STAGGERED PRT WITH GROUND CLUTTER FILTERING

ALGORITHM DESCRIPTION

by Sebastián Torres and David Warde
National Severe Storms Laboratory
March 6, 2009

PREFACE

This document extends the previous Staggered PRT algorithm description from May 2008 by including the SACHI ground clutter filter. It includes a high-level algorithm description with the overall processing logic followed by a detailed explanation of each pre-computation and processing step. In order to fit within the required signal processing pipeline, the output of the SACHI filter has been modified to produce "manufactured" lag-T_1 and lag-T_2 autocorrelations. That is, the magnitudes of the lag-T_1 and lag-T_2 autocorrelations have been adjusted to preserve the filtered spectrum width estimate; while, the phase of the lag-T_1 and lag-T_2 autocorrelations have been aliased to preserve the filtered velocity estimate. Additionally, noise compensation vectors have been added to the algorithm to properly account for the reduced number of spectral components used in the filtered power estimates. Readers familiar with previous descriptions of the SACHI algorithm will notice that the notation in this document has been updated to blend with the previous SPRT algorithm description and to follow usual conventions. Further, to ease implementation and reduce ambiguity, many of the steps in the SACHI algorithm are now described in algorithmic form. Despite these changes, the functionality of the filter as described in our Report 11 remains unchanged. The DC removal ground clutter filter has been retained to operate on those range gates where only long-PRT data is available and ground clutter filtering is needed. Finally, the addition of the SACHI ground clutter filter resulted in a few constraints compared to the previous description. Specifically, the PRT ratio must be 2/3, T_1 must be the short PRT, and the number of samples must be even. Nevertheless, meeting ground clutter filtering requirements makes the SPRT algorithm viable for operational implementation.

ASSUMPTIONS

1) The transmission sequence alternates two pulse repetition times (PRT) as: T_1, T_2, T_1, T_2 … for a total of M pulses.

2) The PRT ratio $T_1/T_2 = 2/3$, where $\kappa_m = 2$, $\kappa_n = 3$ and $T_2 - T_1 = T_u$.

3) All range gates are available and there is a perfect alignment of range gates between the two PRTs (i.e., a given range gate represents the same resolution volume in space for every transmitted pulse). Also, the number of range gates for each PRT is: $N_1 = T_1/\tau_s$ and $N_2 = T_2/\tau_s$, where τ_s is the sampling period.

4) There are no significant echoes beyond the maximum unambiguous range corresponding to T_2 (r_{a2}).

5) The number of staggered PRT samples per range gate (M) is even.

6) The algorithm operates on a radial worth of data at a time.

INPUTS

1) Complex time-series data:

$V(n, m) = I(n, m) + jQ(n, m)$, where $0 \leq n < N_1$ for even m, $0 \leq n < N_2$ for odd m and $0 \leq m < M$. Note that n indexes the range gates and m the sweeps (or pulses).

2) Associated metadata:

λ is the radar wavelength in meters
Noise is the noise power in linear units
dBZ0 is the system calibration constant in dB
ATMOS is the elevation-dependent atmospheric attenuation in dB/km
ΔR is the spacing between range gates in km ($\Delta R = c\,\tau_s/2$)
T_Z is the signal-to-noise ration threshold for reflectivity in dB
T_V is the signal-to-noise ration threshold for velocity in dB
T_W is the signal-to-noise ration threshold for spectrum width in dB
T_O is the overlaid threshold in dB which is sometimes referred to as *TOVER*.

3) Data window:

$d'(m)$, where $0 \leq m < 5M/2$. Note that d' does not need to be normalized or scaled in any way. A tapered data window such as the Blackman window is recommended for best performance of the SACHI ground clutter filter. Otherwise, rectangular window (i.e., no window) should be applied.

4) Ground clutter filter bypass map:

$B(n)$, where n indexes the range bins with the same resolution as the time-series data along a radial, and the map corresponds to the elevation and azimuth of the radial being processed. B is 0 if clutter filtering is required and 1 otherwise.

OUTPUTS

1) Reflectivity, Doppler velocity, and spectrum width:

$Z(n)$ for $0 \leq n < N_2$,
$v(n)$ and $w(n)$ for $0 \leq n < N_2$.

2) Signal-to-noise ratio and overlaid censoring flags:

$NS_Z(n)$, $NS_V(n)$ and $NS_W(n)$ for $0 \leq n < N_2$,
$OV_V(n)$ and $OV_W(n)$ for $0 \leq n < N_2$.

FUNCTIONS AND CONVENTIONS

1) $|\cdot|$ – Returns the absolute value of a complex number or the absolute value of each element of a matrix of complex numbers.

2) arg – Returns the principal phase angle of the input complex number in radians. The algorithm is written to accommodate this phase in the interval $[0, 2\pi)$ or $[-\pi, \pi)$.

3) $\arg\min_{k}$ – Returns the index k to the element in the input vector that has the minimum value.

4) diag – Returns a square matrix with the input vector along the principal diagonal (row index = column index) of the matrix and all other elements not on the principal diagonal equal to zero. The number of rows (columns) of the matrix is equal to the number of elements in the vector.

5) ceiling – Returns the smallest integer value not less than the input number.

6) floor – Returns the largest integer value not greater than the input number.

7) round – Returns the nearest integer to the input number.

8) Italicized names are used to represent scalars (e.g., *Noise*).

9) Bolded names are used to represent vectors or matrices (e.g., **A**). Italicized names with indexing in parentheses are used to represent elements of a vector or matrix [e.g., $A(i,j)$].

HIGH-LEVEL ALGORITHM DESCRIPTION

If first run of SPRT algorithm
    1) Pre-computation of velocity dealiasing rules
    2) Pre-computation of M-independent SACHI filter parameters
End
If the number of samples (M) changed
    3) Pre-computation of window parameters
    4) Pre-computation of M-dependent SACHI filter parameters
End
For each range bin n, where $0 \leq n < N_2$
    If $B(n) = 0$ AND $n < N_1$
        5) SACHI Clutter Filtering
    Else
        If $B(n) = 0$
            6) DC Removal Clutter Filtering
        Else
            7) No Clutter Filtering
        End
        8) Power and correlation computations for each PRT
        9) Combined power computation
    End
End
11) Strong point clutter canceling
For each range bin n, where $0 \leq n < N_2$
    12) Signal power computation
    13) Reflectivity computation
    14) Velocity computation
    15) Spectrum width computation
    16) Determination of significant returns for reflectivity
    17) Determination of significant returns for velocity
    18) Determination of significant returns for spectrum width
End
For each range bin n, where $0 \leq n < N_2$
    19) Determination of overlaid returns for velocity and spectrum width
End

STEP-BY-STEP ALGORITHM DESCRIPTION

1) Pre-computation of velocity dealiasing rules

This method is described in the paper "Design, Implementation, and Demonstration of a Staggered PRT Algorithm for the WSR-88D" by Torres et al. (2004). Herein, VDA_c are the normalized velocity difference transfer function (VDTF) constant values and VDA_p are the normalized number of Nyquist co-intervals for dealiasing.

A set of velocity dealiasing rules could be pre-computed at the initiation of the SPRT algorithm as follows:

(Compute type-I and II positive VDTF discontinuity points. κ_m and κ_n are the integers in the PRT ratio)

$p = 0$
While $2p + 1 < \kappa_m$
    $D_1(p) = (2p + 1)/\kappa_m$
    $TYPE_1(p) = 1$
    $p = p + 1$
End
$q = 0$
While $2q + 1 < \kappa_n$
    $D_2(q) = (2q + 1)/\kappa_n$
    $TYPE_2(q) = 2$
    $q = q + 1$
End

(Create TYPE by combining and sorting and both sets of discontinuity points)

Concatenate D_1 and D_2 to create D with $p + q$ elements.
Concatenate $TYPE_1$ and $TYPE_2$ to create $TYPE$ with $p + q$ elements.
Sort $TYPE$ in a "slave" mode using D as the "master".

(Compute VDTF constants and dealiasing factors for non-negative discontinuity points)

$VDA_c(p + q) = 0$
$VDA_p(p + q) = 0$
For $0 \leq k < p + q$
    If $TYPE(k) = 1$
        $VDA_c(p + q + k + 1) = VDA_c(p + q + k) - 2/\kappa_m$
        $VDA_p(p + q + k + 1) = VDA_p(p + q + k) + 1/\kappa_m$
    Else
        $VDA_c(p + q + k + 1) = VDA_c(p + q + k) + 2/\kappa_n$
        $VDA_p(p + q + k + 1) = VDA_p(p + q + k)$
    End
End

(Compute VDTF constants and dealiasing factors for negative discontinuity points)

For $-(p + q) \leq k < 0$
    $VDA_c(p + q + k) = -VDA_c(p + q - k)$
    $VDA_p(p + q + k) = -VDA_p(p + q - k)$
End

2) Pre-computation of M-independent SACHI filter parameters

This method is described in NSSL Signal Design and Processing Techniques for WSR-88D Ambiguity Resolution (Report 3, Report 9 and Report 11). The SACHI filter parameters could be pre-computed at the initiation of the SPRT algorithm as follows:

(Create 5-by-5 convolution matrix, $\mathbf{C_r}$)

$$\mathbf{C_r} = \begin{bmatrix} C(0) & C(4) & C(3) & C(2) & C(1) \\ C(1) & C(0) & C(4) & C(3) & C(2) \\ C(2) & C(1) & C(0) & C(4) & C(3) \\ C(3) & C(2) & C(1) & C(0) & C(4) \\ C(4) & C(3) & C(2) & C(1) & C(0) \end{bmatrix},$$

where $C(k) = \frac{1}{\sqrt{10}} \sum_{n=0}^{4} c(n) \exp\left(-j2\pi nk/5\right)$; for $0 \le k < 5$ and $\mathbf{c} = [1, 0, 1, 0, 0]$.

(Calculate magnitude deconvolution matrix, $\mathbf{C_{md}}$)
(Note: The following formulas are written in matrix algebra notation)

$$\mathbf{C_{md}} = \left|\mathbf{C_r}\right|^{-1}$$

(Calculate matrices $\mathbf{C_{f1}}$ and $\mathbf{C_{f2}}$ using 1^{st} and 5^{th} columns of $\mathbf{C_r}$, $\mathbf{C_{r,0}}$ and $\mathbf{C_{r,4}}$)

$$\mathbf{C_{f1}} = \mathbf{C}_{\mathbf{r},0}\mathbf{C}_{\mathbf{r},0}^{*T}$$

$$\mathbf{C_{f2}} = \mathbf{C}_{\mathbf{r},4}\mathbf{C}_{\mathbf{r},4}^{*T}$$

(Calculate the correction coefficients ξ_1 and ξ_2 for correction vector $\mathbf{X}$)

$$\xi_k = \left|\mathbf{C_{md}}\left\{\mathbf{C}_{\mathbf{r},k} - \left(\mathbf{C}_{\mathbf{r},0}^{*T}\mathbf{C}_{\mathbf{r},k}\right)\mathbf{C}_{\mathbf{r},0}\right\}\right|^{-1}; k = 1, 2.$$

3) Pre-computation of window parameters

(Calculate the extended number of coefficients)

$M_x = 5M / 2$

(Calculate the number of pulse pairs)

$M_p = M / 2$

(Calculate normalized window d for un-normalized window function d' with M_x points)

$$d(m) = d'(m)\left(\sqrt{\frac{1}{M_x}\sum_{m=0}^{M_x-1}\left[d'(m)\right]^2}\right)^{-1}; \; 0 \le m < M_x.$$

(Calculate window correction factor for lag-1)

$$d_c = \frac{1}{M_x}\sum_{m=0}^{M_x-2} d(m)d(m+1)$$

4) Pre-computation of M-dependent SACHI filter parameters

(Compute correction vector, **X***)*
For $0 \le k < \text{ceiling}(M_p/2)$
$X(k) = 1$
End
For $\text{ceiling}(M_p/2) \le k < \text{ceiling}(M_p/2) + M_p$
$X(k) = \xi_1$
End
For $\text{ceiling}(M_p/2) + M_p \le k < \text{ceiling}(M_p/2) + 3M_p$
$X(k) = \xi_2$
End
For $\text{ceiling}(M_p/2) + 3M_p \le k < \text{ceiling}(M_p/2) + 4M_p$
$X(k) = \xi_1$
End
For $\text{ceiling}(M_p/2) + 4M_p \le k < M_x$
$X(k) = 1$
End

PROCESSING STEPS

5) SACHI Clutter Filtering

The SACHI filter algorithm is used when clutter filtering is required inside the maximum unambiguous range corresponding to T_1 (r_{a1}).

(Form derived time series, V_d, from input time series V)
For $0 \le m < M_p$
$V_d(5m) = V(n, 2m)$
$V_d(5m + 1) = 0$
$V_d(5m + 2) = V(n, 2m+1)$
$V_d(5m + 3) = 0$
$V_d(5m + 4) = 0$
End

(Compute DFT of windowed extended time series power compensated for added zeroes)

$$F(k) = \left(\sqrt{\frac{5}{2}}\right)\left(\frac{1}{M_x}\sum_{m=0}^{M_x-1} V_d(m)d(m)\exp(-j2\pi km/M_x)\right);\ k = 0, 1 \ldots, M_x - 1.$$

(Determine clutter filter width parameter, q)
(Use GMAP to return the number of coefficients identified as clutter, $GMAP_{coef}$. Pass to GMAP the 5^{th} of the Doppler spectrum containing the main clutter replica; i.e., $\{|F(0)|^2, \ldots, |F[\text{ceiling}(M_p/2) - 1]|^2, |F[M_x - \text{floor}(M_p/2)]|^2, \ldots, |F(M_x - 1)|^2\}$; initialize GMAP for spectra with $v_a/5$, and get the number of coefficients identified as clutter to estimate q)
$q = \text{floor}[(GMAP_{coef} + 1)/2]$

(Create clutter filter vectors, $\mathbf{I}_{\mathbf{f1}}$, $\mathbf{I}_{\mathbf{f2}}$, $\mathbf{I}_{\mathbf{1}}$, *and* $\mathbf{I}_{\mathbf{2}}$*)*

For $0 \le k < M_p$
    If $k < q$
        $I_{f1}(k) = 1$
        $I_{f2}(k) = 0$
        $I_1(k) = 0$
        $I_1(k + M_p) = 0$
        $I_1(k + 2M_p) = 0$
        $I_1(k + 3M_p) = 0$
        $I_1(k + 4M_p) = 0$
        $I_2(k) = 1$
        $I_2(k + M_p) = 1$
        $I_2(k + 2M_p) = 1$
        $I_2(k + 3M_p) = 1$
        $I_2(k + 4M_p) = 1$
    ElseIf $k \le M_p - q$
        $I_{f1}(k) = 0$
        $I_{f2}(k) = 0$
        $I_1(k) = 1$
        $I_1(k + M_p) = 1$
        $I_1(k + 2M_p) = 1$
        $I_1(k + 3M_p) = 1$
        $I_1(k + 4M_p) = 1$
        $I_2(k) = 0$
        $I_2(k + M_p) = 0$
        $I_2(k + 2M_p) = 0$
        $I_2(k + 3M_p) = 0$
        $I_2(k + 4M_p) = 0$
    Else
        $I_{f1}(k) = 0$
        $I_{f2}(k) = 1$
        $I_1(k) = 0$
        $I_1(k + M_p) = 0$
        $I_1(k + 2M_p) = 0$
        $I_1(k + 3M_p) = 0$
        $I_1(k + 4M_p) = 0$
        $I_2(k) = 1$
        $I_2(k + M_p) = 1$
        $I_2(k + 2M_p) = 1$
        $I_2(k + 3M_p) = 1$
        $I_2(k + 4M_p) = 1$
    End
End

(Row-wise re-arrange F into a 5-by-M_p matrix, $\mathbf{F_r}$)

For $0 \le k < M_p$

$F_r(0, k) = F(k)$

$F_r(1, k) = F(k + M_p)$

$F_r(2, k) = F(k + 2M_p)$

$F_r(3, k) = F(k + 3M_p)$

$F_r(4, k) = F(k + 4M_p)$

End

(Compute the clutter filtered spectrum matrix, $\mathbf{F_f}$)

(Note: The following formulas are written in matrix algebra notation)

$$\mathbf{F_f} = \mathbf{F_r} - \mathbf{C_{f1}F_r}\,\mathrm{diag}\left(\mathbf{I_{f1}}\right) - \mathbf{C_{f2}F_r}\,\mathrm{diag}\left(\mathbf{I_{f2}}\right)$$

(Magnitude deconvolved matrix, $\mathbf{F_d}$)

$$\mathbf{F_d} = \mathbf{C_{md}}\left|\mathbf{F_f}\right|$$

(Row-wise unfold $\mathbf{F_d}$ into F_{df})

For $0 \le k < M_p$

$F_{df}(k) = F_d(0, k)$

$F_{df}(k + M_p) = F_d(1, k)$

$F_{df}(k + 2M_p) = F_d(2, k)$

$F_{df}(k + 3M_p) = F_d(3, k)$

$F_{df}(k + 4M_p) = F_d(4, k)$

End

(Compute the lag-1 autocorrelation, R_{1df})

$$R_{1df} = \frac{1}{d_c}\sum_{k=0}^{M_x-1}\left|F_{df}(k)\right|^2 \exp\left(j2\pi k / M_x\right)$$

(Compute vector $\mathbf{I_v}$ with M/2 ones centered on $\arg(R_{1df})$)

(Round to the nearest spectral coefficient. Choose symmetric window of coefficients around it)

$$k_{0df} = \mathrm{round}\left[\frac{M_x \arg\left(R_{1df}\right)}{2\pi}\right]$$

If $k_{0df} < 0$

$k_{0df} = k_{0df} + M_x$

End

If $k_{0df} \ge M_x$

$k_{0df} = k_{0df} - M_x$

End

$$k_{1df} = k_{0df} - \mathrm{floor}(M/4)$$

If $k_{1df} < 0$

$k_{1df} = k_{1df} + M_x$

End

$$k_{2df} = k_{0df} + \mathrm{ceiling}(M/4) - 1$$

If $k_{2df} \ge M_x$

$k_{2df} = k_{2df} - M_x$

End

(k_{0df} is the coefficient corresponding to $\arg(R_{1df})$*, k_{1df} and k_{2df} specify the extent of M_p spectral coefficients centered on the mean velocity. If $k_{1df} < k_{2df}$, the ones span from k_{1df} to k_{2df}; otherwise, the ones will span from k_{1df} to $M_x - 1$, and 0 to k_{2df})*

```
If k_1df < k_2df
    For 0 ≤ k < M_x
        If k < k_1df OR k > k_2df
            I_v (k) = 0
        Else
            I_v (k) = 1
        End
    End
Else
    For 0 ≤ k < M_x
        If k < k_1df AND k > k_2df
            I_v (k) = 0
        Else
            I_v (k) = 1
        End
    End
End
```

(Interpolate the elements for the region around zero velocity in F_{df} with linearly interpolated values from S_1 and S_2)

If $q > 0$

$$S_1 = \left|F_{df}(q)\right|^2$$

$$S_2 = \left|F_{df}(M_x - q)\right|^2$$

For $0 \le k < M_x$

If $k < q$

$$F_i(k) = \left[S_2 + (S_1 - S_2)\ (q + k)\ /\ 2q\right]^{1/2}$$

ElseIf $k > M_x - q$

$$F_i(k) = \left[S_2 + (S_1 - S_2)\ (q + k - M_x)\ /\ 2q\right]^{1/2}$$

Else

$$F_i(k) = F_{df}(k)$$

End

End

Else

(Don't interpolate if not needed)

For $0 \le k < M_x$

$$F_i(k) = F_{df}(k)$$

End

End

(Compute the corrected spectrum, F_c)

For $0 \le k < M_x$

$$F_c(k) = F_i(k)\, I_1(k) + F_i(k)\, I_2(k)\, I_v(k)\, X(k)$$

End

(Compute vector $\mathbf{I_c}$ with ones where there's a non-zero spectral component in vector $\mathbf{F_c}$)

For $0 \le k < M_x$

$$I_c(k) = I_1(k) + I_2(k)\, I_v(k)$$

End

(Compute the mean power, P_c, and autocorrelation at lag T_u, R_{1c}, using F_c)

$$P_c = \sum_{k=0}^{M_x-1} \left|F_c(k)\right|^2$$

$$R_{1c} = \frac{1}{d_c} \sum_{k=0}^{M_x-1} \left|F_c(k)\right|^2 \exp\left(j2\pi k/M_x\right)$$

(Retain only M coefficients centered on velocity based on R_{1c} and delete the rest from F_c and I_c)

$$k_{0c} = \text{round}\left[\frac{M_x \arg\left(R_{1c}\right)}{2\pi}\right]$$

If $k_{0c} < 0$
    $k_{0c} = k_{0c} + M_x$
End
If $k_{0c} \geq M_x$
    $k_{0c} = k_{0c} - M_x$
End
$k_{1c} = k_{0c} - M_p$
If $k_{1c} < 0$
    $k_{1c} = k_{1c} + M_x$
End
$k_{2c} = k_{0c} + M_p - 1$
If $k_{2c} \geq M_x$
    $k_{2c} = k_{2c} - M_x$
End

If $k_{1c} < k_{2c}$
    For $0 \leq k < M_x$
        If $k < k_{1c}$ OR $k > k_{2c}$
            $F_m(k) = 0$
            $I_m(k) = 0$
        Else
            $F_m(k) = F_c(k)$
            $I_m(k) = I_c(k)$
        End
    End
Else
    For $0 \leq k < M_x$
        If $k < k_{1c}$ AND $k > k_{2c}$
            $F_m(k) = 0$
            $I_m(k) = 0$
        Else
            $F_m(k) = F_c(k)$
            $I_m(k) = I_c(k)$
        End
    End
End

(Compute the modified mean power, P_m, and autocorrelation at lag T_u, R_{1m}, using F_m)

$$P_m = \sum_{k=0}^{M_x-1} \left|F_m(k)\right|^2$$

$$R_{1m} = \frac{1}{d_c} \sum_{k=0}^{M_x-1} \left|F_m(k)\right|^2 \exp\left(j2\pi k/M_x\right)$$

(Compute noise correction factors)

$$N_c = \frac{1}{M_x} \sum_{k=0}^{M_x - 1} I_c(k)$$

$$N_m = \frac{1}{M_x} \sum_{k=0}^{M_x - 1} I_m(k)$$

(Compute spectrum width power ratio adjustment)

If $P_m > N_m Noise$

$$P_{adj} = \frac{|R_{1m}|}{P_m - N_m Noise}$$

Else

$P_{adj} = 0$

End

(Compute signal power)

If $P_c > N_c Noise$

$S_c = P_c - N_c Noise$

Else

$S_c = 0$

End

(Compute short PRT autocorrelation at lag T_1)

$$R_1(n) = S_c \cdot P_{adj}^4 \exp\left[j2\arg\left(R_{1c}\right)\right]$$

(Compute long PRT autocorrelation at lag T_2)

$$R_2(n) = S_c \cdot P_{adj}^9 \exp\left[j3\arg\left(R_{1c}\right)\right]$$

(Adjust signal power to include noise)

$P(n) = S_c + Noise$

(Note that the outputs of SACHI are $P(n)$, $R_1(n)$ and $R_2(n)$)

6) DC Removal Clutter Filtering

The DC Removal clutter filtering algorithm removes the mean (DC) component of V in those locations where the site-dependent clutter filter bypass map B indicates the need for clutter beyond the maximum ambiguous range corresponding to T_1 (r_{a1}). V_m is the DC component of V computed using only long-PRT sweeps beyond r_{a1}.

(Calculate the mean of the odd pulses)

$$V_m = \frac{2}{M}\sum_{m=0}^{M_p-1} V(n, 2m+1)$$

(Subtract mean from odd pulses. Range gates from even pulses don't exist beyond r_{a1})

For $0 \le m < M_p$

$V_F(2m) = 0$

$V_F(2m+1) = V(n, 2m+1) - V_m$

End

7) No Clutter Filtering

For $0 \le m < M$

$V_F(m) = V(n, m)$

End

8) Power and correlation computations for each PRT

If $n < N_1$

(Compute power from even pulses, if available)

$$P_1 = \frac{2}{M}\sum_{m=0}^{M/2-1} \left|V_F(2m)\right|^2$$

End

If $n < N_2$

(Compute power from odd pulses, if available)

$$P_2 = \frac{2}{M}\sum_{m=0}^{M/2-1} \left|V_F(2m+1)\right|^2$$

End

If $n < N_1$

(Compute lag-1correlations from all pulses, if available)

$$R_1(n) = \frac{2}{M}\sum_{m=0}^{M/2-1} V_F^*(2m)V_F(2m+1)$$

$$R_2(n) = \frac{2}{M-2}\sum_{m=0}^{(M-2)/2-1} V_F^*(2m+1)V_F(2m+2)$$

End

9) Combined power computation

To use as much information as possible, data are extracted from the two power arrays with different rules for each of the three segments depicted in Figure 1. For segment I, data are extracted only from P_1, since P_2 may be contaminated on those range bins with overlaid powers. An average of P_1 and P_2 is extracted for segment II, given that both power vectors are "clean" there. Finally, segment III data are obtained from P_2. In algorithmic form:

If $n < N_2 - N_1$
(Segment I)
$P(n) = P_1$
ElseIf $n < N_1$
(Segment II)
$P(n) = \frac{1}{2}(P_1 + P_2)$
Else
(Segment III)
$P(n) = P_2$
End

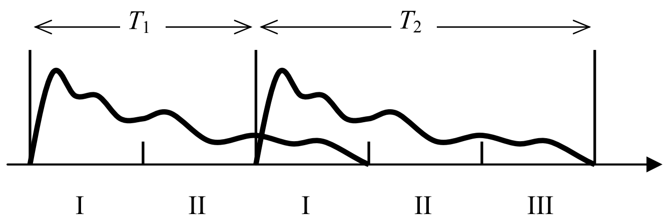

Fig. 1. Signal powers in the staggered PRT algorithm. Roman numerals indicate segment numbers.

10) Strong point clutter canceling

Processing is as in the current system. Strong-point clutter canceling is applied to P, R_1 and R_2 based on radial power continuity in P. For the remainder of the algorithm it is assumed that the outputs of this step are P, R_1 and R_2.

11) Signal power computation

If $P(n) < Noise$
$S = 0$
Else
$S = P(n) - Noise$
End

12) Reflectivity computation

(Range in km)
$R = n\Delta R + \Delta R/2$

(Reflectivity in dBZ. $\log_{10}$ *is the base-10 logarithm)*
If $S > 0$
$Z(n) = 10\log_{10}(S) + dBZ0 + R\ ATMOS + 20\log_{10}(R) - 10\log_{10}(Noise)$,
Else
$Z(n)$ should be set to the smallest possible reflectivity value
End

13) Velocity computation

If $n < N_1$

(Compute Doppler velocities for each PRT using the corresponding correlation estimates)

$$v_1 = -\frac{\lambda}{4\pi T_1}\arg\left[R_1(n)\right]$$

$$v_2 = -\frac{\lambda}{4\pi T_2}\arg\left[R_2(n)\right]$$

(Compute extended Nyquist velocity)

$$v_a = \frac{\lambda}{2T_1}$$

(Dealias velocity using pre-computed rules)

$$l = \arg\min_k \left|v_1 - v_2 - VDA_c(k)v_a\right|$$

$$v(n) = v_1 + 2v_a \ VDA_p(l)$$

(Prevent dealiased velocities outside of the extended Nyquist co-interval)

If $v(n) > v_a$

$v(n) = v(n) - 2v_a$

End

If $v(n) < -v_a$

$v(n) = v(n) + 2v_a$

End

Else

(This value is irrelevant)

$v(n) = 0$

End

14) Spectrum width computation

The spectrum width estimator corresponds to the algorithm implemented in the legacy WSR-88D signal processor.

If $n < N_1$

If $S = 0$ OR $\left|R_1(n)\right| = 0$

(Insert spectrum width of white noise)

$$w(n) = \frac{\lambda}{4\sqrt{3}T_1}$$

ElseIf $S < \left|R_1(n)\right|$

(Insert spectrum width of a constant)

$w(n) = 0$

Else

(Spectrum width computation. ln *is the natural log)*

$$w(n) = \frac{\lambda}{2\sqrt{2}\pi T_1}\sqrt{\ln\left(\frac{S}{\left|R_1(n)\right|}\right)}$$

End

Else

(This value is irrelevant)

$w(n) = 0$

End

15) Determination of significant returns for reflectivity

The non-significant return indicator array (NS_Z) is a binary array where 0 indicates “significant” and 1 indicates “non-significant”

If $S < Noise \cdot 10^{0.1T_Z}$
$NS_Z(n) = 1$
Else
$NS_Z(n) = 0$
End

16) Determination of significant returns for velocity

The non-significant return indicator array (NS_V) is a binary array where 0 indicates “significant” and 1 indicates “non-significant”

If $S < Noise \cdot 10^{0.1T_V}$
$NS_V(n) = 1$
Else
$NS_V(n) = 0$
End

17) Determination of significant returns for spectrum width

The non-significant return indicator array (NS_W) is a binary array where 0 indicates “significant” and 1 indicates “non-significant”

If $S < Noise \cdot 10^{0.1T_W}$
$NS_W(n) = 1$
Else
$NS_W(n) = 0$
End

18) Determination of overlaid returns for velocity and spectrum width

Censoring of velocity and spectrum width data is only necessary in segment I. This is done by analyzing P in segment I (P_1) and P in segment III (P_2) (see Fig. 1). The idea is to determine whether second trip signals mask first trip signals in segment I of P_2. While such overlaid echoes appear in every other pulse and do not bias velocity estimates at those range locations, overlaid powers act as noise. Therefore, when second trip powers in segment I of P_2 are above a preset fraction of their first trip counterparts, the corresponding velocity and spectrum width estimates exhibit very large errors and must be censored. The overlaid indicator arrays (OV_V *and* OV_W) are binary arrays where 0 indicates "not overlaid" and 1 indicates "overlaid".

If $n < N_2 - N_1$
 (Segment I: Range gates that may or may not have overlaid echoes)
 (Check power ratio first)
 If $P(n) > P(n + N_1)\ 10^{0.1T_O}$
  $OV_V(n) = 0$
  $OV_W(n) = 0$
 Else
  (Power ratio not met, but consider non-significant returns as non-existent)
  If $NS_V(n + N_1) = 1$
   $OV_V(n) = 0$
  Else
   $OV_V(n) = 1$
  End
  If $NS_W(n + N_1) = 1$
   $OV_W(n) = 0$
  Else
   $OV_W(n) = 1$
  End
 End
ElseIf $n < N_1$
 (Segment II: Range gates that, based on the assumptions, never have overlaid echoes)
 $OV_V(n) = 0$
 $OV_W(n) = 0$
Else
 (Segment III: Range gates that are always unrecoverable)
 $OV_V(n) = 1$
 $OV_W(n) = 1$
End

(Note that when processing the overlaid and significant return flags, the overlaid flags take a lower priority. That is, if a range bin is tagged as non significant and also as overlaid, the overlaid indication is ignored and the gate is treated as a non-significant return only; e.g., painted black as opposed to purple)