

1 *Computers and Geosciences*
2 Supplementary Information for

3 Small baseline InSAR time series analysis: Unwrapping error
4 correction and noise reduction

5
6 Zhang Yunjun^a, Heresh Fattah^b, Falk Amelung^a
7

8 ^a Rosenstiel School of Marine and Atmospheric Science, University of Miami, Miami, Florida, USA

9 ^b Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California, USA

10 **Content of this file**

11 Section S1. Supplemental figure S1 to S9 and table S1.

12 Section S2. Design matrices.

13 Section S3. Decorrelation noise simulation.

14 Section S4. Additional software features

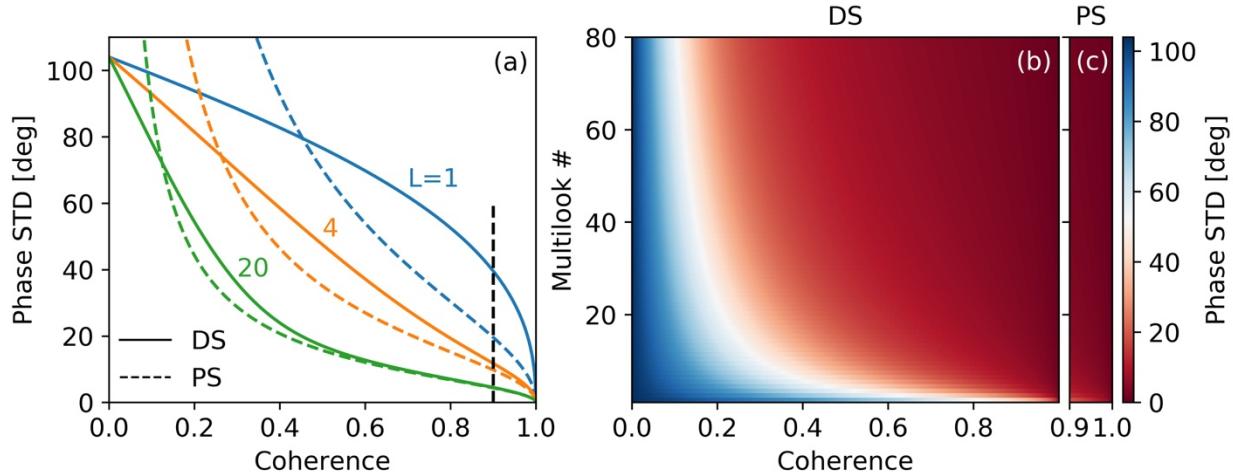
15 Supplemental references.

16 **S1. Supplemental figures and tables**

17 This section provides figures S1 to S9 and table S1. Fig. S1 shows the standard deviation of the
18 interferometric phase as a function of the spatial coherence and number of looks. Fig. S2
19 demonstrates the performance of four weighting functions in different temporal decorrelation
20 settings using the mean RMSE of 10,000 realizations of the inverted phase time-series as a
21 function of the number of looks. Fig. S3 demonstrates the simulation of the unwrapped

22 interferogram for unwrapping error correction with the bridging method, considering the ground
23 deformation, tropospheric turbulence, phase ramps and decorrelation noise. Fig. S4 shows the
24 output percentage of interferograms with unwrapping errors as a function of the LASSO
25 parameter to find its suitable value range. Fig. S5 demonstrates the necessity of adding the step
26 function during the topographic residual correction in the presence of displacement jump using
27 both simulated and real data. Fig. S6 shows the coherence matrix of Sentinel-1 dataset for GPS
28 stations within Sierra Negra. Fig. S7 shows the estimated residual phase time-series. Fig. S8
29 shows the coherence-based network modification for the Sentinel-1 data used in the discussion
30 of the network redundancy in section 6.3. Fig. S9 compares the displacement time-series from
31 the approaches in GIAnT and MintPy with and without unwrapping error correction and
32 weighted network inversion. Table S1 summarizes the information of SAR data used in the paper
33 and their configurations for InSAR stack processing.

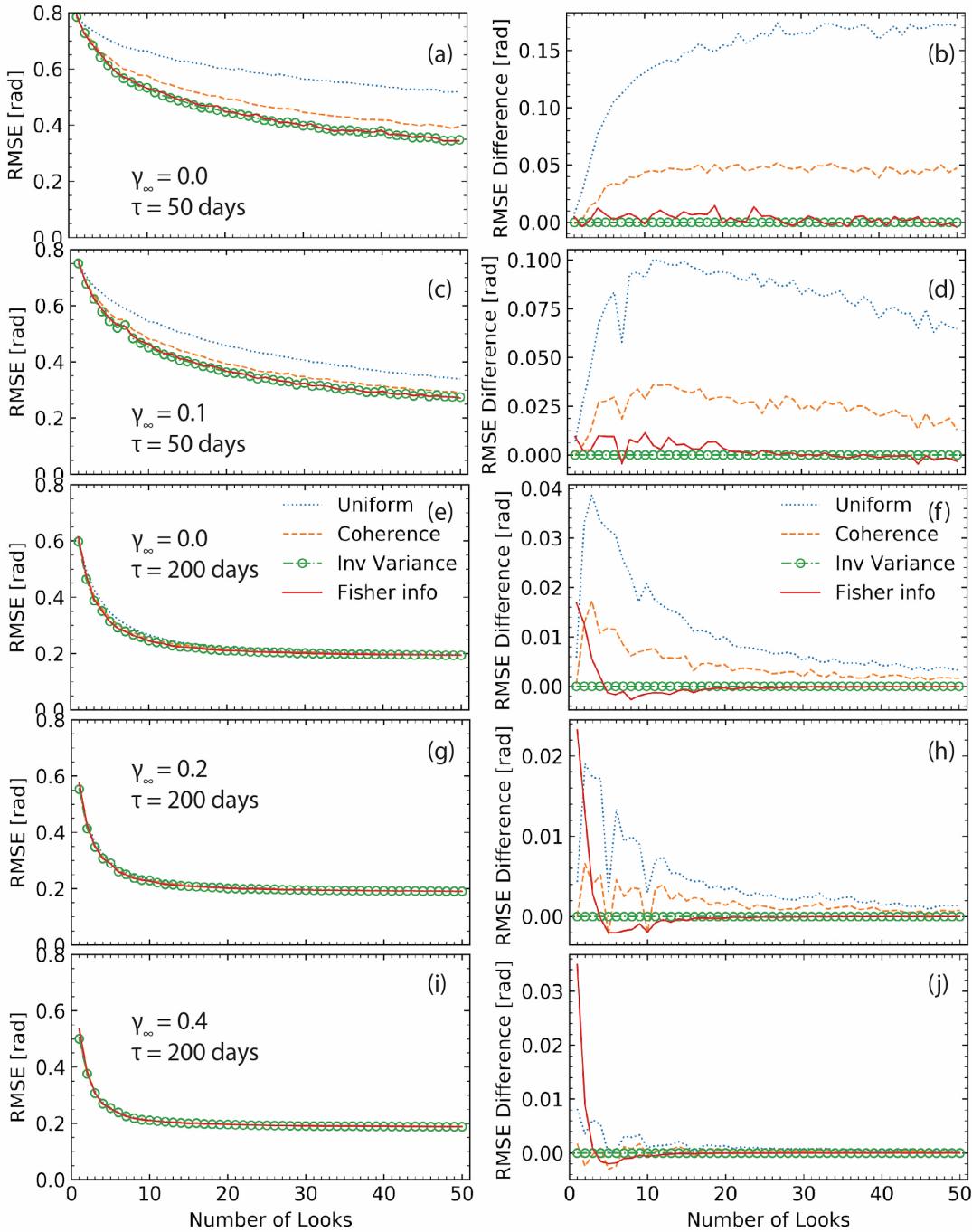
34



35

36 **Figure S1.** Phase standard deviation versus spatial coherence for PS and DS. Related to equation
 37 (6). (a) Standard deviation of interferometric phase as function of coherence for DS (solid lines)
 38 and PS (dashed lines) with 1, 4 and 20 looks. The black dashed line marks the effective boundary
 39 for PS ($0.9 < |\gamma| \leq 1$). (b) Lookup table to convert spatial coherence to phase standard deviation
 40 for number of looks in [1, 80].

41



42

43

44

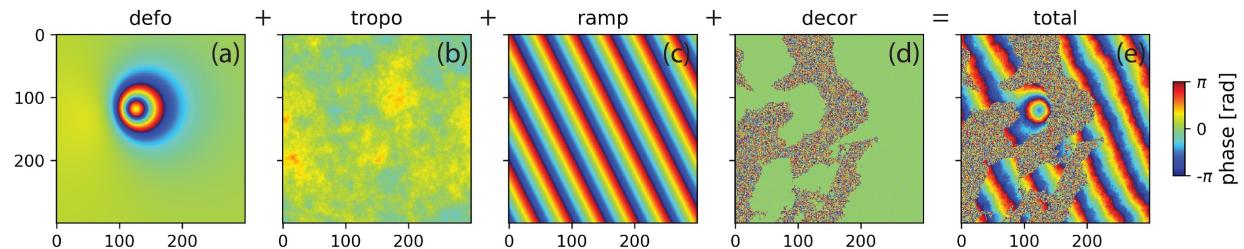
45

46

47

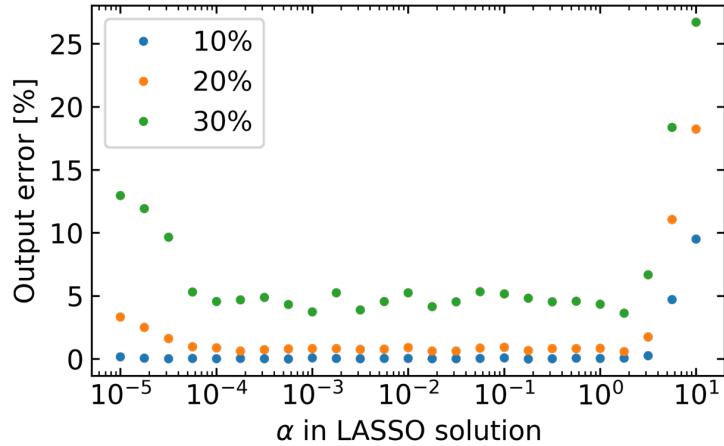
Figure S2. Performance indicator for four weighting functions based on (left panel) the mean RMSE of 10,000 realizations of inverted phase time-series as a function of the number of looks. Related to Fig. 1, which uses $\gamma_\infty = 0.0$ and $\tau = 200$ days. Right panel: same as left panel but shown in differential RMSE with respect to inverse-variance weighting. From top to bottom for different temporal decorrelation settings.

48

49 **Figure S3.** Simulate interferogram for unwrapping error correction with the bridging method.

50 Related to Fig. 2. We consider an area of 300 by 300 pixels with spatial resolution of 62 m in
 51 both directions, illustrated by radar echoes in a Sentinel-1-like geometry in descending orbit
 52 (with an incidence angle of 34 deg and heading angle of -168 deg). (a) Deformation phase
 53 caused by a Mogi source ($x = 120$ row, $y = 120$ col, $z = 2$ km under the free surface with a
 54 volume change of 10^6 m^3), (b) tropospheric turbulence modeled as an isotropic two-dimensional
 55 surface with a power law behavior (the multiplier of spectrum amplitude $p0=1\text{e-}3$, assuming a
 56 flat area without stratified tropospheric delay; Hanssen, 2001), (c) phase ramp modeled as a
 57 linear surface, and (d) simulated decorrelation noise (see section S3). The water body mask is
 58 rescaled from the real DEM in western Kyushu, Japan. We specify the spatial coherence of 0.6
 59 and 0.001 for pixels on land and water respectively with the number of looks of 15 by 5.

60



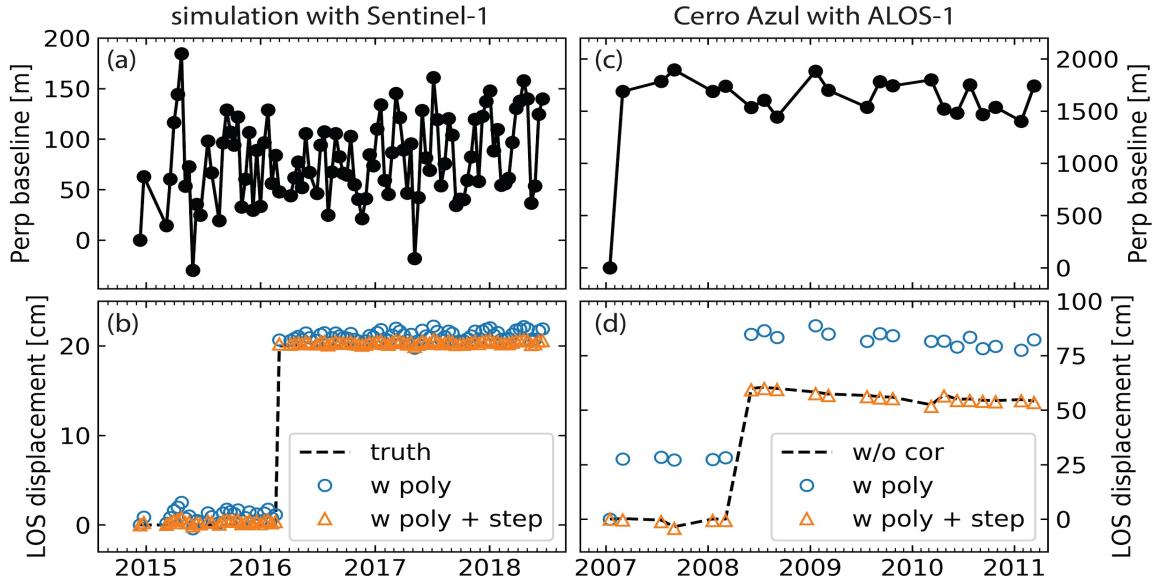
61

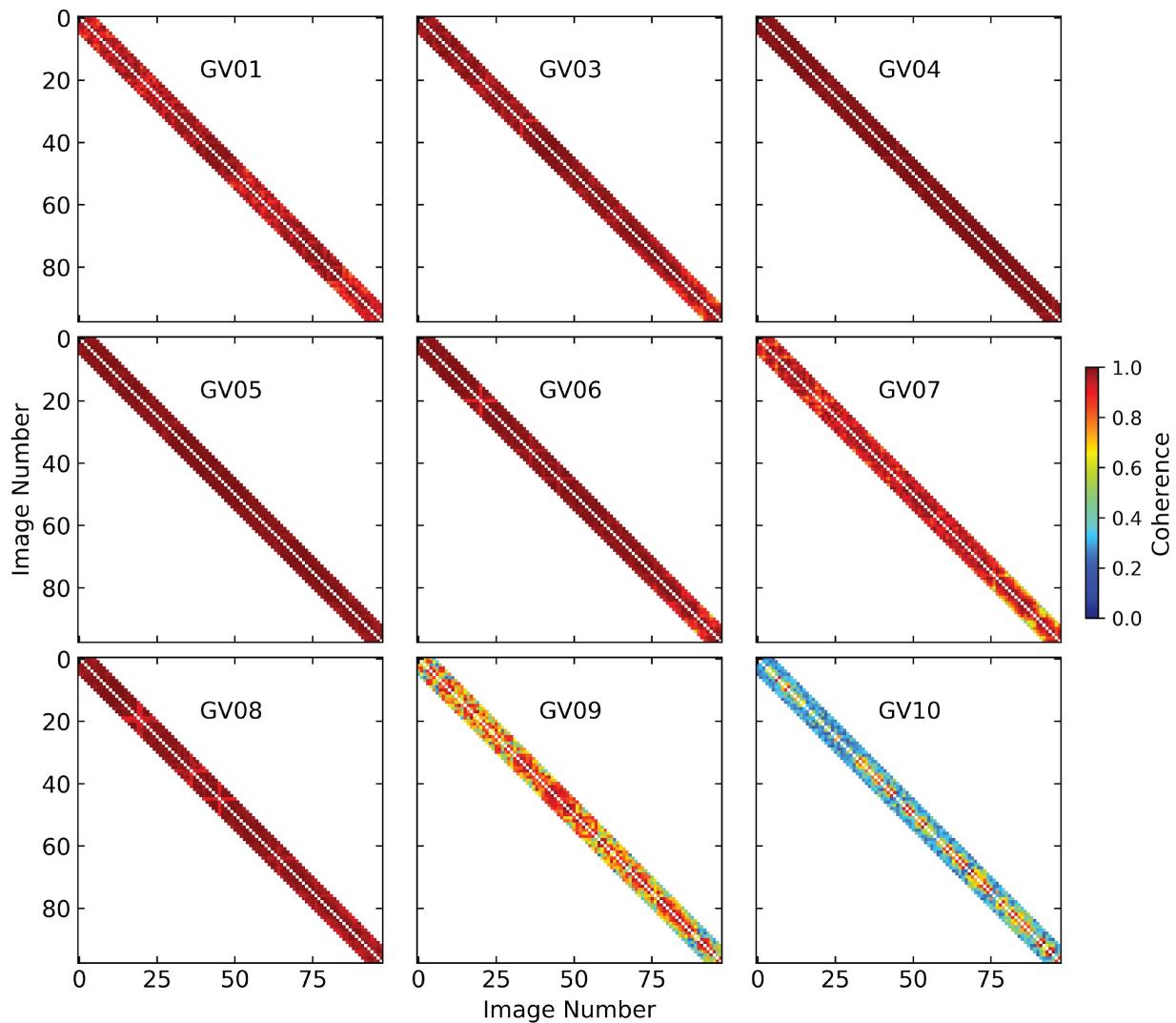
62 **Figure S4.** Simulation for the optimal LASSO trade-off parameter α . Related to equation (11) in
 63 section 3.2. Mean output percentage of 100 realization of interferograms with unwrapping errors
 64 after correction as a function of the nonnegative α value for different input percentage of
 65 interferograms with unwrapping errors. The network of interferograms is the same as Fig. 4a.
 66 The simulation result shows that any number of α in $[10^{-4}, 10^0]$ works. We choose 10^{-2} as default
 67 value.

68

69

70 **Figure S5.** Illustration of the step function in topographic residual correction in presence of
 71 displacement jumps. Related to equation (13) in section 4.8. (a and b) Perpendicular baseline
 72 history (from the Sentinel-1 data of section 5) and an arbitrary displacement time-series using
 73 simulated data (with a permanent displacement jump at 1 March 2016 with a magnitude of 20
 74 cm, shown as the dashed black line in (b), in addition to the topographic residual contribution
 75 from a DEM error of 50 m). Blue empty circles and orange triangles represent displacement
 76 time-series after topographic residual correction assuming quadratic model without and with a
 77 step function, respectively. (c and d) Same as (a and b) but (i) using ALOS-1 data for one pixel
 78 on Cerro Azul located at [W91.270°, S0.928°] and (ii) the black dashed line for the displacement
 79 time-series without topographic residual correction. In both simulated and real data, the
 80 disagreement between the low-frequency quadratic model and the high-frequency displacement
 81 jump leads to biased estimation of the topographic residual (Du et al., 2007) and adding a step
 82 function could effectively eliminate this estimation bias. This estimation bias is amplified in the
 83 first ALOS-1 acquisition by its large perpendicular baseline (the difference between black
 84 dashed line and the blue empty circles in (d)).





85

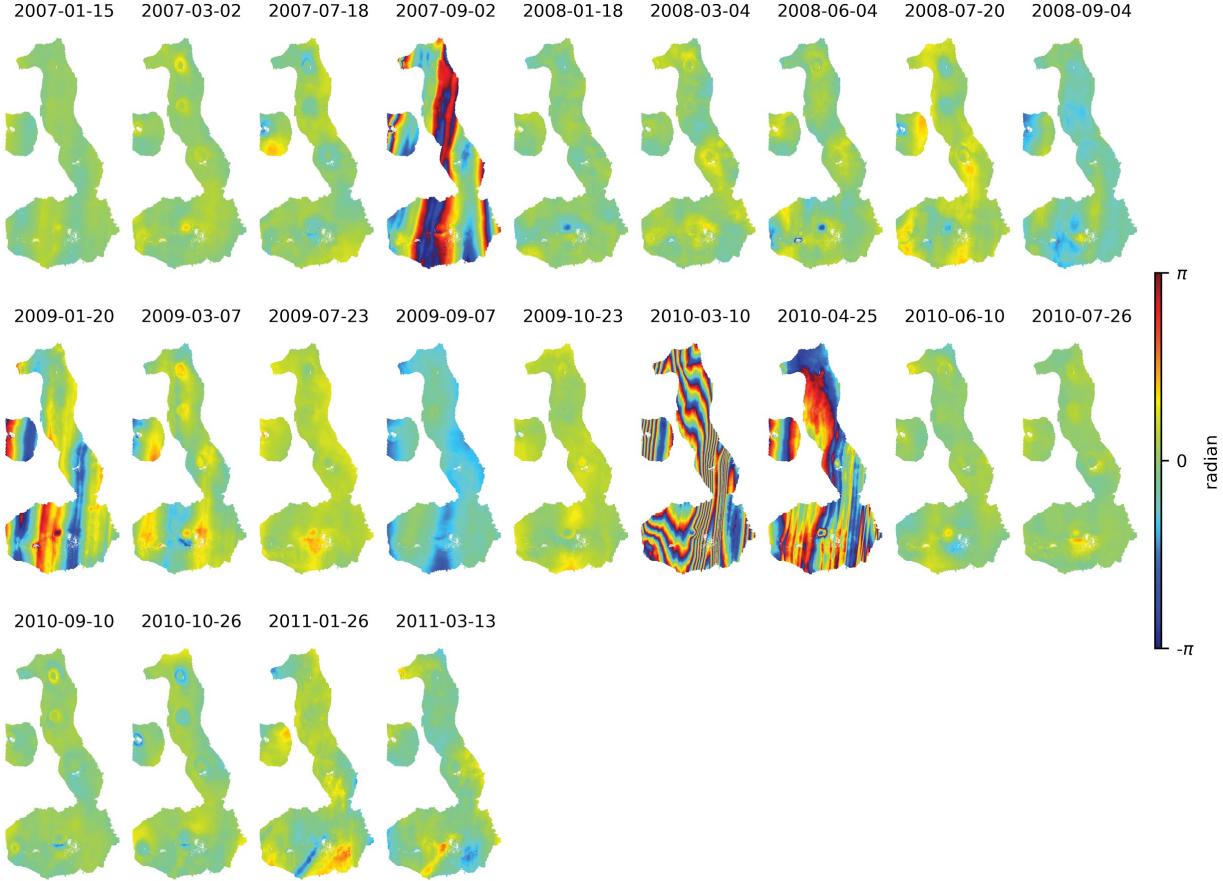
86 **Figure S6.** Coherence matrix of Sentinel-1 dataset for GPS stations within Sierra Negra caldera.

87 Related to Fig. 8 in section 5.1. Both X and Y axis indicate the number of SAR acquisitions.

88 Station GV10 is located in a densely vegetated area outside the caldera on the rim, resulting in

89 fast decorrelation with low spatial coherence on interferograms with more than 2 lags.

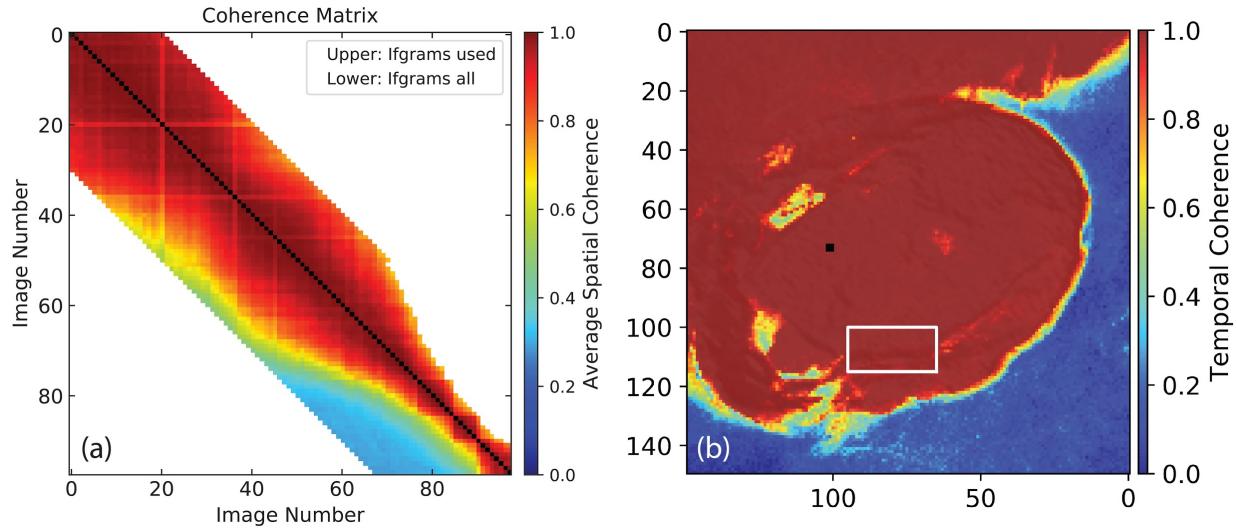
90



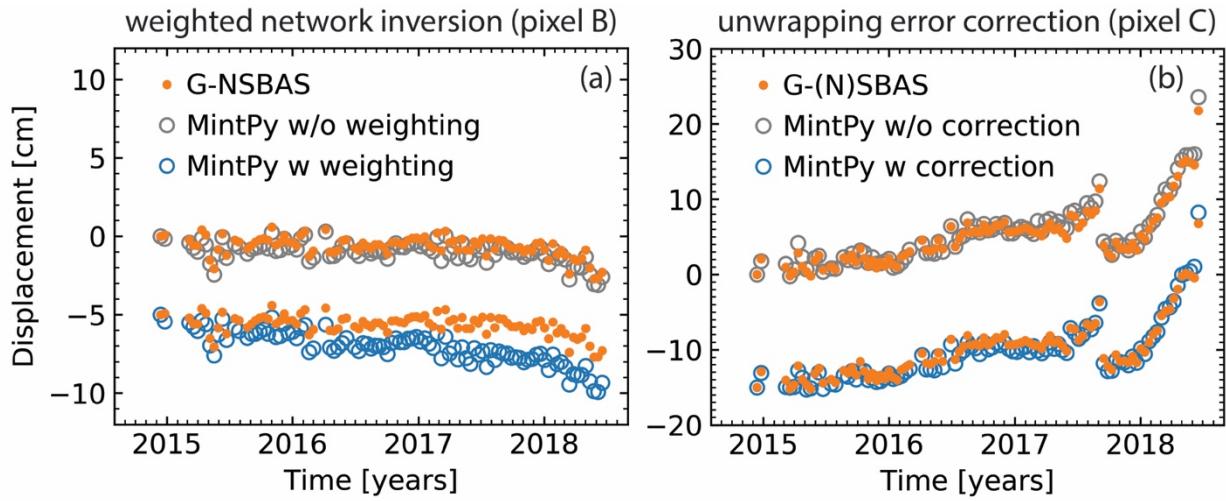
91

92 **Figure S7.** The estimated residual phase time-series $\hat{\phi}_{resid}$ of ALOS-1 dataset. Related to
 93 equation (13-14) in section 4.7 and Fig. 12 in section 5.4. A quadratic phase ramp has been
 94 estimated and removed from each acquisition. This is used in equation (14) to calculate the
 95 residual phase RMS value. Phases on 2 September 2007, 10 March 2010 and 25 April 2010 are
 96 severely contaminated by ionospheric streaks and are automatically identified as outliers. Phase
 97 on 20 January 2009 is contaminated by ionosphere also but is not identified as outlier due to its
 98 relatively small magnitude.

99



100
101 **Figure S8.** Coherence-based network modification for Sentinel-1 data used in section 6.3 in
102 Sierra Negra. Related to Fig. 14 in section 6.3. (a) Coherence matrix of the customized area of
103 interest along the trap door fault within Sierra Negra caldera (marked by the white rectangle in
104 (b)). A network of interferograms with 30 sequential connections (2475 in total) are generated
105 from 98 SAR acquisitions, as shown in the lower triangle. The upper triangle shows the
106 interferogram kept after the network modification. A maximum of 20 connections are shown in
107 Fig. 14 only. (b) Temporal coherence of the network inversion from the interferogram stack with
108 a maximum of 20 connections.
109



110

111 **Figure S9.** Impact of (a) weighted network inversion and (b) unwrapping error correction on the
 112 displacement time-series. Related to Fig. 16 in section 6.5. The comparison within (a) shows that
 113 the difference on pixel B (Alcedo's flank) between MintPy and G-NSBAS is caused by the
 114 weighting during the network inversion. The comparison within (b) shows that the difference on
 115 pixel C (Fernandina's crater) between MintPy and G-(N)SBAS is caused by the unwrapping
 116 error correction.

117

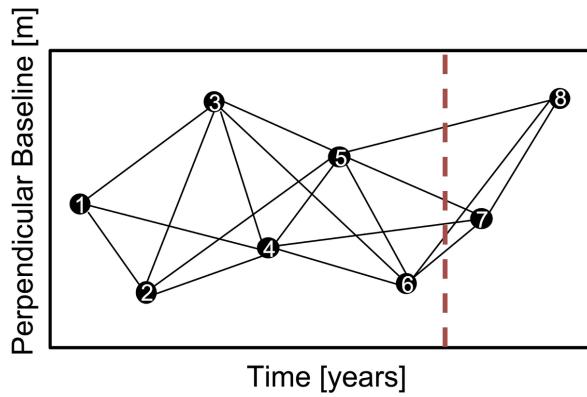
118 **Table S1.** SAR dataset information with parameters used in InSAR stack processing

Satellite	ALOS-1	Sentinel-1A/B
Orbit direction	Ascending	Descending
Track number	133	128 (swath 1 & 2)
Start / end date (# of acquisitions)	2007-01-15 / 2011-03-13 (22)	2014-12-13 / 2018-06-19 (98)
Network selection criteria (# of Interferograms)	$B_{\text{temp}} \leq 1800$ days $B_{\perp} \leq 1800$ m (228)	Sequential with 5 connections (475)
# of looks in range / azimuth direction	8×16	15×5
Ground pixel size in range / azimuth direction (m)	60×51	62×70
InSAR Processor	ROI_PAC	ISCE
Phase Unwrapping	SNAPHU	SNAPHU

119

120 **S2. Design matrices**

121 This section shows examples to generate the design matrices used in the software. A demo set of
122 $N = 8$ SAR images acquired at $[t_1, \dots, t_8]$ is used as the example. A stack of $M = 18$ interferograms
123 is selected using the sequential method with 3 connections. An earthquake or volcanic eruption
124 event occurred between t_6 and t_7 (red dashed line), which caused a permanent ground
125 displacement offset.



126

127 **Figure S10.** Network configuration of the demo dataset. Red dashed line marks the time of a
128 displacement offset due to an earthquake or volcanic eruption.

129 **S2.1 Network inversion**

130 To generate the design matrix \mathbf{A} for network inversion used in equation (1) in section 2.1, we
131 first generate an $M \times N$ matrix. For each row, it consists -1, 0 and 1 with -1 for the reference
132 acquisition, 1 for the secondary acquisition and 0 for the rest. Due to the relative nature of
133 InSAR measurement, the phase on the reference date (the first date by default) cannot be
134 resolved, thus, we can only solve $[\phi^2, \dots, \phi^N]$ instead of $[\phi^1, \dots, \phi^N]$ and the corresponding
135 column (the first column by default) is eliminated in the design matrix \mathbf{A} , which results in size of
136 $M \times (N - 1)$.

137

138

$$\mathbf{A} = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix} \quad (\text{S1})$$

139

140 **S2.2 Phase closure of interferograms triplets**

141 Design matrix \mathbf{C} describe the combination of interferograms to form the triplets used in equation
 142 (10) in section 3.2 for the phase closure unwrapping error correction. An example of \mathbf{C} is shown
 143 below based on the demo network with number of triplets $T = 16$.

144 [1 -1 0 1 0 0 0 0 0 0 0 0 0 0 0 0]

145 [1 0 -1 0 1 0 0 0 0 0 0 0 0 0 0 0]

146 [0 1 -1 0 0 0 1 0 0 0 0 0 0 0 0 0]

147 [0 0 0 1 -1 0 1 0 0 0 0 0 0 0 0 0]

148 [0 0 0 1 0 -1 0 1 0 0 0 0 0 0 0 0]

149 [0 0 0 0 1 -1 0 0 0 1 0 0 0 0 0 0]

150 [0 0 0 0 0 0 1 -1 0 1 0 0 0 0 0 0]

```

151    $\mathbf{C} = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ -1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$  (S2)
152    $[0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ -1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$ 
153    $[0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ -1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0]$ 
154    $[0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ -1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0]$ 
155    $[0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ -1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0]$ 
156    $[0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ -1 \ 0 \ 1 \ 0 \ 0]$ 
157    $[0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ -1 \ 0 \ 1 \ 0]$ 
158    $[0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ -1 \ 0 \ 0 \ 1]$ 
159    $[0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ -1 \ 1]$ 

```

160 **S2.3 Topographic residual correction**

161 Design matrix \mathbf{G} is used in equation (13) for topographic residual correction in section 4.8. It is
162 in size of $N \times (1 + N_{poly} + N_{step})$, where N_{poly} is the user-defined polynomial order N_{poly} (2 by
163 default), N_{step} is the number of Heaviside step functions (0 by default) describing offsets at
164 specific prior selected times. An example of \mathbf{G} is shown below based on the demo network.

165

$$166 \quad \mathbf{G} = \begin{bmatrix} \frac{4\pi}{\lambda} \frac{B_1^1}{rsin(\theta)} & 1 & (t_1 - t_1) & \frac{(t_1 - t_1)^2}{2} & 0 \\ \frac{4\pi}{\lambda} \frac{B_1^2}{rsin(\theta)} & 1 & (t_2 - t_1) & \frac{(t_2 - t_1)^2}{2} & 0 \\ \frac{4\pi}{\lambda} \frac{B_1^3}{rsin(\theta)} & 1 & (t_3 - t_1) & \frac{(t_3 - t_1)^2}{2} & 0 \\ \frac{4\pi}{\lambda} \frac{B_1^4}{rsin(\theta)} & 1 & (t_4 - t_1) & \frac{(t_4 - t_1)^2}{2} & 0 \\ \frac{4\pi}{\lambda} \frac{B_1^5}{rsin(\theta)} & 1 & (t_5 - t_1) & \frac{(t_5 - t_1)^2}{2} & 0 \\ \frac{4\pi}{\lambda} \frac{B_1^6}{rsin(\theta)} & 1 & (t_6 - t_1) & \frac{(t_6 - t_1)^2}{2} & 0 \\ \frac{4\pi}{\lambda} \frac{B_1^7}{rsin(\theta)} & 1 & (t_7 - t_1) & \frac{(t_7 - t_1)^2}{2} & 1 \\ \frac{4\pi}{\lambda} \frac{B_1^8}{rsin(\theta)} & 1 & (t_8 - t_1) & \frac{(t_8 - t_1)^2}{2} & 1 \end{bmatrix} \quad (S3)$$

167 Then equation (13) can be formed as a linear system with N equations as below:

168

169 $\hat{\phi} - \hat{\phi}_{tropo} = \mathbf{G}X + \phi_{resid}$ (S4)

170

171 where $X = [z_\varepsilon, c_0, c_1, c_2, s_7]^T$ is the vector of unknown parameters, $\hat{\phi}$, $\hat{\phi}_{tropo}$ and ϕ_{resid} are the
172 $N \times 1$ inverted raw phase time-series, estimated tropospheric delay time-series and residual
173 phase time-series, respectively. We apply the least squares estimation to obtain the solution as:

174

175 $\hat{X} = (\mathbf{G}^T \mathbf{G})^{-1} \mathbf{G}^T (\hat{\phi} - \hat{\phi}_{tropo})$ (S5)

176 $\hat{\phi}_{resid} = \hat{\phi} - \hat{\phi}_{tropo} - \mathbf{G}\hat{X}$ (S6)

177

178 The estimated residual phase $\hat{\phi}_{resid}$ is used to characterize the noise of phase time-series using
179 equation (14) in section 4.9. The noise-reduced displacement time-series is given as:

180

181 $\phi_{dis}^i = \hat{\phi}^i - \hat{\phi}_{tropo}^i - \frac{-4\pi}{\lambda} \frac{B_\perp^i}{rsin(\theta)} \hat{z}_\varepsilon$ (S7)

182

183 where $i = 1, \dots, N$ and \hat{z}_ε is the estimated DEM error in \hat{X} .

184 **S2.4 Average velocity estimation**

185 For each pixel, the average velocity is estimated as $d^i = vt_i + c$, where $d^i = -\frac{\lambda}{4\pi} \phi_{dis}^i$ is the
186 displacement at t_i in meters, v is the unknown velocity and c is the unknown offset. The solution
187 can be obtained using least squares approximation. An example of the design matrix \mathbf{E} is shown
188 below based on the demo network.

189

190

$$\mathbf{E} = \begin{bmatrix} t_1 - t_1 & 1 \\ t_2 - t_1 & 1 \\ t_3 - t_1 & 1 \\ t_4 - t_1 & 1 \\ t_5 - t_1 & 1 \\ t_6 - t_1 & 1 \\ t_7 - t_1 & 1 \\ t_8 - t_1 & 1 \end{bmatrix} \quad (\text{S8})$$

191

192 For linear displacement, the uncertainty of the estimated velocity σ_v is given by equation (10) in
 193 Fattah and Amelung (2015) as:

194

$$195 \quad \sigma_v = \sqrt{\frac{\sum_{i=1}^N (\phi_{dis}^i - \hat{\phi}_{dis}^i)^2}{(N-2) \sum_{i=1}^N (t_i - \bar{t})^2}} \quad (\text{S9})$$

196

197 where $\hat{\phi}_{dis}^i$ is the predicted linear displacement at i^{th} acquisition \bar{t} is the mean value of time in
 198 years.

199

200 **S3. Decorrelation noise simulation**

201 **S3.1 Coherence model**

202 We simulate the coherence for a stack of interferograms on one pixel using a decorrelation
 203 model with exponential decay for temporal decorrelation. The spatial coherence γ^j of the j_{th}
 204 interferogram can be expressed as (Zebker and Villasenor, 1992; Hanssen, 2001; Parizzi et al.,
 205 2009):

206

207
$$\gamma = \gamma_{geom} \cdot \gamma_{DC} \cdot \gamma_{temporal} \quad (\text{S10})$$

208

209 where γ_{geom} represents the geometric decorrelation, γ_{DC} represents the Doppler centroid
 210 decorrelation, $\gamma_{temporal}$ represents the temporal decorrelation, given by the equations below.
 211 Note that the thermal decorrelation $\gamma_{thermal}$ is served as the instantaneous decorrelation in
 212 temporal decorrelation $\gamma_{temporal}$ (Parizzi et al., 2009).

213

214
$$\gamma_{geom} = \begin{cases} 1 - \frac{|B_{\perp}|}{B_{\perp}^{crit}}, & |B_{\perp}| \leq B_{\perp}^{crit} \\ 0, & |B_{\perp}| > B_{\perp}^{crit} \end{cases} \quad (\text{S11})$$

215
$$\gamma_{DC} = \begin{cases} 1 - \frac{|\Delta f_{DC}|}{B_{az}}, & |\Delta f_{DC}| \leq B_{az} \\ 0, & |\Delta f_{DC}| > B_{az} \end{cases} \quad (\text{S12})$$

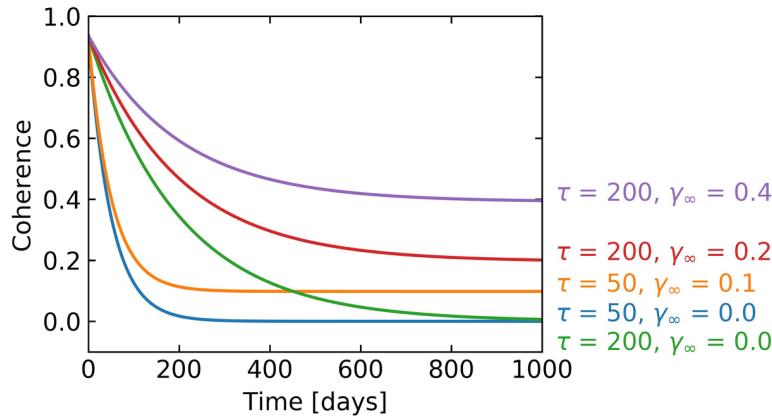
216
$$\gamma_{temporal}(t) = (\gamma_{thermal} - \gamma_{\infty})e^{-t/\tau} + \gamma_{\infty} \quad (\text{S13})$$

217
$$\gamma_{thermal} = \frac{1}{1 - SNR^{-1}} \quad (\text{S14})$$

218

219 The critical perpendicular baseline $B_{\perp}^{crit} = \lambda \frac{B_{rg}}{c} R \cdot \tan(\theta)$ is the baseline causing a spectral
 220 shift equal to the radar bandwidth B_{rg} in range direction (Zebker and Villasenor, 1992; Hanssen,
 221 2001), where λ is the radar wavelength, c is the speed of light, R is the distance between radar
 222 antenna and ground target and θ is the incidence angle, SNR is the thermal signal-to-noise ratio
 223 of radar receiver. τ is the time constant which depends on radar wavelength λ , it's the time for
 224 coherence to drop down to $1/e$, i.e. 0.36, from its initial value (Parizzi et al., 2009; Rocca, 2007).
 225 γ_{∞} is the long-term coherence, or minimum attainable coherence value, which converged over
 226 time, usually with high values in urban area and low values in vegetated area. Note that this
 227 model does not consider the seasonal behavior of temporal decorrelation, volume decorrelation,
 228 and processing-induced decorrelation. For a given set of SAR acquisitions, the geometric and
 229 Doppler centroid decorrelation is almost constant among all pixels. All parameters are deployed
 230 with typical parameters of Sentinel-1 SAR sensor.

231



232

233 **Figure S11.** Simulated coherence as a function of temporal baseline, color coded by different τ
 234 and γ_{∞} settings used in Fig. S2.

235 **S3.2 Simulate decorrelation noise from coherence**

236 For distributed scatterers (DS) in natural, vegetated terrain the interferometric phase exhibits
 237 highly unpredictable speckle characteristics. Its phase can be appropriately modeled by a random
 238 process, complex, stationary, circular Gaussian process in the case of SAR image. Applying the
 239 central limit theorem, the probability density function $pdf(\Delta\phi)$ of interferometric phase is
 240 obtained as (equation (66) from Tough et al., 1995; equation (4.2.23) from Hanssen, 2001):

241

$$242 \quad pdf(\Delta\phi) = \frac{(1-|\gamma|^2)^L}{2\pi} \left\{ \frac{\Gamma(2L-1)}{[\Gamma(L)]^2 2^{2(L-1)}} \times \left[\frac{(2L-1)\beta}{(1-\beta^2)^{L+\frac{1}{2}}} \left(\frac{\pi}{2} + \arcsin\beta \right) + \frac{1}{(1-\beta^2)^L} \right] + D \right\} \quad (S15)$$

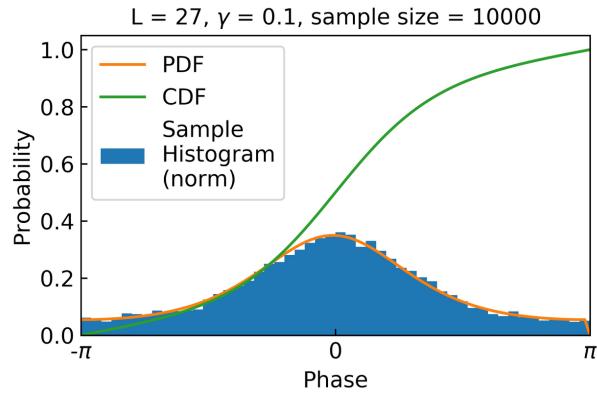
$$243 \quad D = \frac{1}{2(L-1)} \sum_{r=0}^{L-2} \frac{\Gamma(L-\frac{1}{2})}{\Gamma(L-\frac{1}{2}-r)} \frac{\Gamma(L-1-r)}{\Gamma(L-1)} \frac{1+(2r+1)\beta^2}{(1-\beta^2)^{r+2}}$$

244 where $\beta = |\gamma| \cos(\Delta\phi - \Delta\phi_0)$, expected interferometric phase $\Delta\phi_0 = E\{\Delta\phi\}$, gamma function
 245 $\Gamma(L) = \int_0^\infty t^{L-1} e^{-t} dt$, for $L \in R$ and D a finite summation term. Note that D vanishes for
 246 single-look datasets ($L=1$).

247

248 The 10,000 realizations/samples of decorrelation noise of each interferogram (used in section
 249 2.4) is simulated by generating a distribution given by equation (S15) with corresponding
 250 coherence γ and number of looks L . One example with $\gamma = 0.1$ and $L = 9 \times 3$ is shown in Fig.
 251 S12.

252



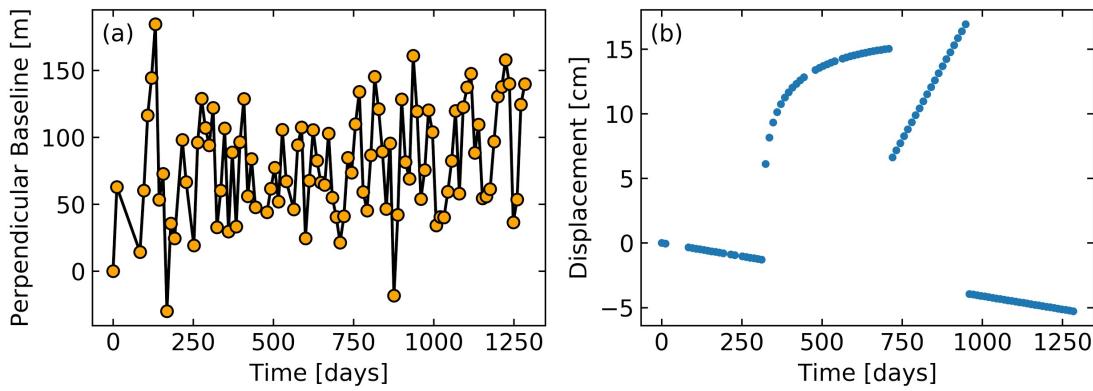
253

254 **Figure S12.** Sampling the decorrelation noise based on phase PDF of distributed scatterers.

255 Blue bars: normalized histogram of sampled decorrelation noises. Orange and green line:

256 phase PDF and cumulative distribution function.

257



258

259 **Figure S13.** Time-series configuration for simulation. (a) Perpendicular baseline history from

260 the 98 Sentinel-1 images of section 5. (b) Specified time-dependent displacement used in section

261 2.4 and 3.2.

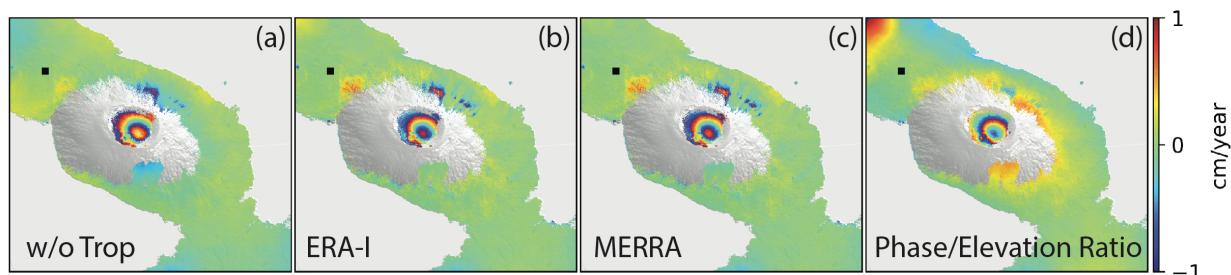
262

263 **S4. Additional software features**

264 **S4.1 Customized workflow beyond `smallbaselineApp.py`**

265 Most scripts in MintPy are stand-alone (summarized in Table S4). Users can apply any phase
266 correction at any time to evaluate the impact. Fig. S14 shows an example, where we use
267 individual scripts ([link on GitHub](#)) to compare velocities estimated from displacement time-
268 series with different tropospheric delay correction methods on Alcedo volcano.

269



270
271 **Figure S14.** Deformation velocity maps on Alcedo volcano from Sentinel-1 (a) without
272 tropospheric correction, with tropospheric correction using (b) ERA-Interim, (c) MERRA-2 and
273 (d) the empirical phase-elevation ratio method.

274

275 **Table S4.** Stand-alone scripts in MintPy

<code>add.py</code>	Generate the sum of multiple input files
<code>asc_desc2horz_vert.py</code>	Project ascending and descending displacement in LOS direction to horizontal and vertical direction
<code>dem_error.py</code>	DEM error (topographic residual) correction
<code>diff.py</code>	Generate the difference of two input files

<code>generate_mask.py</code>	Generate mask file from input file
<code>geocode.py</code>	Resample radar-coded files into geo coordinates, or vice versa.
<code>ifgram_inversion.py</code>	Invert network of interferograms into time-series.
<code>image_reconstruction.py</code>	Reconstruct network of interferograms from time-series
<code>image_math.py</code>	Basic mathematic operation of input file(s)
<code>info.py</code>	Display metadata / structure of input file
<code>load_data.py</code>	Load a stack of interferograms into HDF5 files
<code>load_gbis.py</code>	Load the inversion result from GBIS software
<code>load_hdf5.py</code>	Load the binary file(s) into an HDF5 file
<code>local_oscillator_drift.py</code>	Correct local oscillator drift for Envisat data
<code>mask.py</code>	Mask input data file with input mask file by setting values on the unselected pixels into Nan or zero.
<code>match.py</code>	Merge two or more geocoded files which share common area into one file.
<code>modify_network.py</code>	Modify the network setting of an ifgramStack HDF5 file.
<code>multilook.py</code>	Multilook input file.
<code>plot_coherence_matrix.py</code>	Plot the coherence matrix of one pixel, interactively.
<code>plot_network.py</code>	Plot the network configuration of an ifgramStack file.
<code>plot_transection.py</code>	Plot the value of 2D matrix along a profile.
<code>prep_aria.py</code>	Prepare input data from ARIA GNUW products
<code>prep_gamma.py</code>	Prepare metadata file for GAMMA files.
<code>prep_giant.py</code>	Prepare metadata file for GIAnT files.

<code>prep_isce.py</code>	Prepare metadata file for ISCE files.
<code>prep_roipac.py</code>	Prepare metadata file for ROI_PAC files.
<code>prep_snap.py</code>	Prepare metadata file for SNAP geocoded products.
<code>reference_date.py</code>	Change the reference date of a time-series HDF5 file.
<code>reference_point.py</code>	Change the reference pixel of an input file.
<code>remove_ramp.h5</code>	Remove phase ramps for input file.
<code>save_gbis.py</code>	Save input files in GBIS *.mat file format.
<code>save_gmt.py</code>	Save input file in GMT *.grd file format.
<code>save_hdfeos5.py</code>	Save input time-series into HDF-EOS5 format.
<code>save_kmz.py</code>	Save input file into Google Earth raster image.
<code>save_kmz_timeseries.h5</code>	Save input file into Google Earth points, interactively.
<code>save_roipac.py</code>	Save input file into ROI_PAC style binary file format.
<code>select_network.py</code>	Select interferometric pairs from input baseline file.
<code>smallbaselineApp.py</code>	Routine time series analysis for small baseline InSAR stack.
<code>spatial_average.py</code>	Calculate average in space domain.
<code>spatial_filter.py</code>	Spatial filtering of input file.
<code>subset.py</code>	Generate a subset of (crop) input file.
<code>temporal_average.py</code>	Calculate average in time domain.
<code>temporal_derivative.py</code>	Calculate the temporal derivative of displacement time-series.
<code>temporal_filter.py</code>	Smooth time-series in time domain with a moving Gaussian window

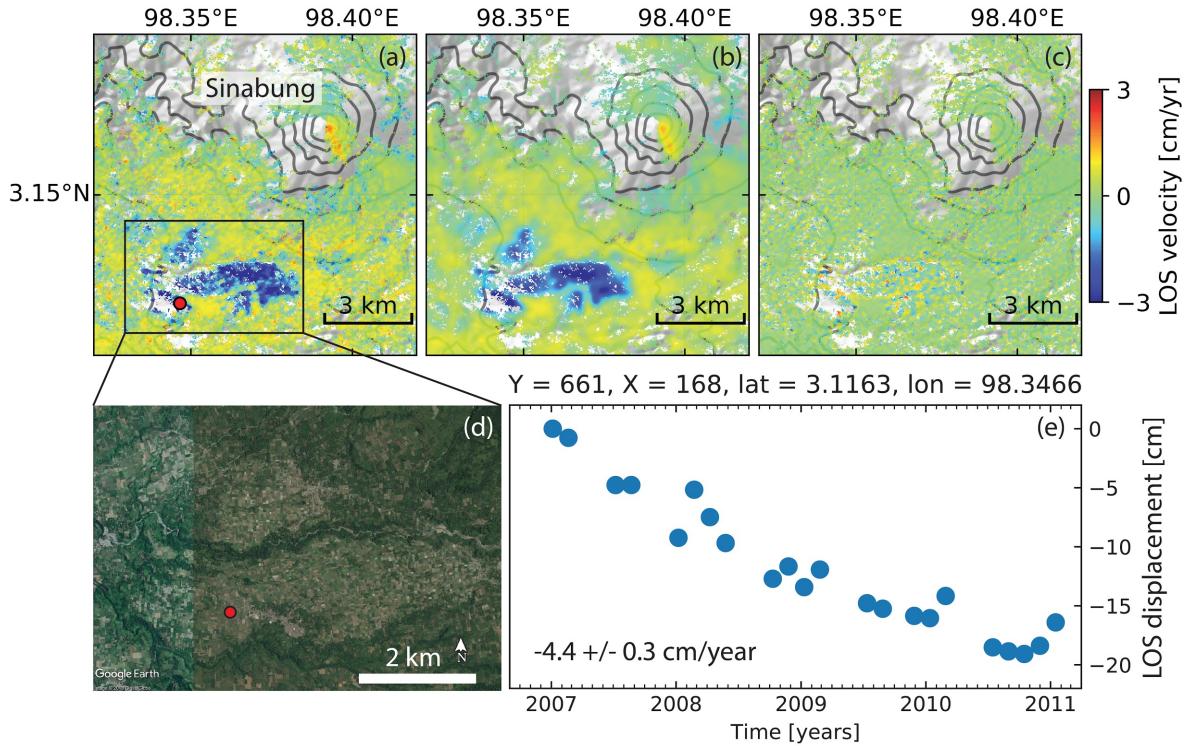
<code>timeseries2velocity.py</code>	Invert time-series for the average velocity.
<code>timeseries_rms.py</code>	Calculate the root mean square for each acquisition of the input time-series file.
<code>tropo_phase_elevation.py</code>	Correct stratified tropospheric delay based on the empirical phase/elevation ratio method.
<code>tropo_pyaps.py</code>	Correct tropospheric delay estimated from global atmospheric model (GAM) using PyAPS software (Jolivet et al., 2011; 2014).
<code>tsview.py</code>	Interactive time-series viewer.
<code>unwrap_error_bridging.py</code>	Correct phase-unwrapping errors with bridging method.
<code>unwrap_error_phase_closure.py</code>	Correct phase-unwrapping errors with the phase closure method.
<code>view.py</code>	2D matrix viewer.

276

277 **S4.2 Filters tools in space and time domain**

278 The software supports filters in space or time domain built on skimage (van der Walt et al.,
 279 2014). Although filtering is not applied in the routine workflow, it is a useful tool to examine the
 280 deformation signal because it allows removing undesired signals. Fig. S15 shows an example,
 281 where we use spatial Gaussian filtering to confirm a patchy, rapid subsidence signal.

282

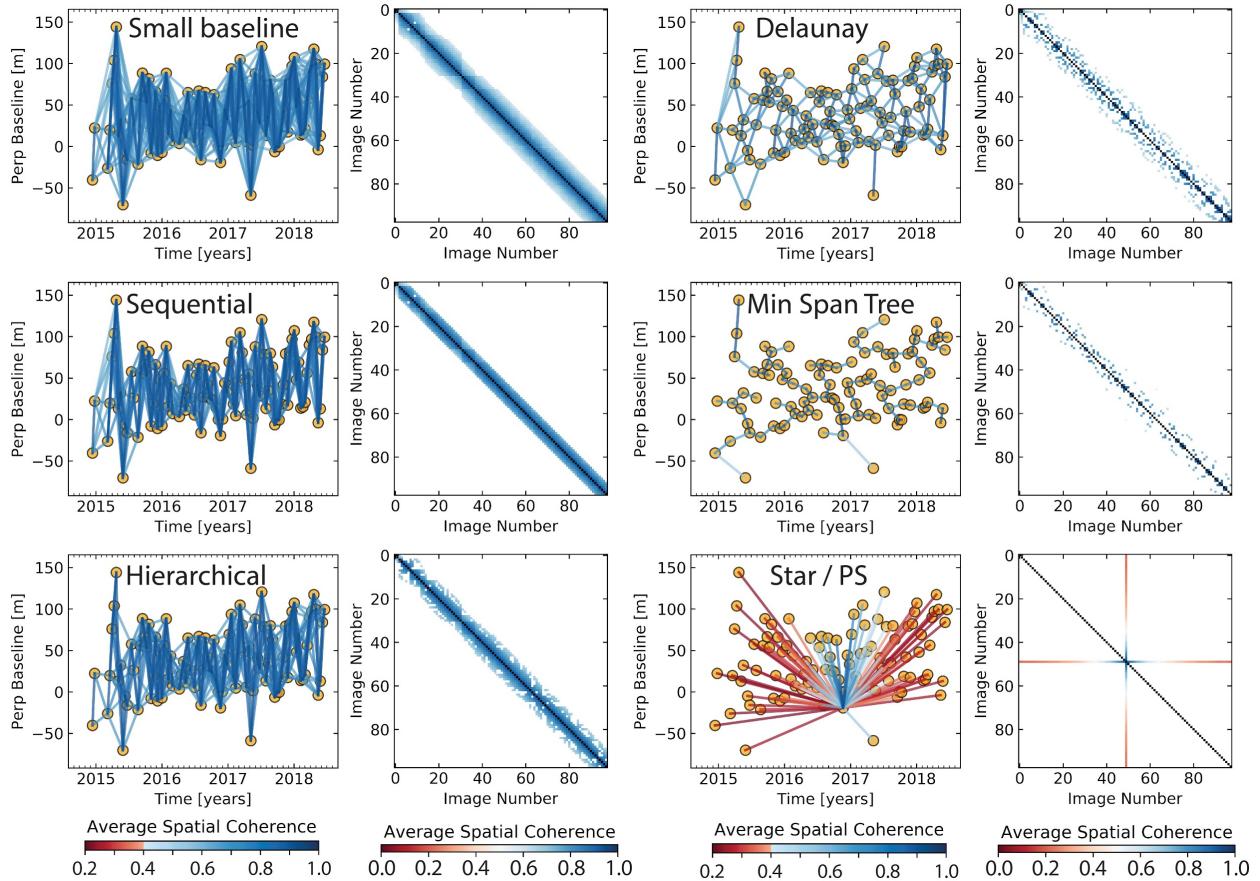


283

284 **Figure S15.** Illustration of the spatial filtering. The LOS velocity from ALOS-1 ascending track
 285 495 acquired over Sinabung volcano, Indonesia during January 2007 to January 2011 is used.
 286 (a) Original velocity in LOS direction, (b and c) velocities after lowpass and highpass Gaussian
 287 filtering with the standard deviation of 3.0. (a) is the sum of (b) and (c). The lowpass filtering
 288 eliminated the very short spatial wavelength features, thus, highlighted the relatively long spatial
 289 wavelength deformation features, such as the volcanic deformation along the Sinabung's
 290 southeast flank and an undocumented patchy, rapid subsidence area (up to -5.6 cm/year) is
 291 found ~6 km to the southwest of the volcano. The spatial pattern of the subsidence signal
 292 correlates well with the agricultural land use, suggesting that subsidence is caused by
 293 groundwater extraction (Chaussard et al., 2013). Reference point is a pixel at [E98.4999°,
 294 N3.1069°] outside of this figure. (d) Google Earth image for the marked rectangle area. (e) LOS
 295 displacement time-series for pixel marked by red circle in (a) at [E98.3466°, N3.1163°].

296 **S4.3 Interferometric pairs selection**

297 The software supports several interferometric pairs selection methods to facilitate the pre-
 298 processing, such as small baseline, sequential, hierarchical, Delaunay triangulation, minimum
 299 spanning tree and star/PS-like methods, as shown in Fig. S16.



300 **Figure S16.** Illustration of interferometric pairs selection. The temporal and perpendicular
 301 baselines are from Sentinel-1 dataset of section 5. For each method, network configuration on
 302 the left and the corresponding coherence matrix on the right. The spatial coherence calculation
 303 is described in section S3.1 with decorrelation rate of 200 days and long-term coherence of 0.2.
 304 The small baseline method selects interferograms with temporal and perpendicular baseline
 305 within the predefined thresholds (120 days and 200 m; Berardino et al., 2002). The sequential
 306 method selects for each acquisition with a predefined number (5) of its nearest neighbors back in
 307 the temporal dimension. The hierarchical method uses a hierarchical clustering scheme to group
 308 images into clusters, and then selects pairs between clusters. The Delaunay and minimum spanning
 309 tree methods use geometric graphs to find the most efficient set of connections. The star/PS method
 310 selects a central reference image and connects it to all other images in the dataset.

308 time (Reeves and Zhao, 1999). The hierarchical method specifies a predefined list of temporal
309 and perpendicular baselines as [6 days, 300 m; 12 days, 200 m; 48 days, 100 m; 96 days, 50 m],
310 each pair of temporal and perpendicular thresholds selects interferograms the same as small
311 baseline method (Zhao, 2017). The Delaunay triangulation method generates triangulations in
312 the temporal and perpendicular baseline domain and selects interferograms within the
313 predefined maximum temporal and perpendicular baseline (120 days and 200 m; Pepe and
314 Lanari, 2006). The minimum spanning tree method calculates a spatial coherence value based
315 on its simple relationship with the temporal and perpendicular baseline and selects N-1
316 interferograms that maximizes the total coherence (Perissin and Wang, 2012). The star-like
317 method selects network of N-1 interferograms with single common reference acquisition (usually
318 in the center of the time period; Ferretti et al., 2001).

319

320 **S4.4 Local oscillator drift correction for Envisat**

321 Data from Envisat's Advanced Synthetic Aperture Radar instrument include a phase ramp in
322 range direction due to timing errors. We correct this local oscillator drift using the empirical
323 model given by Marinkovic and Larsen (2013).

324

$$325 \quad \phi_{LOD}^i = \frac{-4\pi}{\lambda} 3.87 \times 10^{-7} r(t_i - t_1) \quad (\text{S16})$$

326

327 where $(t_i - t_1)$ represents the time difference in years between SAR acquisition t_i and t_1 (see
328 also Fattah and Amelung, 2014). Since this model is independent of the InSAR phase
329 measurement, this correction should be applied before any InSAR data-dependent phase
330 corrections.

331 **Supplemental references**

- 332 Chaussard, E., F. Amelung, H. Abidin, and S.-H. Hong (2013), Sinking cities in Indonesia:
333 ALOS PALSAR detects rapid subsidence due to groundwater and gas extraction, *Remote
334 Sensing of Environment*, 128(0), 150-161, doi:10.1016/j.rse.2012.10.015.
- 335 Du, Y., L. Zhang, G. Feng, Z. Lu, and Q. Sun (2017), On the Accuracy of Topographic
336 Residuals Retrieved by MTInSAR, *IEEE Transactions on Geoscience and Remote Sensing*,
337 55(2), 1053-1065, doi:10.1109/TGRS.2016.2618942.
- 338 Marinkovic, P., and Y. Larsen (2013), Consequences of long-term ASAR local oscillator
339 frequency decay - An empirical study of 10 years of data, paper presented at *Proceedings of
340 the Living Planet Symposium (abstract)*, European Space Agency, Edinburgh, U. K.
- 341 Reeves, S. J., and Z. Zhao (1999), Sequential algorithms for observation selection, *IEEE
342 Transactions on Signal Processing*, 47(1), 123-132, doi:10.1109/78.738245.
- 343 van der Walt, S., J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E.
344 Gouillart, and T. Yu (2014), scikit-image: image processing in Python, *PeerJ*, 2, e453,
345 doi:10.7717/peerj.453.
- 346 Zhao, W. (2017), Small Deformation Detected from InSAR Time-Series and Their Applications
347 in Geophysics, Dissertation thesis, 153 pp, University of Miami, Miami, FL.
- 348