



Wind Simulation Using High-Frequency Velocity Component Measurements

Anton Gavrikov  and Ivan Gankevich  

Saint Petersburg State University, 7-9 Universitetskaya Emb.,
Saint Petersburg 199034, Russia
st047437@student.spbu.ru, i.gankevich@spbu.ru
<https://spbu.ru/>

Abstract. Wind simulation in the context of ships motions is used to estimate the effect of the wind on large containerhips, sailboats and yachts. Wind models are typically based on a sum of harmonics with random phases and different amplitudes. In this paper we propose to use autoregressive model to simulate the wind. This model is based on autocovariance function that can be estimated from the real-world data collected by anemometers. We have found none of the data that meets our resolution requirements, and decided to produce the dataset ourselves using three-axis anemometer. We built our own anemometer based on load cells, collected the data with the required resolution, verified the data using well-established statistical distributions, estimated autocovariance functions from the data and simulated the wind using autoregressive model. We have found that the load cell anemometer is capable of recording wind speed for statistical studies, but autoregressive model needs further calibration to reproduce the wind with the same statistical properties.

Keywords: Load cell · Strain gauge · Anemometer · Three-dimensional ACF · Wind velocity PDF · Autoregressive model · Turbulence

1 Introduction

Wind simulation in the context of ship motion simulation is the topic where multiple mathematical models are possible, and the choice of the model depends on the purpose of the model. In the course of the research where we apply autoregressive model to ocean wave simulation, we decided to investigate whether the same model can be used to simulate wind flow around ship hull.

Wind simulation is studied at different scales and the closest scale for a ship in the ocean is wind turbine. One of the model that is used to describe air flow around wind turbines [12] is similar to Longuet—Higgins model which

Supported by Council for grants of the President of the Russian Federation (grant no. MK-383.2020.9).

is typically used for ocean wave simulations. This model uses wind velocity frequency spectrum to determine coefficients and use them to generate wind velocity vector components.

Wind velocity distribution is described by Kaimal spectrum [10, 13]

$$S(f) = \frac{c_1 u_*^2 z / U(z)}{1 + c_2 (fz / U(z))^{5/3}}, \quad u_* = \frac{kU(z)}{\ln(z/z_0)}, \quad c_1 = 105, \quad c_2 = 33.$$

Here u_* is shear velocity, $k = 0.4$ is von Karman constant, z_0 is surface roughness coefficient, f is frequency, z is height above ground, $U(z)$ is mean speed at height z .

This spectrum is used to simulate each wind velocity vector component at a specified point in space. For each component the same spectrum is used, but the coefficients are different [12]. The spectrum describes wind velocity vector in the plane that is perpendicular to the mean wind direction vector and travels in the same direction with mean wind speed. Time series is generated as Fourier series, coefficients of which are determined from the spectrum, and phases are random variables [12]:

$$V(t) = \bar{V} + \sum_{j=1}^n (A_j \sin \omega_j t + B_j \cos \omega_j t),$$

$$A_j = \sqrt{\frac{1}{2} S_j \Delta \omega \sin \phi_j}, \quad B_j = \sqrt{\frac{1}{2} S_j \Delta \omega \cos \phi_j}.$$

Here S_j is spectrum value at frequency ω_j , ϕ_j is random variable which is uniformly distributed in $[0, 2\pi]$. The result is one-dimensional vector-valued time series, each element of which is velocity vector at a specified point in time and space.

In order to simulate wind velocity vector at multiple points in space, the authors use the function of coherency — the amount of correlation between wind speed at two points in space. This function has a form of an exponent and depends on frequency [13]:

$$\text{Coh}_{jk}(f) = \exp\left(-\frac{C \Delta r_{jk} f}{U(z)}\right),$$

where Δr_{jk} is the distance between i and j points and C is coherency decrement. Time series for each wind velocity vector component are generated independently and after that their spectra are modified in accordance with coherence function (the formulae are not presented here).

In order to simulate wind with autoregressive model it is easier to use autocovariance function instead of spectra and coherence function. We can obtain autocovariance function from spectra as inverse Fourier transform using Wiener—Khinchin theorem. The formula that we obtained this way using various computer algebra programmes is too complex, but can be approximated by a decaying exponent:

$$\gamma(t) = \sigma^2 \exp\left(-0.1 \frac{c_2^{3/5}}{c_1} t\right),$$

where σ^2 is process variance (area under the spectrum).

Unfortunately, this autocovariance function is one-dimensional and there is no easy way of obtaining three-dimensional analogue from the spectra. In order to solve this problem we looked into datasets of wind speed measurements available for the research. However, most of them contain either one or two wind velocity vector components (in a form of wind speed and direction), they are difficult to get and their resolution is very small (we found only one paper that deals with three components [15]). For our purposes we need resolution of at least one sample per second to simulate gusts and some form of turbulence. To summarise, our requirements for the dataset is to provide all three wind velocity vector components and has resolution of at least one sample per second.

We failed to find such datasets and continued our research into anemometers that can generate the required dataset. One of the anemometers that can record all three components is ultrasonic anemometer [1, 6, 14]. As commercial anemometers are too expensive for this research, we built our own version from the generally available electric components. However, this version failed to capture any meaningful data, and incidentally we decided to build an anemometer from load cells and strain gauges (which were originally intended for different research work). This anemometer is straightforward to construct, the electrical components are inexpensive, and it is easy to protect them from bad weather. This anemometer is able to record all three wind velocity vector components multiple times per second.

In this paper we describe how load cell anemometer is built, then we collect dataset of all three wind speed components with one-second resolution, verify this anemometer using commercial analogue, verify measurements from the dataset using well-established distributions for wind speed and direction, and estimate autocovariance functions for autoregressive model from our dataset. Finally, we present preliminary wind simulation results using autoregressive model.

2 Methods

2.1 Three-Axis Wind Velocity Measurements with Load Cell Anemometer

In order to generate wind velocity field using four-dimensional (one temporal and three spatial dimensions) autoregressive model, we need to use four-dimensional wind velocity autocovariance function. Using Wiener—Khinchin theorem it is easy to compute the function from the spectrum. Unfortunately, most of the existing wind velocity historical data contains only wind velocity magnitude and direction. We can use them to reconstruct x and y spectrum, but there is no way to get spectrum for z coordinate from this data. Also, resolution of historical data is too small for wind simulation for ship motions. To solve these problems, we decided to build our own three-axis anemometer that measures wind velocity for all three axes at one point in space multiple times per second.

To measure wind velocity we used resistive foil strain gauges mounted on the arms aligned perpendicular to the axes directions. Inside each arm we placed aluminium load cell with two strain gauges: one on the bending side and another one on the lateral side. Load cells use Wheatstone half-bridge to measure the resistance of the gauges and are connected to the circuit that measures the resistance and transmits it to the microcontroller in digital format. Microcontroller then records the value for each load cell and transmits all of them in textual form to the main computer. The main computer then adds a timestamp and saves it to the database. 3-D anemometer model is shown in Fig. 1.

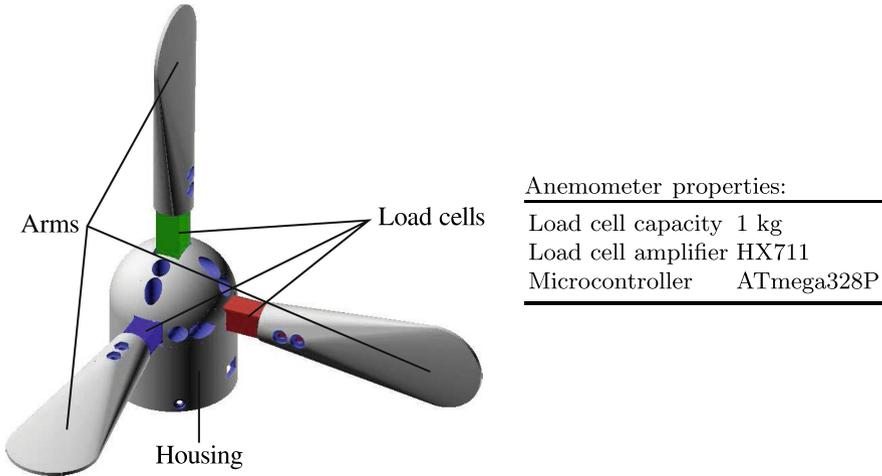


Fig. 1. Three-axis anemometer. Arms are inserted into the housing and fixed using bolts that go through the circular holes. Red, green, blue colours denote load cells for x , y , z axes respectively (Color figure online).

Each load cell faces the direction that is perpendicular to the directions of other load cells. When the wind blows in the direction of particular load cell, only this cell bends. When the wind blows in an arbitrary direction which is not perpendicular to any load cell faces, then all load cells bend, but the pressure force is smaller. Pressures of all load cells are recorded simultaneously, and we can use Bernoulli’s equation to compute wind velocity from them.

Bernoulli’s equation is written as

$$\rho \frac{v^2}{2} + \rho g z = p_0 - p. \tag{1}$$

Here v is wind velocity magnitude, g is gravitational acceleration, z is vertical coordinate, ρ is air density, p is pressure on the load cell and p_0 is atmospheric pressure. Pressure force \vec{F} acting on a load cell is written as

$$\vec{F} = pS\vec{n}, \tag{2}$$

where S is area of the side of the load cell on which the force is applied and \vec{n} is normal vector. Arms in which load cells reside have front and back side with much larger areas than the left and right side, therefore we neglect forces on them. Additionally, on the ground ρgz term vanishes.

For all load cells we have a system of three equations

$$\begin{cases} v_x^2 \propto F_x \\ v_y^2 \propto F_y \\ v_z^2 \propto F_z \end{cases} . \tag{3}$$

Hence $v_x = \alpha_x \sqrt{F_x}$, $v_y = \alpha_y \sqrt{F_y}$, $v_z = \alpha_z \sqrt{F_z}$ where $\alpha_{x,y,z}$ are constants of proportionality. Therefore, to obtain wind velocity we take *square root* of the value measured by the load cell and multiply it by some coefficient determined empirically during anemometer calibration.

2.2 Per-Axis Probability Distribution Function for Wind Velocity

Scalar wind velocity is described by Weibull distribution [9]. Weibull probability distribution function is written as

$$f(v; b, c) = bc(bv)^{c-1} \exp(-(bv)^c). \tag{4}$$

Here $v > 0$ is scalar wind velocity, $b > 0$ is scale parameter and $c > 0$ is shape parameter. This function is defined for positive wind velocity, since scalar wind velocity is a length of wind velocity vector $v = \sqrt{v_x^2 + v_y^2 + v_z^2}$. However, projection of wind velocity vector on x , y or z axis may be negative. Our solution to this problem is to use two Weibull distributions: one for positive and one for negative projection — but with different parameters.

$$f(v_x; b_1, c_1, b_2, c_2) = \begin{cases} b_1 c_1 (b_1 |v_x|)^{c_1-1} \exp(-(b_1 |v_x|)^{c_1}). & \text{if } v_x < 0 \\ b_2 c_2 (b_2 |v_x|)^{c_2-1} \exp(-(b_2 |v_x|)^{c_2}). & \text{if } v_x \geq 0 \end{cases} . \tag{5}$$

Here $b_{1,2} > 0$ and $c_{1,2} > 0$ are parameters of the distribution that control the scale and the shape, v_x is the projection of the velocity vector on x axis. The same formula is used for y and z axis.

2.3 Three-Dimensional ACF of Wind Velocity

Usually, autocovariance is modelled using exponential functions [3]. In this paper we use one-dimensional autocovariance function written as

$$K(t) = a_3 \exp(-(b_3 t)^{c_3}). \tag{6}$$

Here $a_3 > 0$, $b_3 > 0$ and $c_3 > 0$ are parameters of the autocovariance function that control the shape of the exponent.

In order to construct three-dimensional autocovariance function we assume that one-dimensional autocovariance function is the same for each coordinate and multiply them.

$$K(t, x, y, z) = a \exp(- (b_t t)^{c_t} - (b_x x)^{c_x} - (b_y y)^{c_y} - (b_z z)^{c_z}). \tag{7}$$

Here $a > 0$, $b_{t,x,y,z} > 0$ and $c_{t,x,y,z} > 0$ are parameters of the autocovariance function. Parameter a and $\exp(b)$ are proportional to wind velocity projection on the corresponding axis. Parameter c controls the shape of the autocovariance function in the corresponding direction; it does not have simple relationship to the wind velocity statistical parameters.

2.4 Data Collection and Preprocessing

We installed anemometer on the tripod and placed it on the balcony. Then we connected load cells to the microcontroller via HX711 load cell amplifiers and programmed the microcontroller to record the output of each sensor every second and print it on the standard output. Then we connected the microcontroller to the computer via USB interface and wrote a script to collect the data coming from the USB and store it in the SQLite database. We decided to store raw sensor values in the range from 0 to 65535 to be able to calibrate anemometer later.

We calibrated three-axis anemometer using commercial anemometer HP-866A and a fan that rotates with constant speed. First, we measured the wind speed that the fan generates when attached to commercial anemometer. Then we successively placed the fan behind and in front of each arm of our anemometer and measured values that the corresponding load cell reported for each side of the arm with and without the fan. Then we were able to calculate two coefficients for each axis: one for negative and another one for positive wind velocities. The coefficient equals the raw sensor value that is equivalent to the wind speed of 1 m/s (Table 1).

Table 1. Calibration coefficients for each arm of three-axis anemometer: C_1 is for negative values and C_2 is for positive values.

Axis	C_1	C_2
X	11.19	12.31
Y	11.46	11.25
Z	13.55	13.90

Table 2. Dataset properties.

Time span	36 days
Size	122 Mb
No. of samples	3 157 234
No. of samples after filtering	2 775 387
Resolution	1 sample per second

We noticed that ambient temperature affects values reported by our load cells: when the load cell heats up (cools down), it reports values that increase (decrease) linearly in time due to thermal expansion of the material. We removed this linear trend from the measured values using linear regression. The code in R [11] that transforms raw sensor values into wind speed is presented in listing 1.1.

Listing 1.1. The code that transforms raw load cell sensor values into wind speed projections to the corresponding axis.

```
sampleToSpeed ← function(x, c1, c2) {
  t ← c(1:length(x))
  reg ← lm(x~t)
  x ← x - reg$fitted.values # remove linear trend
  x ← sign(x)*sqrt(abs(x)) # convert from force to velocity
  x[x<0] = x[x<0] / c1 # scale sensor values to wind speed
  x[x>0] = x[x>0] / c2 # using calibration coefficients
  x
}
```

Over a period of one month we collected 3.1M samples and filtered out 12% of them because they had too large unnatural values. We attributed these values to measurement errors as they spread uniformly across all the time span and are surrounded by the values of regular magnitude. After that we divided each day into two-hour intervals over which we collected the statistics individually. The statistics for each interval is presented in Fig. 2, dataset properties are presented in Table 2.

Unique feature of three-axis anemometer is that it measures both velocity of incident air flow towards the arms and the turbulent flow that forms behind the arms. Turbulent flow velocity distribution peak is often smaller than the peak of incident flow. To exclude it from the measurements one can choose the side with the largest peak (either positive or negative part of (5)), but in this paper we left them as is for the purpose of the research.

3 Results

3.1 Anemometer Verification

In order to verify that our anemometer produces correct measurements we calculated wind speed and direction from the collected samples and fitted them into Weibull distribution and von Mises distribution respectively. These are typical models for wind speed and direction [4,5]. Then we found the intervals with the best and the worst fit for these models using normalised root-mean-square error (NRMSE) calculated as

$$\text{NRMSE} = \frac{\sqrt{\text{E} \left[(X_{\text{observed}} - X_{\text{estimated}})^2 \right]}}{X_{\text{max}} - X_{\text{min}}}. \quad (8)$$

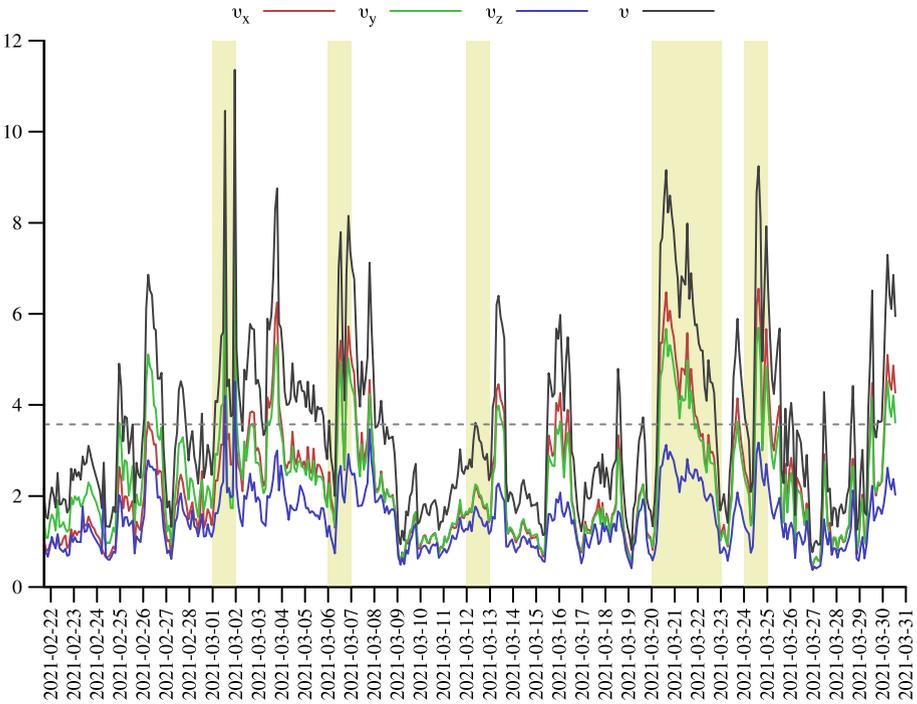


Fig. 2. The statistics for each interval of the collected data. Here v_x , v_y and v_z are mean speeds for each interval for the corresponding axes, v is mean scalar wind speed for each interval. The horizontal line shows overall average speed. Yellow rectangles denote days when EMERCOM of Russia (<https://en.mchs.gov.ru/>) reported wind speeds above average (Color figure online).

Here E is statistical mean, X_{observed} and $X_{\text{estimated}}$ are observed and estimated values respectively.

The wind speed data collected with three-axis anemometer was approximated by Weibull distribution using least-squares fitting. Negative and positive wind speed projections to each axis both have this distribution, but with different parameters. Most of the data intervals contain only one prevalent mean wind direction, which means that one of the distributions is for incident wind flow on the arm of the anemometer and another one is for the turbulent flow that forms behind the arm. For z axis both left and right distributions have similar shapes, for x and y axes the distribution for incident flow is taller than the distribution for turbulent flow. The best-fit and worst-fit distributions for each axis are presented in Fig. 3.

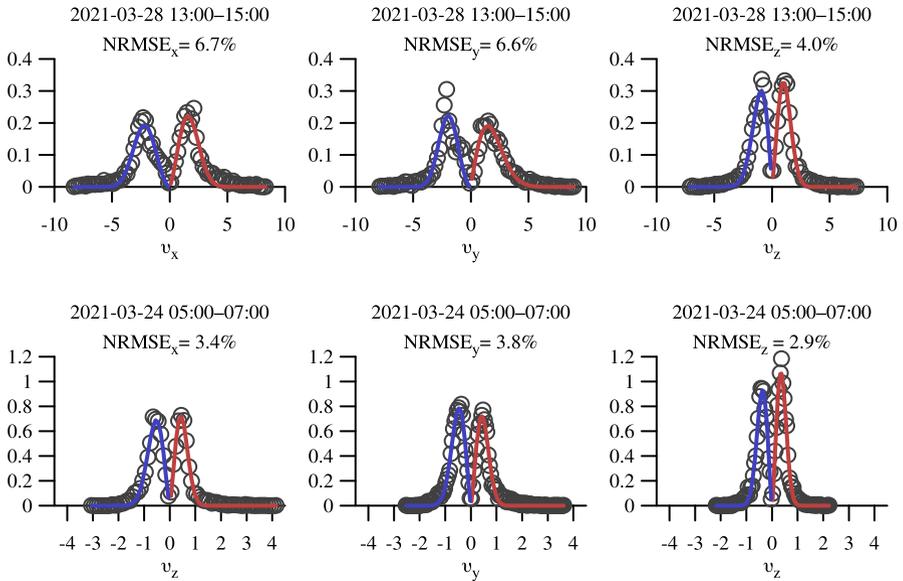


Fig. 3. Per-axis wind velocity distributions fitted into Weibull distribution. Data intervals with the largest error (the first row), data intervals with the smallest error (the second row). Red line shows estimated probability density of positive wind speed projections, blue line shows estimated probability density of negative wind speed projections and circles denote observed probability density of wind speed projections (Color figure online).

Wind direction was approximated by von Mises distribution using least-squares fitting. Following the common practice [7] we divided direction axis into sectors: from -180° to -90° , from -90° to 0° , from 0 to 90° and from 90° to 180° — and fitted each sector independently. We chose four sectors to have one sector for each side of the anemometer. The best-fit and worst-fit distributions for each sector are presented in Fig. 4.

In order to verify that our anemometer produces correct time series we performed synchronous measurements with commercial anemometer HP-866A. Two anemometers were placed in the open field near the shore. The commercial anemometer was directed towards the mean wind direction as it measures the speed only. The data from the two anemometers was recorded synchronously. From the data we selected intervals with wind gusts and compared the

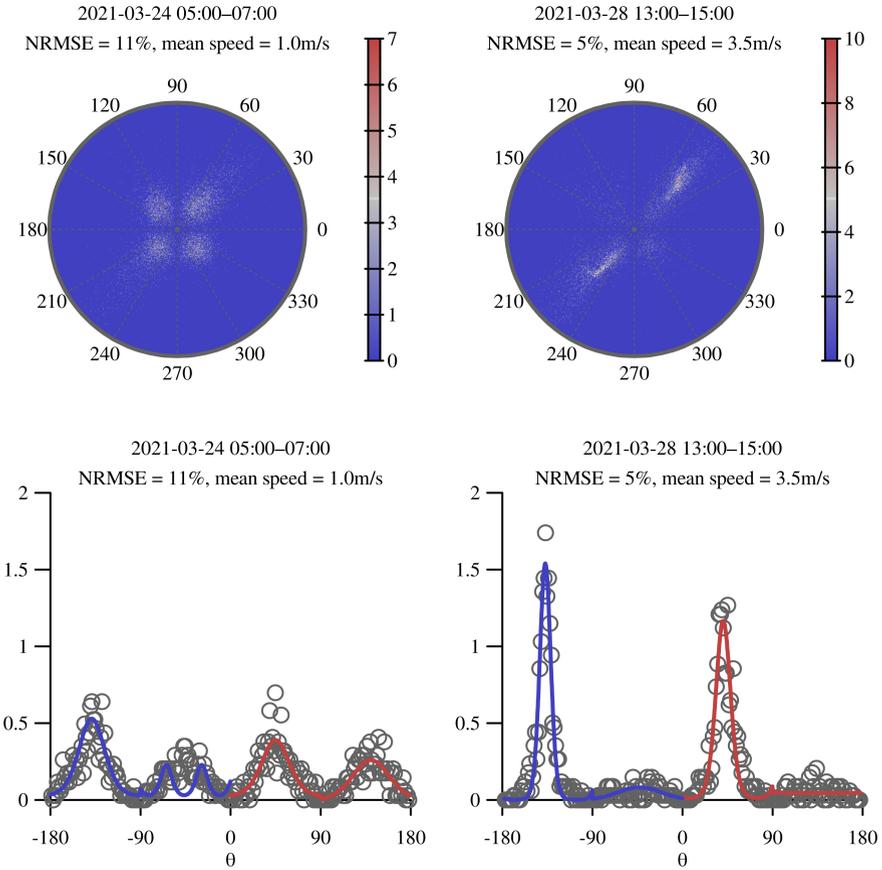


Fig. 4. Bivariate x and y velocity distribution in polar coordinates (first row), wind direction distributions fitted into von Mises distribution (second row). Data interval with the largest error (left column), data interval with the smallest error (right column). Red line shows estimated probability density of positive direction angles, blue line shows estimated probability density of negative direction angles and circles denote observed probability density of direction angles. 0° is north (Color figure online).

measurements of the two anemometers. To compare them we scaled load cell anemometer measurements to minimise NRMSE of the difference between the two time series to compensate for the errors in calibration coefficients. The results showed that there is some correspondence between the measurements of the two anemometers (Fig. 5).

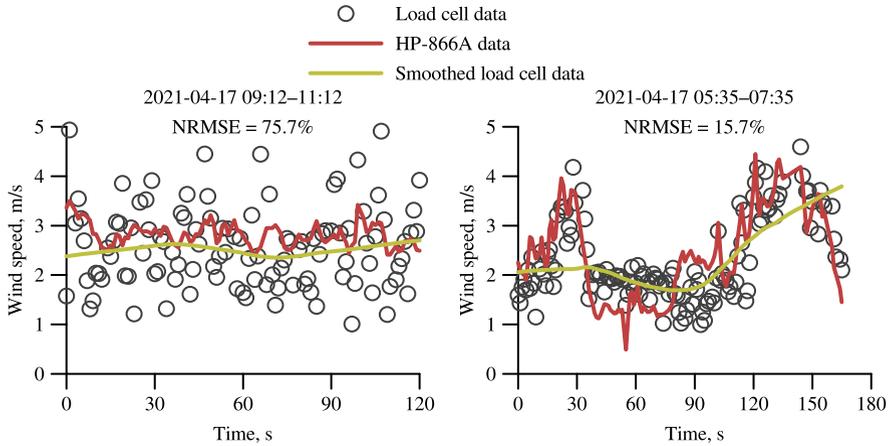


Fig. 5. Comparison of measurements obtained from three-axis and HP-866A anemometers.

Finally, we computed autocovariance for each axis as

$$K(\tau) = E [(X_t - \bar{X})(X_{t-\tau} - \bar{X})] \quad (9)$$

and fitted it into (6). Per-axis ACFs have pronounced peak at nought lag and long tails. The largest NRMSE is 2.3%. Variances for x and y axes are comparable, but ACF for z axis has much lower variance. Parameters a and b from (6) are positively correlated with wind speed for the corresponding axis. The best-fit and worst-fit ACFs for each axis are presented in Fig. 6.

3.2 Turbulence Coefficient

Since our anemometer measures both incident and turbulent flow we took an opportunity to study the speed of the turbulent flow in relation to incident flow. We calculated the absolute mean values of positive and negative wind velocity projections for each time interval of the dataset. We considered the flow with the larger absolute mean value incident, and the other one is turbulent. Then we calculated turbulence coefficient as the ratio of absolute mean speed of turbulent flow to the absolute mean speed of incident flow. We found that the average ratio is close to 60%. It seems, that the ratio decreases as the wind speed increases for wind speeds of 1–5 m/s, but for higher wind speeds we do not have the data. Turbulence coefficient can be used in wind simulation models to control the magnitude of turbulent flow that forms behind the obstacle [8].

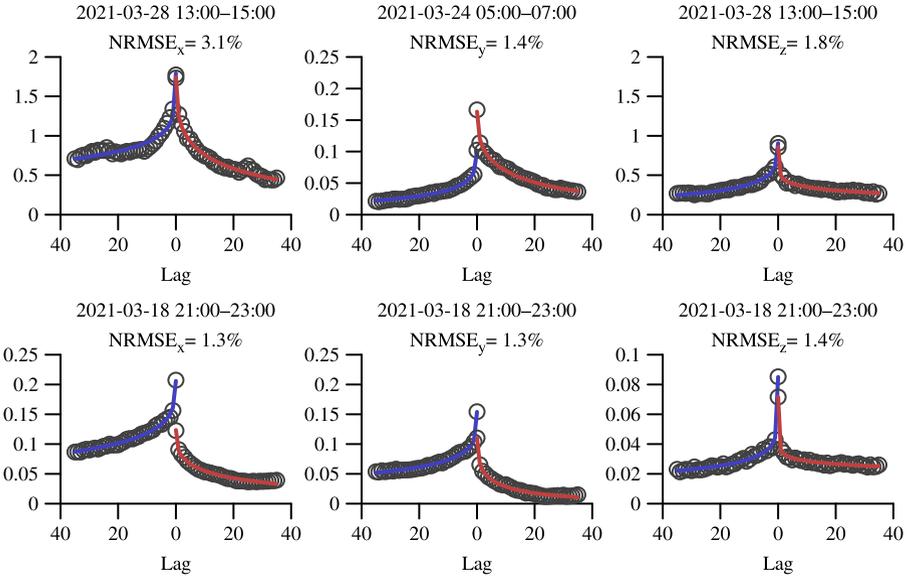


Fig. 6. Per-axis wind velocity ACF fitted into (6). Data intervals with the largest error (the first row), data intervals with the smallest error (the second row). Red line shows estimated ACF of positive wind speed projections, blue line shows estimated ACF of negative wind speed projections and circles denote observed ACF of wind speed projections. Note: graphs use mirrored axes (Color figure online).

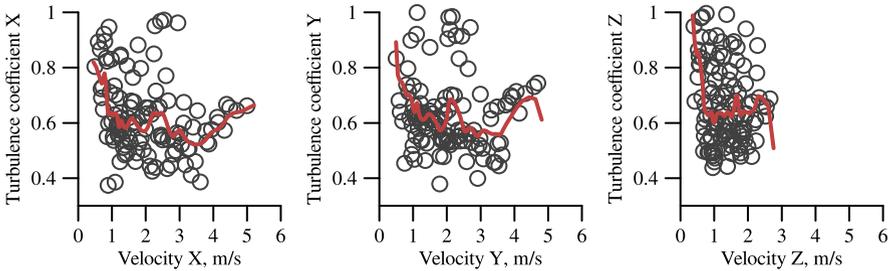


Fig. 7. The ratio of absolute mean speed of turbulent flow to the absolute mean speed of incident flow for each axis.

3.3 Wind Simulation Using Measured ACFs

We simulated three-dimensional wind velocity using autoregressive model implemented in Virtual Testbed [2] — a programme for workstations that simulates ship motions in extreme conditions and physical phenomena that causes them (ocean waves, wind, compartment flooding etc.). Using the data obtained with three-axis anemometer on March 28, 2021, 01:00–03:00 UTC we approximated four-dimensional autocovariance function using (7) by setting the corresponding

parameters from one-dimensional autocovariance functions estimated from the data obtained with anemometer, all other parameters were set close to nought. Non-nought parameters are listed in Table 3. We found that velocity and direction distributions and ACFs of each axis of simulated wind and real wind are similar in shape, but are too far away from each other (see Fig. 8). We consider these results preliminary and will investigate them further in future work.

4 Discussion

NRMSE of wind speed distribution approximation has positive correlation with wind speed: the larger the wind speed, the larger the error and vice versa. Larger error for low wind speeds is caused by larger skewness and kurtosis (see the first row of Fig. 3). Similar approximation errors can be found in [5] where the authors improve approximation accuracy using joint wind speed and direction distributions. Such studies are outside of the scope of this paper, because here we verify anemometer measurements using well-established mathematical models, but the future work may include the study of these improvements.

NRMSE of wind direction distribution approximation has negative correlation with wind speed: the larger the wind speed, the smaller the error and vice

Table 3. Input parameters for AR model that were used to simulate wind velocity.

Axis	ACF <i>a</i>	ACF <i>b</i>	ACF <i>c</i>	Mean velocity, m/s
<i>x</i>	1.793	0.0214	0.2603	-2.439
<i>y</i>	1.423	0.01429	0.2852	-2.158
<i>z</i>	0.9075	0.06322	0.3349	-1.367

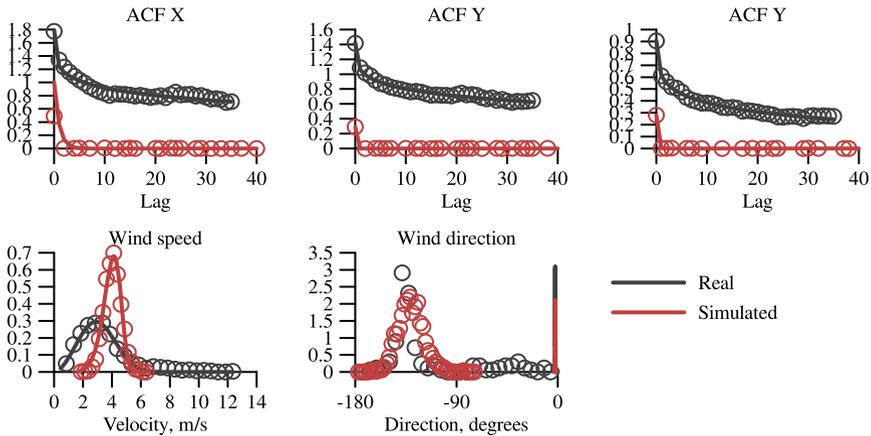


Fig. 8. Comparison of simulated wind velocity and direction distributions and per-axis ACFs between the data from the anemometer and the data from Virtual Testbed (2021, March 28, 01:00–03:00, UTC).

versa. This is in agreement with physical laws: the faster the flow is the more determinate its mean direction becomes, and the slower the flow is the more indeterminate its mean direction is.

Three-axis anemometer disadvantages are the following. The arm for the z axis is horizontal, and snow and rain put additional load on this cell distorting the measurements. Also, thermal expansion and contraction of the material changes the resistance of load cells and distorts the measurements. Pressure force on the arm is exerted by individual air particles and is represented by choppy time series, as opposed to real physical signal that is represented by smooth graph. The first two deficiencies can be compensated in software by removing linear trend from the corresponding interval. The last one makes anemometer useful only for offline studies, i.e. it is useful to gather statistics, but is unable to measure immediate wind speed and direction.

We used a balcony for long-term measurements and open field for verification and calibration. We found no clues that the balcony affected the distributions and ACFs of wind speed. The only visible effect is that the wind direction is always parallel to the wall which agrees with physical laws. Since we measure pressure force directly, the mean wind direction does not affect the form of the distributions, but only their parameters.

5 Conclusion

In this paper we proposed three-axis anemometer that measures wind speed for each axis independently. We analysed the data collected by this anemometer and verified that per-axis wind speeds fit into Weibull distribution with the largest NRMSE of 6.7% and wind directions fit into von Mises distribution with the largest NRMSE of 11%. We estimated autocovariance functions for wind speed for each axis of the anemometer and used this approximations to simulate wind flow in Virtual Testbed. The parameters of these functions allow to control both wind speed and mean direction. The future work is to construct an array of anemometers that is able to measure spatial autocovariance using the proposed anemometer as the base.

Acknowledgements. Research work is supported by Council for grants of the President of the Russian Federation (grant no. MK-383.2020.9).

References

1. Arens, E., et al.: Measuring 3D indoor air velocity via an inexpensive low-power ultrasonic anemometer. *Energy Build.* **211**, 109805 (2020). <https://doi.org/10.1016/j.enbuild.2020.109805>
2. Bogdanov, A., Degtyarev, A., Gankevich, I., Khramushin, V., Korkhov, V.: Virtual testbed: concept and applications. In: Gervasi, O., et al. (eds.) ICCSA 2020. LNCS, vol. 12254, pp. 3–17. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58817-5_1

3. Box, G.E.P., Jenkins, G.M.: *Time Series Analysis: Forecasting and Control*. Holden-Day, San Francisco (1976)
4. Carta, J.A., Bueno, C., Ramírez, P.: Statistical modelling of directional wind speeds using mixtures of von Mises distributions: case study. *Energy Convers. Manage.* **49**(5), 897–907 (2008). <https://doi.org/10.1016/j.enconman.2007.10.017>
5. Carta, J.A., Ramírez, P., Bueno, C.: A joint probability density function of wind speed and direction for wind energy analysis. *Energy Convers. Manage.* **49**(6), 1309–1320 (2008). <https://doi.org/10.1016/j.enconman.2008.01.010>
6. Cosgrove, M., Rhodes, B., Scott, J.B.: Ultra-low-cost logging anemometer for wind power generation feasibility surveys. In: *Proceedings of the 14th New Zealand Electronics Conference*, Wellington, New Zealand, pp. 153–158 (2007). <https://hdl.handle.net/10289/1485>
7. Feng, J., Shen, W.Z.: Modelling wind for wind farm layout optimization using joint distribution of wind speed and wind direction. *Energies* **8**(4), 3075–3092 (2015). <https://doi.org/10.3390/en8043075>
8. Gavrikov, A., et al.: Virtual testbed: simulation of air flow around ship hull and its effect on ship motions. In: Gervasi, O., et al. (eds.) *ICCSA 2020*. LNCS, vol. 12254, pp. 18–28. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58817-5_2
9. Justus, C.G., Hargraves, W.R., Yalcin, A.: Nationwide assessment of potential output from wind-powered generators. *J. Appl. Meteorol. Climatol.* **15**(7), 673–678 (1976). [https://doi.org/10.1175/1520-0450\(1976\)015\(0673:NAOPOF\)2.0.CO;2](https://doi.org/10.1175/1520-0450(1976)015(0673:NAOPOF)2.0.CO;2)
10. Kaimal, J., Wyngaard, J., Izumi, Y., Coté, O.: Spectral characteristics of surface-layer turbulence. *Q. J. R. Meteorol. Soc.* **98**(417), 563–589 (1972)
11. R Core Team: *R: a Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna (2020). <https://www.R-project.org/>
12. Veers, P.: Modeling stochastic wind loads on vertical axis wind turbines. In: *25th Structures, Structural Dynamics and Materials Conference*, p. 910 (1984)
13. Veers, P.S.: *Three-dimensional wind simulation*. Technical Report. Sandia National Labs, Albuquerque (1988)
14. Yakunin, A.G.: 3D ultrasonic anemometer with tetrahedral arrangement of sensors. In: *Journal of Physics: Conference Series*, vol. 881, pp. 12–30. IOP Publishing (2017)
15. Yim, J.Z., Chou, C.R., Huang, W.P.: A study on the distributions of the measured fluctuating wind velocity components. *Atmos. Environ.* **34**(10), 1583–1590 (2000). [https://doi.org/10.1016/S1352-2310\(99\)00414-8](https://doi.org/10.1016/S1352-2310(99)00414-8)