shown in Fig. 6. They have been separated in the picture for the sake of clarity. However, actually, they work overlapped and centered on the same pixel. The concepts explained in [16] and [17] can now be applied over both windows. As a result, omnidirectional slope information is supplied. We have named this enhanced variant as the omnidirectional local phase estimator. The PFP presented in Section IV-B only uses

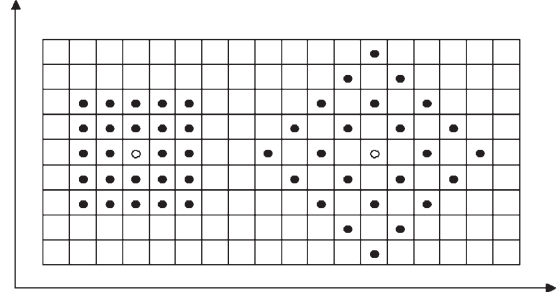


Fig. 6. Windows used by the omnidirectional local phase estimator for computing the spectral mode.

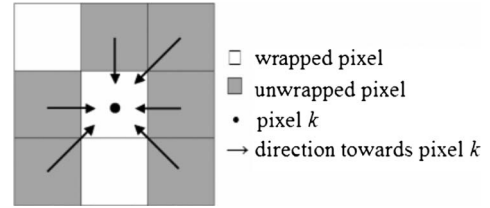


Fig. 7. Example of omnidirectional local phase estimation.

the slope estimations from two contiguous neighbors. On the contrary, the 2PFP solution presented here is able to collect slope information from up to the eight neighbors, providing estimations with higher reliability.

As a consequence of the new information supplied to the algorithm, the stage concerning the evolution model (17) needs to be modified as follows:

$$x_k^i \sim \frac{1}{N_n} \sum_{m=1}^{N_n} p_m(x_k^i | x_m^i) \quad (23)$$

where N_n is the total number of unwrapped neighbors used for the prediction and $p_j(x_k^i | x_m^i)$ is the evolution model to be applied in the direction from pixel m toward pixel k , given by

$$x_k^i = x_m^i + \Delta \hat{\varphi}_m^j + n_m^j \quad (24)$$

where x_m^i refers to the unwrapped phase at pixel m , $\Delta \hat{\varphi}_m^j$ refers to the slope estimation at pixel m in the j -direction (the one toward pixel k), and n_m^j refers to the Gaussian noise sample at pixel m that models the uncertainty introduced by the slope estimation at pixel m in the j -direction. One example of such a situation is shown in Fig. 7. In this case, with $N_n = 6$, the evolution model from each unwrapped neighbor (pixel m) to the pixel k will be applied according to (24). Finally, the composition of all the estimations is computed by using (23).

The necessity of saving and updating a list containing the final particle distribution at every unwrapped pixel that still has a neighbor to unwrap entails an additional computational cost. However, as we will show later with some examples, this path selection and noise filtering combination improves importantly the performance of the algorithm and enables the solution of more complicated situations. The pseudocode shown in Fig. 4 also applies to 2PFP with the difference that phase slopes are estimated according to (24).

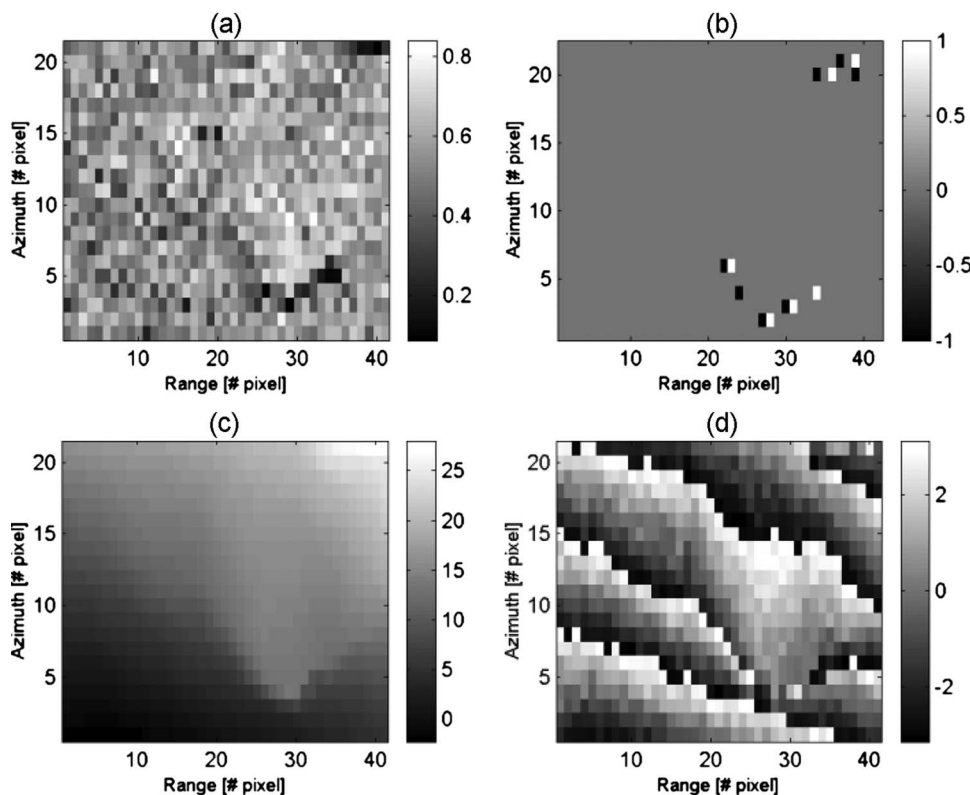


Fig. 8. Longs Peak synthetic interferogram. (a) Coherence. (b) Residues. (c) True absolute unwrapped phase. (d) Noisy input signal (wrapped phase).

V. RESULTS

The results in this section illustrate the performance of different PU solutions as well as the influence of the noisy input signal. The following PU solutions are compared: the branch-cut algorithm proposed by Goldstein [1], Flynn's algorithm [1], the PCG [1], the EKF algorithm described in [16]–[21], two approaches of the GbF proposed in [23] and [24], the row-by-row PF solution (PFPU) proposed in this paper (see Figs. 4 and 5), and the improved 2PFPU algorithm. The two GbF options consist of selecting the state whose weight is the maximum GbF_{mx} or selecting the average state of the distribution GbF_{av} . Algorithms have been applied to different synthetic and real scenarios. Note that all the phase results are presented in radians.

1) *Synthetic Data Over Longs Peak (Colorado) [1]*: We have applied all, but the 2PFPU algorithm, to one example that most of the methods introduced in [1] fail to unwrap: a portion of a synthetic interferogram from a mountainous terrain around Longs Peak, Colorado (Fig. 8). The data set, as detailed in [1], was generated with a high-fidelity InSAR simulator by employing digital terrain elevation models from the U.S. Geological Survey. We have also chosen this synthetic example because the true unwrapped phase is available, so one can compare it with the phase delivered by the PU algorithms.

In the study area, there exist zones where the coherence (i.e., the quality) of the data is very poor. For instance, in the surroundings of rows 3–5 and columns 25–35, it reaches values lower than 0.3 [see Fig. 8(a)]. As a consequence, residues appear [see Fig. 8(b)]. We will illustrate the following examples by showing the final unwrapped phase provided by the PU

algorithms, which can be compared against the true absolute unwrapped phase represented in Fig. 8(c). In addition, the result of wrapping the solution (rewrapped phase) will be analyzed for detecting potential errors introduced by the PU algorithms with respect to the original wrapped phase (i.e., the input signal), which is shown in Fig. 8(d).

The PU algorithm proposed by Goldstein [1] identifies the residues and defines the so-called branch cuts between them, which will not be crossed by the unwrapping paths. This method attempts to minimize the branch-cut lengths. As a result, many areas can be completely isolated and, consequently, not unwrapped consistently with the rest of the interferogram. This is the case shown in Fig. 9.

The results obtained by applying Flynn's algorithm and the PCG method [1] are shown in Figs. 10 and 11, respectively. It can be observed how both methods also fail to unwrap correctly this synthetic data set.

As observed in Fig. 12, the EKF [16]–[21] fails as a consequence of a concentration of residues around the pixel (3, 27). It produces errors in the slope estimates and in the unwrapped phase in this area, thus generating an error which propagates in a diagonal fashion and producing an erroneous phase discontinuity.

In contrast, as shown in Fig. 13 for the same test interferogram, the PFPU solution does not suffer from this error propagation, since the PF is able to fix this problem. Although more expensive than other techniques, the computational cost of the PFPU was only a few seconds to unwrap this area, which is much shorter than the analogous GbF solutions as a consequence of using particles instead of a complete grid of states.

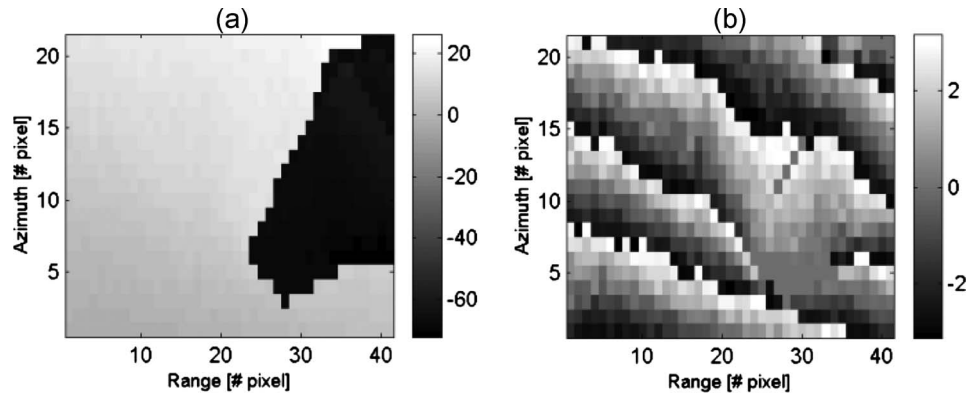


Fig. 9. Goldstein solution for Longs Peak synthetic interferogram. (a) Unwrapped phase. (b) Rewrapped phase.

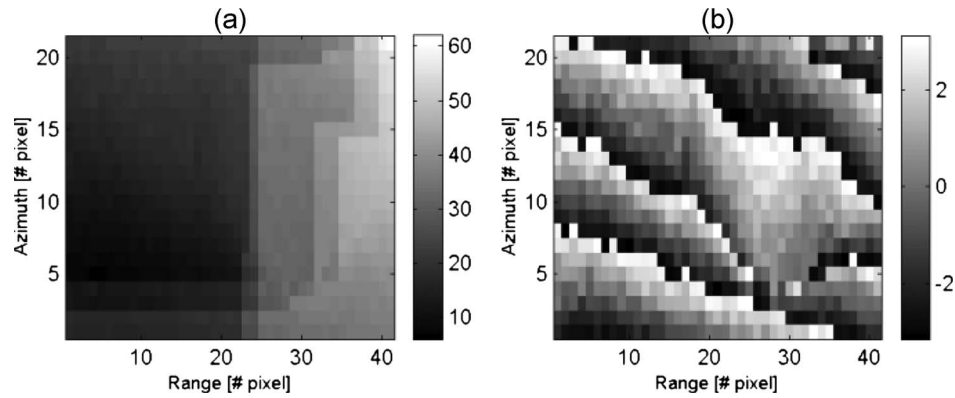


Fig. 10. Flynn solution for Longs Peak synthetic interferogram. (a) Unwrapped phase. (b) Rewrapped phase.

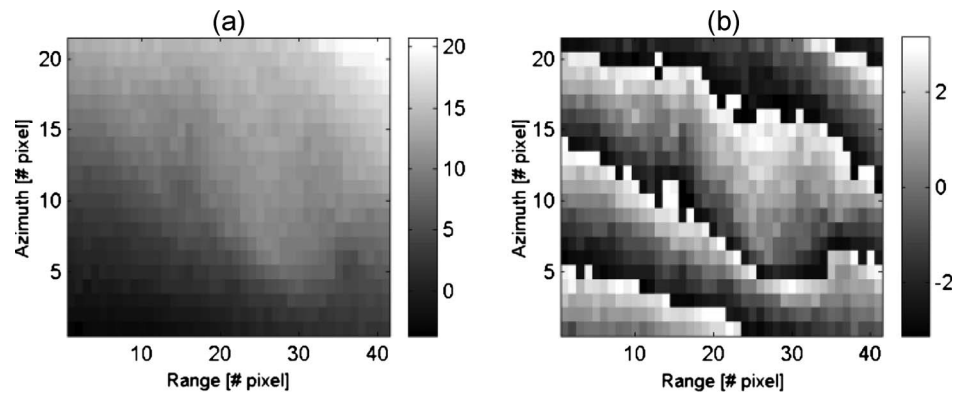


Fig. 11. PCG solution for Longs Peak synthetic interferogram. (a) Unwrapped phase. (b) Rewrapped phase.

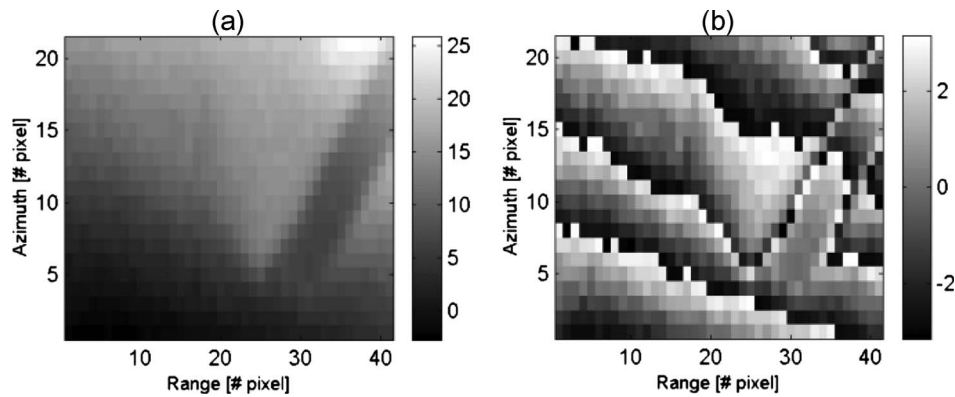


Fig. 12. EKF solution for Longs Peak synthetic interferogram. (a) Unwrapped phase. (b) Rewrapped phase.

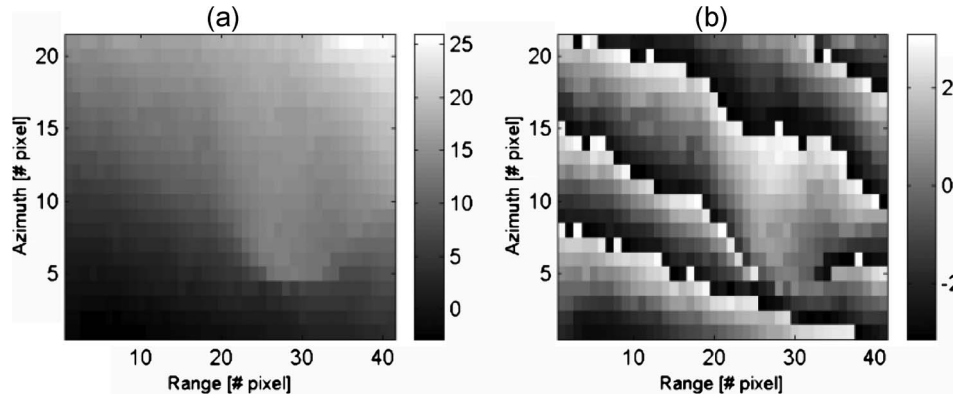


Fig. 13. PFPU solution for Longs Peak synthetic interferogram. (a) $PFPU_{\max}$ unwrapped phase. (b) $PFPU_{\max}$ rewrapped phase. The continuous space contained inside the 2π sliding window has been translated into a discrete state space composed of $N = 100$ cells, and a maximum of $N_s = 50$ particles will be used.

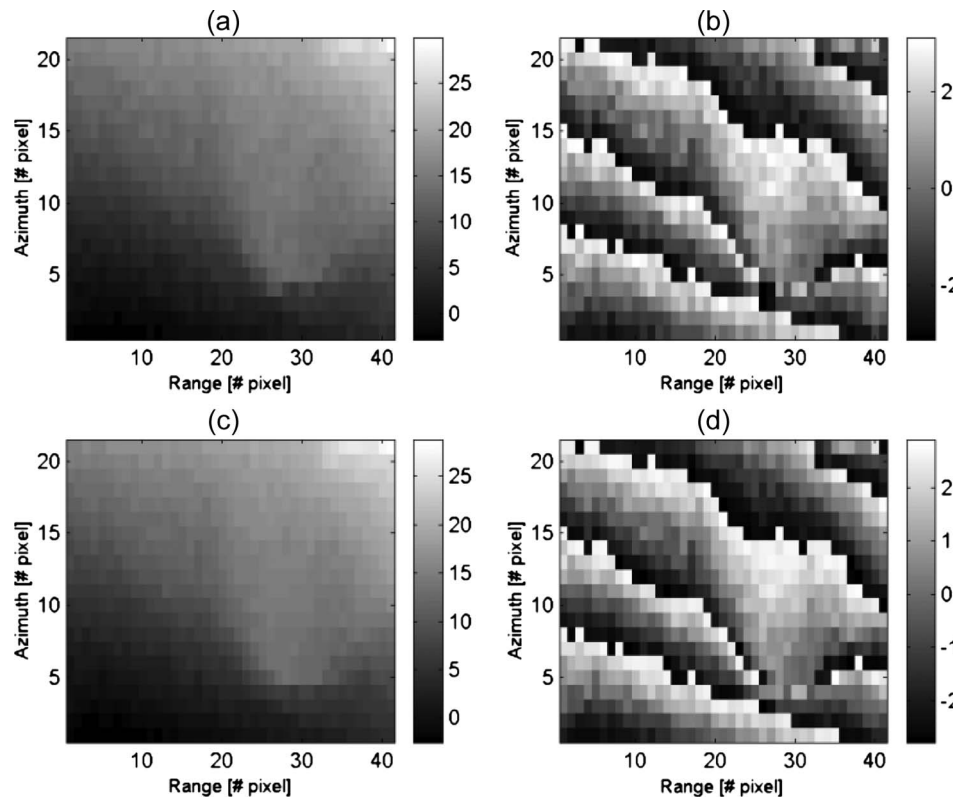


Fig. 14. GbF solution for Longs Peak synthetic interferogram. (a) GbF_{\max} unwrapped phase. (b) GbF_{\max} rewrapped phase. (c) GbF_{av} unwrapped phase. (d) GbF_{av} rewrapped phase. The continuous space contained inside the 2π sliding window has been translated into a discrete state space composed of $N = 100$ cells.

The results provided by both GbF solutions, GbF_{\max} and GbF_{av} , are also shown in Fig. 14. As expected, the results are quite similar to the ones obtained by the PFPU, but they have been obtained with a 35% higher computational effort.

2) *Small Area of a Real ERS-1 and ERS-2 Tandem Interferogram Over Alicante Province:* The following results correspond to a small crop extracted from a real interferogram obtained with images acquired by the ERS-1 and ERS-2 satellites in tandem configuration. The same algorithms as in the previous example have been compared. Coherence and input wrapped phase are shown in Fig. 15. As observed in Fig. 16, the EKF fails in two different areas. First, a concentration of residues around the pixel (40, 80) produces errors in the slope estimates and in the unwrapped phase in this area. Second,

the conjunction of low coherence and high variance of the local phase estimators in pixel (7, 70) produces an error which propagates in a diagonal fashion, generating an erroneous phase discontinuity.

For this example, the rest of tested algorithms show similar results (see Figs. 17–20). The main differences appear for Goldstein's and the PCG solution. Goldstein's algorithm failed to unwrap a small group of pixels around (40, 80), since they appear completely isolated as a result of the combination of the branch cuts [see Fig. 17(a)]. In the case of PCG, by inspecting the rewrapped phase in Fig. 19(b), it can be observed that part of a phase cycle has been lost for a group of pixels around (40, 80). This problem is common in minimum-norm methods. On the other hand, the solution provided by the PFPU

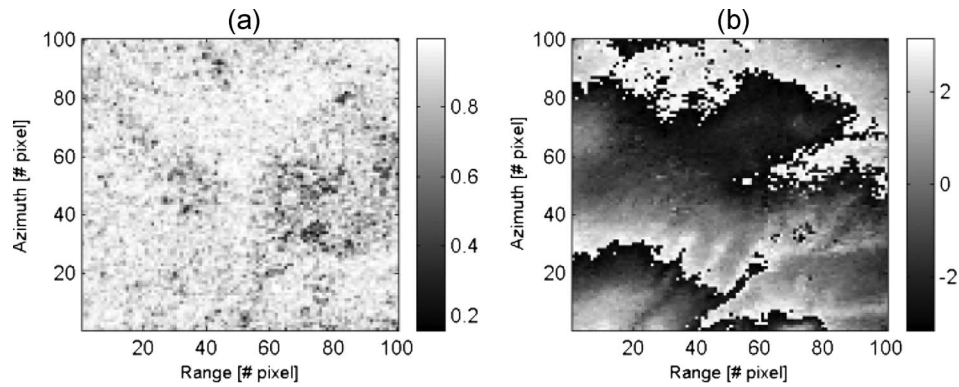


Fig. 15. Small real interferogram with ERS images. (a) Coherence. (b) Input wrapped phase.

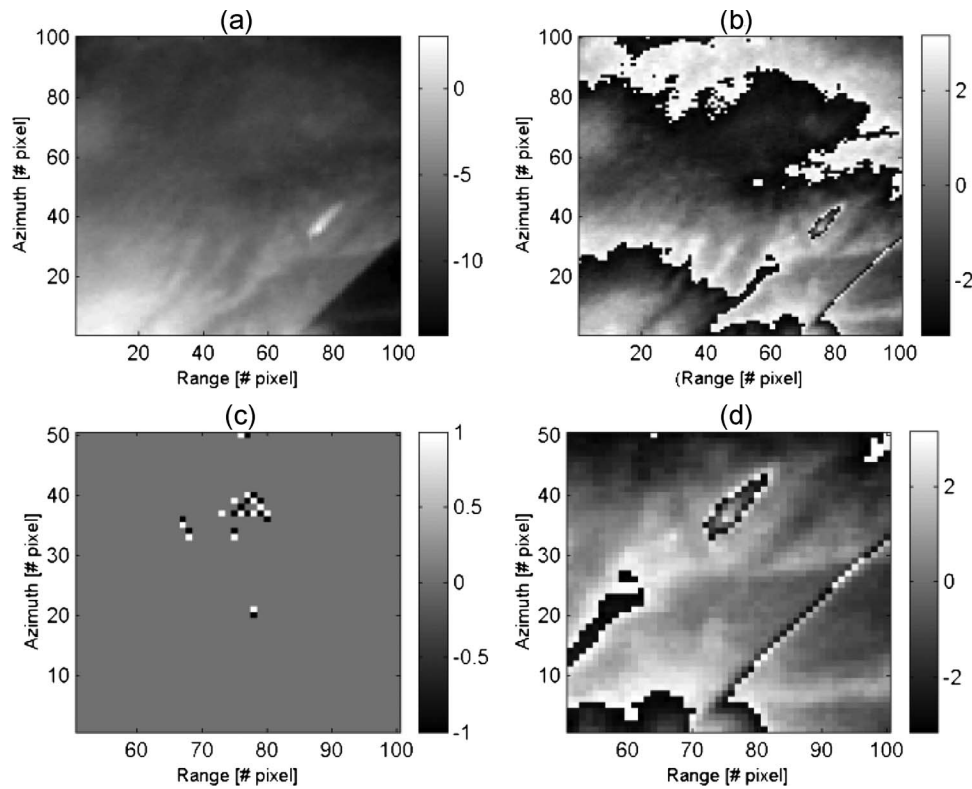


Fig. 16. EKF solution for the small ERS interferogram. (a) Unwrapped phase. (b) Rewrapped phase. (c) Zoom of residues. (d) Zoom of rewrapped phase.

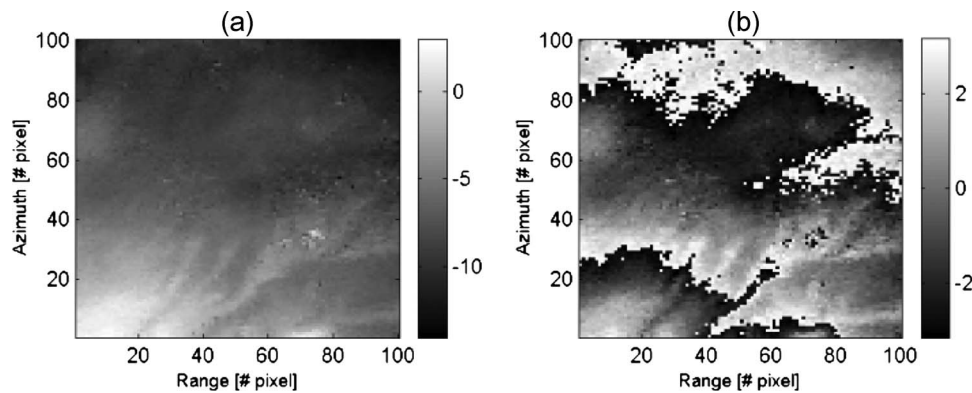


Fig. 17. Goldstein solution for the small ERS interferogram. (a) Unwrapped phase. (b) Rewrapped phase.

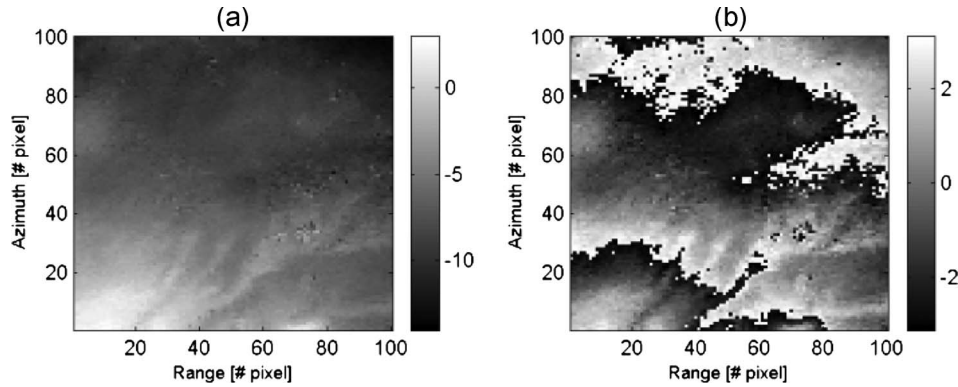


Fig. 18. Flynn solution for the small ERS interferogram. (a) Unwrapped phase. (b) Rewrapped phase.

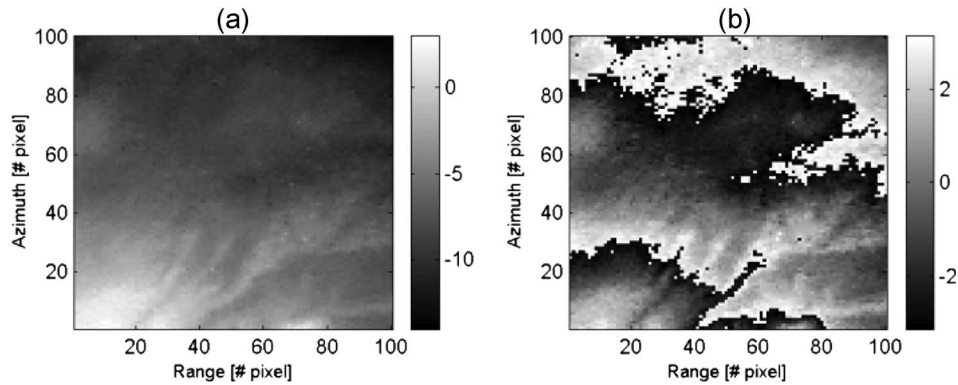


Fig. 19. PCG solution for the small ERS interferogram. (a) Unwrapped phase. (b) Rewrapped phase.

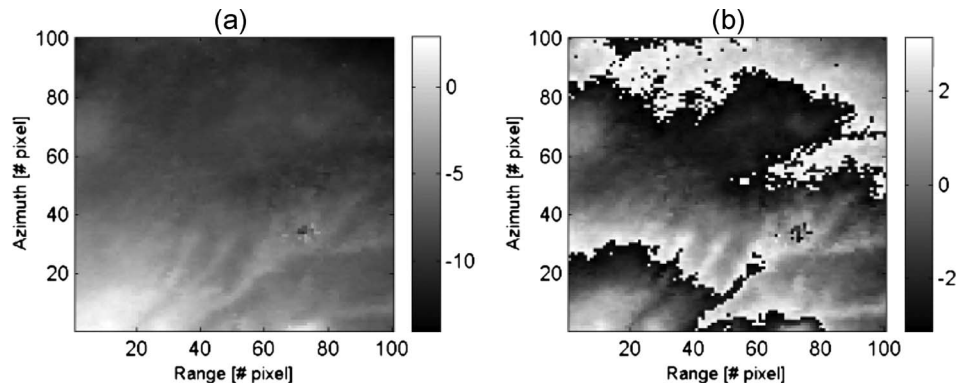


Fig. 20. PFPU solution ($PFPU_{\max}$) for the small ERS interferogram. (a) Unwrapped phase. (b) Rewrapped phase. The continuous space contained inside the 2π sliding window has been translated into a discrete state space composed of $N = 100$ cells, and a maximum of $N_s = 50$ particles will be used.

is smoother than that of Goldstein and Flynn solutions in the upper part of the interferogram due to the filtering capability of this approach.

3) *X-SAR Interferogram Over Mount Etna*: A simple row-by-row PF algorithm (PFPU) has been used so far for computational cost reasons. It has been shown how it can deal with residues and even recover from errors. However, there exist many situations in which zones of very low coherence or with a high concentration of residues are present. In these cases, it would be preferable to unwrap them the last. To do so, an enhanced algorithm (2PFPU) has been introduced in Section IV-C, combining a path-following technique with the PF solution. The evident drawback of this method is that the computational cost grows from the necessity of storing

the particle distribution at each unwrapped pixel that still has any unwrapped neighbors. However, filtering and PU capacities are more powerful than in the row-by-row approach, since the phase is unwrapped moving from the higher quality zones to the lower ones. Just to give an approximate comparison about the relative computational cost of these approaches, it took only a few seconds to process the interferogram by using the Goldstein algorithm, 1 min by using the PCG, nearly half an hour by means of Flynn's algorithm, and nearly 8 h by means of the 2PFPU method.

The data shown in Figs. 21 and 22 are the coherence and input wrapped phase, respectively, extracted from a real interferogram formed by a pair of SLC images with 1718×1992 pixels, acquired by the X-SAR mission over Mount Etna. There exist

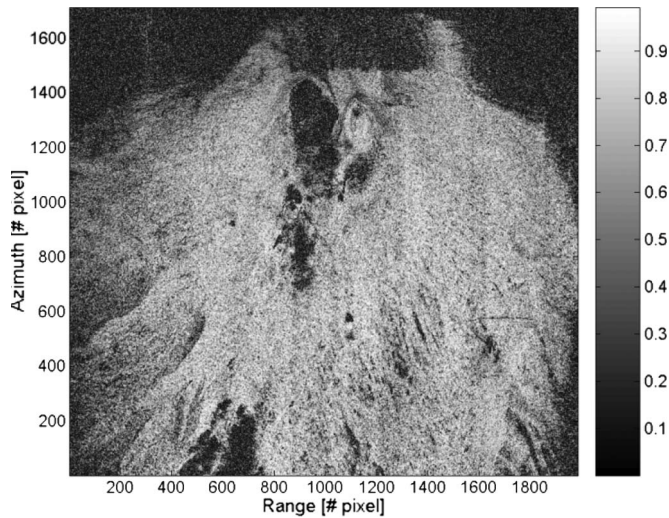


Fig. 21. Coherence of the X-SAR interferogram over Mount Etna.

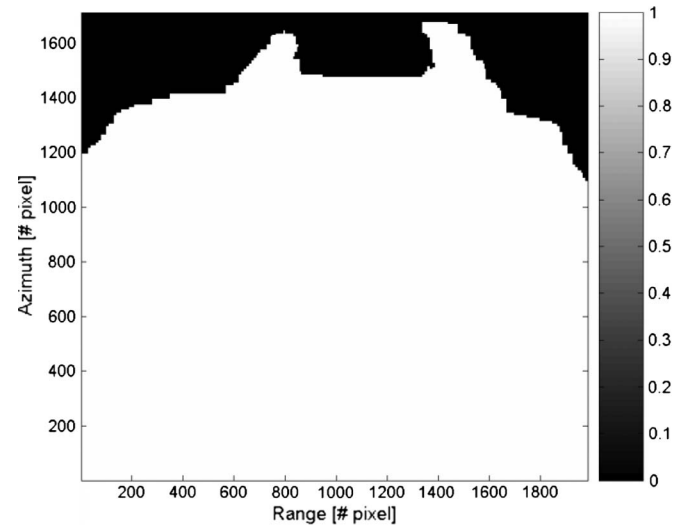


Fig. 23. Mask employed for unwrapping the X-SAR interferogram over Mount Etna. Black areas will not be unwrapped.

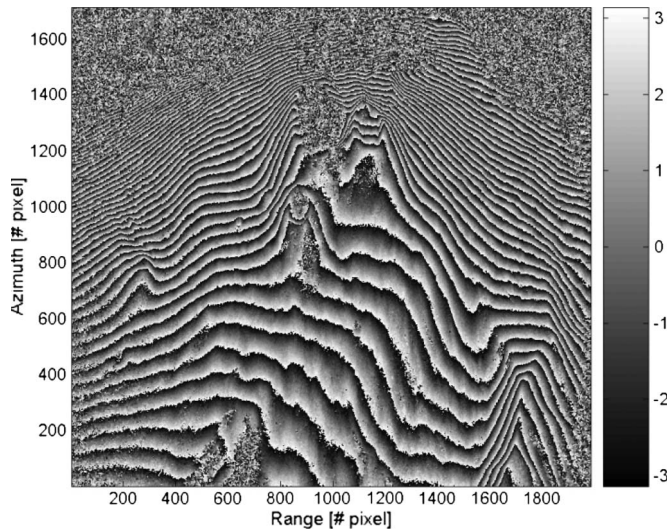


Fig. 22. Input wrapped phase of the X-SAR interferogram over Mount Etna.

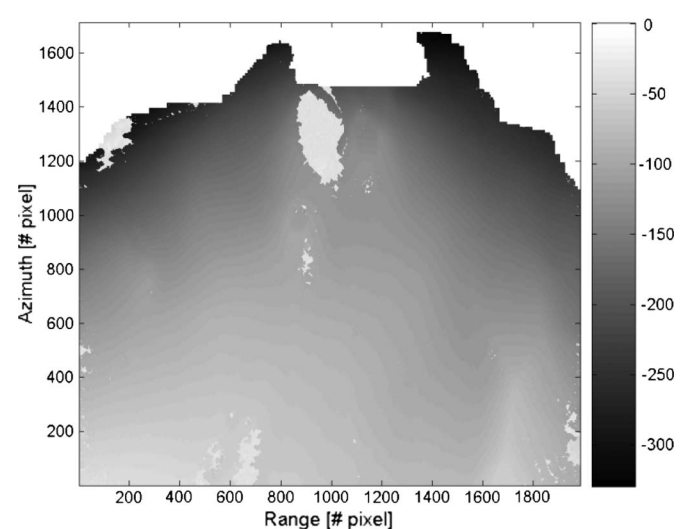


Fig. 24. Goldstein solution for the X-SAR interferogram over Mount Etna: Unwrapped phase.

some areas where the phase information is incoherent, one of them corresponding to the sea (at the top) and others probably due to the presence of vegetation and to shadow areas (as in the volcano crater). We have masked out the sea region but not the other low-quality areas. The mask employed in the subsequent unwrapping is shown in Fig. 23. Black areas will not be unwrapped.

The 2PFPU algorithm will be compared to Goldstein and Flynn solutions, since they share the path-following concept, and also to the PCG solution, as a comparison with minimum-norm methods. In some cases, an additional test has been included here. The difference between the unwrapped phase by each algorithm and the original interferogram has been computed, and the result has been wrapped again. This process should result in a noisy constant, and no bias should appear even for large-size images.

Traditional path-following techniques, Goldstein and Flynn (see Figs. 24 and 25), fail partially to unwrap this interferogram because there appear some isolated regions in the final solution.

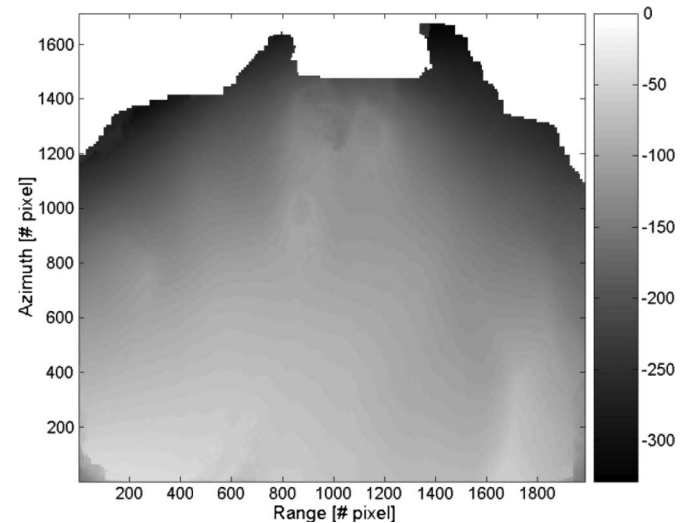


Fig. 25. Flynn solution for the X-SAR interferogram over Mount Etna: Unwrapped phase.

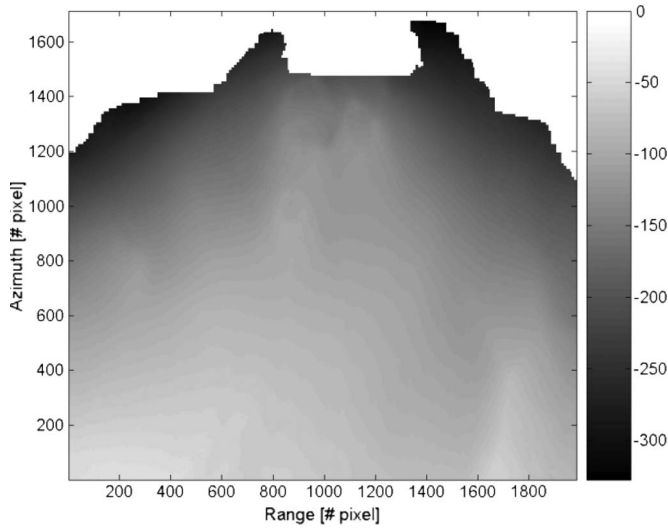


Fig. 26. PCG solution for the X-SAR interferogram over Mount Etna: Unwrapped phase.

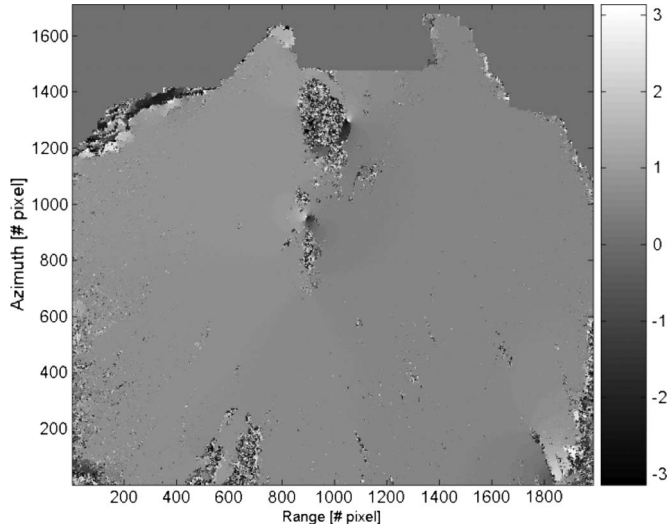


Fig. 27. PCG solution for the X-SAR interferogram over Mount Etna: Wrapped difference between the unwrapped phase and the input wrapped phase.

On the other hand, the PCG approach results in a smooth solution (see Fig. 26) without isolated regions, as expected. In this case, the quality of the solution can be tested by observing the difference of the unwrapped and the original phase, which is shown in Fig. 27. Other than the expected noisy areas, there appear several discontinuities or fringes in this image, so the final solution is not satisfactory.

Finally, Figs. 28 and 29 show the capacities for the enhanced algorithm 2PFPU, combining a path-following technique with a PF and the omnidirectional phase estimator. Fig. 28 shows the unwrapped phase. The quality of this solution is shown by Fig. 29, since the wrapped difference between the solution and the original interferogram exhibits a noisy constant aspect, and no bias has appeared even for this large interferogram. Note that the noisy aspect is due to the filtering incorporated in this technique.

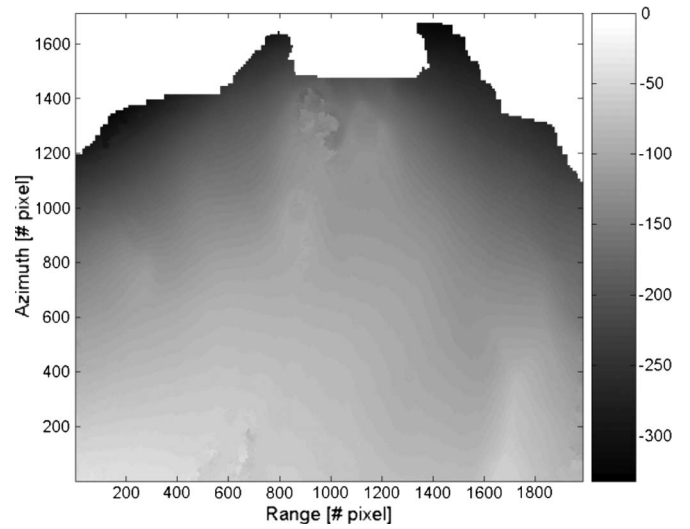


Fig. 28. 2PFPU solution for the X-SAR interferogram over Mount Etna: Unwrapped phase. The continuous space contained inside the 2π sliding window has been translated into a discrete state space composed of $N = 100$ cells, and a maximum of $N_s = 50$ particles will be used.

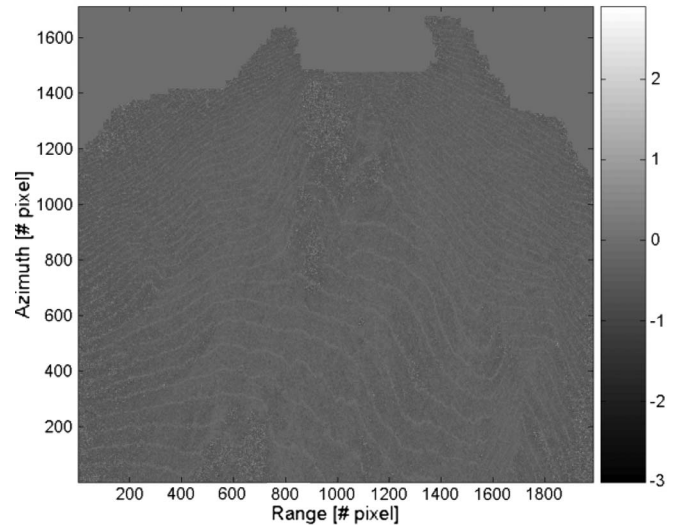


Fig. 29. 2PFPU solution for the X-SAR interferogram over Mount Etna: Wrapped difference between the unwrapped phase and the input wrapped phase.

VI. CONCLUSION

A new solution that simultaneously filters and unwraps the phase contained in an interferogram is presented in this paper.

This solution is based on a PF. Two variants of this method (PFPU and 2PFPU) have been implemented: The first one is a simple row-by-row unwrapping algorithm, and the second one is the result of combining a PF with AI search strategies and an enhanced version of the local phase estimator introduced in [16]–[21]. These algorithms have been compared to some conventional methods like the branch-cut algorithm proposed by Goldstein, Flynn's algorithm, and PCG. This solution performs better in some cases, since PF can deal with zones containing residues.

PF-based solutions have also been compared to the EKF method and the GbF algorithm, which share the same

philosophy of simultaneously filtering and unwrapping. For some situations which the EKF fails to unwrap, a PF solution has shown a better performance and the capacity to recover from errors. It has also been shown that PF solutions perform better than the grid-based ones, since they obtain the same results with the advantage of a lower computational cost.

Our research lines at present and in the near future are focused on the following two main issues: first, utilization of other local phase estimators, for instance, the matrix pencil approach proposed in [28], and second, the combination of the 2PPFU approach with region-growing techniques will be analyzed, since a better performance is expected from the synergy between both strategies.

ACKNOWLEDGMENT

The authors would like to thank Dr. K. P. Papathanassiou (DLR) and Dr. C. Lopez-Martinez (UPC) for providing access to the X-SAR images of Mount Etna. The ERS images used in this paper have been provided by the European Space Agency (ESA) in the framework of the ESA EO Project Cat.1-2494.

REFERENCES

- [1] D. C. Ghiglia and M. D. Pritt, *Two-Dimensional Phase Unwrapping. Theory, Algorithms and Software*. New York: Wiley, 1998.
- [2] M. Costantini, A. Farina, and F. Zirielli, "A fast phase unwrapping algorithm for SAR interferometry," *IEEE Trans. Geosci. Remote Sens.*, vol. 37, no. 1, pp. 452–460, Jan. 1999.
- [3] D. Meng, V. Sethu, E. Ambikairajah, and L. Ge, "A novel technique for noise reduction in InSAR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 2, pp. 226–230, Apr. 2007.
- [4] R. Yamaki and A. Hirose, "Singularity-spreading phase unwrapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 10, pp. 3240–3251, Oct. 2007.
- [5] J.-S. Lee, K. P. Papathanassiou, T. L. Ainsworth, M. R. Grunes, and A. Reigber, "A new technique for noise filtering of SAR interferometric phase images," *IEEE Trans. Geosci. Remote Sens.*, vol. 36, no. 5, pp. 1456–1465, Sep. 1998.
- [6] N. Wu, D.-Z. Feng, and J. Li, "A locally adaptive filter of interferometric phase images," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 73–77, Jan. 2006.
- [7] A. B. Suksmono and A. Hirose, "Adaptive noise reduction of InSAR images based on a complex-valued MRF model and its applications to phase unwrapping problem," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 3, pp. 699–709, Mar. 2002.
- [8] Q. Yu, X. Yang, S. Fu, X. Liu, and X. Sun, "An adaptive contoured window filter for interferometric synthetic aperture radar," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 1, pp. 23–26, Jan. 2007.
- [9] R. Lanari, G. Fornaro, D. Riccio, M. Migliaccio, K. Papathanassiou, J. Moreira, M. Schwabisch, L. Dutra, G. Puglisi, G. Franceschetti, and M. Coltelli, "Generation of digital elevation models by using SIR-C/X-SAR multifrequency two-pass interferometry: The Etna case study," *IEEE Trans. Geosci. Remote Sens.*, vol. 34, no. 5, pp. 1097–1114, Sep. 1996.
- [10] M. G. Kim and H. D. Griffiths, "Phase unwrapping of multibaseline interferometry using Kalman filtering," in *Proc. Image Process. Appl., Conf. Publication No. 465*, 1999, pp. 813–817.
- [11] A. Ferretti, A. Monti Guarnieri, C. Prati, and F. Rocca, "Multi baseline interferometric techniques and applications," in *Proc. ESA Workshop Appl. ERS SAR Interferometry (FRINGE)*, Zurich, Switzerland, Oct. 1996. [Online]. Available: <http://www.geo.unizh.ch/rs/FRINGE96/papers/ferretti-et-al/>
- [12] G. Ferraiuolo, V. Pascasio, and G. Schirinz, "Maximum a posteriori estimation of height profiles in InSAR imaging," *IEEE Geosci. Remote Sens. Lett.*, vol. 1, no. 2, pp. 66–70, Apr. 2004.
- [13] G. Fornaro, A. Pauciuolo, and E. Sansosti, "Phase difference-based multi-channel phase unwrapping," *IEEE Trans. Image Process.*, vol. 14, no. 7, pp. 960–972, Jul. 2005.
- [14] G. Nico, G. Palubinskas, and M. Datcu, "Bayesian approaches to phase unwrapping: Theoretical study," *IEEE Trans. Signal Process.*, vol. 48, no. 9, pp. 2545–2556, Sep. 2000.
- [15] J. Leitão and M. Figueiredo, "Absolute phase image reconstruction: A stochastic nonlinear filtering approach," *IEEE Trans. Image Process.*, vol. 7, no. 6, pp. 868–882, Jun. 1998.
- [16] R. Krämer and O. Löffeld, "Presentation of an improved Phase Unwrapping Algorithm based on Kalman filters combined with local slope estimation," in *Proc. ESA Workshop Appl. ERS SAR Interferometry (FRINGE)*, Zurich, Switzerland, Oct. 1996. [Online]. Available: www.geo.unizh.ch/rs/FRINGE96
- [17] O. Löffeld and R. Krämer, "Phase unwrapping for SAR interferometry—A data fusion approach by Kalman filtering," in *Proc. IEEE IGARSS*, Hamburg, Germany, 1999, pp. 1715–1717.
- [18] O. Löffeld and R. Krämer, "Phase unwrapping for SAR interferometry," in *Proc. IEEE IGARSS*, Pasadena, CA, 1994, pp. 2282–2284.
- [19] O. Löffeld, C. Arndt, and A. Hein, "Estimating the derivative of modulo-mapped phases," in *Proc. ESA Workshop Appl. ERS SAR Interferometry (FRINGE)*, Zurich, Switzerland, Oct. 1996. [Online]. Available: www.geo.unizh.ch/rs/FRINGE96
- [20] O. Löffeld, "Demodulation of noisy phase or frequency modulated signals with Kalman filters," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Adelaide, Australia, 1994, pp. IV-177–IV-180.
- [21] O. Löffeld, H. Nies, S. Knedlik, and W. Yu, "Phase unwrapping for SAR interferometry—A data fusion approach by Kalman filtering," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 1, pp. 47–58, Jan. 2008.
- [22] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.
- [23] J. J. Martinez-Espla, T. Martinez-Marin, and J. M. Lopez-Sanchez, "Using a grid-based filter to solve InSAR phase unwrapping," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 2, pp. 147–151, Apr. 2008.
- [24] J. J. Martinez-Espla, T. Martinez-Marin, and J. M. Lopez-Sanchez, "Introduction of a grid-based filter approach for InSAR phase filtering and unwrapping," in *Proc. IEEE IGARSS*, Barcelona, Spain, Jul. 2007, pp. 4497–4500.
- [25] N. J. Nilsson, *Artificial Intelligence: A New Synthesis*. San Mateo, CA: Morgan Kaufmann, 1998.
- [26] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Norwood, MA: Artech House, 2004.
- [27] W. Xu and I. Cumming, "A region-growing algorithm for InSAR phase unwrapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 37, no. 1, pp. 124–134, Jan. 1999.
- [28] G. Nico and J. Fortuny, "Using the matrix pencil method to solve phase unwrapping," *IEEE Trans. Signal Process.*, vol. 51, no. 3, pp. 886–888, Mar. 2003.
- [29] J. S. Lee, K. W. Hoppel, S. Mango, and A. Miller, "Intensity and phase statistics of multilook polarimetric and interferometric SAR imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 32, no. 5, pp. 1017–1028, Sep. 1994.
- [30] R. J. A. Tough, D. Blacknell, and S. Quegan, "A statistical description of polarimetric and interferometric synthetic aperture radar data," *Proc. R. Soc. Lond. A, Math. Phys. Sci.*, vol. 449, pp. 567–589, 1995.



Juan J. Martinez-Espla was born in Alicante, Spain, in 1973. He received the M.S. degree in telecommunication engineering from the Technical University of Valencia (UPV), Valencia, Spain, in 2000. He is currently working toward the Ph.D. degree in the Department of Physics, System Engineering and Signal Theory, University of Alicante, Alicante, Spain.

Since 2003, he has been bringing his international experience as a Project and Product Manager in telecommunications business, in addition to his research at the Signals, Systems and Telecommunication Group, University of Alicante. His research topics include microwave remote sensing for digital elevation models and advanced state-space techniques for noise reduction, filtering, and phase unwrapping with application in SAR interferometry.



Tomás Martínez-Marin received the B.S. degree in telecommunication engineering from the University of Alcalá, Alcalá de Henares, Spain, in 1990 and the M.S. and Ph.D. degrees in telecommunication engineering from the Technical University of Madrid (UPM), Madrid, Spain, in 1995 and 1999, respectively.

He was with the University of Alcalá as an Assistant Professor in 1990. In 1997, he was with the European University of Madrid (UEM), Madrid, Spain, as an Assistant Professor. Since 2000, he has been with the Department of Physics, System Engineering and Signal Theory, University of Alicante, Alicante, Spain, where he is currently an Associate Professor. His research interests include reinforcement learning, optimal control, intelligent vehicles, and SAR image processing.



Juan M. Lopez-Sanchez (S'94–M'00–SM'05) was born in Alicante, Spain, in 1972. He received the M.S. and Ph.D. degrees in telecommunication engineering from the Technical University of Valencia (UPV), Valencia, Spain, in 1996 and 2000, respectively.

From 1998 to 1999, he was a Predoctoral Grant Holder with the Space Applications Institute, Joint Research Centre, European Commission, Ispra, Italy. Since 2000, he has been leading the Signals, Systems and Telecommunication Group, University of

Alicante, Spain, where he is currently an Associate Professor with the Department of Physics, System Engineering and Signal Theory. His research topics include analytical and numerical models for electromagnetic scattering problems, microwave remote sensing for inversion of biophysical parameters, advanced polarimetric and interferometric techniques, and SAR imaging algorithms. He has coauthored more than 20 papers in international journals and 50 contributions to scientific conferences and symposia.

Dr. Lopez-Sanchez received the INDRA Award for the best Ph.D. thesis about radar in Spain in 2001.