



## **A finite difference approach to despiking in-stationary velocity data - tested on a triple-lidar**

**Meyer Forsting, Alexander Raul; Troldborg, Niels**

*Published in:*  
Journal of Physics: Conference Series (Online)

*Link to article, DOI:*  
[10.1088/1742-6596/753/7/072017](https://doi.org/10.1088/1742-6596/753/7/072017)

*Publication date:*  
2016

*Document Version*  
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

*Citation (APA):*  
Meyer Forsting, A. R., & Troldborg, N. (2016). A finite difference approach to despiking in-stationary velocity data - tested on a triple-lidar. *Journal of Physics: Conference Series (Online)*, 753, Article 072017. <https://doi.org/10.1088/1742-6596/753/7/072017>

---

### **General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

## A finite difference approach to despiking in-stationary velocity data - tested on a triple-lidar

This content has been downloaded from IOPscience. Please scroll down to see the full text.

2016 J. Phys.: Conf. Ser. 753 072017

(<http://iopscience.iop.org/1742-6596/753/7/072017>)

View [the table of contents for this issue](#), or go to the [journal homepage](#) for more

Download details:

IP Address: 192.38.90.17

This content was downloaded on 20/10/2016 at 09:07

Please note that [terms and conditions apply](#).

# A finite difference approach to despiking in-stationary velocity data - tested on a triple-lidar

A R Meyer Forsting<sup>1</sup>, N Troldborg<sup>1</sup>

<sup>1</sup> DTU Wind Energy, Department of Wind Energy, Technical University of Denmark, Risø  
Campus, DK-4000 Roskilde, Denmark

E-mail: alrf@dtu.dk

**Abstract.** A novel despiking method is presented for in-stationary wind lidar velocity measurements. A finite difference approach yields the upper and lower bounds for a valid velocity reading. The sole input to the algorithm is the velocity series and optionally a far-field reference to the temporal variation in the velocity. The new algorithm is benchmarked against common despiking algorithms using a dataset acquired by three synchronised lidars in the upstream area of a full-scale wind turbine rotor and an artificially created space-time series with controlled spike contamination. By accounting for variations in space and time, this approach yields improvements in spike detection for in-stationary lidar measurements of about 25% over other more established stationary methods. Furthermore it proves to be robust even for large numbers of spikes.

## 1. Introduction

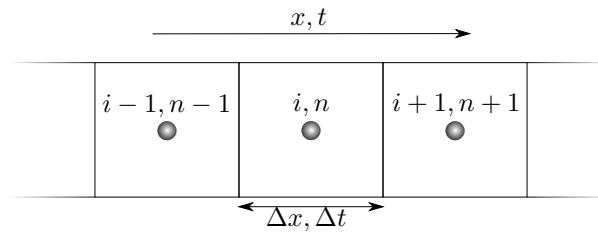
Lidars have been developing into important measurement instruments for the wind industry over the last decade, as they have proven to give accurate measurements and are more versatile than any other classic wind measurement system. As their prices are continuing to drop and their accuracy steadily improves, their market penetration can be expected to increase. Nevertheless the measurement principle of laser Doppler anemometry itself poses serious challenges regarding signal processing. Atmospheric conditions, hard targets, electric noise as well as perturbations in the optical system can heavily influence measurement quality. The various post-processing steps lying in-between the raw Doppler spectra and the final radial velocity usually reject spurious data. However, scanning lidars, like the short-range WindScanners [1], continuously change their focus location to measure velocities over an entire two-dimensional plane, thereby increasing the probability of spikes penetrating into the velocity signal. Undetected and processed these spikes can seriously contaminate the velocity signal; in the worst case making it futile. Avoiding any loss of accuracy and data misinterpretation therefore requires a robust despiking algorithm. Detecting and processing spikes is a widely discussed topic, which has not yet been satisfactorily resolved. The existing despiking methods' performance largely depends on the purpose of the processed data. In the related field of acoustic Doppler velocimetry it has been shown that the 3D phase space method, originally proposed by Goring and Nikora [2] and later modified by Wahl [3], is highly efficient [4] for turbulent flow data. Nevertheless the latter method is only valid for stationary measurements, whereas the scanning lidar measurements are moving spatially. Vickers and Mahrt [5] developed methods for detecting in-stationary flux measurement problems. However, their approach does not directly incorporate the influence of the spatial



movement. Therefore in this paper we present a finite-difference based despiking algorithm, that is derived from considering derivatives in both dimensions, space and time. It is tested and benchmarked against other common despiking methods for an extensive dataset acquired by three synchronised lidars in the upstream area of a wind turbine rotor and on an artificially created signal.

## 2. Finite difference despiking method

The in-stationary measurements represent a discretised form of the function  $u(\mathbf{x}, t)$  in space and time. Considering a point  $x_i$  in one-dimensional space and  $t^n$  in time, the value of the function can be expressed as  $u_i^n = u(x_i, t^n)$ . Defining  $\Delta\bullet$  as the interval of a quantity  $\bullet$ , the point can be expressed as  $x_i = i\Delta x$  and  $t^n = n\Delta t$ . Each consecutive in-stationary measurement point can be thought of as a step in time and space away from the previous point as shown in Figure 1.



**Figure 1.** Discrete measurement points of quantity  $u$ .

Using Taylor expansion around  $x_i^n$ , expressions for  $u_{i-1}^{n-1}$  and  $u_{i+1}^{n+1}$  can be established, where the partial derivative of  $u(x, t)$ ,  $\partial u / \partial \bullet$  is denoted by  $u_\bullet$ :

$$u(x_i + \Delta x, t^n + \Delta t) = u_{i+1}^{n+1} = u_i^n + \Delta x u_x|_i^n + \Delta t u_t|_i^n + \mathcal{O}(\Delta x^2, \Delta t^2) \quad (1)$$

$$= u_{i+1}^n + \Delta t u_t|_i^n + \mathcal{O}(\Delta x^2, \Delta t^2) \quad (2)$$

$$u(x_i - \Delta x, t^n - \Delta t) = u_{i-1}^{n-1} = u_i^n - \Delta x u_x|_i^n - \Delta t u_t|_i^n + \mathcal{O}(\Delta x^2, \Delta t^2) \quad (3)$$

$$= u_{i-1}^n - \Delta t u_t|_i^n + \mathcal{O}(\Delta x^2, \Delta t^2) \quad (4)$$

The first derivative in space can be estimated by a central difference.

$$u_x|_i^n \approx \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} \quad (5)$$

Rearranging equations 2 and 4 and inserting them into the central difference scheme in equation 5 the derivative becomes

$$u_x|_i^n \approx \frac{u_{i+1}^{n+1} - u_{i-1}^{n-1} - 2\Delta t u_t|_i^n}{2\Delta x} \quad (6)$$

The exact time derivative of  $u$  at  $x_i^n$  is unknown. Instead it can be approximated by measurements of  $u$  in homogeneous far-field flow, where  $u$  is solely a function of time. Denoting the maximum possible fluctuation in  $u$  with time as  $f$  and including wind field evolution, the time derivative becomes

$$u_t|_i^n \approx \pm f(t) \quad (7)$$

Note that the fluctuation can either be positive or negative, which stops the  $u_t$  terms from cancelling in the following operation. Inserting equation 7 into 6 and 1 or 3, as well as replacing the spatial derivative in the latter, a final approximation of  $u_i^n$  can be found

$$u_i^n = \frac{u_{i+1}^{n+1} + u_{i-1}^{n-1}}{2} \pm 2\Delta t f(t) \quad (8)$$

Note that  $u_i^n$  in fact becomes a mid-point approximation with an additional term including the fluctuation in  $u$  over time. If the measured velocity at  $x_i^n$  exceeds the bounds of the estimated value determined by equation 8, then it is marked as spike. There are many options for estimating  $f$ . However, for it to capture the temporal fluctuations of the wind field it should be directly linked to the time derivative of the velocity. We propose, based on other common despiking methods, to determine the bounds of an expected fluctuation by the mean of the signal plus  $\alpha$  standard deviations. Note that the magnitude of the derivative needs to be taken in this process. Depending on the length of the reference signal, it should be considered to adapt the period length over which these statistics are computed. Hence, in the benchmarking in section 5.2  $f(t)$  becomes

$$f(t) \approx \langle |u_{t,\infty}(t - t_s < t < t + t_s)| \rangle + \alpha \sigma_{|u_{t,\infty}|}(t - t_s < t < t + t_s) \quad (9)$$

Here  $t_s$  is the time it takes the measurement instrument to revisit the same point in space, equivalent to 15 s for the triple-lidar and  $\alpha$  is a constant. The latter is set to 3.0, such that 99.7% of all data are contained within its range, assuming  $u_{t,\infty}$  is normally distributed.

### 2.1. Replacing the free-stream reference

A free-stream reference might at times not exist, such that a replacement for  $u_{t,\infty}$  in equation 9 is needed. In fact it could be replaced by the measured signal itself, such that  $u_{t,\infty} = u_t$ . However, to avoid spikes from contaminating  $f(t)$  the median is used instead, setting  $\hat{u}_t = \text{median}(|u_t|)$ :

$$f(t) \approx \hat{u}_t(t - t_s < t < t + t_s) + \alpha \sigma_{\hat{u}_t}(t - t_s < t < t + t_s) \quad (10)$$

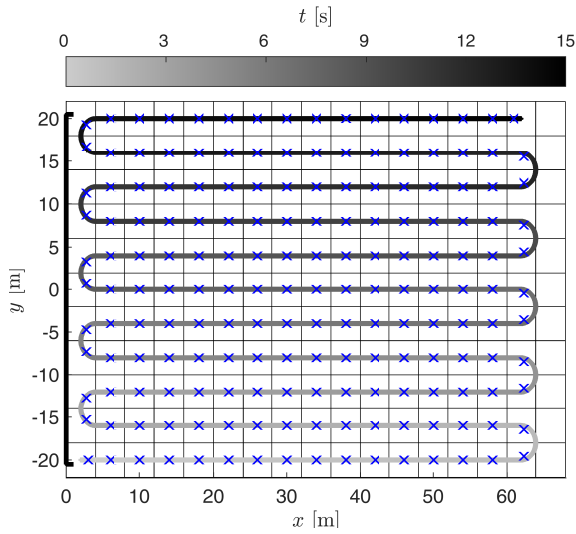
## 3. Despiking methods for benchmarking

The method is benchmarked against three other common despiking methods. A simple acceleration thresholding (AT), 3D phase space (PS) [2] method and one using the interquartile range (IQ). The AT method accentuates the high-frequency content of the signal by determining its time derivative  $\partial u / \partial t$  and dynamically determines a threshold over which a point is identified as spike. For this particular application certain ratios were tuned manually to ensure only local extrema were removed. Clearly this method relies heavily on careful calibration with respect to the measurement scenario and cannot easily be applied universally. Goring and Nikora [2] tried to reduce the parameters governing despiking methods by combining different approaches. They propose to compute the first and second derivatives of a signal and assume any spikes to lie outside an ellipsoid in phase-space. The ellipsoid's axes itself are determined by multiplying the standard deviations of the signal and its second derivative by the Universal threshold [6]. The simple IQ method defines outliers to fall 1.5 times the inter quartile range above or below the upper and lower quartiles, respectively. Finally the method developed by Vickers and Mahrt [5] (MW) employs moving averages and standard deviations on which it bases its threshold criterion. The threshold is first set to 3.5 standard deviations and then increased by 0.1 until no more spikes are detected. The temporal window over which the statistics are calculated is set to 180s, which was determined to be optimal.

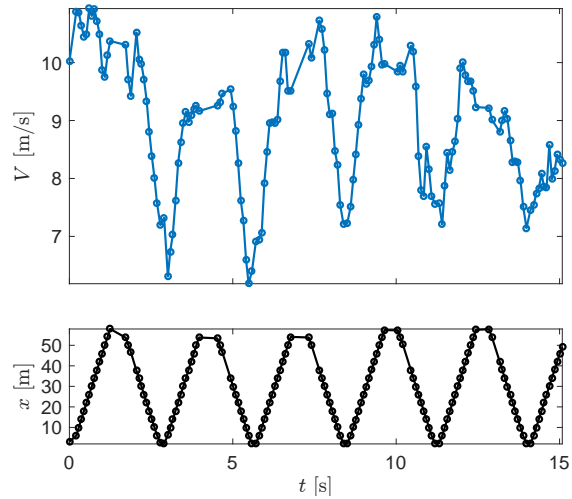
## 4. Test cases

### 4.1. Real

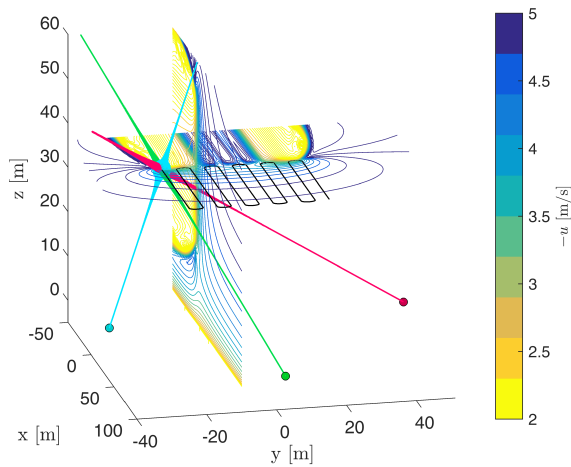
Both, real field measurements and an artificially created space-time series are used for testing the different methods. Ten hours of measurements by a triple-lidar system [7, 8], the so called short-range WindScanner [1], in the upstream area of a wind turbine test the robustness of the algorithms. The stall controlled Nordtank NTK 500 wind turbine with a 41 diameter is located at the DTU Risø campus, just off the Roskilde Fjord (N 55° 41' 04", E 012° 05' 48"). A met



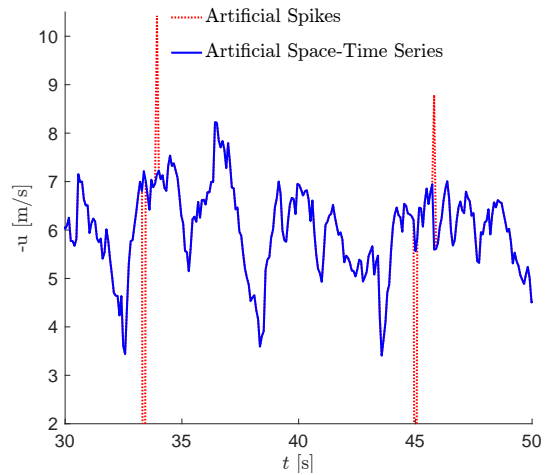
**Figure 2.** Measurement trajectory performed by a triple-lidar system in a horizontal plane at hub height in the upstream area of a wind turbine. The rotor centre is located at (0,0) and the cell-averaged data points are shown as x.



**Figure 3.** Resulting space-time series from triple-lidar measurements following the trajectory in figure 2.



**Figure 4.** CFD-RANS simulation of the flow upstream of the turbine with  $V_\infty = 5$  m/s sampled using a numerical triple-lidar. The lidar beams' thickness indicates the weighting of the velocities.



**Figure 5.** Computationally created triple-lidar space-time series.

most equipped with sonics, cups and vanes at several heights was located 92 m upstream of the turbine along the prevailing wind direction of  $283^\circ$ . The three lidars' focal points followed the trajectory shown in figure 2 simultaneously, such that they completed one horizontal scan at hub height (34 m) in 15 s. The continuous measurements are averaged over each grid cell shown in figure 2 to reduce the noise in the Doppler spectra. The noise filtering method by Angelou *et al.* [9] is subsequently applied to the spectra, determining the line-of-sight velocities. The three

lidar velocity signals allow to compute the velocity vectors in each cell. A similar campaign with a triple-lidar was performed by Simley *et al.* [10] giving a more complete description of the experimental method. In figure 3 an example of the resulting space-time series is shown. The data points closer to the rotor are affected by the rotor induction or blockage, explaining their lower velocities. The resulting troughs in the velocity are easily mistaken as spikes and thus provide a challenging test case, especially on days with large turbulence intensity. An overview of the dataset is given in the first six columns of table 1. It covers average wind speeds from 2.88 – 9.85 m/s and turbulence intensities from 9.31 – 18.2%.

**Table 1.** The first seven columns summarise the triple-lidar measurement periods comprising a total of 10 hours of data. The last four show the percentage of each dataset rejected by the respective despiking method. The meteorological conditions are given by: mean wind speed, turbulence intensity and wind direction. The length of its dataset is given in both, minutes and datapoints.

#	Date [d/m]	$\bar{V}_\infty$ [m/s]	TI [%]	WD [°]	$T$ [min]	Length [ $\times 10^3$ ]	AT [%]	PS [%]	IQ [%]	FD [%]
1	06/08	3.89	18.2	297	28.5	21.3	0.77	3.42	2.02	1.32
2	20/08	9.85	15.9	260	22.7	17.0	0.51	1.60	0.96	0.26
3	21/08	7.31	14.1	247	27.7	20.7	0.79	4.59	0.77	0.44
4	25/08	6.06	13.6	249	26.2	19.6	1.00	3.45	0.62	0.50
5	27/08	5.54	14.4	285	23.1	17.3	1.25	2.37	0.31	0.91
6	25/09	9.64	12.6	271	28.3	21.2	0.17	1.38	0.48	0.22
7	27/09	8.07	14.9	275	86.6	63.8	0.29	1.13	0.05	0.29
8	02/10	6.04	15.3	274	27.4	20.5	0.82	8.97	2.14	3.20
9	29/10	2.88	9.31	270	198	148	8.88	0.00	13.1	2.00
10	30/10	3.10	9.34	274	168	126	3.63	0.00	2.64	6.86

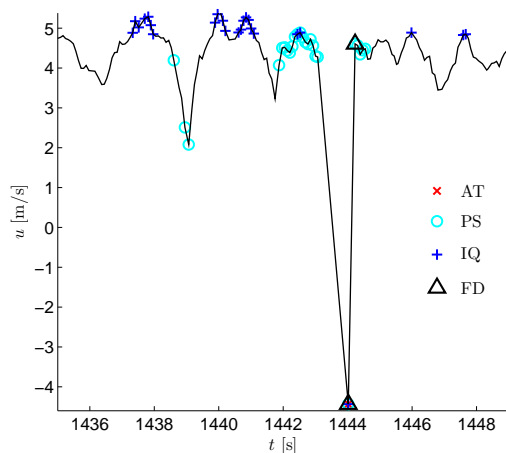
#### 4.2. Artificial

An artificially created space-time series can be contaminated with spikes at will, thus creating a controlled benchmarking environment, where the true solution is known. For this purpose numerical flow-fields are computed using Computational Fluid Dynamics (CFD). The computational method is described in detail in previous publications of the authors [11–13]. Steady-state simulations are performed for free-stream wind speeds between 5 – 12 m/s. Their solutions are subsequently sampled by a numerical triple-lidar, including the effect of volume averaging. This achieved by sampling the CFD flow-field along the entire length of the lidar beam and weighting the velocities according to a function. For the triple-lidar system it is equivalent to the one given by Simley *et al.* [10]. Note that the beam is assumed to have zero thickness. The numerical triple-lidar follows the experimental trajectory as shown in figure 4. As in the experiments all velocities falling into one cell (figure 2) are first averaged for each lidar and only afterwards combined to give the velocity vector. Finally for each free-stream velocity a numerically determined velocity vector is given for each cell. During the field experiment described in the previous section measurements from the met mast served as free-stream reference. To create the artificial space-time series the free-stream velocity data is taken from the #3 dataset, which is 28 minutes long. Interpolating for each measured free-stream velocity between the CFD solutions, taking into account the spatial variation of the triple-lidar measurements in time, the artificial space-time series with 22440 data points is created. An extract of which is presented in figure 5.

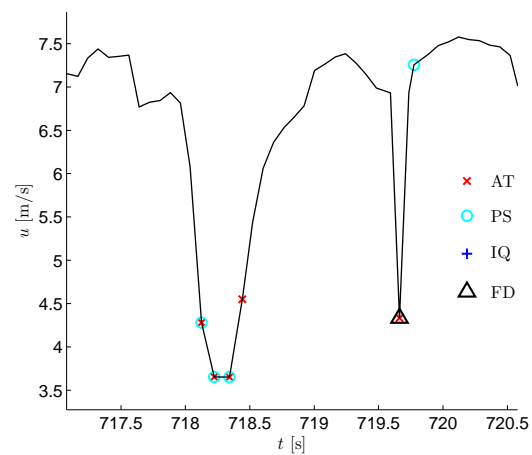
The spikes are injected into the artificial series by randomly selecting points, which are subsequently multiplied by certain factors. These are determined by a normal distribution centred around  $3.5 \cdot \sigma_u / \bar{u}$  with a standard deviation of  $\sigma_u / \bar{u}$ . Here  $u$  denotes the space-time series of the artificially created axial velocity component,  $\bar{u}$  its mean. To identify the sensitivity of each despiking method, the sign of the factors are either all positive, negative or randomly mixed. To create statistically viable results, 100 spiked time series are created on which the despiking methods are tested.

## 5. Results

### 5.1. Real space-time series



**Figure 6.** Extract from dataset #5 stream-wise velocity time series and spikes detected by: AT (x) = Acceleration Thresholding, PS (o) = 3D Phase Space, IQ (+) = Inter Quartile Range, FD ( $\Delta$ ) = Finite Difference algorithm.



**Figure 7.** Extract from dataset #8 stream-wise velocity time series and spikes detected by each despiking method.

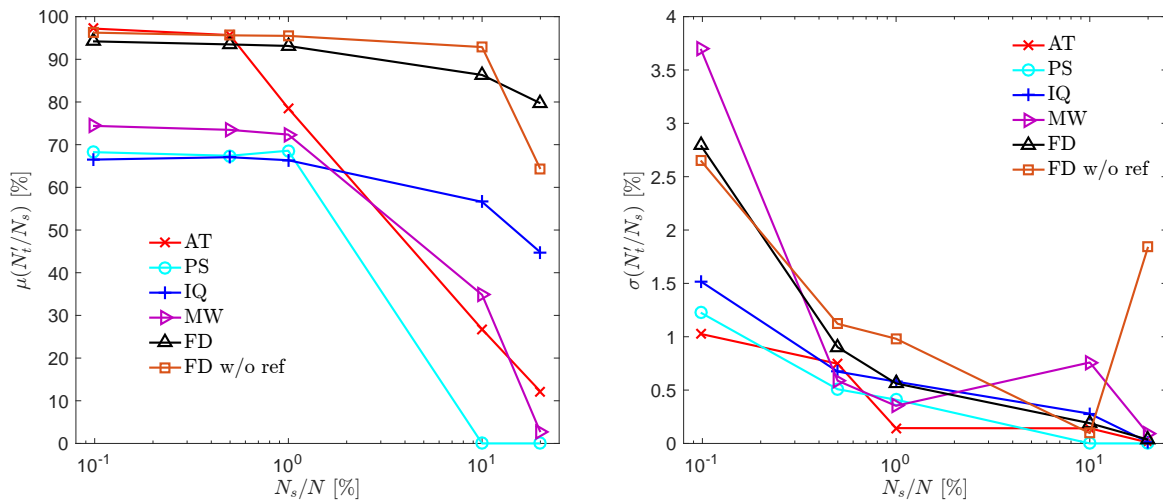
Demonstrations of the spikes detected by selected methods are shown in figures 6 and 7. They highlight the advantage of using the finite difference based algorithm for detecting spikes over the existing methods. All algorithms detect the obvious spike in figure 6 at  $t = 1444$  s, but both the IQ and the PS methods remove a vast amount of valid data. In figure 7 these methods miss the spike at  $t = 719.7$  s completely. Furthermore this figure enforces the difficulty for the algorithms to solely identify spikes, as both AT and PS remove valid readings. The last four columns of table 1 list the percentage of each dataset identified as spikes by the different methods. Clearly all four methods reject quite different amounts. Nevertheless except for a few datasets AT, FD and IQ seem to agree well, whereas PS rejects more points. It fails for the last two sets, but this might be related to the sheer length of the dataset over which some large-scale effects might have occurred. It should also be noted that the turbine was barely running during those days, as the wind speeds were on average below cut-in.

### 5.2. Artificial space-time series

In this section all despiking tools presented throughout the paper are compared by testing them on the 100 spiked time series. This includes two finite-difference methods with differing formulations of  $f$ , following equations 9 and 10. The one without a free-stream reference is



denoted "FD w/o ref". Firstly the four despiking methods are tested on the uncontaminated artificial 2-D series. Remarkably the new FD methods correctly identified none of the points as spikes. For the AT, PS, IQ and MW the points marked as spikes as percentage of the total are 0.70%, 0.79%, 0.23% and 0.46%, respectively. This is result can be deemed acceptable, especially with a turbulence intensity of 14.1 % that day.



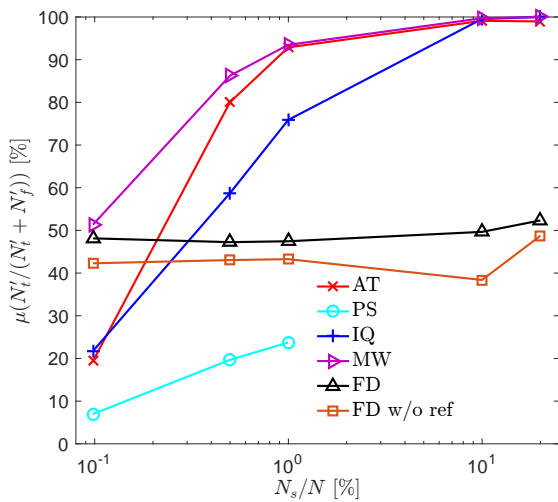
**Figure 8.** Mean detection rate of true spikes. **Figure 9.** Standard deviation of true spikes detected.

The behaviour of the tools is tested for different spike contamination rates. The latter is defined as the ratio between the number of artificial spikes injected over the total number of data points  $N_s/N$ . The ratios were 0.1, 0.5, 1, 10, 20%. In figure 8 the percentage of the injected spikes detected  $N'_t/N_s$  by each method is shown as a function of the contamination rate. Note that the sign of the spike factor is random. The quantities  $N'_{\{t,f\}}$  denote the number of true and false spikes marked by each method, respectively. The spike detection universally reduces with spike contamination. However, whereas the IQ and FD algorithms perform stably, all other despikers' performances deteriorate past a low contamination of 1%. Whilst the detection rate of AT tumbles towards zero, the PS algorithm fails to identify any true spikes any longer. The best performance for this measure has the FD method with free-stream reference, identifying on average 89% across all rates. However, the other FD method without a free-stream reference is actually outperforming it until the contamination rate passes 10%. That all methods except FD exhibit large performance drops above a certain contamination rate is related to the high-frequency content increasing with contamination. This ultimately obscures the existence of spikes, as the standard deviation and median of the signal is giving larger spike thresholds. FD on the other hand only relies on the free-stream standard variation and thus is not affected. Figure 9 presents the standard deviation of the true spikes found across all spiked series. It decreases rapidly with increasing contamination rate, except for FD w/o ref for the largest number of spikes. The larger standard deviation at low contamination rates is related to the sensitivity of the despiking tools to the position of the spikes. With fewer spikes this is more visible.

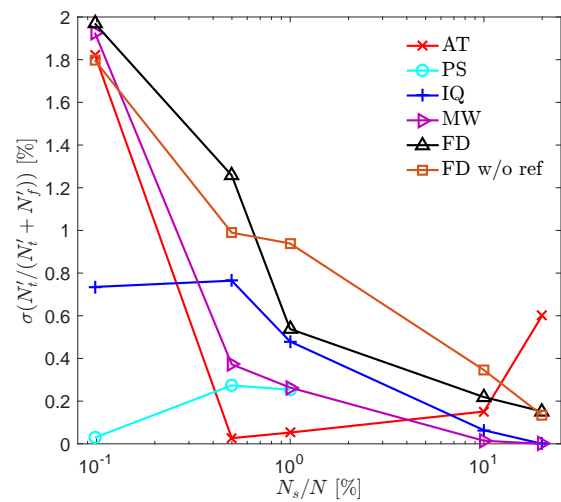
Another measure of performance is the ratio between the true spikes found and the total amount of points marked as spikes  $N'_t/(N'_t + N'_f)$ . This essentially measures whether the algorithm is rejecting the right data. The results are plotted in figure 10. Here the FD methods are the only ones showing constant performance, however they are much worse than the other

methods, except PS. However, the results can easily misinterpreted, as in fact AT, MW and IQ are almost not flagging any points as spike any longer beyond a contamination of 1%. The reason for the FD methods performing worse lies in the mid-point approximation that involves the neighbouring cells. Is a spike a neighbour, a valid point might be identified as spike. This is the case in figure 6. As before the standard deviation decreases with contamination as shown in figure 11. This occurs for similar reasons mentioned before.

Lastly, to test the sensitivity of the algorithms with respect to the sign of the spikes, solutions were obtained for solely positive and negative spike factors. Unsurprisingly all methods are showing some sensitivity, as negative spikes are harder to identify in-between the troughs in the velocity series, caused by the rotor induction. Similarly, it is challenging to identify spikes inside the trough itself. Nevertheless the FD methods perform well with the mean detection rate changing by maximally 2.1% (w/o ref) and 8.4%. AT follows with 11% and MW performs worst with 30%. The other methods fall in-between.



**Figure 10.** Mean of the ratio between true spikes and total points marked as spikes.



**Figure 11.** Standard deviation of the ratio between true spikes and total points marked as spikes.

## 6. Conclusion

The novel finite difference based despiking algorithm accounts for the in-stationary nature of scanning lidar velocity measurements. This allows it to outperform other methods, which have established themselves for despiking stationary velocity time series. Its mean detection rate lies above 70% for spike contamination rates between 0.1–20% and reaches up to 100%. Furthermore its performance is constantly high, whereas all other tested methods fail past a contamination of 1%. It does remove more data then necessary, however. An interesting and simple alternative for despiking in-stationary data seems to be the inter quartile range. It has not performed as well as the new algorithm, but still found about 60% of all spikes. Especially without any information on the free-stream velocity fluctuation, needed in the proposed method, the inter quartile range could be advantageous. In the future it would be interesting to apply the new algorithm to more datasets, thus testing its robustness.

## Acknowledgments

This work forms part of UniTTe ([www.unitte.dk](http://www.unitte.dk)), financed by The Innovation Fund Denmark (1305-00024B). Special thanks to Nikolas Angelou and Andrea Vignaroli from DTU Wind Energy for providing the lidar datasets and Michael Harris from ZephIR Lidar for the valuable input on commercial approaches to lidar signal processing.

## References

- [1] Mikkelsen T K 2015 Windscanner.dk, a new danish remote sensing-based infrastructure for wind energy and turbulence research: Design, establishment and operation 2009-2014 Tech. rep. DTU Wind Energy, Technical University of Denmark
- [2] Goring D G and Nikora V I 2002 *Journal of Hydraulic Engineering* **128** 117–126
- [3] Wahl T L 2007 *Journal of Hydraulic Engineering*
- [4] Mori N, Suzuki T and Kakuno S 2007 *Journal of Engineering Mechanics*
- [5] Vickers D and Mahrt L 1997 *J. of Atmos. Oceanic Tech.* **14** 512–526
- [6] Donoho D L and Johnstone I M 1994 *J. Atmos. Sci.* **155** 377–389
- [7] Angelou N and Sjöholm M 2015 *UniTTe WP3/MC1: Measuring the inflow towards a Nordtank 500kW turbine using three short-range WindScanners and one SpinnerLidar* DTU Wind Energy E (DTU Wind Energy)
- [8] Vignaroli A 2015 *UniTTe- Nordtank Measurement Campaign (Turbine and Met Masts)* DTU Wind Energy E (DTU Wind Energy)
- [9] Angelou N, Abari F F, Mann J, Mikkelsen T and Sjöholm M 2012 Challenges in noise removal from doppler spectra acquired by continuous-wave lidar *Proceedings of the 26th International Laser Radar Conference*
- [10] Simley E, Angelou N, Mikkelsen T, Sjöholm M, Mann J and Pao L Y 2016 *Journal of Renewable and Sustainable Energy* **8**
- [11] Meyer Forsting A R, Troldborg N and Gaunaa M 2016 *Wind Energy* ISSN 1099-1824 we.1991 URL <http://dx.doi.org/10.1002/we.1991>
- [12] Meyer Forsting A and Troldborg N 2015 *Journal of Physics: Conference Series (Online)* **625** ISSN 1742-6596
- [13] Troldborg N, Zahle F, Réthoré P E and Sørensen N 2013 *Wind Energy* 10.1002/we.1757