Key Points:
- A unifying and robust framework for satellite array gradient and curl estimations is introduced
- The adaptive algorithm handles arbitrary numbers of satellites and degenerate arrays
- A numerically efficient implementation of the algorithm in Python is presented

Supporting Information:
- Supporting Information S1
- Supporting Information S2
- Supporting Information S3

Correspondence to:
J. Vogt,
j.vogt@jacobs-university.de

Robust Adaptive Spacecraft Array Derivative Analysis

J. Vogt[1], A. Blagau[1,2], and L. Pick[1,3]

[1]Department of Physics and Earth Sciences, Jacobs University, Bremen, Germany, [2]Institute for Space Sciences, Bucharest, Romania, [3]Deutsches GeoForschungsZentrum, Potsdam, Germany

Abstract Multispacecraft missions such as Cluster, Themis, Swarm, and MMS contribute to the exploration of geospace with their capability to produce gradient and curl estimates from sets of spatially distributed in situ measurements. This paper combines all existing estimators of the reciprocal vector family for spatial derivatives and their errors. The resulting framework proves to be robust and adaptive in the sense that it works reliably for arrays with arbitrary numbers of spacecraft and possibly degenerate geometries. The analysis procedure is illustrated using synthetic data as well as magnetic measurements from the Cluster and Swarm missions. An implementation of the core algorithm in Python is shown to be compact and computationally efficient so that it can be easily integrated in the various free and open source packages for the Space Physics and Heliophysics community.

Plain Language Summary The space environment of the Earth is studied using spacecraft measurements of physical variables. To resolve their variability in both space in time, constellations of several spacecraft such as Cluster, Themis, Swarm, and MMS are needed. This report combines methods to determine the spatial variability from distributed spacecraft observations in a unifying software framework. The approach works for arbitrary numbers of spacecraft in the constellation and adapts to possibly degenerate geometries that otherwise could lead to large errors. The analysis method is validated using synthetic data and then applied to magnetic measurements from the Cluster and Swarm missions. Implementation in the popular numerical software Python is shown to be compact and computationally efficient and presented in a way that facilitates easy integration in existing free and open source software.

1. Introduction

Exploration of the Earth's space environment and the heliosphere is driven by in situ measurements of fields and particles along the trajectories of dedicated spacecraft. Spatiotemporal ambiguities hamper the analysis and the proper interpretation of single-spacecraft data. In the past two decades, this issue has been addressed by a growing number of multisatellite missions such as Cluster, Themis, Swarm, and MMS. In the context of the four-spacecraft ESA mission Cluster, considerable attention was paid to the development of analysis tools for multispacecraft data (e.g., Paschmann & Daly, 1998, 2008). Of particular relevance is the multipoint analysis of spatial gradients and related derivatives such as the divergence and, very importantly, the curl of a vector field which gave rise to the term curlometer (Dunlop et al., 1988). The latter allows for an estimation of the electric current through Ampère's law (see Vogt, 2014, and Dunlop et al., 2018, for recent reviews).

The quality of multipoint gradient estimates crucially depends on geometrical properties of the spacecraft array. Robert et al. (1998a) reviewed geometric quality parameters proposed for the four-spacecraft Cluster constellation and concluded that the most complete geometric characterization is through the eigenvalues of the so-called volumetric tensor R_{vol} formed from dyadic products of the spacecraft position vectors. The array shape is captured by two parameters E (elongation) and P (planarity), defined in terms of eigenvalue ratios of R_{vol}. The impact of the tetrahedral shape parameters E and P on the errors of curl and gradient estimates was studied by Robert et al. (1998b) and Chanteur (2000). De Keyser et al. (2007) and also Hamrin et al. (2008) applied least squares inversion with error control to more general satellite configurations over a finite period of time, with localization enforced through the choice of covariance matrices or weighting functions. Shen et al. (2012) discussed stable gradient and curl estimators in terms of the effective array geometry resulting from the nonzero eigenvalues of the volumetric tensor.

This paper is concerned with a family of gradient and curl estimators that are based on a local least squares principle and formulated in terms of so-called reciprocal vectors q_σ, $\sigma = 1, 2, \ldots, S$, where S denotes the

number of spacecraft in the configuration. Originally formulated for the four-satellite case (Chanteur, 1998), reciprocal vectors were generalized to arrays of more than four spacecraft in noncoplanar geometries (Vogt et al., 2008a), to three-satellite constellations (Vogt et al., 2009), and also to virtual planar arrays with four or more spacecraft (Vogt et al., 2013), a case that is relevant for Swarm measurements in the auroral zone. Using measurements $\boldsymbol{b}_\sigma, \sigma = 1, 2, \ldots, S$ of a vectorial observable $\boldsymbol{b}$, the reciprocal vector concept allows for compact and suggestive estimator representations of spatial derivatives: The gradient matrix is estimated as $\sum_\sigma \boldsymbol{q}_\sigma \boldsymbol{b}_\sigma^{\mathrm{t}}$, the curl through $\sum_\sigma \boldsymbol{q}_\sigma \times \boldsymbol{b}_\sigma$, and the divergence as $\sum_\sigma \boldsymbol{q}_\sigma \cdot \boldsymbol{b}_\sigma$. For two-dimensional configurations, these estimators reproduce only the planar contributions to the spatial derivatives.

In section 2, we derive the *Robust Adaptive Spacecraft Array Derivative Analysis (rasada)* algorithm, that is, a compact recipe that integrates the various members of the reciprocal vector family of gradient and curl estimators and allows for efficient control of errors and degenerate array geometries. As demonstrated in a supplemented jupyter notebook, a fast and efficient implementation of the rasada algorithm in Python requires only a few lines of code, thus facilitating its adoption in free and open software toolboxes dedicated to the analysis of spacecraft data from geospace and heliospheric missions such as SpacePy (Morley et al., 2011), pysat (Stoneback et al., 2018), or HelioPy (Stansby et al., 2019), see also the recent review by Burrell et al. (2018). In section 3, the algorithm is validated and demonstrated through a series of applications to synthetic data and also to Cluster and Swarm measurements. All examples are transparently and reproducibly documented in the supplemented jupyter notebook, in support of the Geoscience Paper of the Future initiative (Gil et al., 2016).

2. Derivation of the Algorithm

Without losing generality and to ease the notation, the spacecraft positions are assumed to be given in a mesocentric reference frame: $\sum_\sigma \boldsymbol{r}_\sigma = \boldsymbol{0}$. The generalized reciprocal vectors $\boldsymbol{q}_\sigma$ are solutions of $\mathsf{R}_{\mathrm{pos}} \boldsymbol{q}_\sigma = \boldsymbol{r}_\sigma, \sigma = 1, 2, \ldots, S$, where $\mathsf{R}_{\mathrm{pos}} = \sum_\sigma \boldsymbol{r}_\sigma \boldsymbol{r}_\sigma^{\mathrm{t}}$ denotes the so-called *position tensor* (Vogt et al., 2008a; Vogt, 2014), that is related to the volumetric tensor through $\mathsf{R}_{\mathrm{vol}} = \frac{1}{S} \mathsf{R}_{\mathrm{pos}}$. Note that elsewhere, the position tensor is written without a subscript, but in this paper we reserve the plain symbol R for the *position matrix*, that is, the $(S \times 3)$ matrix R with the mesocentric spacecraft position vectors as its rows: $R_{\sigma j} = (r_\sigma)_j$, where $\sigma = 1, 2, \ldots, S$ and $j = 1, 2, 3$. This implies that $\mathsf{R}_{\mathrm{pos}} = \mathsf{R}^{\mathrm{t}} \mathsf{R}$.

The various flavors of reciprocal vectors allow for intuitive and compact representations of local linear least squares estimators for spatial derivative operators such as the gradient $\nabla \boldsymbol{b}$, the divergence $\nabla \cdot \boldsymbol{b}$, and the curl $\nabla \times \boldsymbol{b}$ of a vector field $\boldsymbol{b}$ (Chanteur, 1998; Vogt et al., 2008a) using measurements $\boldsymbol{b}_\sigma$ of $\boldsymbol{b}$ at the positions $\boldsymbol{r}_\sigma$:

$$\nabla \boldsymbol{b} \simeq \sum_\sigma \boldsymbol{q}_\sigma \boldsymbol{b}_\sigma^{\mathrm{t}}, \nabla \cdot \boldsymbol{b} \simeq \sum_\sigma \boldsymbol{q}_\sigma \cdot \boldsymbol{b}_\sigma, \nabla \times \boldsymbol{b} \simeq \sum_\sigma \boldsymbol{q}_\sigma \times \boldsymbol{b}_\sigma. \tag{1}$$

Here the gradient matrix $\mathsf{G} = \nabla \boldsymbol{b}$ is defined through the differential $\mathrm{d}\boldsymbol{b} = \mathsf{G}^{\mathrm{t}} \mathrm{d}\boldsymbol{r}$, to be consistent with the gradient vector $\boldsymbol{g} = \nabla h$ of a scalar field h: $\mathrm{d}h = \boldsymbol{g} \cdot \mathrm{d}\boldsymbol{r} = \boldsymbol{g}^{\mathrm{t}} \mathrm{d}\boldsymbol{r}$. In Cartesian coordinates $x_j, j = 1, 2, 3$, the matrix elements of $\mathsf{G} = \nabla \boldsymbol{b}$ are $G_{jk} = \partial_j b_k, k = 1, 2, 3$, and the gradient ∇b_k of a Cartesian component b_k is a column vector.

2.1. Reciprocal Matrix
Suppose the singular value decomposition (SVD) of the position matrix R is given by

$$\mathsf{R} = \mathsf{USV}^{\mathrm{t}}, \tag{2}$$

where U is a $(S \times 3)$ matrix and V a (3×3) matrix satisfying $\mathsf{U}^{\mathrm{t}}\mathsf{U} = \mathsf{E}$ and $\mathsf{V}^{\mathrm{t}}\mathsf{V} = \mathsf{E}$ with the (3×3) identity matrix E. The matrix V further satisfies $\mathsf{VV}^{\mathrm{t}} = \mathsf{E}$ and is thus orthogonal. The elements of the diagonal matrix S are the singular values of R. Then $\mathsf{R}^{\mathrm{t}} = \mathsf{VSU}^{\mathrm{t}}$, and the position tensor can be written in the form

$$\mathsf{R}_{\mathrm{pos}} = \mathsf{R}^{\mathrm{t}}\mathsf{R} = \mathsf{VSU}^{\mathrm{t}}\mathsf{USV}^{\mathrm{t}} = \mathsf{VS}^2\mathsf{V}^{\mathrm{t}}. \tag{3}$$

The column vectors of the orthogonal matrix V are the eigenvectors of $\mathsf{R}_{\mathrm{pos}}$. The corresponding eigenvalues $R_{\mathrm{pos}}^{(1)} \geq R_{\mathrm{pos}}^{(2)} \geq R_{\mathrm{pos}}^{(3)} \geq 0$ (note that $\mathsf{R}_{\mathrm{pos}}$ is positive semidefinite) are the squares of the singular values of R: $\mathsf{S}^2 = \mathrm{diag}\left(R_{\mathrm{pos}}^{(1)}, R_{\mathrm{pos}}^{(2)}, R_{\mathrm{pos}}^{(3)}\right)$ and

$$\mathsf{S} = \mathrm{diag}\left(R^{(1)}, R^{(2)}, R^{(3)}\right) = \mathrm{diag}\left(\sqrt{R_{\mathrm{pos}}^{(1)}}, \sqrt{R_{\mathrm{pos}}^{(2)}}, \sqrt{R_{\mathrm{pos}}^{(3)}}\right). \tag{4}$$

The set of linear equations $R_{pos} \boldsymbol{q}_\sigma = \boldsymbol{r}_\sigma, \sigma = 1, 2, \ldots, S$, is condensed into a single matrix equation $R_{pos} Q^t = R^t$. After transposition and since R_{pos} is symmetric, $QR_{pos} = R$. The rows of the matrix Q are formed by the generalized reciprocal vectors $\boldsymbol{q}_\sigma, \sigma = 1, 2, \ldots, S$. Note that elsewhere, the symbol Q is used to denote the tensor $\sum_\sigma \boldsymbol{q}_\sigma \boldsymbol{q}_\sigma^t$, also called the (generalized) reciprocal tensor. For consistency and to avoid confusion, we write $Q_{rec} = \sum_\sigma \boldsymbol{q}_\sigma \boldsymbol{q}_\sigma^t$ and reserve the plain symbol Q for the matrix of reciprocal vectors defined above, then $Q_{rec} = Q^t Q$. For brevity, we call Q the *reciprocal matrix*. Note that Q is the pseudoinverse of R.

Right-multiplying $QVS^2 V^t = QR_{pos} = R = USV^t$ with V yields

$$QVS^2 = US. \tag{5}$$

This is the defining equation for the reciprocal matrix Q, based on the SVD of the position matrix R, and corresponding to the set of equations $R_{pos} \boldsymbol{q}_\sigma = \boldsymbol{r}_\sigma, \sigma = 1, 2, \ldots, S$.

2.2. Nondegenerate Spacecraft Arrays in Three Dimensions

In the case of a regular three-dimensional array where not all $S \geq 4$ spacecraft are in one plane, the position tensor R_{pos} is invertible, and the generalized reciprocal vectors are $\boldsymbol{q}_\sigma = R_{pos}^{-1} \boldsymbol{r}_\sigma$ (Vogt et al., 2008a). In matrix representation, $Q^t = R_{pos}^{-1} R^t$ or $Q = R R_{pos}^{-1}$. Using $R_{pos}^{-1} = VS^{-2} V^t$, we write

$$Q = R R_{pos}^{-1} = USV^t VS^{-2} V^t = US^{-1} V^t. \tag{6}$$

In order to obtain the matrix form of the estimator $G \simeq \sum_\sigma \boldsymbol{q}_\sigma \boldsymbol{b}_\sigma^t$ for the gradient matrix $G = \nabla \boldsymbol{b}$ of a vectorial observable $\boldsymbol{b}$, the measurements $\boldsymbol{b}_\sigma, \sigma = 1, 2, \ldots, S$, are assumed to form the rows of a *data matrix* B. Since $\sum_\sigma \boldsymbol{q}_\sigma \boldsymbol{b}_\sigma^t = Q^t B$, we find $G \simeq Q^t B$.

The curl $\boldsymbol{c} = \nabla \times \boldsymbol{b}$ is estimated either through the skew-symmetric part $\frac{1}{2} \left(G - G^t \right)$ of the gradient matrix G or, after extraction of the row vectors $\boldsymbol{q}_\sigma$ of Q, directly as $\boldsymbol{c} \simeq \sum_\sigma \boldsymbol{q}_\sigma \times \boldsymbol{b}_\sigma$. The divergence can be obtained as $\mathrm{trace}(G)$ or from $\nabla \cdot \boldsymbol{b} \simeq \sum_\sigma \boldsymbol{q}_\sigma \cdot \boldsymbol{b}_\sigma$.

The gradient vector $\boldsymbol{g} = \nabla h$ of a scalar observable h is estimated as $\boldsymbol{g} \simeq Q^t \boldsymbol{h}$. Here the measurements $h_\sigma, \sigma = 1, 2, \ldots, S$, are assumed to form a data column vector $\boldsymbol{h}$.

2.3. Array Geometry and Error Control

In the two-dimensional case where at least three spacecraft reside in one plane but do not degenerate to a linear configuration, $R_{pos}^{(1)} \geq R_{pos}^{(2)} > 0$ and $R_{pos}^{(3)} = 0$. The third eigenvector is normal to the spacecraft plane and constitutes a singular dimension and without further information the normal gradient component cannot be estimated. Although a full inversion of the set of equations $R_{pos} \boldsymbol{q}_\sigma = \boldsymbol{r}_\sigma, \sigma = 1, 2, \ldots, S$, is not possible, we may compute minimum-norm solutions and still obtain gradient estimates in the spacecraft plane as $\nabla_p \boldsymbol{b} = \sum_\sigma \boldsymbol{q}_\sigma \boldsymbol{b}_\sigma^t$ (Vogt et al., 2009, 2013).

In matrix representation, we first compute a stable pseudoinverse of S through

$$S^- = \mathrm{diag} \left(1/R^{(1)}, 1/R^{(2)}, 0 \right), \tag{7}$$

and then $Q = US^- V^t$ (Vogt et al., 2009, 2013) with row vectors $\boldsymbol{q}_\sigma, \sigma = 1, 2, \ldots, S$, as before.

Finally, for one-dimensional arrays, $R_{pos}^{(1)} > 0$ and $R_{pos}^{(2)} = R_{pos}^{(3)} = 0$. In analogy with the two-dimensional case, we write $Q = US^- V^t$ with $S^- = \mathrm{diag} \left(1/R^{(1)}, 0, 0 \right)$.

In practice, the eigenvalues $R_{pos}^{(2)}$ and/or $R_{pos}^{(3)}$ may be nonzero but still very small, producing unstable estimates of spatial derivatives in the affected directions, that is, results that are very susceptible to instrumental and other random errors. Meaningful estimates can only be constructed in the subspace spanned by the eigenvectors corresponding to the larger eigenvalues, leading to an effective error control strategy closely associated with the geometry of the spacecraft array. Earlier studies following a similar approach include the reports of De Keyser et al. (2007), De Keyser (2008), Hamrin et al. (2008), and Shen et al. (2012).

Since this paper aims at a robust yet simple implementation of error control, we adopt a single tolerance parameter τ to stabilize the inversion of the position matrix. A direction along an eigenvector $\hat{v}_j$ is taken into account only if the corresponding singular value ratio $R^{(j)}/R^{(1)}$ relative to the largest value $R^{(1)}$ is larger than the prescribed tolerance parameter τ. As a canonical value, we suggest $\tau = 0.1$, corresponding to an

elongation of $E = 0.9$, and also a planarity of $P \sim 0.9$ if the first two singular values are of comparable size. In the two-dimensional case where the third direction is excluded a priori by construction (Vogt et al., 2013), the coefficient number of the two-dimensional position tensor corresponds to the inverse square tolerance parameter: $R_{\mathrm{pos}}^{(1)}/R_{\mathrm{pos}}^{(2)} = \tau^{-2} = 100$.

In the case of planar or even linear arrays, measurements do not offer information on spatial derivatives along missing dimensions. Additional information in the form of geometrical or physical constraints can be utilized to construct the missing components (see Vogt et al., 2009, 2013). Integration of constraint equations in the rasada algorithm should be straightforward but is beyond the scope of the present study and thus left for future work.

2.4. Error Estimation

Error analysis of gradient and curl estimators from the reciprocal vector family has been discussed, for example, by Chanteur (1998), Vogt and Paschmann (1998), and Vogt et al., (2008a, 2009, 2013). It turns out that positional errors are typically small compared to instrumental uncertainties. Another important contribution to the statistical error are random deviations from the supposedly linearly varying magnetic field profile, for example, small-scale magnetic fluctuations that do not show a correlation between the different sensors.

Assuming the effective (instrumental plus random deviations from linearity) statistical errors are isotropic and mutually uncorrelated, they can be captured by a scalar quantity δb, and the error covariance matrix of a gradient vector estimate $\boldsymbol{g} = \nabla b_k$ (for the Cartesian component b_k of a vectorial observable $\boldsymbol{b}$) is given by

$$\left\langle \delta\boldsymbol{g}\,\delta\boldsymbol{g}^{\mathrm{t}} \right\rangle = (\delta b)^2 \sum_{\sigma} \boldsymbol{q}_{\sigma}\boldsymbol{q}_{\sigma}^{\mathrm{t}} = (\delta b)^2 \mathsf{Q}^{\mathrm{t}}\mathsf{Q}. \tag{8}$$

The mean square magnitude error is thus

$$\left\langle |\delta\boldsymbol{g}|^2 \right\rangle = (\delta b)^2 \sum_{\sigma} |\boldsymbol{q}_{\sigma}|^2 = (\delta b)^2 \mathrm{trace}(\mathsf{Q}^{\mathrm{t}}\mathsf{Q}) = (\delta b)^2 \|\mathsf{Q}\|^2, \tag{9}$$

where $\|\mathsf{Q}\|$ is the Frobenius norm of the reciprocal matrix Q defined through $\|\mathsf{Q}\|^2 = \sum_{\sigma} |\boldsymbol{q}_{\sigma}|^2 = \mathrm{trace}\left(\mathsf{Q}^{\mathrm{t}}\mathsf{Q}\right)$. The mean square error of the divergence estimator has the same form as the square magnitude error:

$$\left\langle |\delta\nabla \cdot \boldsymbol{b}|^2 \right\rangle = (\delta b)^2 \mathrm{trace}(\mathsf{Q}^{\mathrm{t}}\mathsf{Q}) = (\delta b)^2 \|\mathsf{Q}\|^2. \tag{10}$$

The error covariance matrix of the estimated curl vector $\boldsymbol{c}$ can be written in the form

$$\left\langle \delta\boldsymbol{c}\,\delta\boldsymbol{c}^{\mathrm{t}} \right\rangle = (\delta b)^2 \left(\mathrm{trace}(\mathsf{Q}_{\mathrm{rec}})\mathsf{E} - \mathsf{Q}_{\mathrm{rec}} \right) = (\delta b)^2 \left(\|\mathsf{Q}\|^2\mathsf{E} - \mathsf{Q}^{\mathrm{t}}\mathsf{Q} \right), \tag{11}$$

where E denotes the identity tensor (see Vogt et al., 2019). In the special case of a planar spacecraft array, $\left\langle |\delta c_n|^2 \right\rangle = (\delta b)^2 \|\mathsf{Q}\|^2$ for the mean square error of the curl component normal to the spacecraft plane (Vogt et al., 2019).

2.5. Implementation

The preceding derivation yields the following algorithm for spatial derivative estimators from the reciprocal vector family, derived from a local least squares principle.

1. Store the mesocentric spacecraft position vectors $\boldsymbol{r}_{\sigma}, \sigma = 1, 2, \ldots, S$, as rows of the position matrix R.
2. Compute the SVD of the position matrix: $\mathsf{R} = \mathsf{U}\mathsf{S}\mathsf{V}^{\mathrm{t}}$.
3. Using the nonzero elements of the diagonal matrix S, form its stable pseudoinverse S^-.
4. Compute the reciprocal matrix Q, that is, the stable pseudoinverse of the position matrix as $\mathsf{Q} = \mathsf{U}\mathsf{S}^-\mathsf{V}^{\mathrm{t}}$. The rows of Q are the (generalized) reciprocal vectors $\boldsymbol{q}_{\sigma}, \sigma = 1, 2, \ldots, S$.
5. Assuming that the measurements $\boldsymbol{b}_{\sigma}, \sigma = 1, 2, \ldots, S$, of a vectorial observable $\boldsymbol{b}$ form the rows of a data matrix B, the gradient matrix $\mathsf{G} = \nabla\boldsymbol{b}$ is estimated as $\mathsf{G} \simeq \mathsf{Q}^{\mathrm{t}}\mathsf{B}$. The curl $\boldsymbol{c} = \nabla \times \boldsymbol{b}$ is estimated either using the skew-symmetric part of G or through $\boldsymbol{c} \simeq \sum_{\sigma} \boldsymbol{q}_{\sigma} \times \boldsymbol{b}_{\sigma}$, using the row vectors $\boldsymbol{q}_{\sigma}$ of Q.
6. Isotropic and mutually uncorrelated instrumental errors δb for each component b_k of $\boldsymbol{b}$ produce a gradient vector (∇b_k) error covariance matrix of the form $(\delta b)^2\mathsf{Q}^{\mathrm{t}}\mathsf{Q}$. The gradient magnitude error and, in the case of planar spacecraft arrays, the error of the normal curl component assume the form $\|\mathsf{Q}\|\,\delta b$ where $\|\mathsf{Q}\|$ is the Frobenius norm of the reciprocal matrix Q.
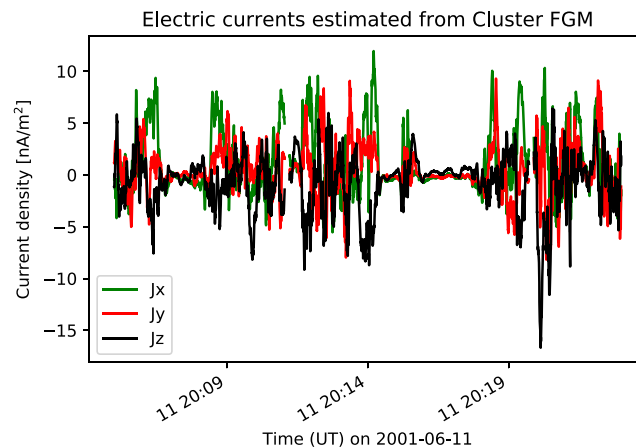
Figure 1. Application of the rasada algorithm to Cluster FGM measurements on 11 June 2001, 20:05–20:23 UT, to estimate the three components of the current density vector $\boldsymbol{j}$.

3. Applications

The rasada algorithm is demonstrated using a jupyter notebook, available as Supporting Information S1. It is implemented in Python. The numerical part of the procedure rests solely on the numpy package, graphical display utilizes matplotlib, CDF files are read using CDFlib, and the pandas package helps to select data and to deal with data gaps.

3.1. Synthetic Data, Nondegenerate Satellite Arrays

The first section of the jupyter notebook introduces the rasada algorithm by means of two types of nondegenerate synthetic array orbit data.

In the first example, the reference four-point array is a regular tetrahedron with unit distances between its vertices, multiplied by a length scale L, and then propagated with constant velocity to produce artificial array orbit data. Following Steps 1–4 of the rasada procedure, the reciprocal matrix Q is constructed. In Step 5, the spatial derivatives are estimated for a linear noise-free vector field $\boldsymbol{b} = \frac{1}{2}\boldsymbol{c} \times \boldsymbol{r}$ where $\boldsymbol{c} = \nabla \times \boldsymbol{b}$ is a constant vector.

The second example illustrates also Step 6, that is, error estimation. Since for the highly regular and symmetric array geometry of the first example all singular values are identical, implying that all potential correlations between the errors of the estimated curl components are zero, a less regular geometry is chosen where two vertices are shortened. The linear field is contaminated by white noise $\delta\boldsymbol{b}$ to produce an ensemble of erroneous measurements of the form $\boldsymbol{b}(\boldsymbol{r}) = \frac{1}{2}\,\boldsymbol{c} \times \boldsymbol{r} + \delta\boldsymbol{b}(\boldsymbol{r})$. Ensemble statistics are compared with the error estimates produced through Step 6 of the rasada algorithm. Within statistical uncertainties, the results are fully consistent.

3.2. Synthetic Data, Partially Degenerate Satellite Arrays

The second section of the jupyter notebook makes use of partially degenerate synthetic satellite arrays to illustrate rasada's geometry control. Starting from a regular tetrahedron, only one spacecraft is allowed to move uniformly toward the plane spanned by the other three, temporarily producing a planar configuration. The effective array dimension parameter, computed as the number of singular values that are kept in rasada Step 3 when the pseudoinverse S^- is constructed, jumps from three to two for the degenerate case, and the in-plane gradient component is obtained. In the same spirit, a transition from a regular quad to a linear configuration is simulated, also producing the expected results. Finally, the important case of normal curl estimation from virtual planar quads is considered as this is relevant for the determination of field-aligned currents (FACs) from the pair of SwA and SwC spacecraft (see Blǎgǎu & Vogt, 2019, and section 3.4). Using ensembles of synthetic measurements, also, the error estimates are tested and confirmed.

3.3. Current Vectors Estimated From Cluster FGM Data

In the third section of the supplemented jupyter notebook, the rasada algorithm is applied to Cluster multisatellite fluxgate magnetometer (FGM) data measured on 11 June 2001 in the time interval 20:05–20:23 UT during a magnetopause crossing. The event was analyzed by Dunlop and Balogh (2005) and is also
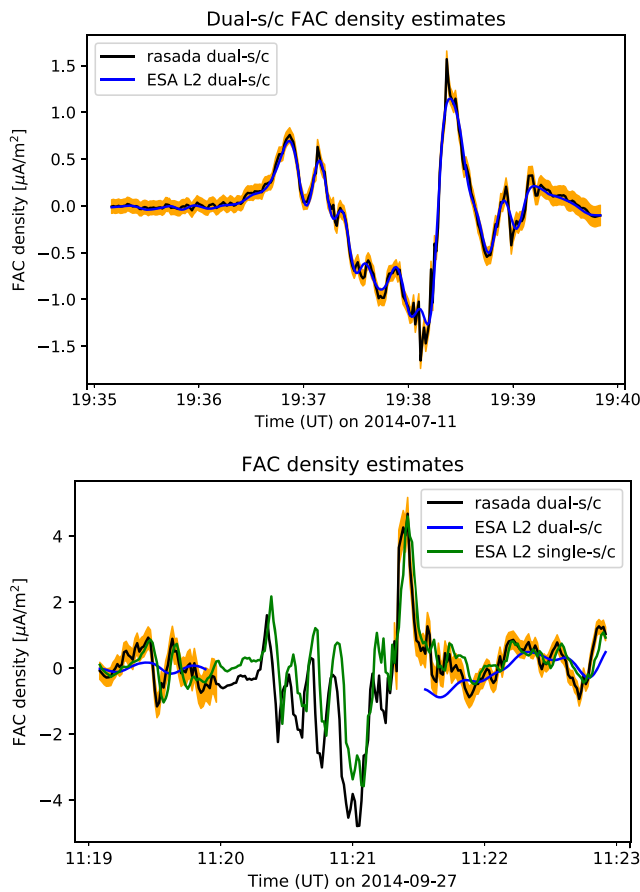
Figure 2. Applications of the rasada algorithm to VFM measurements of the lower pair of Swarm satellites, SwA and SwC, and comparison with the Level-2 FAC data products provided by ESA. The orange-shaded region indicates the statistical error margin for the rasada dual-spacecraft FAC estimate and is missing where the configuration degenerates to a quasi-linear array. (top) 11 July 2014, 19:35–19:40 UT. (bottom) 27 September 2014, 11:19–11:23 UT. The single-spacecraft FAC density profile is produced from the two Level-2 FAC data products for SwA and SwC after suitable smoothing and averaging.

included on the multispacecraft webpage of the Cluster Science Archive (CSA; https://www.cosmos.esa.int/web/csa) as an application of the curlometer technique (Dunlop et al., 1988). CDF files from the CSA contain data gaps that are identified, flagged, and eliminated. The three components of the electric current vector $\boldsymbol{j} = \mu_0^{-1}\nabla \times \boldsymbol{b}$ are computed and displayed (see Figure 1). The current components compare nicely with the results of Dunlop and Balogh (2005) and the plot published on page 18 of the CSA multispacecraft tutorial (ESDC-CSA-TN-001: The Curlometer Technique, Issue 1 Rev 1, 13 December 2016).

3.4. FAC Densities Estimated From Swarm VFM Data

The fourth section of the supplemented jupyter notebook deals with the estimation of auroral FAC densities using vector field magnetometer (VFM) data measured by ESA's Swarm spacecraft. To produce FAC density estimates from planar array data, the component of the curl vector normal to the spacecraft plane (i.e., the only meaningful component in this geometry) is extracted, the normal unit vector is obtained as the third row vector of the matrix V^t, and the inclination of the ambient magnetic field with respect to the normal unit vector is computed. The procedure is summarized in a general-purpose function that can be applied to planar arrays formed by arbitrary numbers of satellites.

The first time interval chosen to demonstrate rasada FAC density estimation from planar satellite configurations is 08:49–08:52 UT on 23 May 2014. The third Swarm spacecraft SwB was sufficiently close to the lower pair SwA and SwC to yield nondegenerate three-point configurations and thus meaningful three-satellite FAC estimates. The resulting profile (displayed in the supplemented jupyter notebook) compares well with the FAC density estimate obtained by Blaǧaǧu and Vogt (2019) who studied the same event using planar reciprocal vectors (Vogt et al., 2009). The only notable difference is a small vertical offset of a few tenths of $\mu A/m^2$ at the beginning of the time interval with the lower satellite pair close to the geographic pole, possibly related to nonlinear contributions from the ambient magnetic field present in the Level-1b magnetic field data (total field without model subtraction) that serve as input data to rasada.

To produce dual-spacecraft FAC estimates from VFM data measured by the two close Swarm satellites SwA and SwC, virtual four-point configurations are constructed using shifted positions and magnetic field data (Ritter & Lühr, 2006; Ritter et al., 2013; Vogt et al., 2013). Figure 2 shows rasada FAC density estimates in comparison with Level-2 FAC data products provided by ESA. The upper diagram shows the results for the time interval 19:35–19:40 UT on 11 July 2014, with stable and nondegenerate virtual four-point configurations. The Level-2 dual-s/c FAC data product is produced from filtered magnetic field data; hence, the overall profile is smoother than the rasada estimate that operates on the raw Level-1b magnetic field measurements without any filtering. Otherwise, the two dual-s/c FAC density estimates compare very well. This is remarkable because the rasada algorithm is applied to magnetic field data not only without any filtering but also without subtraction of a model magnetic field.

The lower diagram of Figure 2 shows a comparison of FAC density estimates for the time interval 11:19–11:23 UT on 27 September 2014. Since the chosen time interval includes the crossover point of the two satellites in the southern hemisphere, the virtual quad configuration temporarily degenerates to a quasi-linear array so that rasada's adaptability to the effective array dimension can be tested. To facilitate a comparison with the single-satellite (SwA and SwC) Level-2 FAC data products, the two time series are condensed into one profile after smoothing and averaging. While the ESA Level-2 dual-satellite FAC data product is not computed in the so-called exclusion zone within four degrees around the geographic pole due to errors caused by array degeneracy, the rasada dual-satellite FAC density estimate is available in the entire time interval and very consistent with average single-spacecraft FAC density (apart from a small vertical offset, possibly related to

differences in the input data). The rasada algorithm, through geometry adjustment based on the degeneracy parameter τ, provides proper dual-satellite estimates where the array can be considered two-dimensional and falls back to a single-spacecraft approach when the virtual quad degenerates into a quasi-linear array.

4. Conclusions and Outlook

In the context of spatial derivative estimation from multisatellite array data, the rasada algorithm introduced in this paper combines all earlier approaches of the reciprocal vector family (Chanteur, 1998; Vogt et al., 2008a, 2009, 2013) in a unifying framework. By construction, the procedure adapts not only to the number of spacecraft in the array but also to its effective geometrical dimension and is robust in the sense that the errors resulting from degenerate configurations are constrained. The chosen implementation in Python is fully vectorized and thus efficient in terms of processing time and computational resources. Integration in Python-based analysis packages such as SpacePy (Morley et al., 2011), pysat (Stoneback et al., 2018), or HelioPy (Stansby et al., 2019) is expected to be straightforward.

Synthetic data are used to demonstrate the core qualities of the rasada procedure, namely, efficient gradient and curl estimation and array geometry control. The error formulas are shown to be consistent with the results of numerical Monte Carlo experiments. The procedure is validated through a reanalysis of Cluster FGM measurements of a magnetopause crossing and then illustrated further using Swarm VFM data to estimate the density of auroral FACs.

The versatile algorithm is not restricted to applications of the type discussed in this report. For instance, using dual-satellite measurements of the Swarm lower pair SwA and SwC, the number of points in the virtual configuration can be easily extended from four to arbitrary multiples of two by adding additional data along the track of each satellite, thus realizing data-adaptive smoothing instead of scale-based filtering as currently employed in the procedure to produce the official ESA dual-satellite FAC data product (Ritter et al., 2013). Also for Swarm three-satellite configurations, two or more position and magnetic field along-track data can be stacked to produce fully three-dimensional array configurations, so that for stationary situations, the full current vector can be obtained as done recently by Dunlop et al. (2015) using the curlometer technique.

In addition to multisatellite spatial derivative estimation, reciprocal vectors are employed beneficially, for example, in discontinuity analysis (Harvey, 1998; Vogt et al., 2011) and in wave vector estimation (Vogt et al., 2008b). It will be straightforward to generalize the rasada algorithm to those multispacecraft analysis tasks.

References

Blagăaău, A., & Vogt, J. (2019). Multi-point field-aligned current estimates with Swarm. *Journal of Geophysical Research: Space Physics*, *124*, 6869–6895. https://doi.org/10.1029/2018JA026439

Burrell, A. G., Halford, A., Klenzing, J., Stoneback, R. A., Morley, S. K., Annex, A. M., et al. (2018). Snakes on a spaceship—An overview of Python in heliophysics. *Journal of Geophysical Research: Space Physics*, *123*, 10,384–10,402. https://doi.org/10.1029/2018JA025877

Chanteur, G. (1998). Spatial interpolation for four spacecraft: Theory. *ISSI Scientific Reports Series*, *1*, 371–393.

Chanteur, G. (2000). Accuracy of field gradient estimations by Cluster: Explanation of its dependency upon elongation and planarity of the tetrahedron. Proceedings of the cluster-ii workshop multiscale/multipoint plasma measurements, imperial college, london, 22-24 september 1999, ESA SP-449 p. 265-268.

De Keyser, J. (2008). Least-squares multi-spacecraft gradient calculation with automatic error estimation. *Annales Geophysicae*, *26*(11), 3295–3316.

De Keyser, J., Darrouzet, F., Dunlop, M. W., & Décréau, P. M. E. (2007). Least-squares gradient calculation from multi-point observations of scalar and vector fields: Methodology and applications with Cluster in the plasmasphere. *Annales Geophysicae*, *25*(4), 971–987.

Dunlop, M. W., & Balogh, A. (2005). Magnetopause current as seen by Cluster. *Annales Geophysicae*, *23*(3), 901–907.

Dunlop, M. W., Haaland, S., Dong, X.-C., Middleton, H. R., Escoubet, C. P., Yang, Y.-Y., et al. (2018). Multipoint analysis of electric currents in geospace using the curlometer technique. In A. Keiling, O. Marghitu, & M. Wheatland (Eds.), *Electric currents in geospace and beyond* (vol. 235, pp. 67–80), Washington DC American Geophysical Union Geophysical Monograph Series. Hoboken: Wiley.

Dunlop, M. W., Southwood, D. J., Glassmeier, K. H., & Neubauer, F. M. (1988). Analysis of multipoint magnetometer data. *Advances in Space Research*, *8*(9-10), 273–277.

Dunlop, M. W., Yang, Y.-Y., Yang, J.-Y., Lühr, H., Shen, C., Olsen, N., et al. (2015). Multispacecraft current estimates at Swarm. *Journal of Geophysical Research: Space Physics*, *120*, 8307–8316. https://doi.org/10.1002/2015JA021707

Gil, Y., David, C. H., Demir, I., Essawy, B. T., Fulweiler, R. W., Goodall, J. L., et al. (2016). Toward the geoscience paper of the future: Best practices for documenting and sharing research from data to software to provenance. *Earth and Space Science*, *3*(10), 388–415.

Hamrin, M., Rönnmark, K., Börlin, N., Vedin, J., & Vaivads, A. (2008). GALS—Gradient analysis by least squares. *Annales Geophysicae*, *26*(11), 3491–3499.

Harvey, C. C. (1998). Spatial gradients and the volumetric tensor. *ISSI Scientific Reports Series*, *1*, 307–322.

Morley, S. K., Koller, J., Welling, D. T., Larsen, B. A., Henderson, M. G., & Niehof, J. T. (2011). Spacepy—A Python-based library of tools for the space sciences. In *Proceedings of the 9th Python in science conference, austin, tx.*, SciPy 2010.

Paschmann, G., & Daly, P. W. (1998). Analysis methods for multi-spacecraft data. The International Space Science Institute, Hallerstr. 6, CH-3012 Bern, Switzerland. ESA Publications Division, Noordwijk, The Netherlands. Copyright: 1998 ISSI/ESA. ISSI Scientific Report SR-001.

Paschmann, G., & Daly, P. W. (2008). Multi-spacecraft analysis methods revisited.

Ritter, P., & Lühr, H. (2006). Curl-b technique applied to Swarm constellation for determining field-aligned currents. *Earth, Planets, and Space*, *58*, 463–476.

Ritter, P., Lühr, H., & Rauberg, J. (2013). Determining field-aligned currents with the Swarm constellation mission. *Earth, Planets, and Space*, *65*, 1285–1294.

Robert, P., Dunlop, M. W., Roux, A., & Chanteur, G. (1998b). Accuracy of current density estimation. *ISSI Scientific Reports Series*, *1*, 395–418.

Robert, P., Roux, A., Harvey, C. C., Dunlop, M. W., Daly, P. W., & Glassmeier, K. H. (1998a). Tetrahedron geometric factors. *ISSI Scientific Reports Series*, *1*, 323–348.

Shen, C., Rong, Z. J., Dunlop, M. W., Ma, Y. H., Li, X., Zeng, G., et al. (2012). Spatial gradients from irregular, multiple-point spacecraft configurations. *Journal of Geophysical Research: Space Physics*, *117*, A11207. https://doi.org/

Stansby, D., Rai, Y., Broll, J., Shaw, S., Aditya, & Badger, T. G. (2019). heliopython/heliopy: Heliopy 0.6.5. Zenodo.

Stoneback, R. A., Burrell, A. G., Klenzing, J., & Depew, M. D. (2018). PYSAT: Python satellite data analysis toolkit. *Journal of Geophysical Research: Space Physics*, *123*, 5271–5283. https://doi.org/10.1029/2018JA025297

Vogt, J. (2014). Analysis of data from multi-satellite geospace missions. In W. Freeden, M. Z. Nashed, & T. Sonar (Eds.), *Handbook of Geomathematics* pp. 1–28). Berlin Heidelberg: Springer.

Vogt, J., Albert, A., & Marghitu, O. (2009). Analysis of three-spacecraft data using planar reciprocal vectors: Methodological framework and spatial gradient estimation. *Annales Geophysicae*, *27*(8), 3249–3273.

Vogt, J., Blagau, A., Bunescu, C., & He, M. (2019). Local least squares analysis of auroral currents. International Space Science Institute (ISSI), to be published in November 2019.

Vogt, J., Haaland, S., & Paschmann, G. (2011). Accuracy of multi-point boundary crossing time analysis. *Annales Geophysicae*, *29*, 2239–2252.

Vogt, J., Narita, Y., & Constantinescu, O. D. (2008b). The wave surveyor technique for fast plasma wave detection in multi-spacecraft data. *Annales Geophysicae*, *26*(7), 1699–1710.

Vogt, J., & Paschmann, G. (1998). Accuracy of plasma moment derivatives. *ISSI Scientific Reports Series*, *1*, 419–447.

Vogt, J., Paschmann, G., & Chanteur, G. (2008a). Reciprocal vectors. *ISSI Scientific Reports Series*, *8*, 33–46.

Vogt, J., Sorbalo, E., He, M., & Blagau, A. (2013). Gradient estimation using configurations of two or three spacecraft. *Annales Geophysicae*, *31*, 1913–1927.