

A RADIO BURSTS DETECTION AND DEDISPERSION METHOD BASED ON HOUGH TRANSFORM

SHIFAN ZUO^{1,2}

¹NAOC

²UCAS

ABSTRACT

We present a very simple and fast method for radio bursts detection and incoherence dedispersion based on the Hough transform, which can be used for both batch and online processing. Different from other methods, we directly detect the f^{-2} curve in the observed time stream data. By using the Hough transform, we show that a point in the observing data maps to a straight line in the Hough transform parameter space, and points on the same f^{-2} curve map to a bundle of lines all cross a same point determined by the parameters of the curve, thus the f^{-2} curve will be transformed as a single high peak in the parameter space. By detection of the peak and its position in the parameter space, we could detect the occurrence of a radio burst and determine its dispersion measure (DM). Our method has a complexity of $\max(N_t, N_f)N_d$ where N_f , N_t and N_d are the numbers of frequency bins, time bins, and dispersion measure bins, respectively. This is much lower than any other existed radio bursts detection and dedispersion method. **The method can be used for both batch and online processing, but it may be more appropriate for the latter due to its low computation complexity and fast speed.** **will revise abstract in next revision**

Keywords: radio burst, Hough transform, dedispersion

1. INTRODUCTION

Astronomical radio pulses are dispersed while traveling through the interstellar medium (ISM) or intergalactic medium (IGM) plasma. At a lower frequency, the wave travels at a lower speed and arrives at a later time. This dispersion of the arrival time significantly decrease the pulse amplitude at a fixed observation time. In order to improve the detection sensitivity, dedispersion of the signal is required to compensate for the time delay induced by dispersion. While a number of sophisticated dedispersion and detection algorithms have been developed over the years for pulsar search and observation, the computation is demanding as it often needs to be done in nearly real time. For the recently discovered fast radio bursts (FRBs), which are bright millisecond radio pulses with unknown origin and mostly non-repeating, this is especially so. Up to now only about 20 FRBs have been discovered, though the inferred rate is fairly high (Lorimer et al. 2007; Thornton et al. 2013; Petroff et al. 2015a; CHIME Scientific Collaboration et al. 2017). The demanding computation for the dedispersion is a factor in the difficulty for discovering more FRBs.

The received signal can be dedispersed by applying frequency dependent time delays to the signal prior to integration, but the difficulty is that usually we do not know the amount of dispersion, so we have to make a large number of trials of different dispersions. This brute force dedispersion procedure requires expensive computations, of complexity $O(N_t N_f N_d)$, which limits its use in many cases. To speed up the dedispersion process, many algorithms are developed, for example, the tree dedispersion algorithm of complexity $O(N_t N_f \log(N_f))$ (Taylor 1974), the Fast Dispersion Measure Transform (FDMT) algorithm of complexity $O(2N_t N_f + N_t N_d \log_2(N_f))$ (Zackay & Ofek 2014), etc. These algorithms, though faster than the brute force method, still have a quite large computation complexity.

Obviously, the optimal dedispersion is the one that maximizes the signal-to-noise ratio of the pulse, which can only be fulfilled by integrating the flux exactly along the dispersion curve in the time-frequency domain. Mathematically, detection of a curve by integrating along the curve can be achieved by a family of transformations, for example, the mostly used Radon transform (Radon 1917) and Hough transform (Hough 1962). The Radon transform of a D -dimensional image $I(\mathbf{x})$ maps a N -dimensional shape $c(\mathbf{p})$ in the image to its parameter space by its integral projection, formulated as

$$\mathcal{R}_{c(\mathbf{p})}\{I\}(\mathbf{p}) = \int_{\mathbf{x} \in c(\mathbf{p})} I(\mathbf{x}) d\mathbf{x} = \int_{\mathbb{R}^D} I(\mathbf{x}) \delta(\mathcal{C}(\mathbf{x}; \mathbf{p})) d\mathbf{x},$$

where \mathbf{p} is a vector of parameters describing the shape, $\mathcal{C}(\mathbf{x}; \mathbf{p})$ is a set of constraint functions that together define the shape, and $\delta(\cdot)$ denotes the Dirac delta function in the above, or the Kronecker delta in the discrete case. The number of constraint functions depends on the dimensionality of the shape: $D - N$ constraints are necessary to describe a N -dimensional shape. For a point that lies on the shape, all the constraint functions evaluate to zero: $\mathcal{C}_i(\mathbf{x}; \mathbf{p}) = 0$ for all i .

The Hough transform is closely related to the Radon transform (van Ginkel & Hendriks 2004), though in its original formulation it is inherently discrete, while the Radon transform is continuous. It was originally designed to detect straight lines in binary images, but it can be extended to detect more general shapes in grey-value images, we describe it in this generalized form. To detect the shape, we set up an N -dimensional accumulator array $A(\mathbf{p})$, each dimension of it corresponding to one of the parameters of the shape to be searched. Each element of this array contains the number of votes for the presence of a shape with the parameters corresponding to that element. The votes are obtained as follows: for each point \mathbf{x}_i with grey value $g_i = I(\mathbf{x}_i)$ in the input image $I(\mathbf{x})$, if the shape passes through it, the vote for this shape parameter is increased by g_i , i.e., let

$$A(\mathbf{p}) \leftarrow A(\mathbf{p}) + g_i \delta(\mathcal{C}(\mathbf{x}_i; \mathbf{p})).$$

If a shape with parameter \mathbf{p} is present in the image, all of the pixels that are part of it will vote for it, yielding a large peak in the accumulator array. The shape detection problem in the image space is transformed to a peak finding problem in the parameter space.

As we usually do not know the dispersion measure, the whole parameter space (in practice a range of dispersion measures) need to be explored. Using the fact that most points in the data are background noise, we could truncate the data according to an appropriate threshold, this will throw away most of the noise (and potentially some pulse signal) below the threshold, thus make the map sparse and greatly reduce the computation of the Hough transform. For this reason, we choose to use the Hough transform for the dedispersion.

The use of Hough transform for radio transients detection and dedispersion was investigated in Fridman (2010), where the author showed that for a sufficiently small bandwidth a transient's pulse track on the time-frequency plane could be approximated by a straight line, and Hough transform could be used to detect this straight line on the time-frequency plane. Before the detection, a threshold given by 1σ above the mean of the data's value was applied to the data array to convert the data to a corresponding binary image, and this

binary image is then Hough transformed. The method was demonstrated with the application of Hough transform to the pulsar B0329+54 data observed by LOFAR in about 10 MHz bandwidth.

In this paper, we present a very simple and fast incoherence dedispersion and burst detection method based on the Hough transform, which has a much lower computational complexity than any of the existing radio pulse detection and dedispersion methods, and can be used for both batch and online processing. Different from Fridman (2010), our method does not depend on the straight line approximation, but detect directly the f^{-2} pulse track curve on the time-frequency plane, hence not limited in the usable bandwidth. We will also truncate the data below a threshold before the Hough transform operation, but use more robust statistical quantities based on the median and median absolute deviation (MAD) to determine the threshold value, which are more reliable and less affected by outliers and strong pulse signals presented in the data. Another difference is that we apply the Hough transform to the truncated gray-value image instead of the binary image, we will show that this can suppress noise and improve the signal-to-noise ration of the signal's peak in the transformed parameter space. Below, we present our algorithm in Sec.2, then we test it with mock and real data in Sec.3.1, and finally concludes in Sec.4.

2. ALGORITHM

We consider the Hough transformation algorithm for dedispersion and pulse search. This is a incoherent dedispersion method, so our input is a time stream of spectral data which is the short time integral of the flux, either from a single dish telescope, or from the synthesized beam of a radio telescope array. Before applying the Hough transform (2.1), the data is first pre-processed by truncation (2.2). We also discuss effects of outliers and intermittent signal (2.3) and computational issues (2.4) in this section.

2.1. Hough Transform

The dispersion delay of the pulse arrival time at a frequency f_1 relative to f_2 is given by

$$\Delta t = t_1 - t_2 = d(f_1^{-2} - f_2^{-2}). \quad (1)$$

where t_i is the arrival time of signal at frequency f_i in units of ms, $d \equiv 4.15 \times \text{DM}$, DM is the dispersion measure in units pc cm $^{-3}$, f_i is frequencies measured in GHz. Equation 1 is a differential relation between the arrival time and the measurement frequency, we can get the function of the curve by integration as

$$t = d f^{-2} + t_0, \quad (2)$$

where t_0 is a integration constant, represent the time offset of the curve. Each dispersed pulse signal is a f^{-2} curve, which can be uniquely determined by the two parameters (d, t_0) .

For a data point (t_i, f_i) on the curve defined by Eq. (2), we have the relation of the two parameters as

$$t_0 = -f_i^{-2}d + t_i, \quad (3)$$

which we see is just a straight line with slope $-f_i^{-2}$ and interception t_i , so in this space a point on the curve defined in Eq. (2) in the observing data maps to a straight line, and points on this curve map to a bundle of lines, which all cross at the same point (t_0, d) . The problem of detecting a f^{-2} curve in the observing data is transformed to a peak detection problem, which is both easier and more robust. Furthermore, the presence of discontinuity and outliers have little effect on the peak detection in the parameter space, as long as there are enough identifiable points on the curve. Outliers, even ones in the form of a line, will not generate peaks as high as the one corresponding to the curve since they do not have the f^{-2} function form.

To take advantage of the Hough transform, we first subtract the mean of the background noise and then truncate the data to throw away values below an appropriately chosen threshold T , i.e., we only preserve points (t_i, f_i) with value $|I(t_i, f_i)| > T$, this makes the input image very sparse (we denote this image as I_m), only those remaining points in the input image need to be considered and Hough transformed to the parameter space, thus greatly reduce the computation.

To do the Hough transform, we initialize an all zero accumulator matrix $A(t_0, d)$. For each point (t_i, f_i) in I_m , we accumulate a straight line given by Equation 3 with strength $I(t_i, f_i)$ to the accumulator A , i.e.,

$$A \leftarrow A + I(t_i, f_i) \delta(f_i^{-2}d - t_i + t_0). \quad (4)$$

Shouldn't we accumulate I_m ? For any point (t_i, f_i) in I_m , $I_m(t_i, f_i) = I(t_i, f_i)$ This will take $O(N_d)$ operations. We see lines corresponding to points that are on the pulse curve Eq. (3) will all cross at the point (t_0, d) , generating a high peak at this point in A , with value about $N_s\mu$ where N_s is the number of points on the curve and μ is the mean value of these points.

We initialize a $N_{t_0} \times N_d$ accumulator $A(t_0, d)$, with DM range $[d_{\min}, d_{\max}]$. If the source dispersion value d_s is known a prior, as in the case of known pulsars, this range can be very narrow, otherwise a wide range should be chosen to cover possible dispersion for the searched signal. Once we have chosen the appropriate

range $[d_{\min}, d_{\max}]$, the range of t_0 is

$$t_{0,\min} = -d_{\max} f_{\min}^{-2} + t_{\min}, \quad (5)$$

$$t_{0,\max} = -d_{\min} f_{\max}^{-2} + t_{\max}, \quad (6)$$

and $N_{t_0} \approx N_d$. The attainable resolution of d is determined by the time and frequency resolution: from Equation 1, for neighboring frequency $d \approx \frac{1}{2} \Delta t f^3 / \Delta f$, so

$$\Delta d = \frac{1}{2} \Delta t \frac{3f^2 \Delta f}{\Delta f} = \frac{3}{2} f^2 \Delta t \sim f_{\min}^2 \Delta t. \quad (7)$$

2.2. Background Subtraction

To reduce the amount of computation, we pre-process the data by subtracting the mean and then apply a threshold to filter out most of the data before doing the Hough transformation. For a background noise with a Gaussian distribution $N(\mu, \sigma^2)$ (a good approximation for radio noise in a short period of time of order of seconds), the threshold can be set as $T = \tau\sigma$, which will remove $\sim 95\%$ and 99.7% of the data for $\tau = 2.0$ and 3.0 respectively. But in the truncation process, some pulse signal may also be thrown away, especially for low signal-to-noise ratio (SNR) data. The threshold T should be chosen to achieve optimal detect sensitivity.

If T is set appropriately such that the pulse is mostly preserved, and suppose the pulse has a maximum length in the image $I(t_i, f_i)$, i.e., it pass the two points (t_{\min}, f_{\min}) and (t_{\max}, f_{\max}) , then the number of points on the curve of the pulse is about $\max(N_t, N_f)$, where N_f and N_t are the numbers of time bins and frequency bins, respectively. Plus the remaining noise and maybe outliers, the number of points remain in the sparse I_m is of order $O(\max(N_t, N_f))$, for each point, we have $O(N_d)$ computation of the straight line in the parameter space, thus lead to a total computation complexity of order $O(\max(N_t, N_f)N_d)$. But how do we estimate the mean μ and the standard deviation σ of the noise? This may not be trivial in the presence of the pulse signal in the data if its SNR is high. For example, if the signal has a distribution $N(\mu_s, \sigma_s^2)$ and if $\mu_s \neq 0$, then the estimate of μ is $\hat{\mu} \approx \mu + \mu_s \max(N_t, N_f) / (N_t N_f)$ by simple average of the data, thus biased. Since the mean is subtracted, $\mu_s = 0$? Also, I don't quite understand what is the conclusion we draw from this? This is before the background subtraction. What I mean here is that, in the presence of high SNR signal or strong RFIs, the estimate of the background mean (and also standard deviation) could be biased, so more appropriate method is needed to get robust estimate of them.

If strong outliers are present in the data, the Gaussian model of noise may not be valid. We decide to use the more robust statistical quantities: the median and the median absolute deviation

(MAD), i.e., we set $\hat{\mu} = \text{median}(I)$, $\hat{\sigma} = \text{MAD}(I) = \text{median}(|I - \text{median}(I)|) / 0.6745$. Median and median absolute deviation as more robust statistical quantities of the mean and std in the presence of outliers are often used in literatures, see e.g., Hampel (1974); Rousseeuw & Croux (1993); Leys et al. (2013); Friedman (2008). But what is the basis of this choice? And would this increase the computation? There are some fast median finding algorithm with $O(n)$ computation time, for example, see <https://www.quora.com/What-is-the-fastest-way-to-find-the-median-of-an-unsorted-array> <https://www.quora.com/What-is-the-most-efficient-algorithm-for-finding-the-median> <https://cs.stackexchange.com/questions/1914/how-to-find-the-median-of-an-unsorted-array>.

After the truncation, there is still some residue noise left in the data, for random noise their positions are uniformly distributed in random. There is a small chance that some points of the noise, say $(t_{n1}, f_{n1}), (t_{n2}, f_{n2}), \dots, (t_{nN}, f_{nN})$ will lie on a curve $t = d_n f^{-2} + t_{0n}$ with parameters (d_n, t_{0n}) . After the Hough transform these points will generate a bundle of lines which all cross at point (d_n, t_{0n}) in the accumulator matrix, $A_n(t_{0n}, d_n) = \sum_{i=1}^N I(t_{ni}, f_{ni})$. Noting that the background mean has already been subtracted, the remaining noise has a zero mean truncated Gaussian distribution, so the expectation value of $\mathbb{E}(A_n) = 0$, because the positive and negative values of the noise will cancel out in the accumulation. This is a bonus from using the gray-value image, for if we convert this truncated data to the corresponding binary image as done in Friedman (2010), then it will have $I(t_{ni}, f_{ni}) = 1$ for $i \in 1, \dots, N$, so $A_n(t_{0n}, d_n) = N$, i.e., the peak due to the remaining noise would be much higher, thus hamper the detection of the pulse's peak.

2.3. Outliers and Intermittent Signal

Outliers (e.g. those produced by radio frequency interference, RFI) are often present in the data, which may be much stronger than the astronomical radio pulse signal. How do they affect the peak detection after the Hough transform? In most cases, the outliers would appear as vertical (short pulse in time) or horizontal (narrow frequency band) lines in the observing data $I(t, f)$. Such kinds of outliers will not have much influence on the detection of the real pulse signal, as such data points will be mapped into lines which will cross at $d = 0$ (no time delay) or $d = \infty$ (infinitely large dispersion), which would not contribute to the accumulator matrix in the reasonable range of $[d_{\min}, d_{\max}]$. It is not very likely that the shape of the outliers happens to appear as a f^{-2} curve with parameter (d, t_0) in the right range, though in rare coincidence such event could be produced, e.g.

as in the case of the so called "peryton" (Petroff et al. 2015b). A simple automated algorithm as discussed here may not be able to identify such cases, but hopefully the algorithm could filter out most outliers such that only a small number of pulse events remain and can be further investigated in detail, possibly with human intervention.

In the real data the pulse signal may also be intermittent, i.e. it is not continuous along the entire curve, but consists a few segments with gaps between them in the original data, or in the truncated data due to the truncation process. This may cause some problems for methods that depend on the continuity of the curve, but the Hough transform method described in this paper obvious has no such limitations, though it may loss some signal-to-noise ratio in the peak. As long as enough points remained on the curve, and their sum has a significantly higher value than the background level, they will be accumulated to a high peak and be detected.

I commented out the DM index part, the deviation from f^{-2} is perhaps due to noise, so I don't think we really need to discuss searching for trackers with $\alpha \neq -2$. OK, seems no transit sources has a DM index significantly deviating from -2.

2.4. Computational Issues

The Hough transform method could be applied to the original background mean subtracted time-frequency data without truncation, in this case, every point in the data is to be transformed, leading to a worst case computational complexity of $O(N_t N_f N_d)$, which is the same as that of the brute force dedispersion procedure. This is clearly not the optimal, because most of the data are just background noise, pulse signal, if present, only take a very small portion of the data. We could truncate the data by an appropriately selected threshold. After experimentation with the real data, we found that, in most cases, for a value of $\tau = 3.0$, pulse signal could be successfully detected. Some data with lower SNR may need a lower $\tau = 2.5$ or 2.0 . Very few pulses needs $\tau = 1.0$ for detection, see subsection 3.2 for some examples. For $\tau = 3.0$, we only need to compute the Hough transform for about 0.3% of the data, for the very low $\tau = 1.0$, we need to compute the Hough transform for about 32% of the data. Thus an appropriate truncation will save a lot of computations. To see the real effect of the truncation to the computation savings, we have tested several truncation thresholds for some real observing data, and compared their performance to the brute force method, two of the results (one for the FRB 010125 with data dimension of 96×500 , and the other for FRB 110220 with data dimension of 1024×500 , where the first dimension is the number of frequency channels, and the second is

the number of time samples) are shown in Figure 1. For these tests, we have implemented the brute force dedispersion and the Hough transform method in C programming language, and run each method 10 times by a single thread CPU (i.e., no parallel speedup) with the same range and resolution of dispersion measure and time for both methods, the time reported is the average time for these 10 runs. For the computation of the median (and also the MAD) in the Hough transform method, we have just simply implemented it by first sorting the array, which is not a very fast method for median computation, more effective methods exists, for example, the median of medians algorithm Blum et al. (1973), which finds an approximate median in just linear time only. If some of the faster median computing methods are used, the performance of the Hough transform method can be further improved. But even for this simple slow implementation, we see the computation time of the Hough transform method with a truncation threshold $\tau \geq 1.0$ is less than that of the brute force method, and the computation times decays exponentially with the increasing truncate threshold τ .

But this doesn't mean that higher truncation threshold is better, there is a tradeoff between the speed and detection sensitivity, which can be controled by the truncation threshold τ . Higher τ makes the detection faster but also increase the rate of miss detection or error detection, as we will see in subsection 3.2 some weak pulse can only be successfully detected under a very low truncation threshold $\tau = 1$. Maybe counter-intuitively, a too-low truncation threshold or no truncation at all may also hamper the pulse detection by our method. More on this will be discussed in subsection 3.1. In reality, the appropriate truncation threshold depends on the distribution property and relative strength of the pulse and the noise background, and also depends on the specific experiment requirement, for example, if low computation complexity and high speed is more desire, a higher τ can be chosen, but if lower miss detection is required, then a lower τ must be chosen.

The method presented above is assumed to be applied to a block of data. For online processing, we could apply the method to succession of partially overlapping blocks. Due to the truncation process, the detection accuracy of the DM may be lower than some other method though for its advantage of high speed, the method may be more appropriate for online processing, in which mode, once a high enough peak was detected in the accumulator, this block of the data can be triggered to be saved for further analysis.

The Hough transform method can be easily parallelized to speed up the computation. The Hough trans-

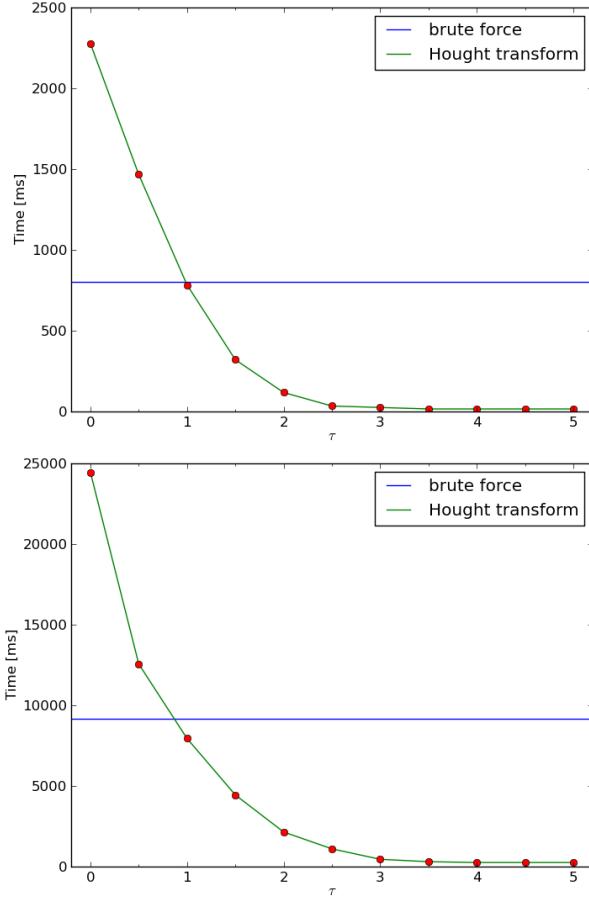


Figure 1. Comparison of the computation time for the Hough transform method with different truncation threshold τ and the brute force method. Top is for FRB 010125, bottom is for FRB 110220. See text for more information.

form is done for each data point in the truncated time-frequency data image, and their transformed result are accumulated in the accumulator array. To do the computation in parallel, we can partition the points in the truncated image to N parts and do Hough transform for these points in each part i independently and the results are accumulated to its own accumulator A_i , the pulse detection can be done by peak finding in the total accumulator A , which is the sum of all A_i s, $A = \sum_{i=1}^N A_i$.

Should add some computation time example in this subsection See Figure 1.

3. APPLICATION TEST

In this section we test how our algorithm works with data. We first use test it with simulation data (3.1), then with real pulsar and FRB data (3.2).

3.1. Simulation Data

We generate a mock sample of observing data as a superposition of Gaussian noise and a dispersed FRB sig-

nal. The noise is independent and identically distributed (iid) Gaussian with a distribution $N(0, \sigma_n^2)$, and the signal is also iid Gaussian with distribution $N(\mu, \sigma_s^2)$. The parameters for the simulated data are set as $\sigma_n = 1.0$, $\mu = 3.0$, $\sigma_s = 3.0$, and $DM = 1000 \text{ pc cm}^{-3}$. The observing frequency range is 400 – 800 MHz, with 2048 frequency bins. The simulated data $I(t, f)$ is shown in the top panel of Figure 2. We truncate the simulated data at a threshold T with $\tau = 3.0$ to throw away most of the noise. The truncated data I_m is a very sparse one, as shown in the bottom panel of Figure 2, only noise with an absolute value greater than the T and most of the signal are preserved on the masked data.

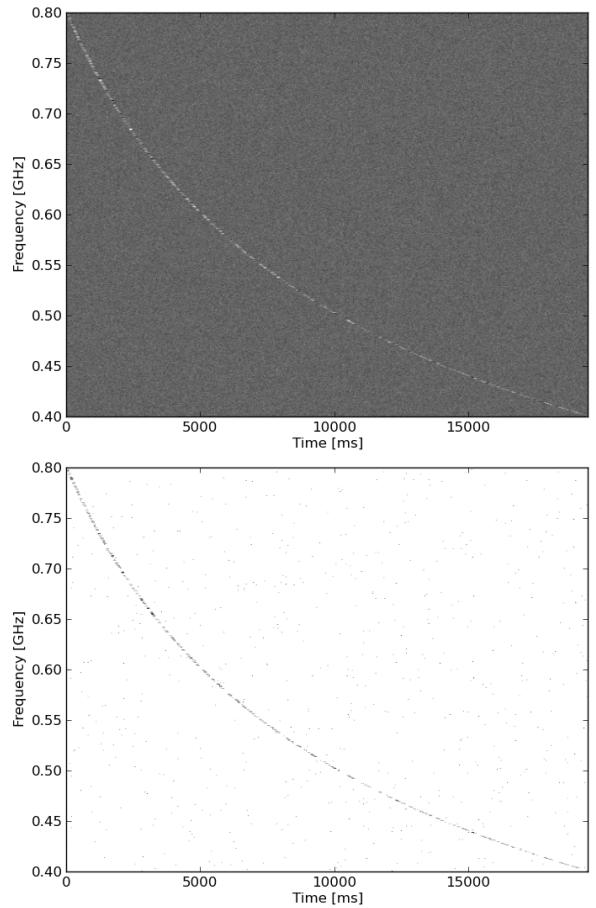


Figure 2. Top: simulated data; Bottom: truncated data (with $\tau = 3.0$) which masks out most of the noises. Both panels are hard to say, please increase point size. Maybe also should show truncated data with a few different thresholds

The Hough transform of the truncated data is shown in the range of $DM \in [800, 1200] \text{ pc cm}^{-3}$ in Figure 3, from which we see the peak is just at the right location $DM = 1000 \text{ pc cm}^{-3}$.

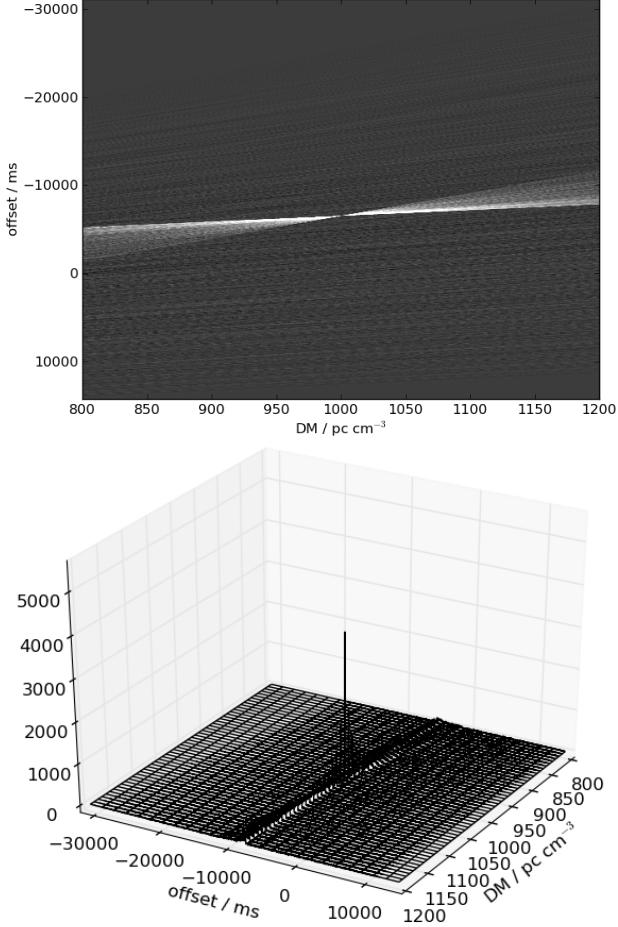


Figure 3. The Hough transform of the data shown in 2d (top) and 3d (bottom) plot.

In this subsection, should discuss performance for different SNR and different thresholds, needs add more material

We further explore effects of different truncation threshold τ and noise level σ_n to our method. To do this, we define a quantity Peak-to-Median Ratio (PMR) as $\text{PMR} = \max(A)/\text{median}(A_{\geq 0})$ as a measure of the relative strength of the highest peak to the noise background level in the accumulator A , where $A_{\geq 0} := \{a \in A \mid a \geq 0\}$. Higher PMR means the signal peak is more significant, thus easier to be successfully detected. To see the effects of different truncation thresholds, we fix all other parameters as set in the above simulation but with τ varies from 0.0 to 6.0 with step 0.5, do Hough transformation for each τ to get their corresponding accumulator A and calculate the PMRs, the result is shown in Figure 4, then we varies the noise standard deviation σ_n from 0.5 to 6.0 with step 0.5 but keep others fixed to see the effects of different noise level (or equivalently different signal-to-noise

ratio), the result is shown in Figure 5. From the results we see PMR is monotonically decreasing with the increased σ_n just as expected, but it has a non-trivial variation with the truncation threshold τ , and the result shows that for this specific simulation, an optimal truncation threshold is about $\tau \sim 3.5$, not the lowest value 0, i.e., with no truncation at all. That may seem a little counter-intuitive at first sight, but understandable, because the truncation process throws away much of the noise but relatively little of the pulse signal, so the PMR is possible to get larger for some appropriate τ . This shows that the truncation process is not just a way to lower the computational complexity and speed the pulse detection, but may also improve its detection.

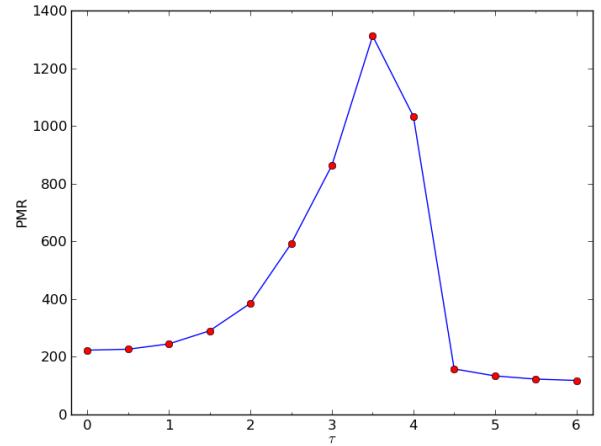


Figure 4. The variation of Peak-to-Median Raio (PMR) with different truncation threshold τ .

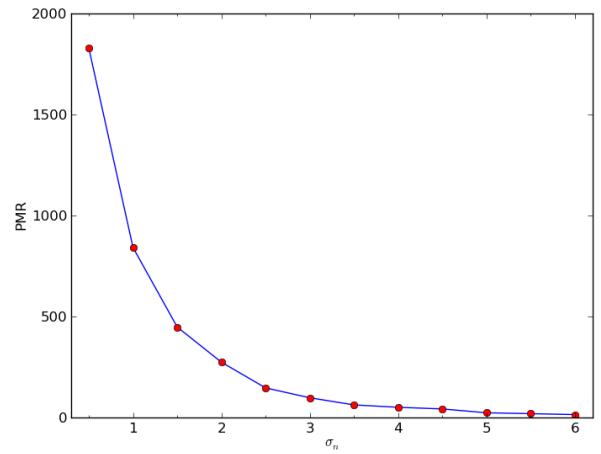


Figure 5. The variation of Peak-to-Median Raio (PMR) with different noise standard deviation σ_n .

3.2. Real Pulsar and FRB Data

We first apply our method to real observation data of three pulsars, i.e. B0329+54, B1929+10, and B2319+60 taken by the Green Bank Telescope (GBT). Any paper reference? Also, is there some standard bench mark data for such programs? Please check some papers on dedispersion algorithm. These three pulsars were not first observed by GBT, I only use the section of GBT's data observed in the project GBT14B-339 which has the pulsar signals in it. I didn't see standard benchmark data used, different papers used different pulsar observation data. Some may prefer to use Parkes pulsar data, because it has observed most of the pulsars and has a pulsar data archive, but I find its archive data seems difficult to use. We chose a threshold $\tau = 3.0$ for truncating the data, and used a 4000×2000 accumulator $A(t_0, d)$ with a DM range $[0, 100] \text{ pc cm}^{-3}$. From the data we extracted ~ 2 seconds data which contains the pulse signal, the data and their corresponding Hough transform results are given in the top and bottom rows in Figure 6 respectively, where we have also marked the detected peaks by red + in the transformed images.

In the data shown in the plot, there is one single pulse track for B2319+60 (right column) with DM = $94.591 \text{ pc cm}^{-3}$, while for B0329+54 (left column) there are three tracks with DM = $26.7641 \text{ pc cm}^{-3}$, and many tracks for B1929+10 (middle row) with DM = $3.18321 \text{ pc cm}^{-3}$. The DM values quoted here are taken from `psrcat` by Manchester et al. (2005). We see that each pulse track in the observed data has been transformed to a bundle of lines crossing at the same point. If there are more than one cross points corresponding to more than one pulse track in the observed data, they all have the same DM value, and the DM values of these detected peaks are very close to the values taken from `psrcat` by Manchester et al. (2005). Is there any RFI in the real data? It would be nice to illustrate that the RFI is automatically ignored after Hough transformation. If there is RFI, should also mark them and discuss here. You can see from the figure Figure 6, there are two strong narrow frequency band RFIs in the middle of each data, but they do not show in the corresponding Hough transformed images.

We also apply our method to several FRB event data, including eight FRBs observed by the Parkes telescope taken from the FRB Catalogue compiled by Petroff et al. (2016)¹, and one FRB event data (FRB 110523) observed by the GBT (Masui et al. 2015)². These FRBs are listed in Table 1.

For all these FRBs, we have initialized a 4000×2000 accumulator $A(t_0, d)$ with an appropriate DM range for each. What is the appropriate DM range? If we simply use, say, 10-2000, can we detect them? We should pretend not to know their DM when we do this simulation! They can be detected, the used range will have no influence to their detection as long as the DM is within this range and the resolution of the DM is enough. A wider range only increases the computation time. Also a wider range makes the Hough transformed image visually worse. We tried a few different truncation thresholds, then apply the Hough transform to search for them. We found that in most cases a threshold $\tau = 3.0$ is sufficient, but for a few weaker ones lower thresholds are required, specifically, $\tau = 2.5$ for FRB 120127, $\tau = 2.0$ for FRB 010724, and $\tau = 1.0$ for FRB 110626. Observing data of these FRBs are shown in Fig.7, and their corresponding Hough transformed result are shown in Fig.8, with the detected peaks marked by red + signs in the transformed image.

We see from Table 1 that FRB 120127 and FRB 110626 have very low peak intensity $S_{\text{peak,obs}}$ and integrated intensity F_{obs} relative to other ones, so lower thresholds are needed to detect them. The case of FRB 010724 is somewhat different, its pulse signal is fairly strong, but it could also only be detected with a lower threshold, say $\tau = 2.0$. We think this may be related to its non-uniform background noise, as can be seen clearly in Figure 7, the background noise has big difference between the left and right part of the pulse track, this affects the background mean subtraction, and further makes the pulse detection harder. Nevertheless, all FRBs can be successfully detected by using the Hough transform method, and the DM values obtained from the detected peaks are very close to the public values as listed in Table 1.

Note that in our processing, we have not make any special treatment for the RFIs or any other outliers before the Hough transform. For all data except for the FRB 110523, we have used the raw data, the only processing besides the truncation and Hough transform described above is rebinning in time direction to reduce the amount of data. For FRB 110523 the pre-processed data is available to us, which has been calibrated and RFI flagged. We can see from the data images, many of the data has RFIs or outliers in them, usually single frequency or narrow band RFIs, some are much stronger than the pulse signals. As we discussed in Section. 2.3, they should not have much effect on the detection based on the Hough transform, and the results indeed confirmed that they do not.

¹ <http://www.astronomy.swin.edu.au/pulsar/frbcat/>

² <http://www.cita.utoronto.ca/~kiyo/release/FRB110523/>

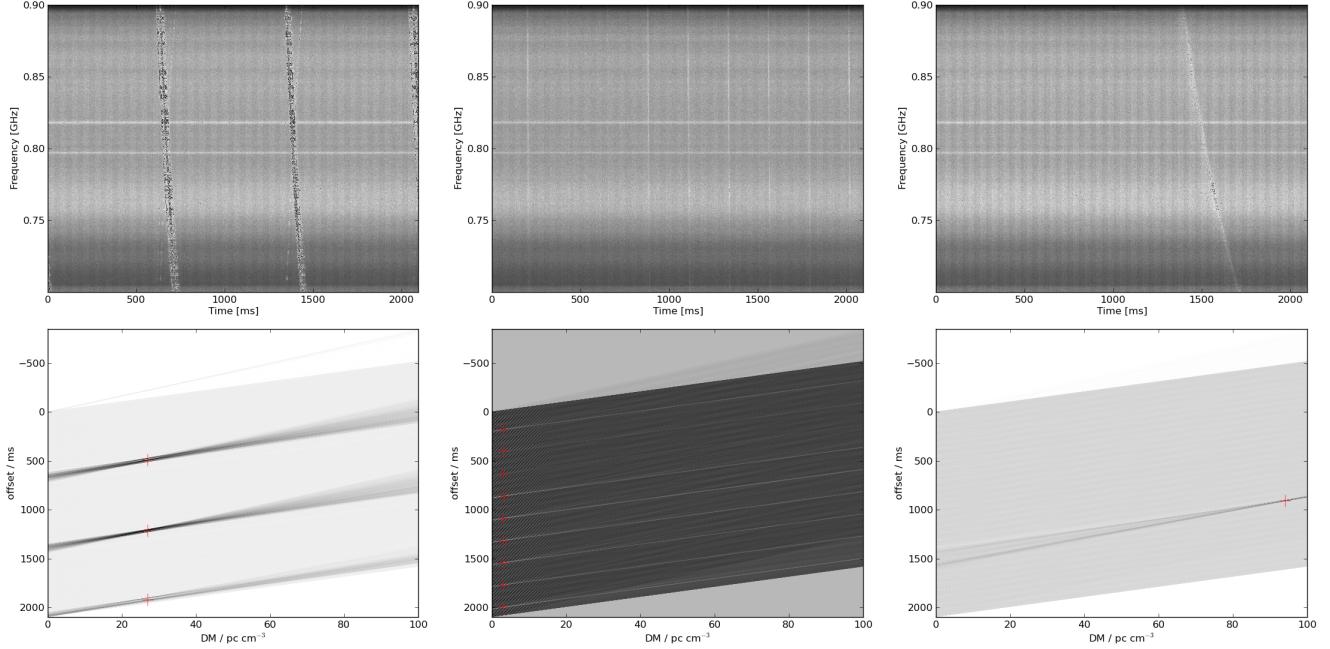


Figure 6. Observing data (top) and its Hough transform (down) of pulsars B0329+54, B1929+10, and B2319+60. The detected peak is shown as a red + in the transformed image. The symbol size should be increased, and if multiple tracks exist, they should all be marked.

Event	Telescope	DM [pc cm $^{-3}$]	S _{peak,obs} [Jy]	F _{obs} [Jy ms]	Ref
FRB 010125	parkes	790(3)	0.30	2.82	Burke-Spolaor & Bannister (2014)
FRB 010621	parkes	745(10)	0.41	2.87	Keane et al. (2011)
FRB 010724	parkes	375	$>30.00^{+10.00}_{-10.00}$	>150.00	Lorimer et al. (2007)
FRB 110220	parkes	944.38(5)	$1.30^{+0.00}_{-0.00}$	$7.28^{+0.13}_{-0.13}$	Thornton et al. (2013)
FRB 110523	GBT	623.30(6)	0.60	1.04	Masui et al. (2015)
FRB 110626	parkes	723.0(3)	0.40	0.56	Thornton et al. (2013)
FRB 110703	parkes	1103.6(7)	0.50	2.15	Thornton et al. (2013)
FRB 120127	parkes	553.3(3)	0.50	0.55	Thornton et al. (2013)
FRB 140514	parkes	562.7(6)	$0.47^{+0.11}_{-0.08}$	$1.32^{+2.34}_{-0.50}$	Petroff et al. (2015a)

Table 1. FRBs used in this paper and some of their parameters.

Also note that, in the above applications, we define a pulse peak is successfully detected in the accumulator A if an only if the central value of the peak is the largest of all elements of A and its position is consistent with the corresponding DM value, i.e., we have only used the information of the central value of the peak. This is clearly not the optimal way as we see in the figures, the peak corresponding to a pulse signal has a very specific shape, if we could also use its shape information, we could more robustly detect the peak even if its central value may be lower than some other values in the accumulator A due to random fluctuations or some other interferences, in this case, the truncation threshold τ may be set higher to still successfully detect the pulse, thus the detection

could be faster. But this is less related to the Hough transform method and is out scope of this paper, we do not discuss it further.

4. CONCLUSION

We have presented a simple and fast radio bursts detection and incoherence dedispersion method based on Hough transform, which has a low computational complexity of $O(\max(N_t, N_f)N_d)$, and can be used for both batch and online processing. We have shown its effectiveness by simulation and application to the real pulsar and FRB observing data. We hope the new method could be applied to future radio pulses (pulsars, FRBs,

etc.) detections and dedispersion tasks. **will add some more discussion in next revision.**

REFERENCES

- Blum, M., Floyd, R. W., Pratt, V., Rivest, R. L., & Tarjan, R. E. 1973, Journal of Computer and System Sciences, 7, 448 . <http://www.sciencedirect.com/science/article/pii/S002200073800339>
- Burke-Spolaor, S., & Bannister, K. W. 2014, ApJ, 792, 19
- CHIME Scientific Collaboration, Amiri, M., Bandura, K., et al. 2017, ArXiv e-prints, arXiv:1702.08040
- Fridman, P. A. 2008, AJ, 135, 1810
- . 2010, MNRAS, 409, 808
- Hampel, F. R. 1974, Journal of the American Statistical Association, 69, 383. <http://www.tandfonline.com/doi/abs/10.1080/01621459.1974.10482962>
- Hough, P. 1962, METHOD AND MEANS FOR RECOGNIZING COMPLEX PATTERNS, , , uS Patent 3,069,654
- Keane, E. F., Kramer, M., Lyne, A. G., Stappers, B. W., & McLaughlin, M. A. 2011, MNRAS, 415, 3065
- Leys, C., Ley, C., Klein, O., Bernard, P., & Licata, L. 2013, Journal of Experimental Social Psychology, 49, 764 . <http://www.sciencedirect.com/science/article/pii/S0022103113000668>
- Lorimer, D. R., Bailes, M., McLaughlin, M. A., Narkevic, D. J., & Crawford, F. 2007, Science, 318, 777. <http://science.scienmag.org/content/318/5851/777>
- Manchester, R. N., Hobbs, G. B., Teoh, A., & Hobbs, M. 2005, AJ, 129, 1993
- Masui, K., Lin, H.-H., Sievers, J., et al. 2015, Nature, 528, 523
- Petroff, E., Bailes, M., Barr, E. D., et al. 2015a, MNRAS, 447, 246
- Petroff, E., Keane, E. F., Barr, E. D., et al. 2015b, MNRAS, 451, 3933
- Petroff, E., Barr, E. D., Jameson, A., et al. 2016, PASA, 33, e045
- Radon, J. 1917, Berichte Schsische Akademie der Wissenschaften, Leipzig, Math.-Phys. Kl., 69, 262
- Rousseeuw, P. J., & Croux, C. 1993, Journal of the American Statistical Association, 88, doi:10.2307/2291267. <http://dx.doi.org/10.2307/2291267>
- Taylor, J. H. 1974, A&AS, 15, 367
- Thornton, D., Stappers, B., Bailes, M., et al. 2013, Science, 341, 53. <http://science.scienmag.org/content/341/6141/53>
- van Ginkel, M., & Hendriks, C. L. L. 2004
- Zackay, B., & Ofek, E. O. 2014, ArXiv e-prints, arXiv:1411.5373

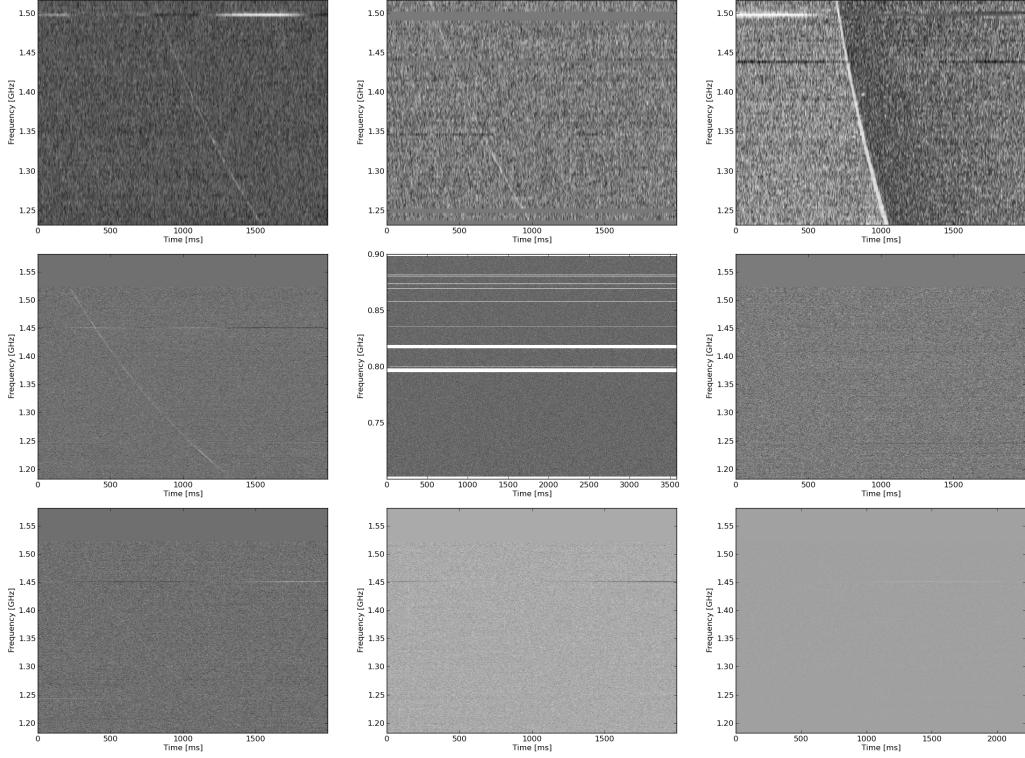


Figure 7. The observation data for FRB 010125, FRB 010621, FRB 010724(left to right), FRB 110220, FRB 110523, FRB 110626 (middle row, left to right), and FRB 110703, FRB 120127, FRB 140514(left to right).

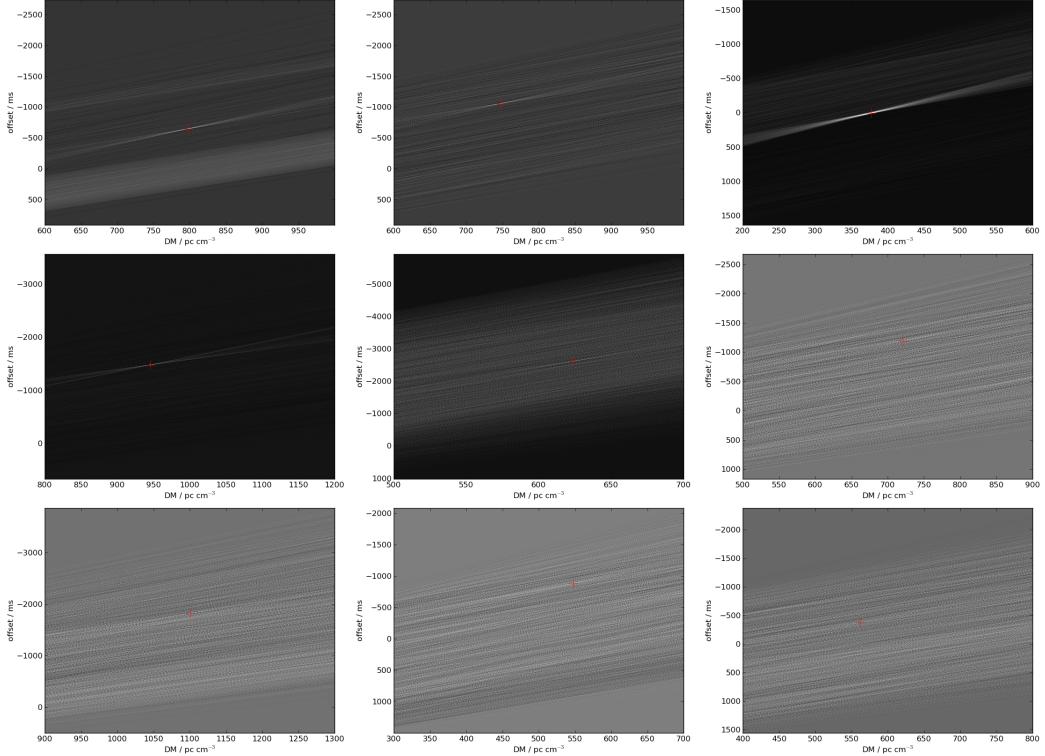


Figure 8. The Hough transform matrix A for the 9 FRBs shown in the figure above, in the same order. The detected peaks are marked by a red + in the transformed image. **Can we show the 3d plot instead?**