

## A Self-Contained Identification Scheme for Eddies in Drifter and Float Trajectories

MATTHIAS LANKHORST

*IFM-GEOMAR, Kiel, Germany*

(Manuscript received 14 March 2005, in final form 3 April 2006)

### ABSTRACT

It is becoming increasingly recognized that the eddy field plays an important—possibly dominating—role for oceanic motions in many aspects (e.g., transport of properties and risk assessment in the case of extreme events). This motivates the study of individual eddy events. In the Lagrangian coordinate system, vorticity possibly associated with eddies appears in two forms: as shear vorticity between neighboring particles, and as curvature of the trajectory of a single particle. Typical field experiments in physical oceanography using surface drifters or subsurface floats do not reach data densities high enough to produce enough encounters of drifters to calculate shear vorticity between them. However, curvature in individual tracks is easily observed. This study presents a methodology that extracts segments from within a trajectory that are “looping,” which will be interpreted as a drifter being caught in an eddy. The method makes use of autoregressive processes, a simple type of stochastic processes, which easily enables a fit to the nonperfectly shaped trajectory data usually expected from field experiments. These processes also deliver frequency and persistence of the detected eddies by a very simple calculation, which makes the methodology highly suited for automatized scanning of larger datasets.

### 1. Introduction

The imprint of eddy vorticity in particle trajectories can be by shear and curvature. With field experiments in mind, data density is typically low and thus insufficient to deduce synoptic shear vorticity. The search for eddies in such trajectories is then restricted to analyzing curvature. With the aim of finding coherent vortices and deducing their internal structure, one would want to identify several revolutions around an eddy center, in contrast to shorter arclike “curves” (which may well be caused by a coherent eddy nearby). The whole issue of identifying coherent eddy structures in trajectories is then reduced to finding “looping” segments within a trajectory, and the main difficulty is that data from field experiments is noisy and of irregular shape.

The most natural way to deal with the issue is obviously inspection by eye, which has been applied frequently in oceanographic research (e.g., Shoosmith et al. 2005; Fratantoni and Richardson 2006). However, this is time-consuming and possibly subject to human

error or bias. A different approach is made by Boebel et al. (2003), who use a computer algorithm to derive curvature from the (smoothed) tracks and count revolutions accordingly. However, they use threshold values for curvature and duration of the eddy event, and allow for equivalently defined exceptions from these. All of these thresholds are likely to depend on the flow characteristics of the region under study. Roughly at the same time, a software package called Drifter Analysis (DANA) was developed by Lankhorst (2003), which contains an eddy detection algorithm based mainly on the fit of a stochastic process [an autoregressive (AR) process]. This algorithm has now received further upgrades, tuning, and validation using the work by Shoosmith et al. (2005, from now on referred to as SRBR), and is the subject of this manuscript.<sup>1</sup> Using the AR process has the advantage that the fundamental behavior of the underlying time series (i.e., oscillating or not, including persistence and period time scales) is easily read off from the fit parameters. The goal is to derive a method that is completely self-contained and requires no initial guess for any parameter (like threshold curvatures or initial guesses for time scales), al-

---

*Corresponding author address:* Matthias Lankhorst, Leibniz-Institut für Meereswissenschaften (IFM-GEOMAR), Düsternbrookweg 20, 24105 Kiel, Germany.  
E-mail: mlankhorst@ifm-geomar.de

---

<sup>1</sup> For interested readers, the DANA software is freely available upon request to the author (it is a set of routines for MatLab).

though the method of course uses such thresholds internally. The mathematical (i.e., less tied to physical quantities) character of these thresholds gives rise to the hope that the method is useful for very different flow regimes. In fact, it has produced sensible results during test runs with RAFOS and SOFAR float data from depths between 700 and 2600 m from the North and tropical Atlantic. These test runs covered the relatively wide ranges of Lagrangian integral time scales from approximately 3–10 days and eddy kinetic energies between 5 and 100 cm<sup>2</sup> s<sup>-2</sup> (Lankhorst and Zenk 2006).

## 2. Methodology

In this section, the methodology used to detect looping parts within a trajectory (called “loopers”) is described. Because of their fundamental importance to the method, AR processes are introduced first. This is followed by a listing of all criteria that need to be fulfilled before a trajectory segment is considered to be looping. A comparison with the study by SRBR, who identified eddies in float trajectories by visual inspection, finishes this section.

### a. About AR processes

Autoregressive processes are a class of linear stochastic processes. An autoregressive process of order  $n$  (AR $n$  process) is one that describes a process variable  $z_t$  by an equation such as

$$z_t = \phi_1 z_{t-1} + \phi_2 z_{t-2} + \dots + \phi_n z_{t-n} + \epsilon_t \quad (1)$$

The  $\phi_i$  are constant coefficients,  $t$  is the time (in discrete, equally separated steps), and  $\epsilon_t$  is a random noise process as a forcing. We are therefore looking at *time series* of a variable  $z_t$ . The name *autoregressive* comes from the fact that  $z_t$  is expressed in terms of itself at previous times. For interested readers, a very detailed discussion of time series analysis is given by Box et al. (1994). The following paragraphs try to summarize the aspects of AR processes that are important to the use presented here.

An AR process is *stationary* (i.e., all statistical moments are constant with time) exactly if all (possibly complex) solutions  $L$  of the *characteristic equation*

$$1 - \phi_1 L - \phi_2 L^2 - \dots - \phi_n L^n = 0 \quad (2)$$

are outside of the unit circle. This shall be assumed true from now on.

The normalized autocorrelation function  $R_z(\tau)$ ,

where  $\tau$  is the time lag, satisfies the associated homogeneous equation [cf. Eq. (1) without the forcing term]:

$$R_z(\tau) = \phi_1 R_z(\tau - 1) + \phi_2 R_z(\tau - 2) + \dots + \phi_n R_z(\tau - n). \quad (3)$$

The power spectral density  $p_z$  as a function of frequency  $f$  is then given by

$$p_z(f) = \frac{2\overline{\epsilon^2}}{|1 - \phi_1 e^{-2\pi i f} - \phi_2 e^{-4\pi i f} - \dots - \phi_n e^{-2\pi i n f}|^2}$$

with  $0 \leq f \leq \frac{1}{2}$ . (4)

Note that the shape of both the autocorrelation and spectrum is independent of the forcing term  $\epsilon_t$ , which can only influence the overall level as a constant factor.

Equations (1) and (3) can also be rewritten as linear difference equations of order  $n$ , after introducing some  $\Delta_z$  and  $\Delta_t$  terms. This emphasizes the close relationship to linear differential equations, which the reader may be more familiar with. The linearity (i.e., additivity and homogeneity of the solutions) is valid for both the difference and the differential versions, which is the underlying reason why these processes are called linear.

It is therefore not surprising that a second-order autoregressive (AR2) process resembles a harmonic oscillator with stochastic forcing  $\epsilon_t$ . Depending on the coefficients  $\phi_{1,2}$ , the solutions to the homogeneous Eq. (3) can be damped exponentials only (overdamped oscillator), exponentially damped sinusoids (underdamped oscillator), or the aperiodic, degenerated solution in between (critically damped oscillator). In terms of the characteristic polynomial [Eq. (2)], this means two real roots, two complex conjugates, or a degeneration into one root, respectively. The latter is defined by the parabola

$$\phi_1^2 + 4\phi_2 = 0 \quad (5)$$

for an AR2 process.

Let us now restrict ourselves to stationary, oscillating (i.e., underdamped) AR2 processes. Their coefficients are “below” the parabola of Eq. (5), and their autocorrelation is described by

$$R_z(\tau) = e^{-d\tau} \cos(2\pi f_0 \tau),$$

where

$$d = -\ln(\sqrt{-\phi_2}) \quad \text{and} \quad f_0 = \frac{1}{2\pi} \arccos\left(\frac{\phi_1}{2\sqrt{-\phi_2}}\right). \quad (6)$$

The two coefficients  $\phi_{1,2}$  can be transformed into two time scales: a damping time scale  $d$  and an oscillation frequency  $f_0$ . Here  $d$  is the decay scale of the enveloping exponential and a measure of how long an oscillating motion may be expected to exist coherently. Note that  $d$  is real, because  $\phi_2 < -\frac{1}{4}\phi_1^2 \leq 0$ . Figure 1 shows two examples of AR2 processes with identical  $f_0$  but different  $d$ . Both time series oscillate at frequencies near  $f_0$ , but the one with weaker damping does so in a clearer and more energetic way. This behavior is also apparent in the autocorrelation and spectrum (also shown in Fig. 1).

Stationary AR2 processes are those with coefficients within a triangle in  $\phi_1$ - $\phi_2$  space, the corners of which are  $(-2, -1)$ ,  $(2, -1)$ , and  $(0, 1)$ . This triangle and the parabola [Eq. (5)] are indicated in the inset in Fig. 2. The remaining figure discusses numerous algorithms to fit an AR process to a given time series (i.e., to find the coefficients  $\phi$ ... and the strength of the forcing,  $\epsilon^2$ , that best match the original time series). The Signal Processing Toolbox [(SigProcTbx) from MatLab (more information available online at [www.mathworks.com](http://www.mathworks.com))] software contains four such algorithms (called aryule, arburg, arcov, armcov), of which the first two use fairly common techniques [the Yule-Walker equations (see Box et al. 1994) and Burg's algorithm (see Press et al. 1993)]. The performance of these algorithms was tested by challenging them with artificial AR2 time series with known coefficients. The coefficients were chosen to be in a range that would be expected from oceanic float data: both  $d$  and  $f_0$  are assumed to be small, which leads to coefficients in the lower-right-hand corner of the "stationarity triangle." It turned out that the simple aryule method performs worse than the other three routines (between the results of which there was almost no difference), so that arburg was selected for the eddy detection process. It is interesting to note that all fitting routines seem to reproduce the frequency better than the damping scale. This can be seen from the clouds of scattered points around the true values, which are aligned parallel to the parabola of Eq. (5) (lines of equal  $f_0$  are also roughly parallel to this parabola).

For the eddy detection process, it was necessary to apply a smoothing filter to the data prior to the AR2 fit to ensure that the fit detects the desired signal. Repeated failures to do so were often related to high-frequency "noise" in the raw data (e.g., poor positioning or unresolved tidal-inertial oscillations), or simply two superimposed eddylike features (a small, high-frequency eddy orbiting a large, low-frequency object). In almost every case, a simple low-pass filter was sufficient to suppress the unwanted signal, and has therefore been applied as a standard method. However, this

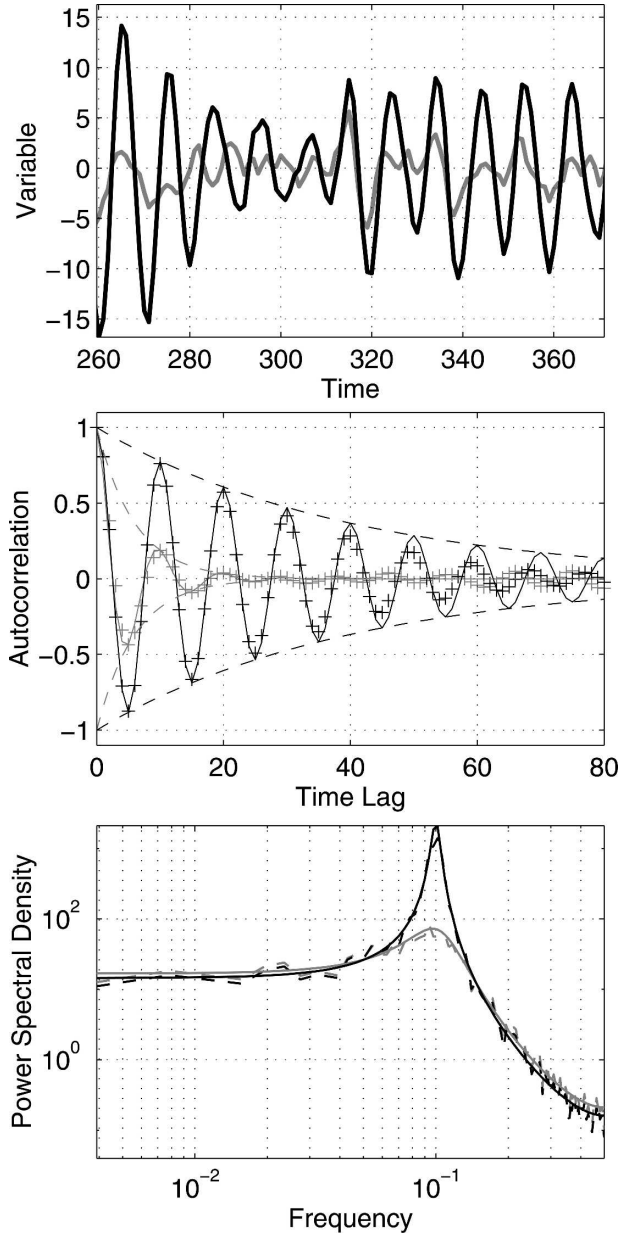


FIG. 1. Example of two artificially generated time series of AR2 processes. Forcing and fundamental frequency ( $f_0 = (1/10)$ ) were identical for both, but decay time varied [gray: $d = (1/6)$ , black: $d = (1/40)$ ]. (top) Close-up view of the time series. (middle) Sample (crosses) and theoretical [solid lines, from Eq. (3)] autocorrelation functions [dashed: enveloping exponentials from Eq. (3)]. (bottom) Sample (dashed) and theoretical [solid, from Eq. (4)] spectra.

carries an inconsistency with it: the spectrum of an AR2 process has a distinct shape (one spectral line with a decay proportional to  $f^{-2}$  at the high-frequency end), which is possibly very different from that of the filtered time series. As an ad hoc validation of the procedure, an AR2 fit was applied to artificial time series with

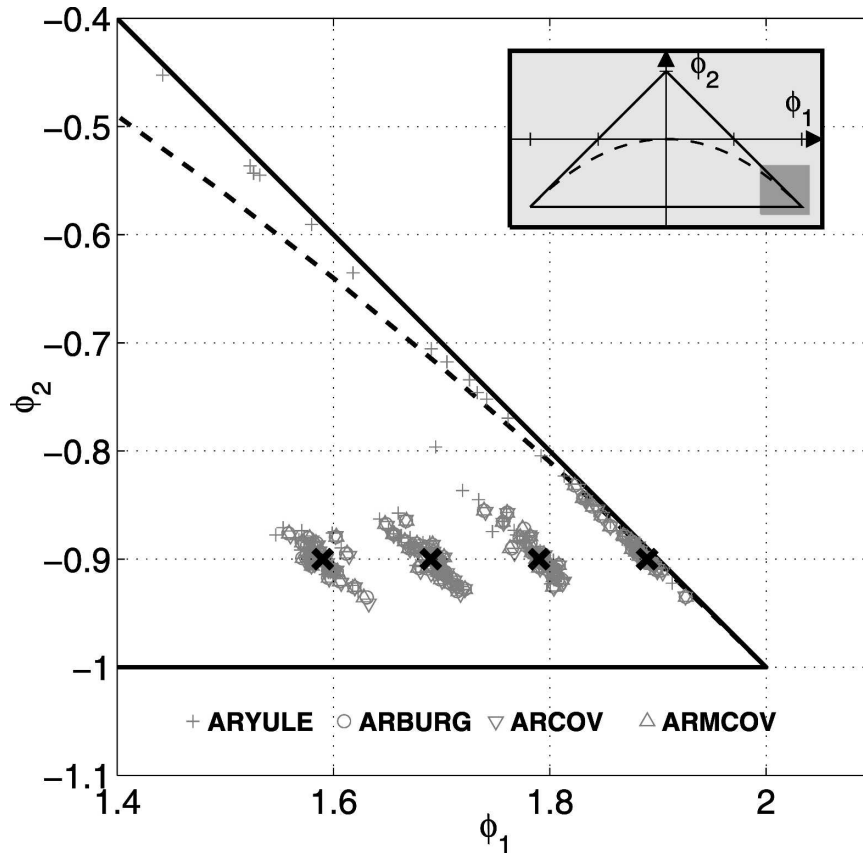


FIG. 2. Ad hoc test of AR-fitting routines. Time series with known AR2 coefficients (black diagonal crosses) were created, and four fit routines were challenged to recreate these coefficients (different gray symbols). The aryule algorithm produced the worst results, while the results from the other three routines were almost identical. The arburg algorithm was chosen for the eddy detection process. Black solid lines indicate the triangle in which AR2 processes are stationary, and the dashed line is the parabola that separates periodic (below the parabola) from aperiodic (above it) solutions. The inset in the upper-right-hand corner shows an overview of the entire stationarity triangle.

known AR2 parameters. Then, a filter was applied to the time series, and the fit routine was applied again. The results of this test are shown in Fig. 3: in all cases, the fundamental frequency of the AR2 process was found with great accuracy. The damping scale is reduced because filtering removes some of the noise. Nevertheless, the spectrum calculated from Eq. (4) using the coefficients from the fit to the filtered time series resembles that of the original time series (emphasizing the spectral line at  $f_0$  due to reduced noise), which generates exactly the behavior wanted for the eddy detection procedure. To create Fig. 3, a high-order Butterworth filter with a bandpass between  $\frac{1}{4}$  and 4 times the fundamental frequency of the AR2 time series was used. For the eddy detection (cf. section 2b), a much weaker filter will be applied, so that this may be considered safe.

### b. Eddy detection criteria

The actual search for looping parts within a trajectory works as follows and is included in the DANA software by the author.

- 1) The position data of the trajectory is smoothed with a digital filter. As a careful default, a third-order Butterworth low-pass filter with a cutoff period of  $(1/20)$  of the window length (cf. next list item) is applied. The MatLab SigProcTbx routines to create and apply the filter are `butter` and `filtfilt`. From the smoothed trajectory, zonal ( $u$ ) and meridional ( $v$ ) velocities are computed by simple differences.
- 2) The trajectory is split into overlapping windows. As a default for the window length, four passes with lengths of 2, 4, 8, and 16 times the Lagrangian integral time scale  $T_i$  are suggested. Here  $T_i$  should be

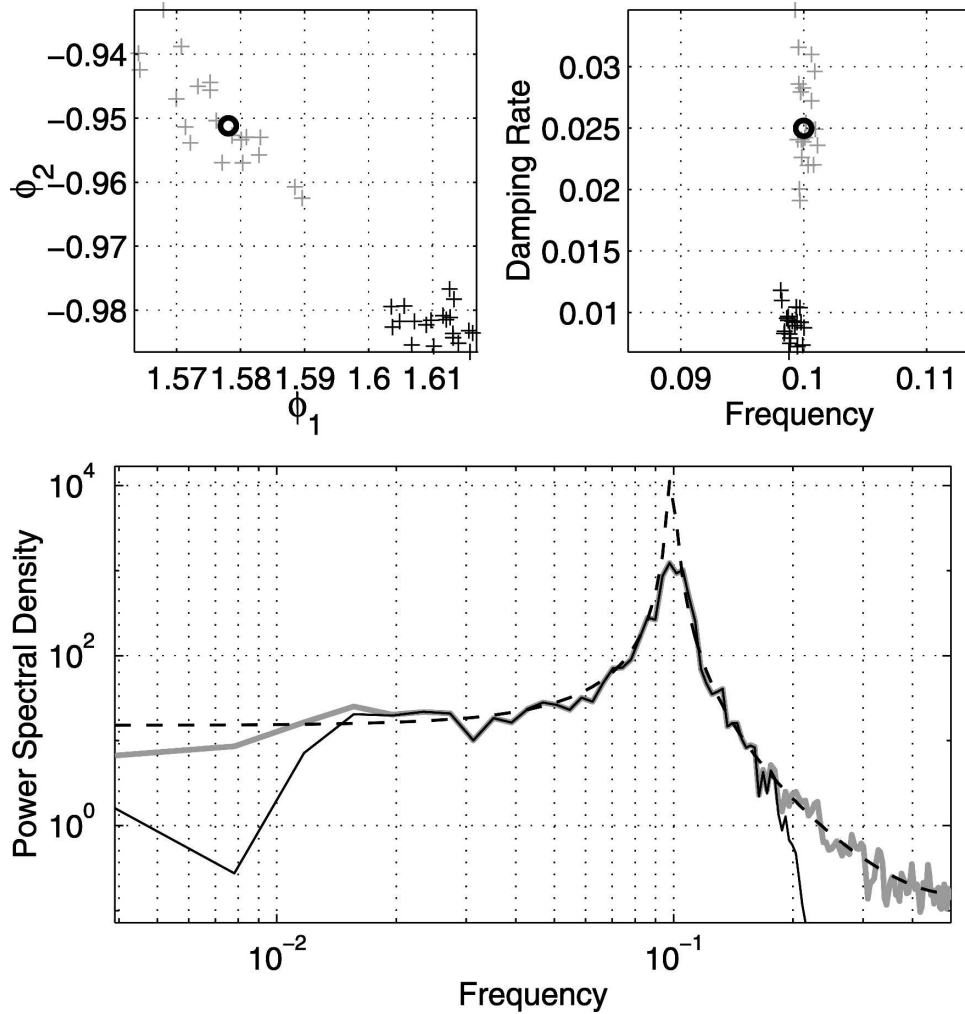


FIG. 3. Test of AR2 fits applied to filtered time series. (top left) Time series with known AR2 (black circles) parameters were created, and a fit routine applied to reconstruct them (gray crosses). Then, a high-order bandpass filter was applied to the time series, and the AR2 fit routine was applied again (black crosses). (top right) The fundamental frequency was still found with very good accuracy, while the damping scale was reduced due to the fact that the filter reduced some of the noise. (bottom) For one realization, spectra are shown [gray solid: sample spectrum of the original time series, black solid: sample spectrum of the filtered time series, black dashed: theoretical spectrum from Eq. (4) using the coefficients from the fit to the filtered time series]. The overall impression is that filtering does not affect analysis of the fundamental frequency, and that the AR2 fit works mainly on the parts of the spectrum that carry energy, thereby ignoring the steep cutoff implied by the filter. This behavior is necessary for the eddy detection process.

computed as an average over all trajectories first. It is advisable that this quantity be reasonably homogeneous throughout the dataset one wishes to study. Note that this step automatically adapts the algorithm to the particularities of the flow field, thereby eliminating the need for human input.

- 3) For each of these windows, means and trends of  $u$  and  $v$  are removed, and AR2 processes are fit to both  $u$  and  $v$  using the arburg algorithm (see section 2a). This returns coefficients  $\phi \dots$ , from which

damping scales  $d_{u,v}$  and frequencies  $f_{u,v}$  are derived by Eq. (6). Additionally, auto- and cross correlations of  $u$  and  $v$  are computed for all windows, and the revolution angle of the trajectory is found by applying MatLab functions `unwrap` and `angle` to  $u + iv$ .

- 4) The trajectory segment will be considered looping if all of the following criteria apply.

Oscillating: The fit shows oscillations (i.e., both  $f_{u,v}$  are real).

Equal persistence: Here  $d_u$  and  $d_v$  are (almost) equal. This has been expressed as follows:

$$|\log_2\left(\frac{d_u}{d_v}\right)| < 1.02. \quad (7)$$

Together with the next criterium, this ensures that the fits in  $u$  and  $v$  directions have the same characteristics, which is likely for coherent vortices but unlikely for random noise.

Equal frequency: Equivalently,  $f_u$  and  $f_v$  are (almost) equal:

$$|\log_2\left(\frac{f_u}{f_v}\right)| < 0.15. \quad (8)$$

Duration long versus period: The duration time scale is long compared to the detected period for both  $u$  and  $v$  components:

$$\frac{d_u}{f_u} < 0.9 \quad \text{and} \quad \frac{d_v}{f_v} < 0.9. \quad (9)$$

Quality of the AR2 fit: The AR2 process explains most of the actual variance; that is, the noise level  $\bar{\epsilon}^2$  that the AR2 fit returns is below the sample variances  $\bar{u}^2$ ,  $\bar{v}^2$  times a scaling factor:

$$\frac{\bar{\epsilon}_u^2}{\bar{u}^2} < 0.59 \quad \text{and} \quad \frac{\bar{\epsilon}_v^2}{\bar{v}^2} < 0.59. \quad (10)$$

Minimum revolution angle: The revolution angle  $\theta$  is at least two complete revolutions:

$$\frac{1}{|\theta|} < \frac{1}{4\pi}.$$

Phase lag 90°: The phase lag between  $u$  and  $v$  is near 90° or 270°, to extract near-circular motions. This is expressed by the cross-correlation  $R_{uv}$  between  $u$  and  $v$ , which must be near 0 for lag 0:

$$\frac{|R_{uv}(0)|}{\max(|R_{uv}|)} < 0.36. \quad (11)$$

The eddy period is then taken from  $f_{u,v}$  and the sense of rotation from the sign of  $\theta$ . With geostrophic motions in mind, counterclockwise loops will be considered cyclonic eddies in the Northern Hemisphere and clockwise loops anticyclonic, and vice versa for the Southern Hemisphere. If a loop crosses the equator, this obviously fails, and the DANA software issues a warning.

The numeric values on the right-hand sides in the above list are initial threshold values that define the behavior of the fit routine. However, a sensitivity study indicates that the method is relatively inert to poorly

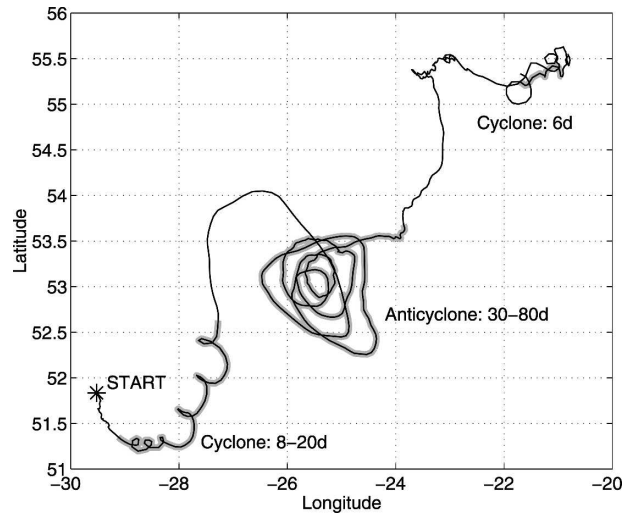


FIG. 4. Sample trajectory of a RAFOS float at 1500-m depth in the northeastern Atlantic with three eddy features. These demonstrate several difficulties in the detection process: eddy translation (both cyclones), irregular shape (the anticyclone), and varying revolution period as the float changes its distance from the eddy center (all). The shading indicates what the automatized algorithm of this manuscript detected as eddies. To include the large anticyclone, an additional time window of length 160 days was applied in this computation.

chosen parameters (cf. validation of the method in section 2c). The parameters in Eqs. (7), (8), (9), (10), and (11) have been found to yield best results in this sensitivity study.

The typical application that the algorithm was developed for is subsurface RAFOS float trajectories at intermediate (e.g., 500–3000 m) depth and position fixes in regular intervals of typically 1 day. In fact, the above numbers have worked reasonably well for two further datasets of dynamically rather different regimes (RAFOS floats at 1500 m in the northeastern Atlantic and at 800 m in the tropical Atlantic), although no comparison to a study like SRBR has been made in these cases. An example of such a trajectory with three eddy features is given in Fig. 4. These nicely demonstrate the difficulties one encounters during the automatic detection: advection, varying periods, and irregular shapes of the eddies. Near the start of this trajectory, there is a cyclonic eddy that translates in a curve toward the northeast, while the float slowly leaves the eddy. Thus, the revolution period from the float track increases from approximately 8 to 20 days before the eddy is left. Later, the float is caught in a large stationary anticyclone, in which the trajectory is of a rather irregular, noncircular shape. Periods vary between 30 and 80 days. Toward the end, there is a small cyclone (period circa 6 days) that moves erratically. All three eddy features are detected by the algorithm using the above

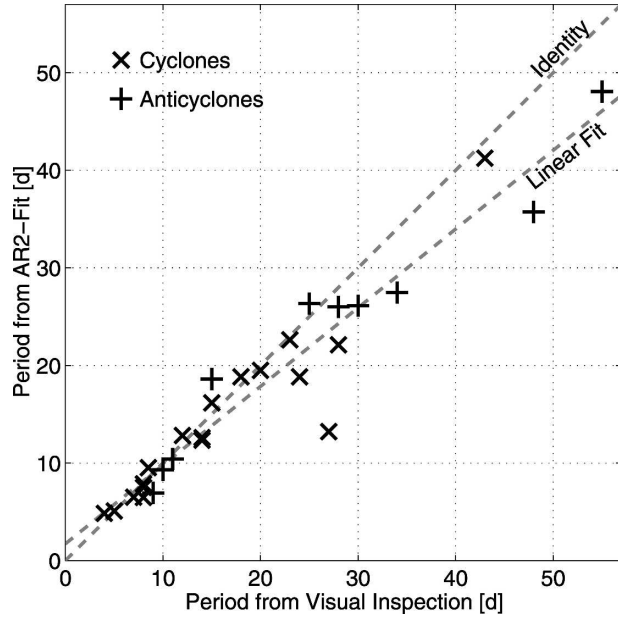


FIG. 5. Validation of the detected eddy period. A number of float trajectories from different depths (700–1500 m) and regions (subpolar and tropical Atlantic) was inspected visually for eddies. Their periods were estimated and compared to those from the automatic detection scheme using the AR2 fit. As suggested in Fig. 3, the periods are found with great accuracy by the AR2 fit (cf. the results to the identity function indicated).

methodology (with the exception that another time window of 160-day length was applied to include the large anticyclone).

### c. Validation of the method

This section will briefly review the performance of the AR2 fits with respect to the detected periods, which is followed by a comparison of the entire eddy detection algorithm with the work by SRBR.

For a scientific analysis of the eddy parameters, knowledge of the period is essential. Information on the eddy size is then easily obtained by multiplying the (detrended or, alternatively, RMS) float velocity with the period. As suggested by Figs. 2 and 3, the AR2 fit routines are fairly precise when detecting the fundamental frequency. To verify this further, periods of a random selection of eddies from RAFOS float trajectories collected during Leibniz-Institut für Meereswissenschaften (IFM-GEOMAR) projects were estimated visually, and these estimates were compared to the results from the DANA software, which uses the algorithm described in section 2b. The results of this comparison are shown in Fig. 5, which verifies a good performance of the algorithm. At long periods, the algorithm tends toward underestimation, which can be explained as follows: the longest window length sets an

TABLE 1. Dependence of the eddy detection on the number of completed revolutions. The first column indicates how many loopers are found by the reference SRBR, the second column shows how many of these loopers are also identified by the DANA software. Column 3 contains the ratio of these numbers, and column 4 the number of revolutions completed as indicated in SRBR (their appendixes A and B).

Loopers by SRBR	Loopers common	Ratio	No. of revolutions
108	60	55.6%	All
93	60	64.5%	>2.0
57	47	82.5%	>3.0
45	41	91.1%	>4.0
35	34	97.1%	>5.0

upper limit to what can be detected. If eddy periods near this length exist, the automatic algorithm will only find those parts that are short enough to fit in, which could be bad frequency estimates, while the better estimates are ruled out by the criterion that two complete revolutions are required. In such a case, the visual inspection will detect the “true” values better. The slope of the linear fit indicated in the figure is approximately 0.81; it increases to 0.97 if only eddies with periods below 25 days are considered, which is highly satisfactory considering the noisiness of the data.

As a thorough validation of the methodology as a whole, a comparison to the results by SRBR was carried out. There, a dataset from the northeastern Atlantic is inspected visually for eddies, which results in a total of 108 loopers, of which 52 are cyclonically looping and 56 anticyclonically looping [(52/56)  $\approx$  0.93]. Analyzing the same dataset with DANA using the methodology of section 2b, a total of 105 loopers are found, distributed among 62 cyclonic and 43 anticyclonic loopers [(62/43)  $\approx$  1.44].

Of the 108 loopers identified by SRBR, 60 are recognized by DANA (56%). However, this ratio increases substantially if events with very few revolutions are not considered, because most of the discrepancies between the DANA algorithm and SRBR occur with such short-lived events. In fact, if only those loopers with more than three completed revolutions (as indicated by SRBR’s appendixes A and B) are considered, DANA identifies 47 of 57 events (83%). Further details on this effect are listed in Table 1. The explanation for this is that the eddy detection criteria of this manuscript are naturally fulfilled better the longer the looping record is (agreement reaches 97% when considering only loopers with more than five revolutions). Furthermore, of the eddies with fewer than three revolutions observed, at least eight do not complete two 360° circles and can therefore not be found by DANA, while they

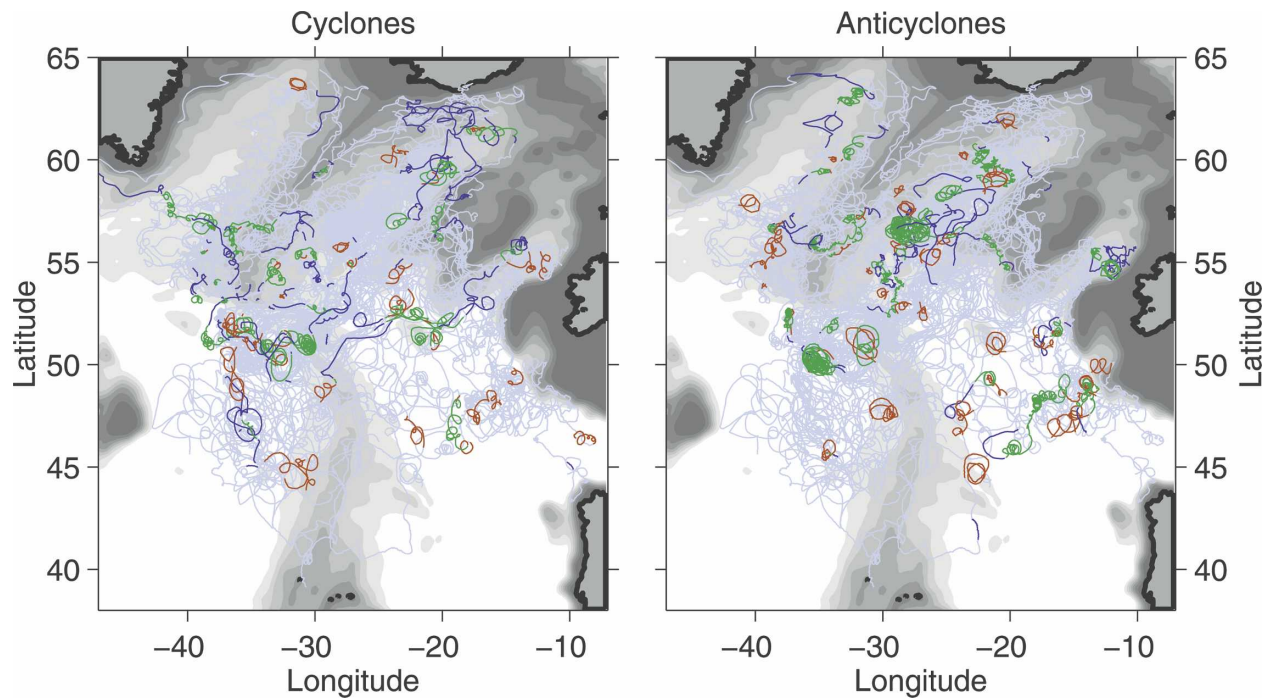


FIG. 6. Comparison between identified loopers according to the algorithms of SRBR and DANA. (light blue) No loopers detected by either method. (dark blue) Loopers detected by DANA only. (red) Loopers detected by SRBR only. (green) Loopers detected by both methods.

are recognized by SRBR because two closed loops are completed before reaching a  $720^\circ$  revolution angle. Vice versa, DANA will detect cycloidal cusps as loopers even if they are elongated so that they do not have closed loops, because the data are detrended first. These cases are not considered by SRBR. Several cases are indeed very difficult to decide on, both by eye and by the automatism, and the overall impression is that the agreement is good enough to be limited by the capabilities of both the automatic scheme and the human eye (and not only the former).

To find best agreement between DANA and SRBR, the parameters of Eqs. (7), (8), (9), (10), and (11) have been tuned by testing more than 10 000 combinations of different values (the other criteria refer to physical properties that are not supposed to be varied). The goal was to find parameters that would maximize the common times spent in loopers in both methods, while yielding equal total looper times in each method. With the above numbers, DANA finds 9073 looper days, of which 5268 (58%) are common with the 9086 found by SRBR. Although the parameters were varied throughout a large domain [by factors of up to 50% for Eq. (9) and exceeding 300% for all others] in the sensitivity study, this ratio of common looper time to total looper time of either method never fell below 42%, indicating that the method is relatively insensitive to poorly cho-

sen parameters. It should be noted that these numbers include all data, in particular those loopers for which different behavior between DANA and SRBR can be explained as in the previous paragraph. Therefore, the agreement of 58% must be considered highly satisfactory.

To demonstrate overall agreement, Figs. 6 and 7 show an overview of the results by SRBR and the present study. In Fig. 6, all trajectories are shown (“spaghetti diagram”), with the looping parts color coded according to which of the two methods identified them. The 5268 days commonly identified are shown in green, while the data labeled by none or only one method is in different colors. It is obvious at a glance that DANA identifies features with strong advective motion (wavy lines without closed loops), which SRBR do not consider. However, visual inspection revealed that in several such cases there was clearly a pattern that one could consider to be a quickly moving eddy feature. This may be the reason why DANA finds more cyclones than SRBR, as there seem to be more small cyclones which are rapidly advected when compared to corresponding anticyclones (this agrees well with findings by Lankhorst and Zenk 2006). Figure 7 shows that the discrepancy between the cyclone–anticyclone ratios is worst in the western basins, which are known to contain the strong North Atlantic and Irminger Currents possibly advecting the eddies around. It obviously re-



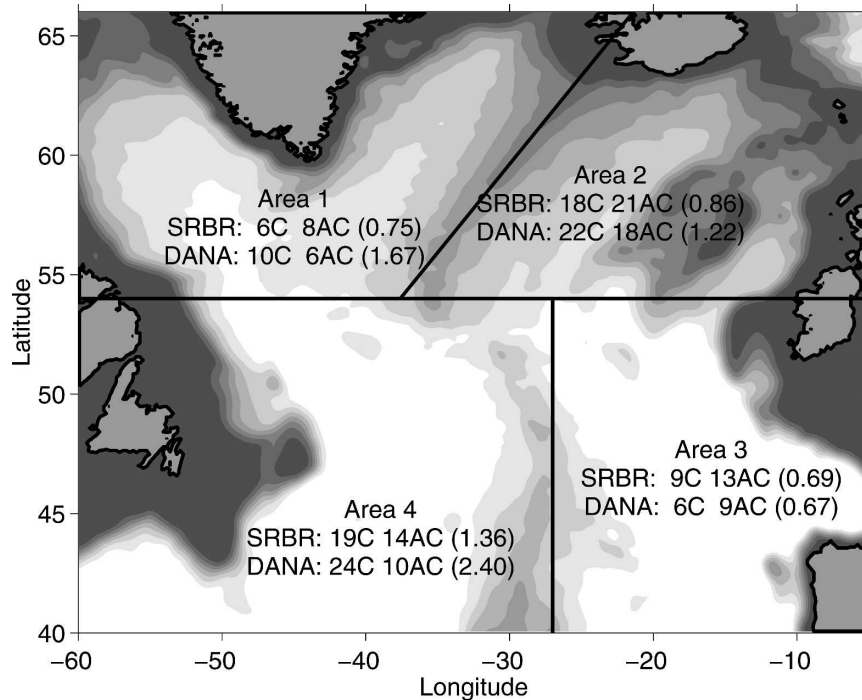


FIG. 7. Comparison of “looper” detection by SRBR (data are from their Table 1 and their Fig. 4) and by the DANA algorithm within a RAFOS float dataset at middepth in the North Atlantic. In four areas, cyclonically (C) and anticyclonically (AC) looping trajectory segments are counted. The numbers in brackets are quotients of these.

mains beyond the scope of the DANA detection algorithm to differentiate between eddies and other circulation patterns (e.g., wave motions) that may both create the same cycloidal Lagrangian drift tracks. There are also many cases where the record lengths of the loopers identified by DANA do not match those by SRBR, which is natural but decreases the percentage of agreement significantly. Otherwise, agreement between the two methods is as described in the previous paragraphs and is expected to satisfy scientific needs.

### 3. Conclusions

A self-contained identification scheme for looping segments within float or drifter trajectories is presented. The underlying scientific question is the analysis of coherent eddies, which will result in such loops if a drifter or float is caught in an eddy for a sufficient time. The detection algorithm splits each trajectory up into overlapping time windows, and for every window a set of criteria is applied to decide whether this trajectory segment is considered looping or not. The criteria are based on fundamental statistic properties such as auto- and cross correlations, and in particular on fits of second-order autoregressive (AR2) processes to the ve-

locity time series. From the AR2 coefficients, it is easily read off whether a time series is oscillating, and periods are derived with good accuracy. Validation has been carried out for the suitability of the AR2 fit routines for this purpose. A comparison to the study by SRBR, who identified looping trajectories by visual inspection, yielded good agreement to the results found here using the same float dataset. As a limitation, the eddy field parameters (i.e., typical eddy periods and sizes) should be reasonably homogeneous throughout the dataset under study. The algorithms to formulate the various criteria in computer language are readily available, and a software package called Drifter Analysis (DANA), first presented by Lankhorst (2003), has the methodology of this study implemented in MatLab code. The advantages of having an automatized computer algorithm for the eddy detection are obviously the aspect of saving time and using objective criteria rather than the human eye. In a recent study (Lankhorst and Zenk 2006), DANA has been applied to reveal a distinct asymmetry in the abundances of cyclones versus anticyclones in another float dataset of the subpolar North Atlantic. It may be interesting to reanalyze other datasets with this question in mind, which is now possible in a rather effective manner.

*Acknowledgments.* This work was initiated as part of the analysis of float trajectories from the *Sonderforschungsbereich 460 A3*, funded by the Deutsche Forschungsgemeinschaft, Bonn, Germany, under supervision by Rolf Käse and Walter Zenk. Deb Shoosmith and Heather Furey provided helpful comments and access to the data from the comparison study.

## REFERENCES

- Boebel, O., J. Lutjeharms, C. Schmid, W. Zenk, T. Rossby, and C. Barron, 2003: The Cape Cauldron: A regime of turbulent inter-ocean exchange. *Deep-Sea Res. II*, **50**, 57–86.
- Box, G. E. P., G. M. Jenkins, and G. C. Reinsel, 1994: *Time Series Analysis: Forecasting and Control*. 3d ed. Prentice-Hall, 598 pp.
- Fratantoni, D. M., and P. L. Richardson, 2006: The evolution and demise of North Brazil Current Rings. *J. Phys. Oceanogr.*, **36**, 1241–1264.
- Lankhorst, M., 2003: Klassifizierung charakteristischer Eigenschaften von Float-Trajektorien in unterschiedlichen dynamischen Regionen. Diploma thesis, Mathematisch-Naturwissenschaftliche Fakultät, Christian-Albrechts-Universität Kiel, Germany, 84 pp.
- , and W. Zenk, 2006: Lagrangian observations of the mid-depth and deep velocity fields of the northeastern Atlantic Ocean. *J. Phys. Oceanogr.*, **36**, 43–63.
- Press, W. H., B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, 1993: *Numerical Recipes in FORTRAN 77: The Art of Scientific Computing*. 2d ed. Cambridge University Press, 933 pp.
- Shoosmith, D. R., P. L. Richardson, A. S. Bower, and H. T. Rossby, 2005: Discrete eddies in the northern North Atlantic as observed by looping RAFOS floats. *Deep-Sea Res. II*, **52**, 627–650.