

## Algorithm for computing infinitesimal strain rate between three non-colinear GPS stations, given their N-S and E-W velocities, with a worked example

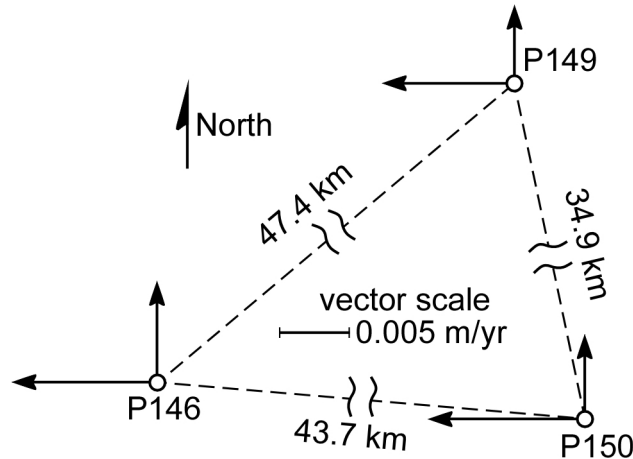
**Introduction.** This algorithm explains how to compute the mean infinitesimal horizontal strain (or instantaneous horizontal strain rate) in the crust between three GPS sites. The theory on which this algorithm is based is presented in the instantaneous strain primer that accompanies this module. This algorithm is implemented in strain calculators written in *Excel*, *Mathematica* and *MatLab*.

The analysis described in this document involves several assumptions. All significant strain is assumed to occur in the horizontal plane, so no vertical velocities are used. One of the principal strain axes is assumed to be vertical. Deformation is assumed to be homogeneous across the area between the three GPS sites. In the description of the worked example, no attempt is made to round values to the appropriate significant figures until the end of the analysis.

---

**Step 1.** Acquire location and velocity data for three GPS stations that form a triangle in which none of the interior angles is less than  $\sim 30^\circ$ . Usually, the location of GPS stations is expressed in geographic coordinates (latitude and longitude). To be compatible with the following explanation, the location data should be converted to UTM coordinates within the same UTM zone, and the velocity data should consist of N-S and E-W vectors with magnitudes expressed in m/yr relative to a stable-North-America reference frame such as NAM08. If the stations straddle a zone boundary, it is best to compute the coordinates of the site(s) in the eastern zone as if they were in the western zone. You can use the web-based conversion utility offered by the National Geodetic Survey via <https://www.ngs.noaa.gov/NCAT/>. Alternatively, the Excel spreadsheet "gps\_strain\_calculator\_excel.v3.xls" (available via [https://d32ogoqmya1dw8.cloudfront.net/files/getsi/teaching\\_materials/gps\\_strain/gps\\_strain\\_calculator\\_excel.v3.xls](https://d32ogoqmya1dw8.cloudfront.net/files/getsi/teaching_materials/gps_strain/gps_strain_calculator_excel.v3.xls)) implements code in rows 41-88 that is patterned after a conversion spreadsheet by Steven Dutch that converts from geographic coordinates (WGS84) to UTM (or vice versa).

Site No.	UTM easting (meters)	UTM northing (meters)	E velocity (m/year)	Uncert (m/year)	N velocity (m/year)	Uncert (m/year)
P146	712245.807	4357118.796	-0.01031	0.00001	0.00625	0.00001
P149	748566.739	4387604.015	-0.00942	0.00003	0.00520	0.00003
P150	755806.255	4353418.458	-0.01086	0.00003	0.00592	0.00003



**Figure 1.** Map of 3 non-colinear GPS sites, part of the Plate Boundary Observatory. The east-west and north-south velocities of each site relative to the Stable North American Reference Frame are indicated by arrows.

**Step 2.** Find the center of the triangle formed by the three GPS stations. The  $x$  (east-west) coordinate of the triangle center is the sum of the three  $x$  coordinates divided by 3. In this example, we have

$$\frac{712,245.8 + 748,566.7 + 755,806.3}{3} = 738,872.9$$

The  $y$  (north-south) coordinate is the sum of the three  $y$  coordinates divided by 3. For this example,

$$\frac{4,357,118.8 + 4,387,604.0 + 4,353,418.5}{3} = 4,366,047.1$$

**Step 3.** Transform the three points into a coordinate system in which the center of the triangle has the  $\{x, y\}$  coordinates  $\{0, 0\}$ . To do this, subtract the coordinates of the center of the triangle from the coordinates of each of the stations. The transformed coordinates of the three stations are as follows:

- station 146 :  $\{(712245.8 - 738872.9), (4357118.8 - 4366047.1)\} = \{-26627.1, -8928.3\}$
- station 149 :  $\{(748566.261 - 738872.9), (4387604.0 - 4366047.1)\} = \{9693.8, 21556.9\}$
- station 150 :  $\{(755806.266 - 738872.9), (4353418.5 - 4366047.1)\} = \{16933.4, -12628.6\}$

The reason for this step, which was suggested by J.C. Savage, is that the translation vector that we will soon compute relates to the translation of the origin of the coordinate system. It is simply more meaningful to compute the translation of the center of the area we are studying than to compute the translation of the origin of the UTM grid we are in, which in this case is more than 700 km west and 4,350 km south of our area of interest.

**Step 4.** Make a matrix called **m1** in which the x (east-west) coordinates of the GPS sites form the first column, and the y (north-south) coordinates form the second column. In our example, the first row is formed by the Cartesian coordinates of GPS site 146, expressed in UTM coordinates.

$$m1 = \begin{bmatrix} -26627.1 & -8928.3 \\ 9693.8 & 21556.9 \\ 16933.4 & -12628.6 \end{bmatrix}$$

Rather than keep typing those long numbers, we can refer to the value "-12628.6" in row 3, column 2 of matrix **m1** using subscripts: **m1**<sub>32</sub>.

**Step 5.** Use the values in matrix **m1** to fill-in the values in matrix **m2** that are not filled with 0s and 1s, as defined by the following:

$$m2 = \begin{bmatrix} 1 & 0 & -m1_{12} & m1_{11} & m1_{12} & 0 \\ 0 & 1 & m1_{11} & 0 & m1_{11} & m1_{12} \\ 1 & 0 & -m1_{22} & m1_{21} & m1_{22} & 0 \\ 0 & 1 & m1_{21} & 0 & m1_{21} & m1_{22} \\ 1 & 0 & -m1_{32} & m1_{31} & m1_{32} & 0 \\ 0 & 1 & m1_{31} & 0 & m1_{31} & m1_{32} \end{bmatrix}$$

So, in this example, the matrix **m2** would look like this if we wrote-in the actual values.

$$m2 = \begin{bmatrix} 1 & 0 & -(-8928.3) & -26627.1 & -8928.3 & 0 \\ 0 & 1 & -26627.1 & 0 & -26627.1 & -8928.3 \\ 1 & 0 & -(21556.9) & 9693.8 & 21556.9 & 0 \\ 0 & 1 & 9693.8 & 0 & 9693.8 & 21556.9 \\ 1 & 0 & -(-12628.6) & 16933.4 & -12628.6 & 0 \\ 0 & 1 & 16933.4 & 0 & 16933.4 & -12628.6 \end{bmatrix}$$

Note that matrix **m2** is a square matrix, which means that it has the same number of columns as rows.

**Step 6.** Invert matrix **m2** to form matrix **m3**.

$$m3 = m2^{-1}$$

This can be done by hand, but frankly you would be a little bit nuts to do so.

- In Microsoft *Excel*, there is a function called MINVERSE (for Matrix INVERSE) that will invert a square matrix. While your computer is connected to the web and *Excel* is open, select Excel Help from the Help menu, and type MINVERSE in the search box. An explanation for the MINVERSE function from the Help Center will appear, and illustrate

how to use the function in your version of *Excel*. The utube video at <http://www.youtube.com/watch?v=G8w-d9U3PJM> might also be helpful.

- The command to invert a square matrix **m2** in *MatLab* is **inv(m2)** (see <http://www.mathworks.com/help/techdoc/ref/inv.html>).
- The command to invert a square matrix **m2** in *Mathematica* is **Inverse[m2]** (see <http://reference.wolfram.com/mathematica/ref/Inverse.html> and <http://www.wolframalpha.com/examples/Matrices.html>).
- You might find a matrix calculator online that will provide you with the result you need.

In this example, matrix **m3** looks like this:

$$m3 = \begin{bmatrix} 0.333333 & 0 & 0.333333 & 0 & 0.333333 & 0 \\ 0 & 0.333333 & 0 & 0.333333 & 0 & 0.333333 \\ 2.475 \times 10^{-6} & -1.169 \times 10^{-5} & -1.489 \times 10^{-5} & 1.265 \times 10^{-6} & 1.242 \times 10^{-5} & 1.042 \times 10^{-5} \\ -2.338 \times 10^{-5} & 0 & 2.530 \times 10^{-6} & 0 & 2.085 \times 10^{-5} & 0 \\ -2.475 \times 10^{-6} & -1.169 \times 10^{-5} & 1.489 \times 10^{-5} & 1.265 \times 10^{-6} & -1.242 \times 10^{-5} & 1.042 \times 10^{-5} \\ 0 & -4.951 \times 10^{-6} & 0 & 2.979 \times 10^{-5} & 0 & -2.484 \times 10^{-5} \end{bmatrix}$$

**Step 7.** Create a column matrix called **m4** that contains the velocity data for the three GPS sites, as shown below:

$$m4 = \begin{bmatrix} P146 v_x \\ P146 v_y \\ P149 v_x \\ P149 v_y \\ P150 v_x \\ P150 v_y \end{bmatrix} = \begin{bmatrix} -0.01031 \\ 0.00625 \\ -0.00942 \\ 0.0052 \\ -0.01086 \\ 0.00592 \end{bmatrix}$$

In the matrix above,  $P146 v_x$  is the east velocity of PBO site P146 in m/yr, and  $P146 v_y$  is the north velocity of that site. Matrix **m4** is a 6x1 column matrix – it has 6 rows and 1 column.

**Step 8.** Multiply matrix **m4** by matrix **m3** to yield a 6x1 column matrix called **m5**.

$$m5 = m3 \cdot m4 = \begin{bmatrix} t_x \\ t_y \\ \Omega \\ \epsilon_{xx} \\ \epsilon_{xy} \\ \epsilon_{yy} \end{bmatrix} = \begin{bmatrix} -0.0101967 \\ 0.00579 \\ -2.48541 \times 10^{-8} \\ -9.21367 \times 10^{-9} \\ 1.53178 \times 10^{-8} \\ -2.30811 \times 10^{-8} \end{bmatrix}$$

**Step 9.** Find the infinitesimal translation vector for the triangle defined by the three GPS stations, as measured in the Stable North American Reference Frame (SNARF). The first two values of matrix  $m5$  (that is, the values in the first two rows of  $m5$ , or  $m5_{11}$  and  $m5_{21}$ ) are the coordinates of the mean translational velocity vector of the sites.

$$\text{translational vector} = \{t_x, t_y\} = \{m5_{11}, m5_{21}\}$$

In this example,

$$\text{translational vector} = \{-0.0101967, 0.00579\}.$$

The length of the translation vector is the translational speed in meters per year, which we can find using the Pythagorean Theorem.

$$\text{translational speed} = \sqrt{t_x^2 + t_y^2}$$

In this example,

$$\text{translational speed} = \sqrt{(-0.0101967)^2 + 0.00579^2} = 0.0117259 \text{ m/yr}$$

The  $x$  coordinate of the translation vector has a negative value, so the vector is directed toward the west rather than toward the east. The  $y$  coordinate has a positive value, so the vector is directed toward the north rather than toward the south. To find the azimuth of the translation vector, we must first find the angle between the translation vector and a unit vector that is pointing due north, which would have coordinates  $\{0, 1\}$ . (A unit vector is a vector that has a length of 1.)

$$\text{unit north vector} = \hat{n} = \{0, 1\}$$

We can find the unit vector that is coincident with the translation vector by dividing each coordinate of the translation vector by the translational speed. In this example,

$$\text{unit translation vector} = \hat{t} = \left\{ \frac{-0.0101967}{0.0117259}, \frac{0.00579}{0.0117259} \right\}$$

$$\text{unit translation vector} = \{-0.869587, 0.49378\}$$

To find the angle  $\theta_1$  in degrees between the north vector and the unit translation vector, take the inverse cosine (or arc cosine or  $\cos^{-1}$ ) of the dot product of the two unit vectors, and multiply the result by the conversion factor from radians to degrees:  $(180^\circ/\pi)$ .

$$\theta_1 = \left( \cos^{-1}(\hat{n} \cdot \hat{t}) \right) \left( \frac{180^\circ}{\pi} \right)$$

In this example,  $\theta_1 = \sim 60.41^\circ$ . Then we use  $\theta_1$  to find the azimuth of the translation vector. If the value of the  $x$  coordinate of the translation vector ( $t_x$ ) is a negative number, we subtract  $\theta_1$  from  $360^\circ$  to yield the azimuth. If  $t_x$  is not a negative number, the azimuth is simply equal to  $\theta_1$ . In this example,  $t_x$  is a negative number, so the azimuth of the translation vector is

$$360^\circ - \theta_1 = 360^\circ - 60.41^\circ = \sim 300^\circ.$$

We had already determined that the translation vector would be directed toward the north and west, so an azimuth between 270° and 360° makes sense. The centroid of our triangle of GPS stations is translating toward ~300° at a rate of 0.0117259 meters per year.

**Step 10.** Find the value of the rotational velocity,  $\Omega$ , in matrix **m5**: the value in **m5**<sub>31</sub>.

$$\Omega = -2.4854 \times 10^{-8} \text{ radian per year.}$$

This is the rate at which the principal axes of strain are rotating relative to the original orientation of those lines. In the more convenient units of nano-radian per year, the rotational velocity is

$$\Omega = -24.8541 \text{ nano-radian per year.}$$

The negative value for  $\Omega$  indicates that the axes of the strain ellipse (and lines of material particles that coincided with the principal strain axes prior to deformation) are rotating in a clockwise direction.

**Step 11.** Define the 2-D Lagrangian strain tensor ( $\epsilon_{ij}$ ) and call it matrix **m6**.

$$\epsilon_{ij} = \begin{bmatrix} \epsilon_{xx} & \epsilon_{xy} \\ \epsilon_{yx} & \epsilon_{yy} \end{bmatrix}, \text{ so}$$

$$\mathbf{m6} = \begin{bmatrix} \epsilon_{xx} & \epsilon_{xy} \\ \epsilon_{yx} & \epsilon_{yy} \end{bmatrix} = \begin{bmatrix} m5_{41} & m5_{51} \\ m5_{51} & m5_{61} \end{bmatrix}$$

In this example,

$$\mathbf{m6} = \begin{bmatrix} -9.2137 \times 10^{-9} & 1.5318 \times 10^{-8} \\ 1.5318 \times 10^{-8} & -2.3081 \times 10^{-8} \end{bmatrix}.$$

**Step 12.** Determine the orientations of the principal infinitesimal strain axes by finding the eigenvectors of the symmetrical Lagrangian infinitesimal strain tensor **m6**. Determine the magnitudes of the principal infinitesimal extensions by finding the eigenvalues (the magnitudes of the eigenvectors) of the Lagrangian strain tensor.

Explanations of how to determine eigenvectors and eigenvalues are plentiful in books on tensors and on the web. Two good sources of information are <http://mathworld.wolfram.com/Eigenvalue.html> and <http://controls.engin.umich.edu/wiki/index.php/EigenvaluesEigenvectors>.

- If you are using *MatLab*, a general explanation of the process is available online via <http://www.mathworks.com/help/techdoc/ref/eig.html> and <http://www.mathworks.com/help/techdoc/math/f4-2852.html>. If you supply two output variables (e.g., **V** for a matrix of eigenvectors and **D** for a matrix in which the corresponding eigenvalues are along the diagonal of a square matrix), *MatLab* will return eigenvalues and the eigenvectors as unit vectors: **[V,D]=eig(m6)**.
- If you are using *Mathematica*, you input the following command to acquire the eigenvectors of matrix **m6**: **Eigenvectors[m6]**. The unit vectors that coincide with the principal strain axes will be returned in order of descending magnitude, so the vector

associated with the eigenvalue with the largest *absolute* value will be listed first. To find the eigenvalues of **m6** in *Mathematica*, input **Eigenvalues[m6]**. Or just type **EigenSystem[m6]** and the output will be a list in which the first group of elements are the eigenvalue, followed by the unit vectors that coincide with the eigenvectors.

- Microsoft *Excel* does not have a built-in function to find eigenvectors or eigenvalues. Some plug-ins that can compute eigenvalues and eigenvectors are advertised by third-party developers on the web.

One work-around is to use the eigensystem calculator from Wolfram Alpha, available online via

[http://www.wolframalpha.com/entities/calculators/eigenvalue\\_calculator/gv/9h/bi/](http://www.wolframalpha.com/entities/calculators/eigenvalue_calculator/gv/9h/bi/) or <http://www.wolframalpha.com/widgets/view.jsp?id=ca851e9f71b25d2d588b7e0d4de3d627>.

The Excel spreadsheet "gps\_strain\_calculator\_excel.v3.xls" (available via [https://d32ogoqmya1dw8.cloudfront.net/files/getsi/teaching\\_materials/gps\\_strain/gps\\_strain\\_calculator\\_excel.v3.xls](https://d32ogoqmya1dw8.cloudfront.net/files/getsi/teaching_materials/gps_strain/gps_strain_calculator_excel.v3.xls)) implements code in rows 155-162 that determine the unit eigenvectors associated with  $\mathbf{e}_1$  and  $\mathbf{e}_2$ , respectively, in a horizontal plane aligned with the geographic coordinates (+Y is toward north, +X is toward east). The explanation of the relationships implemented in that code is explained in the following text.

In the 2D case in which the strain matrix is a 2x2 matrix, the eigenvalues are the results of the following expressions

$$eigenvalues = \frac{(\epsilon_{xx} + \epsilon_{yy}) \pm \sqrt{4\epsilon_{xy}\epsilon_{yx} + (\epsilon_{xx} - \epsilon_{yy})^2}}{2}$$

with the greater value corresponds to the greater principal extension  $\mathbf{e}_1$  along the greater instantaneous horizontal strain axis  $\mathbf{S}_{1H}$ . The smaller eigenvalue is the lesser principal extension  $\mathbf{e}_2$  along the lesser instantaneous horizontal strain axis  $\mathbf{S}_{2H}$ . In this example,

$$\mathbf{e}_1 = 6.6663 \times 10^{-10}, \text{ and}$$

$$\mathbf{e}_2 = -3.2961 \times 10^{-8}$$

Use one of the techniques mentioned above to find the corresponding eigenvectors. In this example, the unit vector that coincides with the greater principal strain axis is

$$\hat{\mathbf{S}}_1 = \{-0.8404, -0.5420\}$$

To find the angle  $\theta_2$  in degrees between the unit north vector ( $\hat{\mathbf{n}}$ ) and the unit vector that coincides with the greater principal strain axis ( $\hat{\mathbf{S}}_1$ ), take the inverse cosine of the dot product of the two unit vectors, and multiply the result by the conversion factor from radians to degrees:  $(180^\circ/\pi)$ .

$$\theta_2 = \left( \cos^{-1}(\hat{\mathbf{n}} \cdot \hat{\mathbf{S}}_1) \right) \left( \frac{180^\circ}{\pi} \right)$$

In this example,  $\theta_2 = \sim 123^\circ$ . Then we use  $\theta_2$  to find the azimuth of the  $\mathbf{S}_{1H}$  axis. If the value of the  $x$  coordinate of the  $\mathbf{S}_1$  eigenvector is a negative number, we subtract  $\theta_2$  from  $360^\circ$  to yield the azimuth. If it is not a negative number, the azimuth is simply equal to  $\theta_2$ . In this example, the  $x$  coordinate of the  $\mathbf{S}_1$  eigenvector is a negative number, so

$$\text{azimuth } S_{1H} \text{ axis} = 360^\circ - \theta_2 = 360^\circ - 123^\circ = \sim 237^\circ.$$

The  $S_{1H}$  axis will trend toward  $\sim 237^\circ$  (or  $57^\circ$ ), and the  $S_{2H}$  axis will be  $90^\circ$  from  $S_{1H}$ , trending  $327^\circ$  (or  $147^\circ$ ).

**Step 13.** Determine the magnitude of the maximum infinitesimal shear strain ( $\gamma_{\max}$ ) at  $45^\circ$  to the maximum principal strain axis. Maximum infinitesimal shear strain is given by the following

$$\gamma_{\max} = 2\sqrt{\left(\frac{\epsilon_{xx} - \epsilon_{yy}}{2}\right)^2 + (\epsilon_{xy})^2}$$

$$\gamma_{\max} = 2\sqrt{\left(\frac{m\epsilon_{11} - m\epsilon_{22}}{2}\right)^2 + (m\epsilon_{12})^2}$$

In this example,

$$\gamma_{\max} = 2\sqrt{\left(\frac{(-9.2137 \times 10^{-9}) - (-2.3081 \times 10^{-8})}{2}\right)^2 + (1.5318 \times 10^{-8})^2} = 3.3628 \times 10^{-8}$$

The maximum infinitesimal shear strain is  $3.3628 \times 10^{-8}$ . There are two other ways of computing the maximum shear strain. The difference between principal extensions  $\mathbf{e}_1$  and  $\mathbf{e}_2$  is equal to the maximum shear strain. In this example, the difference between  $\mathbf{e}_1$  and  $\mathbf{e}_2$  is

$$\mathbf{e}_1 - \mathbf{e}_2 = (6.6663 \times 10^{-10}) - (-3.2961 \times 10^{-8}) = 3.3628 \times 10^{-8}$$

The principal stretches ( $\mathbf{S}_1$  and  $\mathbf{S}_2$ ) are given by

$$\mathbf{S}_1 = 1 + \mathbf{e}_1, \text{ and}$$

$$\mathbf{S}_2 = 1 + \mathbf{e}_2.$$

The quadratic elongations  $\lambda_1$  and  $\lambda_2$  are given by

$$\lambda_1 = \mathbf{S}_1^2 = (1 + \mathbf{e}_1)^2 \text{ and } \lambda_2 = \mathbf{S}_2^2 = (1 + \mathbf{e}_2)^2$$

Finally, the maximum infinitesimal shear strain is given by

$$\gamma_{\max} = \frac{\lambda_1 - \lambda_2}{2\sqrt{\lambda_1\lambda_2}}$$

All three methods give us the same answer:  $3.3628 \times 10^{-8}$  or 33.628 nano-strain.

**Step 14.** Determine the areal strain rate between the GPS sites. A unit circle (*i.e.*, a circle with a radius of 1) located within the triangle prior to deformation will be an ellipse after deformation. The change in area between the circle and ellipse, divided by the area of the circle, is the area strain. The greater principal stretch ( $\mathbf{S}_1 = 1 + \mathbf{e}_1$ ) is the major axis of the ellipse, while the lesser principal stretch ( $\mathbf{S}_2 = 1 + \mathbf{e}_2$ ) is the minor axis.

$$\text{circle area} = \pi$$

$$\text{ellipse area} = \pi \mathbf{S}_1 \mathbf{S}_2$$



$$\text{area strain} = (\text{ellipse area} - \text{circle area})/\text{circle area}$$

In this example, the equation above yields an answer of  $-3.2295 \times 10^{-8}$ . Another way of computing the areal strain is

$$\text{area strain} = (\mathbf{S}_1 \mathbf{S}_2) - 1$$

In this example, the equation above yields the same answer as the first equation:  $-3.2295 \times 10^{-8}$ . The area strain is also equal to the first invariant of the strain tensor. An invariant of a tensor is a combination of tensor components that does not change with different coordinate systems. The first invariant of the 3-D strain tensor is

$$\text{volume strain} = \mathbf{e}_1 + \mathbf{e}_2 + \mathbf{e}_3 = \varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33}$$

The first invariant in the 2-D case is given by

$$\text{area strain} = \mathbf{e}_1 + \mathbf{e}_2 = \varepsilon_{11} + \varepsilon_{22}$$

In this example,

$$\begin{aligned} \text{area strain} &= \mathbf{e}_1 + \mathbf{e}_2 \\ \text{area strain} &= (6.6663 \times 10^{-10}) + (-3.2961 \times 10^{-8}) = -3.2295 \times 10^{-8} \\ \text{area strain} &= m\mathbf{6}_{11} + m\mathbf{6}_{22} \\ \text{area strain} &= (-9.2137 \times 10^{-9}) + (-2.3081 \times 10^{-8}) = -3.2295 \times 10^{-8} \end{aligned}$$

In this example, the area strain is  $-3.2295 \times 10^{-8}$  or  $-32.295$  nano-strain. The negative number denotes a decrease in area.

---

**Step 15.** Identify/compute the second and third invariants of the strain tensor. The first invariant of the 2-D strain tensor is the dilation or area strain (step 15) and is equal to the sum of the diagonal terms of the stress tensor. It is also known as the *trace* of the strain tensor. The second invariant of the 2-D strain tensor is

$$\begin{aligned} (\varepsilon_{11} \varepsilon_{22}) - \varepsilon_{12}^2 &= \mathbf{e}_1 \mathbf{e}_2 \\ (m\mathbf{6}_{11} m\mathbf{6}_{22}) - m\mathbf{6}_{12}^2 &= \mathbf{e}_1 \mathbf{e}_2 \end{aligned}$$

or, in this example,

$$\begin{aligned} ((-9.2137 \times 10^{-9})(-2.3081 \times 10^{-8})) - (1.5318 \times 10^{-8})^2 &= -2.19731 \times 10^{-17} = -2.19731 \times 10^{-8} \text{ nano-strain} \\ ((6.6663 \times 10^{-10})(-3.2961 \times 10^{-8})) &= -2.19731 \times 10^{-17} = -2.19731 \times 10^{-8} \text{ nano-strain} \end{aligned}$$

The third invariant of the 2-D strain tensor is the determinant of the strain tensor

$$\text{determinant}[\varepsilon_{ij}] = \mathbf{e}_1 \mathbf{e}_2$$

or, in this example

$$((6.6663 \times 10^{-10})(-3.2961 \times 10^{-8})) = -2.19731 \times 10^{-17} = -2.19731 \times 10^{-8} \text{ nano-strain}$$

**Step 16.** Compute the uncertainties

The inverse of the data covariance matrix is matrix  $m7$ , whose only non-zero terms are along the diagonal of the matrix. The diagonal terms are the reciprocals of the squares of the uncertainties in the site velocities.

$$m7 = \begin{bmatrix} \frac{1}{P146E \sigma^2} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{P146N \sigma^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{P149E \sigma^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{P149N \sigma^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{P150E \sigma^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{P150N \sigma^2} \end{bmatrix}$$

In our example,

$$m7 = \begin{bmatrix} \frac{1}{0.00001^2} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{0.00001^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{0.00003^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{0.00003^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{0.00003^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{0.00003^2} \end{bmatrix}$$

$$m8 = m2^T = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ -m1_{12} & m1_{11} & -m1_{22} & m1_{21} & -m1_{32} & m1_{31} \\ m1_{11} & 0 & m1_{21} & 0 & m1_{31} & 0 \\ m1_{12} & m1_{11} & m1_{22} & m1_{21} & m1_{32} & m1_{31} \\ 0 & m1_{12} & 0 & m1_{22} & 0 & m1_{32} \end{bmatrix}$$

or, in our example,

$$m8 = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 8928.3 & -26627.1 & -21556.9 & 9693.77 & 12628.6 & 16933.4 \\ -26627.1 & 0 & 9693.77 & 0 & 16933.4 & 0 \\ -8928.3 & -26627.1 & 21556.9 & 9693.77 & -12628.6 & 16933.4 \\ 0 & -8928.3 & 0 & 21556.9 & 0 & -12628.6 \end{bmatrix}$$

Matrix  $m9$  is the inverse of the matrix product  $[m8 \cdot m7 \cdot m2]$

$$m9 = [m8 \cdot m7 \cdot m2]^{-1}$$

In some cases, this inversion might result in a poorly conditioned matrix, in which case it is preferable to employ a Moore-Penrose matrix inverse. This is implemented in Mathematica using the built-in function **PseudoInverse**, as in **PseudoInverse** $[m8 \cdot m7 \cdot m2]$ .

The values along the diagonal of matrix  $m9$  are the squares of the uncertainties for the east-west ( $m9_{11}$ ) and north-south ( $m9_{22}$ ) components of translation, the rotation ( $m9_{33}$ ),  $\epsilon_{xx}$  ( $m9_{44}$ ),  $\epsilon_{xy}$  ( $m9_{55}$ ) and  $\epsilon_{yy}$  ( $m9_{66}$ ). In our example, the uncertainties are  $\sqrt{m9_{11}} = 0.00001453$ ,  $\sqrt{m9_{22}} = 0.00001453$ ,  $\sqrt{m9_{33}} = 6.7227 \times 10^{-10}$ ,  $\sqrt{m9_{44}} = 6.7197 \times 10^{-10}$ ,  $\sqrt{m9_{55}} = 6.7227 \times 10^{-10}$ , and  $\sqrt{m9_{66}} = 1.1646 \times 10^{-9}$ .

### Step 17. Summarize the input and the output data

- Cartesian coordinates of infinitesimal translation vector (meter):  $\{-0.010197 \pm 0.0000145, 0.00579 \pm 0.0000145\}$
- Azimuth of infinitesimal translation vector:  $300^\circ$
- Magnitude of infinitesimal translation vector: 0.0117259 meter per year
- Instantaneous angular velocity:  $-24.8541 \pm 0.6723$  n-radian/yr
- Infinitesimal Lagrangian strain tensor ( $\epsilon_{ij}$ ):
 
$$\begin{bmatrix} -9.2137 \pm 0.6720 \times 10^{-9} & 1.5318 \pm 0.06723 \times 10^{-8} \\ 1.5318 \pm 0.06723 \times 10^{-8} & -2.3081 \pm 0.1165 \times 10^{-8} \end{bmatrix}$$
- Greatest infinitesimal horizontal extension (-ve value is contraction): 0.66663 n-strain/yr
- Azimuth of  $S_{1H}$  axis:  $57^\circ$  or  $237^\circ$
- Least infinitesimal horizontal extension (-ve value is contraction): -32.9614 n-strain/yr
- Azimuth of  $S_{2H}$  axis:  $147^\circ$  or  $327^\circ$
- Infinitesimal maximum shear strain at  $45^\circ$  to  $S_{1H}$ : 33.628 n-strain/yr
- Areal strain (-ve means constriction): -32.2948 n-strain/yr
- First invariant of the strain tensor: -32.2948 n-strain/yr
- Second invariant of the strain tensor:  $-2.1973 \times 10^{-8}$  n-strain/yr
- Third invariant of the strain tensor:  $-2.1973 \times 10^{-8}$  n-strain/yr

## Some useful resources and references

Information about the EarthScope Plate Boundary Observatory is available online via <http://pbo.unavco.org>

General information about reference frames can be found at <https://www.unavco.org/software/visualization/GPS-Velocity-Viewer/GPS-Velocity-Viewer.html>

The Universal Transverse Mercator coordinate system is described at [https://en.wikipedia.org/wiki/Universal\\_Transverse\\_Mercator\\_coordinate\\_system](https://en.wikipedia.org/wiki/Universal_Transverse_Mercator_coordinate_system) and <https://pubs.usgs.gov/fs/2001/0077/report.pdf> and <https://gisgeography.com/utm-universal-transverse-mercator-projection/> among other references.

The National Geodetic Survey's NGS Coordinate Conversion and Transformation Tool (NCAT) is available via <https://www.ngs.noaa.gov/NCAT/>

Information about UNAVCO is available online via <http://www.unavco.org/>

The full public data holdings of UNAVCO are available via their "Data Archive Interface Version 2" at <http://www.unavco.org/data/gps-gnss/data-access-methods/dai2/app/dai2.html>.

Information about the GPS-GNSS data holdings in particular is available via <http://www.unavco.org/data/gps-gnss/gps-gnss.html>

An online eigensystem calculator is available from Wolfram|Alpha at [http://www.wolframalpha.com/entities/calculators/eigenvalue\\_calculator/gv/9h/bi/](http://www.wolframalpha.com/entities/calculators/eigenvalue_calculator/gv/9h/bi/)

A Wolfram|Alpha widget for evaluating a number of characteristics of a given matrix, including its eigensystem, is available online at <http://www.wolframalpha.com/widgets/view.jsp?id=ca851e9f71b25d2d588b7e0d4de3d627>

A stand-alone Macintosh application called SSPX performs this same analysis in a more comprehensive manner for 3 or more GPS sites, including estimates of uncertainty. SSPX is available for academic or research use via the Mac App Store (<https://itunes.apple.com/us/app/sspx/id850389108>); also visit Nestor Cardozo's website at <http://www.ux.uis.no/~nestor/work/programs.html>.

Allmendinger, R.W., Cardozo, N, and Fisher, D.M., 2012, **Structural geology algorithms, vectors and tensors**: Cambridge University Press, 289 p., ISBN 978-1-107-40138-9.

Cardozo, N., and Allmendinger, R.W., 2009, SSPX -- A program to compute strain from displacement/velocity data: *Computers and Geosciences*, v. 35, p. 1343-1357, doi: 10.1016/j.cageo.2008.05.008.

Means, W.D., 1976, **Stress and strain – Basic concepts of continuum mechanics for geologists**: Englewood Cliffs, New Jersey, Prentice-Hall, Inc., 339 p.

Savage, J.C., Gan, W., and Svarc, J.L., 2001, Strain accumulation and rotation in the Eastern California Shear Zone: *Journal of Geophysical Research*, v. 106, B10, P. 21,995-22,007.