UNRAVEL: A Robust Modular Velocity Dealiasing Technique for Doppler Radar

VALENTIN LOUF

School of Earth, Atmosphere and Environment, Monash University, Melbourne, Victoria, Australia

ALAIN PROTAT

Research and Development Branch, Bureau of Meteorology, Melbourne, Victoria, Australia

ROBERT C. JACKSON AND SCOTT M. COLLIS

Division of Environmental Science, Argonne National Laboratory, Lemont, Illinois

JONATHAN HELMUS

Anaconda Inc., Austin, Texas

(Manuscript received 31 January 2019, in final form 19 February 2020)

ABSTRACT

Unfold Radar Velocity (UNRAVEL) is an open-source modular Doppler velocity dealiasing algorithm for weather radars. UNRAVEL is an algorithm that does not need external reference velocity data, making it easily applicable. The proposed algorithm includes 11 core modules and 2 dealiasing strategies. UNRAVEL is an iterative algorithm. The goal is to build the dealiasing results starting with the strictest possible continuity tests in azimuth and range and, after each step, relaxing the parameters to include more results from a progressively growing number of reference points. UNRAVEL also has modules that perform 3D continuity checks. Thanks to this modular design, the number of dealiasing strategies can be expanded in order to optimize the dealiasing results. While the first driver dealiasing Doppler velocity from each tilt independently from one another, the second driver also performs a three-dimensional continuity check of the velocity using successive elevations. The proposed dealiasing algorithm is tested using severe weather data from an S-band Doppler radar that have been aliased to mimic aliased radial velocity patterns that would be observed by a C-band Doppler radar. Artificially aliasing S-band data permits creation of a reference to which the performance of various dealiasing techniques can be compared. Comparisons show that UNRAVEL consistently outperforms other established dealiasing algorithms for the test period selected in this work.

1. Introduction

Doppler radars allow observations of the internal dynamics within life-threatening storms. They are the only remote sensing instruments that can measure the radial velocities of wind inside regions of heavy precipitation. Doppler radars play thus a crucial role in atmospheric science, from meteorological hazard warnings, to nowcasting, and to the initialization of numerical weather prediction models. Yet, to achieve these tasks with the adequate accuracy, a robust quality control must be performed on the data. The two major issues in radar data quality control are the

removal of noise and clutter, and the correction of aliased radial velocities.

Velocity aliasing is described in detail in the literature (Sauvageot 1992; Doviak and Zrnić 2006; Skolnik 2008, among others), so only a brief description is given here. Because the discrete sampling of the phase occurs within a fixed time interval [or pulse repetition frequency (PRF)], there is an ambiguity between the actual shift in velocity and an alias caused by the frequency of the PRF. The extent of unambiguous velocities is called the Nyquist cointerval, $[-V_N, V_N]$, where (Ray and Ziegler 1977)

$$V_N = \frac{\lambda(\text{PRF})}{4}, \quad (1)$$

Corresponding author: Valentin Louf, valentin.louf@monash.edu

DOI: 10.1175/JTECH-D-19-0020.1

© 2020 American Meteorological Society. For information regarding reuse of this content and general copyright information, consult the [AMS Copyright Policy \(www.ametsoc.org/PUBSReuseLicenses\)](https://www.ametsoc.org/PUBSReuseLicenses).

with λ being the radar wavelength; V_N is often called the Nyquist velocity. Any shift in velocity will be within the Nyquist cointerval, meaning that any frequency shift exceeding $\pm\text{PRF}/2$ will be aliased by $2nV_N$ (with $n \in \mathbb{Z}$). For example, if the true Doppler velocity is 16 m s^{-1} , and the Nyquist cointerval is $\pm 14 \text{ m s}^{-1}$, then the Doppler velocity will be measured at -12 m s^{-1} .

The estimation of the true Doppler velocity V_T in relation to the observed velocity V_O is given by (Doviak and Zrnić 2006)

$$V_T = V_O + 2nV_N. \quad (2)$$

Because aliasing causes unrealistic gradients (discontinuity) in the field of observed velocities, dealiasing, also often called unfolding, is the process of finding n for each V_O in a way that the V_T field is continuous. While continuity of the velocity field is a necessary condition to infer dealiasing success, it is not a sufficient one. The ultimate goal is to obtain a velocity field that is the most accurate representation of the underlying Doppler velocities.

The smaller the Nyquist cointerval the more challenging the dealiasing. Therefore, multi-PRF techniques have been implemented, which alternate measurements at two or more values of PRF to increase the Nyquist cointerval (e.g., Zrnić and Mahapatra 1985; Holleman and Beekhuis 2003; Tabary et al. 2006; Tahanout et al. 2015). Although such techniques can produce artifacts (Joe and May 2003), they are generally robust, except when applied to regions of large wind shear or turbulence or velocity data at low signal-to-noise ratio. In Australia, the operational radar network still includes a lot of old radars, for which the multi-PRF techniques do not perform well. Also, aliased velocities can still be found in extreme weather such as east coast lows and tropical cyclones, even for radars where a multi-PRF technique was implemented. This indicates that robust dealiasing techniques are still needed to produce Doppler velocities that can be used with confidence in an operational context.

Most Doppler velocity dealiasing techniques are based on spatial continuity, one-dimensional (Ray and Ziegler 1977), two-dimensional (e.g., Zhang and Wang 2006), three-dimensional (Bergen and Albers 1988), and even four dimensions by adding temporal continuity (James and Houze 2001; Lim and Sun 2010). However, the main issue is that regions with no significant returns that are censored or removed from displays disrupt the continuity of the velocity field, and strong wind shears associated with severe weather make it difficult to correct for velocity aliasing. Various methods have been developed when no preceding

neighbors are available, notably using upper-air soundings as reference (Eilts and Smith 1990) or velocity–azimuth displays (VAD) winds (Yamada and Chong 1999; Tabary et al. 2001; Gong et al. 2003).

Most dealiasing schemes tend to perform poorly on data coming from weather radars of the Australian Bureau of Meteorology, notably when dealing with the heavy convective rainfalls in the northern Australian tropics, or the east coast lows on the New South Wales coast (Speer et al. 2009). Tropical convective precipitating systems in northern Australia tend to produce “popcorn” convection, that is, strong convective towers scattered away from one another, with large areas in between with no radar returns, which is a challenge for velocity dealiasing due to data gaps. Also, strong wind shear is often a problem, due to sudden change in velocity from one radar gate to the next. A concrete example of strong wind shear are the east coast lows, on the eastern coast of Australia, generally associated with extreme rainfalls. Moreover, most of the Australian radar network is composed of C-band radars (51 C band and 14 S band), meaning that the Nyquist velocity is generally around $\sim 13 \text{ m s}^{-1}$. Large areas with no radar returns, strong wind shears, and small Nyquist velocities are all major challenges for dealiasing techniques.

More robust dealiasing techniques have been developed recently (Zhang and Wang 2006; Li et al. 2014; Xu and Nai 2017). He et al. (2012b) developed the S-band–China New Generation Weather Radar (CINRAD) improved dealiasing algorithm (S-CIDA) for the S-band CINRAD radars. He et al. (2012a) then changed the S-CIDA algorithm to take into account the characteristics of C-band radar data from CINRAD. Our new scheme, Unfold Radar Velocity (UNRAVEL), expands from this work. It is composed of 11 core modules, most of them being transcriptions or variations on the aforementioned dealiasing techniques, like Bergen and Albers (1988), Eilts and Smith (1990), Jing and Wiener (1993), and Zhang and Wang (2006), among others. These modules are called by a driver, that is, an application that is able to call the different modules in a particular order, specific to that driver, to resolve the aliasing problem. This driver is made to be user-friendly, it takes as input the radar object from the Python ARM Radar Toolkit (Py-ART) (Helmus and Collis 2016) and does the dealiasing automatically. The current driver is based upon Li et al. (2014), which itself is an extension of He et al. (2012a) and Zhang and Wang (2006) algorithms.

In section 2 we introduce the dataset that is used in this study. In section 3, UNRAVEL dealiasing scheme is described in detail. In section 4, dealiasing performances and error rates of several dealiasing techniques,

including UNRAVEL, are discussed using a week of data from several heavy rainfall events from the operational S-band dual-polarization Doppler radar covering the Sydney region. These radar data are artificially aliased to simulate C-band Doppler measurements (roughly half the V_N of S-band radars) to test the performance of various dealiasing techniques. Finally, the results are summarized and discussed in [section 5](#).

2. Instruments

In the following sections, data coming from two ground radars are used to illustrate UNRAVEL: the C-band radar of Gunn Point (-12.245°N , 131.045°E) and the S-band radar of Terrey Hills (-33.701°N , 151.209°E).

The C-band polarimetric radar (CPOL) is a dual-polarization Doppler radar, working at a frequency of 5.6 GHz with a pulse repetition frequency of 1000 Hz and a beamwidth of 1° . CPOL is located at Gunn Point, about 25 km northeast from Darwin airport ([Keenan et al. 1998](#)). CPOL performs a set of scans with an update time of 10 min. This includes, nominally, a volume scan, vertically pointing scan, and two RHI scans. The scan comprises 15 elevations: 0.5° , 0.9° , 1.3° , 1.8° , 2.4° , 3.1° , 4.2° , 5.6° , 7.4° , 10.0° , 13.3° , 17.9° , 23.9° , 32.0° , and 43.1° . An additional series of scans at 90° is also performed regularly. The maximum unambiguous range of the volume scan is 140 km with a range gate spacing of 250 m and an azimuthal beam spacing of 1° . Because of its location in the tropics and long observational record, CPOL is a unique tool for research. The 17 years of CPOL dataset underwent recently a series of calibration, quality control, and processing that are described in detail in [Louf et al. \(2019\)](#). The processing and quality control on CPOL are what kickstarted the development of UNRAVEL. Various algorithms were tried to dealias CPOL velocities, like the publicly available four-dimensional dealiasing (4DD) ([James and Houze 2001](#)), and region based ([Helmus and Collis 2016](#)), but also those that can be implemented quickly (e.g., [Bergen and Albers 1988](#); [Eilts and Smith 1990](#)). The dealiasing errors that these algorithms made on CPOL Doppler velocity were so frequent that we decided to develop our own dealiasing technique, with the requirement that this new technique could be tailored by the user to the specificities of the radar being used.

The second radar that we use in this study is the S-band dual-polarization Doppler radar of Terrey Hills, near Sydney, Australia, that started to operate in late September 2017. This radar performs a series of 14 tilts

(between 0.5° and 32°) every 6 min. There is also a long-range scan at 0.5° . The range gate spacing is of 250 m and an azimuthal beam spacing is of 1° .

3. UNRAVEL description

In principle, a dealiasing algorithm compares two velocities and requires that

$$|V_R(\theta, r) - V_i(\theta, r)| < \alpha V_N, \quad (3)$$

where r is the range, θ is the azimuth, V_R is a velocity used as reference for the comparison, V_i is the velocity being treated, and $\alpha \in (0, 2]$ is an adjustable parameter. The operation of UNRAVEL modules is twofold: 1) find a robust way to define V_R that is representative of the underlying field, and 2) find the factor n from Eq. (2) for V_i that satisfies Eq. (3).

UNRAVEL is based upon the works of [Zhang and Wang \(2006\)](#), [He et al. \(2012a\)](#), and [Li et al. \(2014\)](#). The multiple passes and the module concepts are drawn from these algorithms while improving them. UNRAVEL is open source, and is released as part of the Python Package Index (named unravel). The exact version of the UNRAVEL code presented herein is available here: [Louf and Jackson \(2020\)](#). Note that we are currently working on the integration of UNRAVEL in the Py-ART ([Helmus and Collis 2016](#)). UNRAVEL modules fall into three categories: initialization, dealiasing, and checking. While the order of these three categories must be respected, the modules inside the dealiasing and checking categories can be executed in any order. In this section, the individual dealiasing modules are first described, and then, the default driver, meaning the sequence of modules used by default by UNRAVEL to dealias the velocity field, is explained.

UNRAVEL is by default a multipass technique. The philosophy behind the multipass is simple. We start with stringent continuity tests, and after each pass we progressively relax the constraint on continuity to retrieve more and more solutions using dealiased velocities from the previous passes as references.

UNRAVEL works with a flag array, which records the state of each velocity point at any given step of the algorithm. Within each step of the algorithm, the radar gates are marked with four different flags to track the processes that each point has gone through. The possible values are -3 for missing value, 0 for unprocessed value, 1 for processed and unchanged value, and 2 for processed and dealiased value. There is no difference in treatment between a gate flagged 1 or 2,

the reason why we have kept these two different flags for processed value is for debugging purposes. The initialization modules identify noisy data (velocities exhibiting large standard deviation, nonmeteorological echoes) and missing values and set the corresponding pixels to the -3 value in the flag array. The initialization modules also do a first pass at dealiasing and find the radials of reference for the dealiasing modules to work. The dealiasing modules exclusively process the velocities flagged 0. Each time a solution is found, the dealiasing modules flag the corresponding velocity point as unchanged or dealiased (1 or 2). Once a velocity has been flagged 1 or 2, this point cannot be modified anymore by subsequent dealiasing module. Only the checking modules can modify a point that has been flagged 1 or 2. It can also be used as a new reference in the subsequent steps. Note that a dealiasing module does not necessarily have to find an answer and flag a velocity; if no answer is found within a tolerable velocity range controlled by a stringency parameter α [Eq. (3)], it just skips that particular point (flag 0), and lets the subsequent module find the solution for it. Finally, the checking modules are used to verify the consistency in the Doppler velocity of each processed velocities with its surroundings. While this flagging process was proposed in James and Houze (2001) and Zhang and Wang (2006), UNRAVEL behaves differently due to the immutability of values flagged 1 or 2 by the dealiasing modules here, hence the importance to be as strict as possible during the initialization above. The different modules are discussed below in more detail.

a. Initialization

1) PREPROCESSING

The main objective of this module is to identify and flag artifacts or noisy data as missing values (flag: -3). The subsequent initialization and dealiasing modules consider that all the values not flagged -3 in the array are true signals that need to be processed. Failure to identify artifacts or noisy data may cause problems to the subsequent modules when trying to solve Eq. (3). These problems could be that the modules interpret the value of a “wrong” gate as a V_i that needs to be dealiased, in a region without aliasing, or that the subset of velocities chosen to compute V_R contains one or more of these high-variability anomalies, and thus causing V_R itself to not be representative of the underlying velocity field.

The main difficulty encountered by this module is that high-variability gates are not necessarily “wrong” gates. High-variability gates can be due to residual

nonmeteorological echoes that need to be removed, aliased velocities that need to be dealiased (notably on the edges of regions where the aliasing shift happen), or regions of wind shear that need to be processed. To deal with these conditions, the preprocessing module uses a modified z -score method within a running window of 10 gates. Each radial is filtered independently, starting from the first gate, and for each gate in the window the following is estimated:

$$z = \frac{v_i - \tilde{v}}{\text{MAD}(\tilde{v})}, \quad (4)$$

where v_i is the velocity being processed, $\tilde{v}$ is the median of the previous 10 gates window, and $\text{MAD}(\tilde{v})$ is the median absolute deviation of the 10 gates window (e.g., Rousseeuw and Croux 1993). Missing data are ignored. If all the data within the window are missing, then the window moves to the next gate. The advantage of the modified z -score method is that regions of wind shear generally correspond to a low z value. A cutoff value for z is used to determine if v_i is a high-variability anomaly. For UNRAVEL, the cutoff value is 2 by default, meaning that if $|z| \leq 2$, then v_i is left as is; if $|z| > 2$, then it is either an aliased gate flagged for processing or an artifact flagged as a missing value.

For dealiasing, V_R is chosen to be the average of all gates in the window of the same sign as $\tilde{v}$ in Eq. (4). In other words, if $\tilde{v} > 0$, then $V_R = \overline{v_+}$, and if $\tilde{v} < 0$, then $V_R = \overline{v_-}$. If after dealiasing the value between the gate and the reference is still greater than αV_N [Eq. (3)], then it is compared to the other average velocity ($\overline{v_+}$ if $\overline{v_-}$ is the reference, and conversely). If after that the value of that gate is still αV_N away, then that gate is flagged as a missing value.

Figure 1 is the velocity field of tropical cyclone Blanche from CPOL radar ($V_N = 13.3 \text{ m s}^{-1}$) at the first elevation (0.5°) from 1500 UTC 4 March 2017. This case is interesting because there are several isolated clouds with corresponding Doppler velocities that are fully aliased. There is a lot of clear-air, high-variability data and ground-clutter residues near the radar. There is also a large part on the west side of the radar domain that is fully aliased and where the first gates encountered by the radar are aliased. Figure 1a shows the raw velocity field, while Fig. 1b shows the velocity field after the preprocessing. Preprocessing has removed the noise, despeckled the data (i.e., elimination of isolated pixels), and dealiased the edges of aliased areas. In our experience, this first guess at dealiasing really helps the subsequent modules to find the correct answer as the regions of aliasing are now well defined.

FIG. 1. Velocity field from CPOL at elevation 0.5° from 1500 UTC 4 Mar 2017. Raw velocity (a) before and (b) after the preprocessing module. Isoradials are plotted every 50 km. The Nyquist velocity of CPOL is 13.3 m s^{-1} .

2) FINDING AND PROCESSING THE REFERENCE RADIALS

UNRAVEL starts its dealiasing process by searching for two reference radials, generally 180° apart, in a similar fashion as [He et al. \(2012a\)](#). The aliased

FIG. 2. Normalized sum of the absolute values of measured velocities at all the valid gates along each radial (blue circles). The orange points correspond to the radials that satisfy the first criterion; i.e., the number of valid data points must be at least two-thirds of the average number of valid gates. Conversely, the blue points fail to satisfy the first criterion. The vertical dashed lines are the reference radials. The one at 9° of azimuth is the first reference radial.

velocities are then dealiased in a sector that extends clockwise and counterclockwise to the midpoint between each of the reference radials. The concept of reference radials allows UNRAVEL to work without the need of an external reference for the velocity, as is required by many existing algorithms. This concept was first proposed by [Yamada and Chong \(1999\)](#), and the current implementation in UNRAVEL is a modified version of [Zhang and Wang \(2006\)](#). The initial reference radial only needs to be in the vicinity of the region where velocities have the smallest magnitudes, that is, where the average value of the absolute velocities $[|V_T(r)| \text{ at } \theta \text{ constant}]$ is minimum. The most likely position for this to occur is where the wind direction is orthogonal to the direction of the antenna.

[He et al. \(2012b\)](#) originally used one reference radial for S-band radar data, and later proposed to use two reference radials for C-band radars ([He et al. 2012a](#)). This is because, for the same wind field, the number of aliased velocities is nearly doubled for C band (for the same PRF, the Nyquist cointerval is about half of that of S-band radars). This leads to an increase in complexity of the observed wind field, and a greater probability of dealiasing mistakes.

[Figure 2](#) illustrates the procedure to find the reference radials by showing the normalized sum of the absolute values of measured velocities along each radial. In orange are radials with at least two-thirds of the average number of valid gates per radial, that is, radials with a larger-than-average quantity of data

points. To find the reference radials, the following criteria must be satisfied:

- He et al. (2012a) proposed that in the initial radials, the number of valid data points must be at least two-thirds of the average number of valid gates in all the radials at any azimuth (the orange points on Fig. 2 satisfy this criterion, the blue points do not). This ensures that the initial radials comprise a large enough number of points to act as reference for the next modules.
- Individual velocity values in the radial must not be in close range of the Nyquist velocity $|V|_i < \beta V_N$. This corresponds to the conditional mean velocity β parameter in Zhang and Wang (2006). While $\beta = 0.3$ in Zhang and Wang (2006), we have found that we needed to increase the β parameter from 0.3 to 0.6 in order to find a reference, as most of the time $\beta = 0.3$ tends to be too strict.
- The reference radials are minima in the curve of the mean absolute values of the measured velocities at all the valid gates along each radial (dashed lines on Fig. 2). This criterion also minimizes the risk of encountering aliased velocities in the reference radials.
- They must have the biggest difference between minima in the curve and the adjacent maxima.
- The two initial reference radials must be separated by at least 120° . More than two radials can meet the previous criterion, but only two will be separated by about 180° and also have the largest difference between the minima and the adjacent maxima of the curve (He et al. 2012a). The value of 120° is simply to allow for some tolerance.

If no reference radials are found, then UNRAVEL tries again by looking only at the minima in the curve of the mean absolute values, meaning that UNRAVEL will tolerate a smaller number of points (not looking at the first criterion) as long as there is a minimum in the curve of the mean absolute values (so that it mitigates the risk of finding aliased velocities in the reference radials). Even if there is no precipitation in the radar domain, there is still clear-air data, and clear-air signal usually satisfies this criterion. However, this is a trivial case scenario for dealiasing.

Once the reference radials are found (vertical dashed lines on Fig. 2), the two neighboring radials of each reference radial are dealiased, using continuity along the azimuth. This initialization module dealiases each velocity gate in the two adjacent radials using azimuthal continuity constraints against gates at the same range from the initial radial. If a reference velocity is available in the initial radial, then the corresponding

gate in the neighbor radial is processed and flagged as 1 (or 2 if the gate was identified as aliased). Otherwise, the gate in the neighbor radial is skipped and its processing is postponed to a later pass.

3) AZIMUTHAL CONTINUITY DEALIASING, FIRST PASS

For the first pass, we use the azimuthal continuity between radials. This module is similar to the dealiasing technique in Zhang and Wang (2006), except that we use the average of the previous three radials as reference to mitigate data with high variability.

Once the two reference radials are found, four sectors are defined. Each sector starts at one of the two reference radials and the sector limit is the midpoint between the two reference radials. UNRAVEL starts the dealiasing from these two radials in the clockwise and the counterclockwise directions, processing data inside these sectors. This strategy restricts the possible generated errors to within a single sector. Figure 3a illustrates the effect of the first pass. The solid black line is the first reference radial and the dotted line is the second reference. The dashed and dot-dashed lines are the sector limits. The order of the azimuthal continuity dealiasing is solid line to dot-dashed line, solid to dashed line, dotted to dot-dashed line, and dotted to dashed line. This sector strategy is similar to He et al. (2012b), with the exception that the sectors are not of equal size as we allow for the reference radials to be separated by $180^\circ \pm 45^\circ$. After the first pass, 30.3% of velocities on Fig. 3a have been processed.

Along the azimuth direction, each velocity gate is compared to the average of three velocities in preceding radials at the same range. We use the average of three gates as reference to limit the impact of high-variability data. To obtain a valid processed velocity, at least two of the three preceding gates in the azimuth direction must exist (i.e., not missing). If Eq. (3) is satisfied for $n = 0$ then a flag 1 is assigned, if it is satisfied for $n \neq 0$, meaning that the observed velocity is aliased, then a flag 2 is assigned.

The value of α in Eq. (3) is one of the parameters used to control the strictness of the dealiasing process. Zhang and Wang (2006) proposed to use $\alpha = 0.8$. After performing a sensitivity test on how α impacts dealiasing, we found that $0.6 \leq \alpha \leq 0.8$ gave the best results (cf. section 4b). For the first pass, $\alpha = 0.6$ by default, meaning that for the first pass the difference between two adjacent velocities must be below 60% of the Nyquist velocity (30% of the Nyquist cointerval) to be flagged as a valid processed value. In the last dealiasing modules, α can reach a value of 0.8. For $\alpha = 0.6$ and

CPOL 0.5 Deg. 2017-03-04T15:00:06Z

FIG. 3. Processed velocity field from CPOL at elevation 0.5° from 1500 UTC 4 Mar 2017, only the gates flagged as processed (value of 1 or 2) are shown, the unprocessed or missing gates (value of 0 or -3) are filled in white. (a) After the azimuthal dealiasing module (30.3% processed), (b) after the range continuity module (57% processed), (c) after the box continuity module (97.1% processed), and (d) after the least squares regression module (100% processed). Isoradials are plotted every 50 km. The solid line is the first reference radial, the dotted line is the second one. The dashed and the dot-dashed lines denote the sector limits of the radial by radial dealiasing when starting from one of the reference radials.

$V_N = 13.3 \text{ m s}^{-1}$, the maximum allowed difference between V_i and V_R is $\sim 7.9 \text{ m s}^{-1}$.

b. Dealiasing modules

The dealiasing modules only look at the unprocessed (flag: 0) velocities. While many strategies for dealiasing are made possible through the choice of dealiasing modules and their order of execution, here we present

the default strategy. The default strategy, when first designed, was following the module sequence of [He et al. \(2012a\)](#) and has been modified empirically later. After each step, the ratio of unprocessed to processed velocity is computed, if it reaches 100% after any given dealiasing module, remaining dealiasing modules are skipped and the dealiased velocity field is sent to the checking modules. All the dealiasing modules have

common input parameters: the coordinates (r , θ , and ϕ , the elevation, when applicable), the input aliased velocity field, the currently processed velocity field, the flag array, V_N , and α from Eq. (3).

1) DEALIASING ALONG A RADIAL

We start by processing velocities along a radial (θ constant), like in Ray and Ziegler (1977), or Eilts and Smith (1990). This is done by checking the radial continuity forward (from $r = r_0$ to $r_{\max}$) and then backward (from $r = r_{\max}$ to r_0), where r_0 is the closest gate and $r_{\max}$ is the farthest gate from the radar for a given radial. This module has a “window” parameter, that is, a parameter that allows to use a certain number of gates to compute V_R . If that “window” parameter is equal to 1 then the module will only use the immediate previous gate (r_{n-1} when onward, r_{n+1} backward) as reference V_R . If the window is more than 1, then V_R is the mean of all of the already valid and processed gates within the window (flag: 1 or 2). If none of these gates are flagged 1 or 2, then the current gate is skipped and its processing postponed. Figure 3b shows the effect of the range continuity module. After this step, the number of processed gates is around 57% for this case.

2) AZIMUTHAL CONTINUITY DEALIASING

This is the same technique as the first pass module. However, while this module starts processing at the first reference radial, it does not split the scan into sectors and does a full sweep, first clockwise, and then counterclockwise. Like the previous dealiasing module, it has a parameter controlling the window size, but this time over θ , keeping r constant. In the initialization module, three radials are used as reference, in the dealiasing processing driver, the default value is six. The median value of the velocities (flagged 1 or 2) within the window is taken as V_R . At least one gate flagged 1 or 2 is required; otherwise, the current gate is skipped.

3) DEALIASING INSIDE A BOX

This module dealias velocities based on the median of an area of corrected velocities surrounding, or preceding, the gate being processed. This module looks at a p number of previously dealiased gates in the q previous radials; the values of p and q being controlled by the user. It is similar to the dealiasing technique from Bergen and Albers (1988). Bergen and Albers (1988) originally took a box along a vertex of the gate being processed. In UNRAVEL, two strategies are available to the user. One along the vertex, like Bergen and Albers (1988), and another one where the box is surrounding the gate being processed. The reason for this second strategy is that this module is generally called

after the range and azimuthal continuity modules and thus, the processed velocities used to compute V_R are not necessarily in the space “behind” the gate to be processed, but in the space that surrounds that gate.

4) CLOSEST REFERENCE

The closest reference module will find a solution for all the remaining unprocessed velocity. This module looks for the processed velocity, in a 2D plane, that is the closest from the unprocessed velocity. Once that closest velocity is found, a box around that velocity is taken and the median of these processed velocities is used as reference for the dealiasing of the unprocessed one. Because this module computes the distance of all valid points against the leftover ones, this is the most time consuming module. For this module to be executed, it means that no reference was available within 90 km of the unprocessed velocities. This usually happens for the lowest tilts in large areas with no radar returns. This module is similar to the dealiasing techniques in Bergen and Albers (1988).

This module will find a solution for all the unprocessed points. Therefore, after the execution of this module, no dealiasing module will work (no more “0” flag in the flag array) and only the error checking module can have an impact on the velocity field.

c. Error checking modules

The aim of the error checking modules is to verify that a processed gate (flag: 1 and 2) is not at odd with its local environment, that is, along a radial, inside a box, or between successive elevations. The error checking modules can change the value of a gate that has been previously processed.

1) LINEAR LEAST SQUARES ERROR CHECK ALONG RADIALS

For the purpose of this check, the wind field can be assumed to follow a linear relationship along the range (He et al. 2012a); that is,

$$V_T = a_\theta r + b_\theta, \quad (5)$$

where the coefficients a_θ and b_θ are obtained by solving a linear least squares equation for each radial. Then, as previously, each gate is adjusted to be within the Nyquist cointerval value predicted by the least squares regression.

2) BOX CHECK ERRORS

This module is similar to the previous “dealiasing inside a box” module. To compute V_R , this module uses the z score of a quality subset of the surrounding

velocities. Unlike the preprocessing module where the modified z score is used, here, because the subset of velocities is much larger, the “normal” z score is used (the mean $\bar{v}$ instead of the median $\tilde{v}$ and the standard deviation σ_v instead of MAD). Only the gates one standard deviation from the mean ($|z| \leq 1$) are kept.

3) CONTINUITY CHECK BETWEEN ELEVATIONS

While all the previous modules treat each elevation slices independently, this module checks the continuity between adjacent elevations. Gates at all elevations are projected to the ground using the azimuthal equidistant projection. This projection is able to determine what set of gates on slice $n - 1$ is directly below a given gate on slice n . Thus, the gates of slice $n - 1$ are used as reference to check to continuity with elevation n . In other words, $V_i(r, \theta, \phi_n)$ is compared to $V_{\text{ref}}(r \pm w, \theta \pm w, \phi_{n-1})$, where w is the user-defined window, used to find reference velocities and ϕ the elevation. Like the previous module, the z score within one standard deviation from the mean is used to determine V_R from V_{ref} .

d. UNRAVEL driver

UNRAVEL is a modular technique, meaning that each module can be called in a particular order designed by the user to tailor the dealiasing technique to find the best solution. This flexible dealiasing strategy is done by a driver. Figure 4 shows the flowchart for the default driver of UNRAVEL. The driver starts with the initialization, that is, the modules described in section 3a. This initialization part is mandatory and should always start the dealiasing process. Then, for the default process, the radial-by-radial continuity and gate-by-gate continuity check are executed twice. After each pass, the window for finding the reference value for dealiasing is increased. In the “long range” driver, this sequence is executed several times, with the window of reference value alternating between a large window (12, 24, 48, and 96 gates) and a small window (6 gates). This strategy allows for the continuity modules to look farther and farther away for a reference, and take each time more and more points as reference. This is particularly efficient when the radar scene is made of small (in area) echoes scattered all around the radar scan without background continuity. The same strategy applies to the next module being executed, the dealiasing inside a box. This module is called four times, each time the window is increased: (5, 5), (20, 10), (40, 10), (80, 20). The biggest window has a length of about 20 km in range and a width in azimuth varying from ~ 350 m to 40 km, based on the size of the box for the resolution of the CPOL radar. The goal is to progressively increase the size of the area where gates can be used as reference. Figure 3c shows

FIG. 4. Flowchart of the module sequence for the default driver of UNRAVEL.

the effect of the box dealiasing. After each iteration of this module, 85.7%, 87.5%, 91.9%, and finally 97.1% of velocities have been processed.

The next module (closest reference) will find a solution for all remaining points that have not been yet processed; however, the closest reference module is a real bottleneck in term of speed, especially if there are many unprocessed gates left. We found through development and experimentation that it is more efficient, in terms of speed, to call several times the “dealiasing inside a box” module and to increase the window to look for reference values. This reduces the number of gates to process and thus reduce the processing load in the subsequent module. After the execution of the dealiasing modules, the checking modules are being called. The driver starts with the radial least squares check, and then performs the check inside a box. Once these checking modules are done, the driver starts again the whole process with another elevation of the velocity field. Once all elevations are dealiasied, the 3D continuity check is performed, checking the continuity between velocities of adjacent elevations.

4. Statistical assessment of UNRAVEL against several dealiasing techniques

a. Dealiasing algorithms

In this section, we compare how UNRAVEL performs against the 4DD algorithm (James and Houze 2001), the unwrap algorithm (Herráez et al. 2002), the region-based algorithm from Py-ART (Helmus and Collis 2016), and the multipass algorithm from Zhang and Wang (2006).

To quantitatively demonstrate the performance of these various algorithms, we use a week of data coming from the S-band ($V_N = 26.6 \text{ m s}^{-1}$) dual-polarization Doppler radar of Terrey Hills that are artificially aliased to a Nyquist cointerval of $[-13.3, 13.3] \text{ m s}^{-1}$ to simulate the Doppler velocity that would be measured by a C-band radar. The procedure is 1) alias S-band Doppler velocities within the C-band Nyquist interval $[-13.3, 13.3] \text{ m s}^{-1}$, 2) run the various dealiasing algorithms on these artificially aliased velocities, and 3) use the S-band velocity field as a reference to compare the results of the different algorithms. We used Py-ART's region-based algorithm on the aliased S-band data first and checked manually that these few aliased points were correctly dealiasied (UNRAVEL would have removed noise and artifacts from the data).

The data coming from Terrey Hills' radar are measurements from heavy rainfall events from 2 to 11 October 2018. This period corresponds to 1149 volumetric scans (14 elevation angles). An animation

of this week of data is available online.¹ This week of rainfall is interesting because it covers a variety of cases: large precipitating systems, small precipitating systems with large areas with nonsignificant returns (popcorn convection), strong wind shear, spiraling velocities, clear-air data, and while not visible on the animation, important ground clutter above the blue mountains ($\sim 50 \text{ km}$ west from Sydney). Figure 5 is an example of data from 0542 UTC 7 October 2018. Figure 5a is the raw S-band velocity field, Fig. 5b is the artificially aliased velocity field to $V_N = 13.3 \text{ m s}^{-1}$, and Fig. 5c is the result given by UNRAVEL.

1) 4DD

The 4DD algorithm is the one used operationally at the Australian Bureau of Meteorology to dealias the Doppler velocity. 4DD uses the four-dimensionality available in an operational Doppler radar data stream. It examines one tilt angle at a time, beginning at the highest elevation. It then dealiasies each tilt from top to bottom elevation until the entire radial velocity volume is corrected. 4DD requires the dealiasing results of the previous radar scan, or the wind profile from radiosoundings.

For the purpose of this study, we gave as input to 4DD the results coming from the S-band measurements to initialize the process. Because the data coming from the S-band measurements are accurate, and not approximations of the wind fields, this should have a positive impact on the 4DD results. For all the following scans, the previous dealiasing results produced by 4DD are given as an input to the 4DD. These are the ideal conditions for the 4DD algorithm and do not represent the reality in an operational context. The results shown for 4DD in this study should therefore be considered as the highest possible accuracy that can be achieved with this method.

2) REGION BASED

The region-based algorithm is part of Py-ART (Helmus and Collis 2016). This dealiasing algorithm was created to mimic the decisions an operator would make when hand dealiasing a volume. Specifically, the algorithm aims to identify continuous regions of similar velocities and then dealiasies these regions against each other to minimize the difference in velocities along the boundaries. To identify these continuous regions, the algorithm first defines intervals to group the velocities. While this is specifiable by the user, the

¹ <http://www.theweatherchaser.com/radar-loop/IDR71I-sydney-terrey-hills/2018-10-03-00/2018-10-12-00>.

FIG. 5. Data from S-band Doppler radar at Terrey Hills at 0542 UTC 7 Oct 2018. (a) Raw velocity field, the original V_N is $26.6 m s^{-1}$, (b) the artificially aliased velocity field to $V_N = 13.3 m s^{-1}$, (c) UNRAVEL dealiasing results of the artificially aliased velocity field. Because of the despeckling and filtering

default version of the algorithm splits the Nyquist cointerval into three equal intervals.

Once the regions have been identified, the velocity gates along the boundary of each adjacent region are compared. If this comparison finds that the average difference between the two sets of gates is greater than V_N , the smaller of the two regions is adjusted by a factor of $2nV_N$ to minimize the average difference. Once two regions have been dealiased, the regions are merged and the algorithm continues. Regions are compared and merged according to the size of the boundary between gates. Regions that have the largest boundaries, as determined by counting adjacent gates, are dealiased and merged first. The algorithm ends when all regions have been merged or no boundaries exist between separate regions.

3) MULTIPASS

The Zhang and Wang (2006) algorithm is an automated 2D multipass velocity dealiasing scheme, this algorithm is the seminal work upon which UNRAVEL, and also the He et al. (2012a) algorithm, are based. This algorithm being the backbone of UNRAVEL, we just conceived a new driver that calls the different dealiasing modules the way described by Zhang and Wang (2006).

4) UNWRAP

The unwrap algorithm is a fast two-dimensional phase-unwrapping algorithm that uses a reliability function and does not follow a continuous path to perform the unwrapping operation. It has not been developed specifically for radars, it comes from the field of applied optics, but it is used to solve a similar problem of data aliasing inside a Nyquist cointerval. It is also available in Py-ART.

b. Sensitivity test on parameter α

As stated previously, dealiasing consists in finding n from Eq. (2) that satisfies Eq. (3). All the various dealiasing modules presented in section 3 depends on a parameter called α that represents the required quality of the fit (or agreement) between the reference velocity and the dealiased one. $\alpha = 0$ only allows for a perfect fit, $\alpha = 1$ is the maximum possible value as it corresponds to velocity differences of exactly half the Nyquist cointerval, while larger values of α allow for larger mismatches between the reference velocity and the dealiased one. Thus, α is a normalized parameter varying

←

processing made by UNRAVEL, some gates may appear missing; these gates are generally individual pixels of noise (high-variability data). Isoradials are plotted every 50 km.

FIG. 6. Mean error-rate profile of a week of data from 4 to 11 Oct 2018 of heavy rainfall events by the Doppler radar at Terrey Hills at each tilts for the UNRAVEL for $\alpha = 0.4$, $\alpha = 0.6$, and $\alpha = 0.8$. The number of volume used is 1149 and the sample size (of actual data, excluding missing values) is around 6.4×10^8 .

between 0 and 1. The risk is that if α is too large, then some aliased points may be considered as valid values. On the other hand, if α is too small, then the dealiasing modules might incorrectly dealias Doppler velocities associated with high gradients.

Zhang and Wang (2006) proposed $\alpha = 0.8$. Thanks to the dataset from Terrey Hills S-band Doppler radar, a sensitivity test on the parameter α can be made to see what value of α gives the best dealiasing results. Figure 6 shows the error rates at each elevation, that is, the total number of error points divided by the total number of existing points (i.e., not being censored) for a given

elevation across all volumes for $\alpha = 0.4$, $\alpha = 0.6$, and $\alpha = 0.8$. The best results at all elevations unequivocally occur with $\alpha = 0.6$ and $\alpha = 0.8$, which is in line with what Zhang and Wang (2006) proposed. The stricter $\alpha = 0.4$ only performs well at the lowest elevation and at higher elevations too, but fails more often than larger α for elevations between 0.9° and 7.4° . The lowest elevation is the most impacted by noise, artifacts, and clutter, so it can be understood under these conditions that a strict value can perform better. However, the strictness tends to be a hindrance as a strict valid value condition gives rise to more false alarms. Thus, as proposed by Zhang and Wang (2006), the default value for UNRAVEL is $\alpha = 0.8$, as Fig. 6 shows that it performs well at all elevations, including the lowest.

c. Error rates

The five dealiasing algorithms are being compared by computing the average error rate profile at the different elevation angles (Fig. 7a). A gate is considered an error if the absolute difference between the dealiasing result and the reference is more than 1 m s^{-1} . When a dealiasing algorithm fails, it generally fails by a multiple of $2V_N$. Figure 7a shows the error rates for each radar elevation. Different error rates are estimated in the following to better understand error sources in the different algorithms: the total number of wrongfully dealiasing gates E_t divided by the total number of gates N_t (Fig. 7a); the total number of wrongfully processed gates that should have been dealiasing E_a divided by the total number of gates that were aliased N_a (Fig. 7b); and also the total number of wrongfully processed gates that should not have been dealiasing $E_t - E_a$ divided by the total number of gates that were not aliased $N_t - N_a$ (Fig. 7c).

UNRAVEL outperforms the other techniques with a total error rate lower than 0.2%, followed by 4DD, region-based, the multipass algorithm of Zhang and Wang (2006), and far behind the unwrap algorithm. UNRAVEL not only makes the smallest number of errors, by far, in the lowest elevation scans; it is also the most consistent technique for all elevation tilts. All the algorithms have similar trends for elevations above 7.4° , they all perform well. As the lower tilts tend to be noisier and contain more artifacts it means that UNRAVEL is capable to identify and filter high-variability data better than the other algorithms. UNRAVEL has the best average performance of the evaluated algorithms. UNRAVEL and multipass have the lowest rate of “false alarm,” that is, gates that should not have been dealiasing compared to the other algorithms. This is in part due to the iterative construction of the dealiasing and also the checking modules that force a certain

FIG. 7. Mean error-rate profiles for each elevation: (a) for all gates E_t/N_t , (b) only for gates that needed to be dealiased outside of the Nyquist counterval E_a/N_a , and (c) only for gates that did not need to be dealiased $(E_t - E_a)/(N_t - N_a)$. N_t is the total number of gates, N_a is the number of radar gates that need to be dealiased, E_t

consistency of the velocity unfolding between successive elevations.

The results of the unwrap algorithm have been divided by a factor 10 on Figs. 7a and 7c; otherwise, they would have been out of scale. Because of the poor performance of the unwrap algorithm, it is not discussed further in this study.

The most common sources of dealiasing errors are due to noisy data on the edges of convective systems, range bins with residual ground clutter contributions, all producing artificially high-velocity gradients and spurious Doppler measurements. Dealiasing algorithms also tend to have problems in the lowest elevation scans, where most of the data are present. Figure 8 shows the error rate (i.e., the total number of errors divided by the sample size, E_t/N_t) as a function of the signal-to-noise ratio (SNR; in dB) and the elevation angle for UNRAVEL (Fig. 8a), region based (Fig. 8b), 4DD (Fig. 8c), and multipass (Fig. 8d). Similarly, Fig. 9 shows the error rate made over the points that should have been dealiased (E_a/N_a), and Fig. 10 shows the error rate made over the points that were correct before dealiasing and should not have been dealiased.

UNRAVEL main source of errors are the high-SNR gates at higher elevations (Fig. 8a). However, Fig. 8 demonstrates that UNRAVEL fails much less often than the other algorithms. UNRAVEL handles particularly well the low-SNR gates (i.e., noisy data). Moreover, Fig. 9 shows that UNRAVEL correctly identifies and dealiases area of aliasing much more often than the other algorithms. The only notable source of errors made over aliased area are for high-SNR values, but even then, UNRAVEL fares much better than the other algorithms.

The region-based algorithm handles relatively well the low-elevation scans; however, it clearly underperforms for elevation above 13.3° (Fig. 8b) for all SNR. This is possibly because there are more isolated regions at high elevations, and thus region based is probably more prone to errors in such conditions. Also, Fig. 10b shows that the main source of error made by the region-based technique

←

is the total number of gates wrongly corrected, and E_a is the total number of wrongly corrected gates that needed to be dealiased. For (a) and (c), the error rates of the unwrap algorithm have been divided by a factor of 10 in order to be comparable to the other algorithms. Average profiles coming from a week of data from 4 to 11 Oct 2018 (1149 volumetric scans) of heavy rainfall events by the Doppler radar at Terrey Hills. For the UNRAVEL, region-based (Helmus and Collis 2016), unwrap (Herráez et al. 2002), 4DD (James and Houze 2001), and the multipass algorithm from Zhang and Wang (2006).

FIG. 8. Error rate E_t/N_t as a function of the signal to noise ratio (dB) and the elevation for (a) UNRAVEL, (b) region-based, (c) 4DD, and (d) multipass dealiasing algorithms for a week of data from 4 to 11 Oct 2018 (1149 volumetric scans, sample size of $\sim 6 \times 10^8$) of heavy rainfall events by the Doppler radar at Terrey Hills.

is actually coming from “false alarms,” that is, velocities that should not have been dealiased.

The 4DD algorithm fails for low SNR at all elevations, and even more particularly for the low elevation. It also tends to fail more frequently for high SNR at high-elevation. Because 4DD starts by dealiasing the highest elevation, it may explain why the 4DD error rate is so high at the highest elevation. Figure 8c shows that 4DD has trouble managing the noise at the lowest elevation. Figure 9c notably shows that 4DD manages quite poorly the high-SNR regions, at almost all elevations, and notably the lowest ones. These regions are of prime

importance are they correspond to regions of intense convection. However, 4DD tends to work well for the in-between elevations.

Figure 8d shows that the multipass algorithm has a “failure” behavior similar to UNRAVEL. Multipass tends to make more mistakes at higher elevation for large SNR values, but also handles well the lowest elevation scans. However, multipass fails more often (Fig. 8d) than UNRAVEL. Because UNRAVEL is based in part on Zhang and Wang (2006) algorithm, these similarities are not unexpected. Nonetheless, UNRAVEL contains more dealiasing and checking

FIG. 9. As in Fig. 8, but for the error rate made on aliased gates E_a/N_a .

strategies than multipass, these additional strategies allow for a better dealiasing performance for UNRAVEL (0.2% average failure rate for UNRAVEL compared to ~1% for multipass). Figures 9 and 10 show that both UNRAVEL and multipass are more robust than the other dealiasing techniques. They handle better aliased regions of high SNR (Fig. 9), and they create much fewer “false alarms” (Fig. 10).

We also looked (not shown) at how the error rate behaved as a function of $dv/d\theta$ or dv/dr (velocity gradients along the azimuthal and range directions, respectively). All the algorithms fail more often for the gradients exceeding the maximum allowed gradient [i.e., the maximum allowed difference between V_R and

V_i from Eq. (3)] in UNRAVEL or very low gradient values (hence, not related to the velocity gradient itself), for all elevations. However, we did not find any notable relationship when looking at the error rate as a function of both the SNR and the velocity gradients.

5. Conclusions

UNRAVEL is an open-source, modular, Doppler velocity dealiasing algorithm. UNRAVEL is an automated scheme that finds reference velocities in the input velocity field and does not require external data sources. UNRAVEL is the best-performing algorithm

FIG. 10. As in Fig. 8, but for the error rate made nonaliased gates $(E_t - E_a)/(N_t - N_a)$.

with the smallest error ratio when compared to the 4DD algorithm (James and Houze 2001), the unwrap algorithm (Herráez et al. 2002), the region-based algorithm from Py-ART (Helmus and Collis 2016), and the multipass algorithm from Zhang and Wang (2006) for the test period selected in this work.

The reasons behind the good performance of UNRAVEL is that it takes what most algorithms of the past two decades have to offer and constrain them to a certain domain of competence by putting checks and limitations on each module. It uses a flag array, like James and Houze (2001), to record the state of the dealiasing process of each gate. It is a multipass algorithm like Zhang and Wang (2006), the first pass requiring strict

continuity, and after each pass, to build more and more results, the strictness of the algorithm is relaxed by looking at a larger area for reference, and by progressively allowing a larger gradient between the surrounding reference and the velocity gate to dealias. Most importantly, it is an algorithm that will perform checks once the entire velocity field is dealiased.

UNRAVEL has been tested against S- and C-band data across more than 1000 volume scans of data coming from several radars of the Australian radar network. It is freely available on GitHub and work is undergoing to make UNRAVEL part of the Py-ART package. Because of its design, UNRAVEL can be run on multiple threads. This is the principal reason why

UNRAVEL is entirely written in Python. While it is perceived as a relatively slow programming language, it also scales up processing quite easily. For example, the entire archive used in this study was processed in about 35 min by UNRAVEL (16 processes at the same time, 2 GB of memory for each process) while the same archive was processed in 20 h by 4DD (4DD is an inherently single-threaded algorithm). Depending on how much data there are to dealias, UNRAVEL takes from 30 to 50 s on one volumetric scan (150-km range, 250-m gate spacing).

Acknowledgments. This work has been supported by the U.S. Department of Energy Atmospheric Systems Research Program through Grant DE-SC0014063. The contribution of Scott Collis through Argonne National Laboratory was supported by the U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research, under Contract DE-AC02-06CH11357. This work was partly support by the Climate Model Development and Validation activity of the Department of Energy office of Science, Office of Biological and Environmental Research. The authors would also like to thank Michel Chong from Météo France for his valuable insights on dealiasing techniques.

REFERENCES

- Bergen, W. R., and S. C. Albers, 1988: Two- and three-dimensional de-aliasing of Doppler radar velocities. *J. Atmos. Oceanic Technol.*, **5**, 305–319, [https://doi.org/10.1175/1520-0426\(1988\)005<0305:TATDDA>2.0.CO;2](https://doi.org/10.1175/1520-0426(1988)005<0305:TATDDA>2.0.CO;2).
- Doviak, R., and D. Zrnić, 2006: *Doppler Radar and Weather Observations*. Dover Publications, 562 pp.
- Eilts, M. D., and S. D. Smith, 1990: Efficient dealiasing of Doppler velocities using local environment constraints. *J. Atmos. Oceanic Technol.*, **7**, 118–128, [https://doi.org/10.1175/1520-0426\(1990\)007<0118:EDODVU>2.0.CO;2](https://doi.org/10.1175/1520-0426(1990)007<0118:EDODVU>2.0.CO;2).
- Gong, J., L. L. Wang, and Q. Xu, 2003: A three-step dealiasing method for Doppler velocity data quality control. *J. Atmos. Oceanic Technol.*, **20**, 1738–1748, [https://doi.org/10.1175/1520-0426\(2003\)020<1738:ATDMFD>2.0.CO;2](https://doi.org/10.1175/1520-0426(2003)020<1738:ATDMFD>2.0.CO;2).
- He, G., G. Li, X. Zou, and P. S. Ray, 2012a: A velocity dealiasing scheme for synthetic C-band data from China's new generation weather radar system (CINRAD). *J. Atmos. Oceanic Technol.*, **29**, 1263–1274, <https://doi.org/10.1175/JTECH-D-12-00010.1>.
- , —, —, and —, 2012b: Applications of a velocity dealiasing scheme to data from the China new generation weather radar system (CINRAD). *Wea. Forecasting*, **27**, 218–230, <https://doi.org/10.1175/WAF-D-11-00054.1>.
- Helmus, J. J., and S. M. Collis, 2016: The Python ARM Radar Toolkit (Py-ART), a library for working with weather radar data in the Python programming language. *J. Open Res. Software*, **4**, e25, <https://doi.org/10.5334/jors.119>.
- Herráez, M. A., D. R. Burton, M. J. Lalor, and M. A. Gdeisat, 2002: Fast two-dimensional phase-unwrapping algorithm based on sorting by reliability following a noncontinuous path. *Appl. Opt.*, **41**, 7437, <https://doi.org/10.1364/AO.41.007437>.
- Holleman, I., and H. Beekhuis, 2003: Analysis and correction of dual PRF velocity data. *J. Atmos. Oceanic Technol.*, **20**, 443–453, [https://doi.org/10.1175/1520-0426\(2003\)20<443:AAOCODP>2.0.CO;2](https://doi.org/10.1175/1520-0426(2003)20<443:AAOCODP>2.0.CO;2).
- James, C. N., and R. A. Houze, 2001: A real-time four-dimensional Doppler dealiasing scheme. *J. Atmos. Oceanic Technol.*, **18**, 1674–1683, [https://doi.org/10.1175/1520-0426\(2001\)018<1674:ARTFDD>2.0.CO;2](https://doi.org/10.1175/1520-0426(2001)018<1674:ARTFDD>2.0.CO;2).
- Jing, Z., and G. Wiener, 1993: Two-dimensional dealiasing of Doppler velocities. *J. Atmos. Oceanic Technol.*, **10**, 798–808, [https://doi.org/10.1175/1520-0426\(1993\)010<0798:TDDODV>2.0.CO;2](https://doi.org/10.1175/1520-0426(1993)010<0798:TDDODV>2.0.CO;2).
- Joe, P., and P. T. May, 2003: Correction of dual PRF velocity errors for operational Doppler weather radars. *J. Atmos. Oceanic Technol.*, **20**, 429–442, [https://doi.org/10.1175/1520-0426\(2003\)20<429:CODPVE>2.0.CO;2](https://doi.org/10.1175/1520-0426(2003)20<429:CODPVE>2.0.CO;2).
- Keenan, T., K. Glasson, F. Cummings, T. S. Bird, J. Keeler, and J. Lutz, 1998: The BMRC/NCAR C-band polarimetric (C-POL) radar system. *J. Atmos. Oceanic Technol.*, **15**, 871–886, [https://doi.org/10.1175/1520-0426\(1998\)015<0871:TBNCBP>2.0.CO;2](https://doi.org/10.1175/1520-0426(1998)015<0871:TBNCBP>2.0.CO;2).
- Li, G., G. He, X. Zou, and P. S. Ray, 2014: A velocity dealiasing scheme for C-band weather radar systems. *Adv. Atmos. Sci.*, **31**, 17–26, <https://doi.org/10.1007/s00376-013-2251-8>.
- Lim, E., and J. Sun, 2010: A velocity dealiasing technique using rapidly updated analysis from a four-dimensional variational Doppler radar data assimilation system. *J. Atmos. Oceanic Technol.*, **27**, 1140–1152, <https://doi.org/10.1175/2010JTECHA1300.1>.
- Louf, V., and R. C. Jackson, 2020: UNRAVEL: A modular Doppler velocity dealiasing technique for weather radars. Zenodo, <https://doi.org/10.5281/zenodo.3674157>.
- , A. Protat, R. A. Warren, S. M. Collis, D. B. Wolff, S. Raunyar, C. Jakob, and W. A. Petersen, 2019: An integrated approach to weather radar calibration and monitoring using ground clutter and satellite comparisons. *J. Atmos. Oceanic Technol.*, **36**, 17–39, <https://doi.org/10.1175/JTECH-D-18-0007.1>.
- Ray, P. S., and C. Ziegler, 1977: De-aliasing first-moment Doppler estimates. *J. Appl. Meteor.*, **16**, 563–564, [https://doi.org/10.1175/1520-0450\(1977\)016<0563:DAFMDE>2.0.CO;2](https://doi.org/10.1175/1520-0450(1977)016<0563:DAFMDE>2.0.CO;2).
- Rousseeuw, P. J., and C. Croux, 1993: Alternatives to the median absolute deviation. *J. Amer. Stat. Assoc.*, **88**, 1273–1283, <https://doi.org/10.1080/01621459.1993.10476408>.
- Sauvageot, H., 1992: *Radar Meteorology*. Artech House, 366 pp.
- Skolnik, M., 2008: *Radar Handbook*. 3rd ed. McGraw-Hill Education, 1328 pp.
- Speer, M., P. Wiles, and A. Pepler, 2009: Low pressure systems off the New South Wales coast and associated hazardous weather: Establishment of a database. *Aust. Meteor. Oceanogr. J.*, **58**, 29–39, <https://doi.org/10.22499/2.5801.004>.
- Tabary, P., G. Scialom, and U. Germann, 2001: Real-time retrieval of the wind from aliased velocities measured by Doppler radars. *J. Atmos. Oceanic Technol.*, **18**, 875–882, [https://doi.org/10.1175/1520-0426\(2001\)018<0875:RTROTW>2.0.CO;2](https://doi.org/10.1175/1520-0426(2001)018<0875:RTROTW>2.0.CO;2).
- , F. Guibert, L. Perier, and J. P. du Chatelet, 2006: An operational triple-PRT Doppler scheme for the French radar network. *J. Atmos. Oceanic Technol.*, **23**, 1645–1656, <https://doi.org/10.1175/JTECH1923.1>.

- Tahanout, M., A. E. H. Adane, and J. P. D. Chatelet, 2015: An improved m-PRT technique for spectral analysis of weather radar observations. *IEEE Trans. Geosci. Remote Sens.*, **53**, 5572–5582, <https://doi.org/10.1109/TGRS.2015.2425549>.
- Xu, Q., and K. Nai, 2017: Mesocyclone-targeted Doppler velocity dealiasing. *J. Atmos. Oceanic Technol.*, **34**, 841–853, <https://doi.org/10.1175/JTECH-D-16-0170.1>.
- Yamada, Y., and M. Chong, 1999: VAD-based determination of the Nyquist interval number of Doppler velocity aliasing without wind information. *J. Meteor. Soc. Japan*, **77**, 447–457, https://doi.org/10.2151/jmsj1965.77.2_447.
- Zhang, J., and S. Wang, 2006: An automated 2D multipass Doppler radar velocity dealiasing scheme. *J. Atmos. Oceanic Technol.*, **23**, 1239–1248, <https://doi.org/10.1175/JTECH1910.1>.
- Zrnić, D., and P. Mahapatra, 1985: Two methods of ambiguity resolution in pulse Doppler weather radars. *IEEE Trans. Aerosp. Electron. Syst.*, **AES-21**, 470–483, <https://doi.org/10.1109/TAES.1985.310635>.