National Renewable Energy Laboratory

Innovation for Our Energy Future

TurbSim User's Guide

B.J. Jonkman and M.L. Buhl, Jr.

TurbSim User's Guide

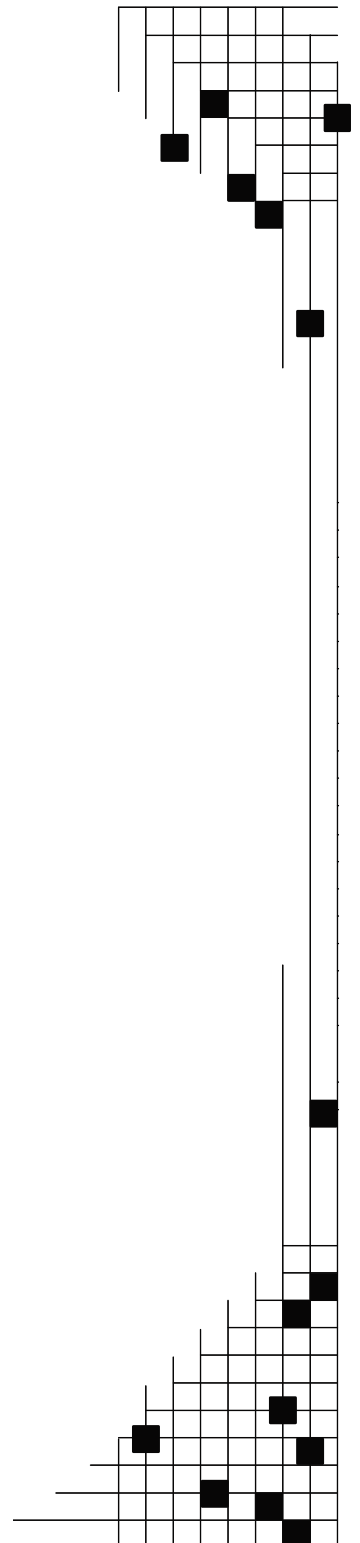B.J. Jonkman and M.L. Buhl, Jr.

Prepared under Task No. WER6.2104

National Renewable Energy Laboratory
1617 Cole Boulevard, Golden, Colorado 80401-3393
303-275-3000 • www.nrel.gov

Operated for the U.S. Department of Energy
Office of Energy Efficiency and Renewable Energy
by Midwest Research Institute • Battelle

Contract No. DE-AC36-99-GO10337

Table of Contents

List of Abbreviations

BLAS   Basic Linear Algebra Subprograms
CoRA   Colorado Research Associates
CXML   Compaq Extended Math Library
DNS   Direct Numerical Simulation
ETM   Extreme Turbulence Model
EWM   Extreme Wind Model
FF   full-field time series
FFT   Fast Fourier Transform
FFTPACK   FFT Package
HH   hub-height
IEC   International Electrotechnical Commission
LAPACK   Linear Algebra Package
LES   Large-Eddy Simulation
LLLJP   Lamar Low-Level Jet Project
NCAR   National Center for Atmospheric Research
NREL   National Renewable Energy Laboratory
NTM   Normal Turbulence Model
pRNG   pseudorandom number generator
TI   Turbulence Intensity

Introduction

TurbSim is a stochastic, full-field, turbulent-wind simulator. It numerically simulates time series of three-dimensional wind velocity vectors at points in a vertical rectangular grid. TurbSim output can then be used as input into AeroDyn-based [1] codes such as FAST [2], YawDyn [3], or MSC.ADAMS® [4].

The underlying theory behind this method of simulating time series assumes a stationary process. To simulate non-stationary components, TurbSim also has the ability to superimpose coherent turbulent structures onto the time series it generates.

History

In 1988, Paul Veers of Sandia National Laboratories wrote a program called SNLWIND [5] that could generate full-field turbulent wind for the u component only. In 1992, Neil Kelley of the National Renewable Energy Laboratory (NREL) added several spectral models to SNLWIND and modified it to generate the v and w components [6]. SNLWIND-3D was the result. During the next five years, NREL researchers added further modifications, including the Kaimal and von Karman spectral models specified by the International Electrotechnical Commission (IEC). Gary Desroachers modified it to run on many different platforms by including C-Preprocessor directives for conditional compilation. In 1997, Marshall Buhl added the ability to generate binary files that are compatible with Garrad Hassan's *BLADED* turbine design code [7].

Both SNLWIND and SNLWIND-3D were written in FORTRAN 77 and required recompilation for different grid densities and run lengths. This made using and supporting the programs difficult.

In January 2000, Neil Kelley removed all of the spectral models except the two defined by the IEC, and Marshall Buhl modernized the remaining code by rewriting all but the Fast Fourier Transform (FFT) routines in Fortran 95 and eliminating the C-Preprocessor directives. This included using dynamic-memory allocation for the big arrays. He found and fixed a few bugs, modified the input and output file format, streamlined processes, and added the ability to generate hub-height files in AeroDyn format. Because the changes were substantial, Marshall renamed the program SNwind (Sandia/NREL Wind) [8].

In 2003, NREL researchers updated the code to allow for the addition of coherent turbulent structures and results from the Lamar Low-Level Jet Project (LLLJP). Bonnie Jonkman put the spectral models from SNLWIND-3D back in and replaced the FORTRAN 77 FFT routines with more modern routines from the Compaq Extended Math Library (CXML). She also made changes to the Cholesky factorization algorithm, which sped up the code and allowed for a significant reduction in the memory required to run the program. She eliminated the requirement that the grid be an even number of points in each direction, and allowed the grid height to be different from its width.

Once these enhancements were in place, the researchers modified it to generate coherent structures with realistic temporal and spatial scaling, and coherent turbulence time step files became another output option. The code was then renamed TurbSim (Turbulence Simulator) because of its ability to generate coherent turbulence.

Neil Kelley has written a companion document, *Overview of the TurbSim Stochastic Inflow Turbulence Simulator* [9], to discuss how TurbSim has been developed and some of the theory behind that development.

Retrieving Files from the Archive

Download the TurbSim archive from the NREL web server page at http://wind.nrel.gov/designcodes/. The file will have a name like "TurbSim_v101.exe." Create a TurbSim folder somewhere on your file system and put this file there. You can double click on it from Windows Explorer or type "TurbSim_v101" (or the file name) at a command prompt with the TurbSim folder as the current directory. This will create some files and folders. Please see Marshall Buhl's paper *Installing NWTC Design Codes* [10] for information on how to set up TurbSim to run in any folder.

You will also need to download the coherent structures archive from NREL's web server page if you wish to generate coherent turbulence with TurbSim. The file will have a name like "TSM_structures.exe." Create a folder somewhere on your file system and put this file there. Execute the program by double clicking on it or by typing "TSM_structures" at a command prompt with the folder you created as the current directory. When executed, this archive will create the files and folders used to define coherent structures. You will need to type the name of the path to these folders in TurbSim and AeroDyn input files.

Distributed Files

See Table 1 for a list of the files included in the TurbSim archive.

Certification Test

Before using TurbSim, run the certification-testing program. It is a batch file called "CertTest.bat" and is located in the "Test" folder. To test the installation, edit "CertTest.bat" and set the environment variables found near the top of the file to settings that are compatible with your system. You will probably have to change only the "Editor" variable. Then, open up a command window, go to the Test folder, and enter "CertTest" or "CertTest MATLAB" if you have access to MATLAB® [11] and would like to see plots of the data.

TurbSim will run several times. The test procedure will compare the new results to those stored in the "Test\TstFiles" folder, and it will write the differences between the output files to a file called "CertTest.out." If you have specified the "MATLAB" option, MATLAB will also open and plot many results. You must close the MATLAB program before the test procedure can continue. Before finishing, the test procedure will automatically open the "CertTest.out" file with the editor you specified with the "Editor" variable. Scan through the file; the only

Table 1. Files in the TurbSim Archive.

File(s)	Description
ArcFiles.txt	The list of files that are written to the archive.
ArcFiles.bat	The batch file that creates the archive.
ChangeLog.txt	The list of changes to TurbSim.
RunTurbSim.pl	A sample perl script used to run TurbSim, using a different seed each time.
Source*.*	The Fortran source code for TurbSim.
TurbSim_AD.ctp	A sample AeroDyn coherent turbulence parameter input file.
TurbSim.exe	The TurbSim program for 32-bit Windows® platforms.
TurbSim.inp	A sample input file.
TurbSim.pdf	The user's guide in PDF format.
TurbSimOverview.pdf	An overview of TurbSim in PDF format.
Test*.*	Files used to run and manage the certification tests and MATLAB® scripts for reading TurbSim data.
Test\EventData*.*	Files used to run the certification tests with coherent turbulence.
Test\TstFiles*.*	My results for the certification tests.

differences should be the date and time stamps in the headers of the files and the CPU time in the summary files. If you recompiled TurbSim with another compiler, there may be some slight differences in the last digit of many of the numbers.

Compiling TurbSim

You should not need to compile TurbSim unless you want to make changes to the code. The archive contains Fortran code specific to TurbSim as well as the Fortran FFTPACK version 4.1 [12], LAPACK version 3.0 [13], BLAS [14], [15], and RanLux [16] routines that TurbSim uses. TurbSim also uses the NWTC Subroutine Library version 1.12, [17] which you will need to download from the NREL web server page at http://wind.nrel.gov/designcodes/.

The code has been written primarily for the Intel® Visual Fortran compiler. If you want to port TurbSim to another platform or compiler, you may have to make changes in the NWTC Subroutine Library's SysVF.f90 file and possibly the BLAS LSAME() function. If your compiler has trouble with non-standard compiler directive prefixes, you may also have to remove the code between lines beginning with `!MS$if` and `!MS$endif` in the TurbSim source files. If you have access to an optimized BLAS library, you are encouraged to link your code with it instead of using the reference BLAS routines included in the TurbSim archive.

Using TurbSim

To run TurbSim, enter "turbsim [/h] [<RootName.ext>]", where /h and <RootName.ext> are optional.

Examples:

    turbsim

> This will start TurbSim and open "turbsim.inp". This is equivalent to entering "turbsim turbsim.inp".

    turbsim myroot.tsm

> This will start TurbSim and open the input file "myroot.tsm".

All output files will have the specified root file name with different extensions.

Wind components are as follows:

u	downwind
v	to the left when looking downwind
w	up

Input Files

Do not add or remove lines from the sample input files in the archive. TurbSim assumes that parameters are located on specific lines. See "Appendix A: Sample TurbSim Input File" for an example of a TurbSim input file.

Runtime Options

The first two input values tell TurbSim which one of three pseudorandom number generators (pRNGs) to use and how to initialize it. The random numbers generated by these pRNGs are used to generate random phases in all of the spectral models and to determine a number of randomized boundary conditions and coherent turbulence parameters in the non-IEC spectral models. The first input value is a random seed that must be an integer between -2147483648 and 2147483647 (inclusive). The second input value is either a second random seed or the strings "RanLux" or "RNSNLW" which are not case sensitive.

If the second input is a random seed, it must also be an integer between –2147483648 and 2147483647 (inclusive). TurbSim will then use the two seeds to initialize the intrinsic pRNG, which uses two separate congruential generators together to produce a period of about 10^{18}. This intrinsic pRNG is based on an algorithm by Pierre L'ecuyer [18] and is identical to the one found in SNwind.

If the second input is the string "RNSNLW," TurbSim will generate random numbers using the algorithm found in SNLWIND and SNLWIND-3D. It will be initialized with only one seed.

Finally, if the second input is the string "RanLux," TurbSim will use Lüscher's level 3 "Luxury Pseudorandom Numbers" [19], [20]. This pRNG is based on a subtract-and-borrow algorithm with a period on the order of 10^{171} and modified by throwing numbers away to destroy correlations. It will be initialized with only one seed. We recommend that you use RanLux since our initial tests show that it seems to be the best behaved of the three generators.

The next eight fields tell TurbSim what you want for output. Valid entries for these flags are the strings "true" and "false." They are not case sensitive, and you can use any combination of output types. See the "Output files" section of this paper for more complete descriptions of these file types.

The first output option is to generate hub-height (HH) time series in binary form. This format contains time series of HH wind data and turbulence parameters. The second file type also contains time series of HH wind data and turbulence parameters, but they are stored in formatted (human-readable) form. The third is HH time series in AeroDyn format.

The fourth and fifth options generate binary, full-field (FF) time series in formats readable by AeroDyn. The fourth option is a format unique to TurbSim and provides an encoding scheme designed to give maximum resolution. It contains all the information necessary to decode the time series in one file, but it is not compatible with *BLADED*. *AeroDyn v12.60, which will soon be released, will support this new file format; until then, use the BLADED-compatible FF files.* Option number five uses an encoding scheme that provides less resolution in some cases, but it is compatible with *BLADED*.

The sixth option is binary tower time series, containing points in a line at the tower centerline from the bottom of the rectangular grid to the ground. If you have chosen to output FF time series in the format unique to TurbSim, these tower points will be added to the FF binary output file. Otherwise, a separate tower points binary file will be created, using an encoding scheme similar to the BLADED FF file format.

The seventh output option is FF time series in SNLWIND-3D human-readable format, and the last output option is coherent turbulence time steps in AeroDyn form. The coherent turbulence feature works only with non-IEC spectral models when the gradient Richardson number is greater than –0.05. TurbSim also creates binary FF time series in AeroDyn form when a coherent turbulence time step file is requested. If no FF time series has been specified, TurbSim will create *BLADED*-style binary FF files.

The last parameter in this section is a flag to tell whether the turbine rotates in the clockwise direction when looking downwind. This feature is probably useful only if you want to use FF results with

BLADED and compare them to results using AeroDyn. This flag does not have any affect on results when used with AeroDyn.

Turbine/Model Specifications

The first two parameters are the number of grid points to generate in the vertical and horizontal directions. Unlike SNwind, which accepted only even numbers, TurbSim allows both even and odd grid-point sizes.

The third parameter in this section is the time step. We set it to 0.05 seconds and recommend that you do not change it since the time step determines the maximum frequency used in the inverse FFT.

The next parameter is the length in seconds of the data to analyze. This number dictates how many frequencies will be used to generate the output time series. We recommend that this number be at least 600 seconds. In order to speed up the inverse FFT computations, TurbSim may add a few extra time steps to ensure that the number of analysis time steps is a product of small prime numbers.

The usable length of the data to output in seconds comes next. This differs slightly from the actual amount of data that TurbSim will output. Because AeroDyn requires that there be data both upwind and downwind of the tower in case the turbine is yawed, it insists that there be extra data in the FF files to shift the data enough to ensure that the turbine will reside entirely within the wind-data domain. TurbSim always adds the amount of time equal to the grid width divided by the mean HH wind speed to the requested amount of usable time; this is the length of the output time series. If the analysis time is less than the output time, TurbSim will use the length of the output time series as the analysis time.

The hub height of the turbine is the sixth parameter in this section. TurbSim works in the metric system so enter the value in meters.

The next two parameters are the height and width of the grid. TurbSim assumes the diameter of the rotor disk is the smaller of these two values. If the grid height is larger than its width, TurbSim will place the top of the rotor disk at the top of the grid. Otherwise, the hub is centered both vertically and horizontally on the grid. (See Figure 1.) Keep in mind that AeroDyn will test to make sure that the rotor radius plus the shaft length (the distance from the yaw axis to the hub) is less than half the grid width. The height and width must be large enough to ensure that no part of the blade can lie outside the grid.

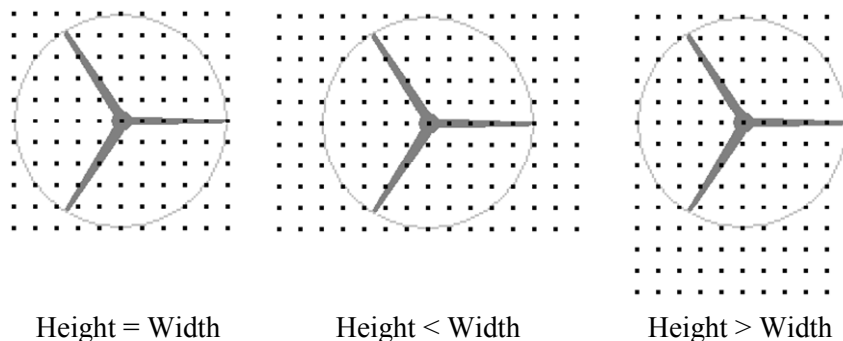Height = Width   Height < Width   Height > Width

Figure 1. Example grid and rotor placements.

The last two input parameters in this section are the mean flow angles. These angles will add mean crosswise and vertical wind directions, which are constant across the entire grid. You should use caution when using these inputs since AeroDyn marches FF grids through the turbine in the x-direction at the

mean hub-height wind speed, regardless of the flow angles. This can possibly give strange results if the mean flow angles are not small.

The vertical mean-flow angle, parameter nine in this section of the input file, allows you to simulate wind blowing uphill or downhill. Enter the angle in degrees and do not exceed 45 degrees in magnitude. A positive value means the wind is blowing uphill.

The last parameter in this section of the input file, the horizontal mean-flow angle, is the wind direction. Positive angles (enter them in degrees) shift the wind to the left when looking downwind. In the transformation, the horizontal mean-flow angle is applied before the vertical mean-flow angle. If you are generating full-field wind files for AeroDyn, we recommend that you introduce a yaw error in your aeroelastic simulation instead of using a non-zero horizontal mean-flow angle.

Meteorological Boundary Conditions

The first parameter tells TurbSim which spectral model it should use. Valid values are "IECKAI" for the Kaimal model, "IECVKM" for the Von Karman model, "SMOOTH" for the Risø smooth terrain model, "WF_UPW", "WF_07D", and "WF_14D" for the NREL Wind Farm upwind, seven rotor-diameters downwind, and 14 rotor-diameters downwind models, "NWTCUP" for the NREL Wind Farm upwind model modified for NWTC conditions, and "GP_LLJ" for the NREL Great Plains Low-Level Jet model. The values are not case sensitive. For more information on these models, see the section "Spectral Models" in this report.

The second input parameter tells TurbSim which IEC standard to use. Enter "1" to use the scaling from the IEC 61400-1 [21] standard or enter "2" or "3" to use the scaling from the IEC 61400-2 [22] or -3 [23] standards. If you want to use the scaling parameters from the second edition of the IEC 61400-1 standard [24], follow your input with the string "-ED2" (i.e. "1-ED2"). Likewise, if you want to use the scaling parameters from IEC 61400-1, edition 3 [21], input the string "1-ED3". If the 61400-1 edition number is not specified, TurbSim will use the scaling from the third edition of IEC 61400-1 for the Kaimal model and scaling from the standard's second edition for the von Karman model, which is not defined in the newer edition. This input is ignored if the spectral model is not "IECKAI" or "IECVKM."

The third parameter tells TurbSim what turbulence intensity (TI) you want to use with the IEC Kaimal or von Karman models. Input values of "A", "B", or "C" correspond to the standard IEC Normal Turbulence Model (NTM) turbulence characteristics, with "A" being the most turbulent. You can also specify the TI in percent instead of choosing the standard turbulence characteristics.

If you enter the string "KHTEST" in the third parameter, TurbSim will allow you to test the effects of a KH billow. In order to use this feature, you must specify the "NWTCUP" spectral model. TurbSim will then override the inputs for Richardson number, power law coefficient, and billow type, size, and location. An LES-type billow centered on the rotor disk will be scaled so that the coherent TKE is at least $30 \ \mathrm{m^2/s^2}$ and the billow achieves a bandwidth of 25 Hz. This billow will last at least half of the usable length of the output time series, and will start a quarter of the way through the time series. This parameter is not used in the other spectral models.

The fourth parameter is the type of IEC wind to use. Valid entries are "NTM" for the Normal Turbulence Model (NTM), "*x*ETM" for the Extreme Turbulence Model (ETM), "*x*EWM1" for the turbulent Extreme Wind Speed Model (EWM) using the 10 minute average wind speed with a recurrence period of 1 year, and "*x*EWM50" for the turbulent EWM using a wind speed with a recurrence period of 50 years. The *"x"* in all three extreme wind models must be replaced with the number 1, 2, or 3, indicating the respective wind turbine class. The definitions of these types and the wind turbine classes can be found in

the 61400-1 Edition 3 standard [21]. This input is used only with the IEC spectral models. If the turbulence intensity was specified as a percent instead of the standard turbulence characteristics, the wind model must be "NTM."

The fifth parameter in this section tells TurbSim how to calculate the mean wind profile. Valid entries are "IEC" for the profile used in previous versions of TurbSim and SNwind (which uses the power law over the rotor disk and logarithmic profile above and below the rotor), "LOG" for the logarithmic profile, "PL" for the power law wind profile, and "JET" for the low-level jet wind profile. The "JET" wind profile is available only with the "GP_LLJ" spectral model. You may also enter the string "default" here for TurbSim to pick a wind profile based on the spectral model. Please see the "Wind Profiles" section of this report for more details on the different wind profile types.

The sixth parameter in this section specifies the height of the reference wind speed (in meters). This allows you to specify the mean wind speed at a height other than the hub height. TurbSim uses this reference height with the wind profile type to calculate the HH mean wind speed.

The seventh parameter in this section is for the mean, streamwise wind speed at the reference height. It will be the mean total wind speed, which is the vector sum of the u, v, and w components. Its value must be in meters per second. If you use the "JET" wind profile, you may also enter the string "default" here for TurbSim to calculate a wind speed based on the jet height.

The eighth parameter is the height of the low-level jet in meters. Enter the approximate height at which the low-level jet wind profile reaches its maximum wind speed, or enter the string "default" if you would like TurbSim to calculate a jet height for you. This value must be between 70 and 490 meters, but it is not used when the wind profile type is not "JET."

The next parameter is the power law exponent, which is used to compute the mean u-component wind speeds across the rotor disk for the power law wind profile (for both "IEC" and "PL" wind profile types). Enter a positive number to increase the wind speed with height or enter the string "default" if you would like TurbSim to use a default value that is based on the specified spectral model.

The surface roughness length is the last parameter in this section. This length is the extrapolated height at which the mean wind speed becomes zero in a neutral atmosphere, assuming a logarithmic vertical wind profile. Enter the length in meters, or enter the string "default" if you would like TurbSim to use a default value that is based on the specified spectral model.

Non-IEC Meteorological Boundary Conditions

If you have specified either the Kaimal or von Karman spectral model, TurbSim will stop reading the input file at this point. The other spectral models require these additional meteorological boundary conditions. All of the inputs in this section, with the exception of the gradient Richardson number, can be replaced with the string "default" which is not case sensitive.

The first parameter in this section is the site latitude in degrees. The latitude is used only to calculate the default mixing layer depth. The latitude cannot be between -5 and 5 degrees.

Next is the gradient Richardson number. This parameter is the only one in this section that does not accept the value "default." Enter zero for neutral conditions, a negative value for unstable conditions, or a positive number for stable atmospheric conditions.

The friction or shear velocity, u_*, comes next. Its value must be in meters per second. If you have entered "default" for the reference wind speed, you must *not* enter "default" for the u_* input parameter.

Following the friction velocity is the depth of the mixing layer. Enter the depth in meters.

Next are the cross-correlation coefficients for *u'w'*, *u'v'*, and *v'w'*. TurbSim uses these coefficients to correlate the time series of the three wind components. You may have to adjust these numbers to obtain the desired calculated cross-correlation coefficients or mean Reynolds stresses ($\overline{u'w'}$, $\overline{u'v'}$, and $\overline{v'w'}$) in the TurbSim summary file.

The coherence decrements for the three wind components are the next three parameters in this section. They must be positive numbers indicating the degree of spatial coherence between points on the grid. The decrements should be based on the *coherence-squared* function.

The coherence exponent is the last parameter in this section. Please see the section "Spatial Coherence Models" of this report for details on how the decrements and exponent for the spatial coherence function(s) are used.

Coherent Turbulence Scaling Parameters

The coherent turbulence scaling parameters are used with non-IEC spectral models when the gradient Richardson number is greater than -0.05 and the option to create coherent turbulence time step files has been selected. TurbSim uses empirical values to calculate when and how coherent events should be added to the background turbulence. It uses these values to create a coherent turbulence time step file that AeroDyn will read.

The first parameter in this section is the name of the path that contains the coherent event definition files. This directory should contain files called "Events.les," "Events.dns," and "Events.xtm" as well as one or more files named "Event*xxxxx*.dat" (with the *xxxxx* replaced with digits). These event definition files and the associated binary data files that AeroDyn will read are provided in the coherent structure archive on the TurbSim website (in folder EventData).

The second parameter tells TurbSim which type of coherent event files to use. To have TurbSim randomly choose between the two included types of event files, type the string "Random" here. You may also set this parameter to either of the strings "les" or "dns" to use a specific type of event file. TurbSim will automatically use LES events when the "KHTEST" option is being used.

Set the next parameter to "true" if you wish to randomize the location of the coherent structures in the rotor disk. A value of "true" will override the next three input parameters (the disturbance scale and the location of the wave within the rotor disk). Instead, the coherent structures will be randomly chosen to cover the full rotor disk (75% of the time), only the lower half of the disk (12.5% of the time), or only the upper half (12.5% of the time).

The fourth parameter in this section is the disturbance scale. This parameter determines the size of the coherent event data set relative to the rotor disk. It is the ratio of the height of the coherent dataset to the (assumed) rotor diameter. A value of one will make the coherent structures the height of the rotor disk, 0.5 will make them half the height of the rotor disk. If the "KHTEST" option is used, TurbSim automatically sets this parameter to 1.0.

The next two parameters in the input file position the coherent structures on the rotor disk. The first input is the fractional location of the tower centerline from the right to left side (looking downwind) of the

coherent event dataset. The coherent structures are periodic in this direction so they will cover the grid horizontally, regardless of the location of the tower centerline. The second parameter is the fractional location of hub height from the bottom of the dataset. To center the events at the hub, use 0.5 for both parameters. The structures are constant above and below the top and bottom of the dataset. See Figure 2 for a picture of how the structures are scaled. TurbSim automatically sets both values to 0.5 when the "KHTEST" option is used.

The last parameter in this section is used to determine where the first coherent structure will be placed in the time step file. TurbSim will make sure that the first event in the coherent time step file will not occur before the time entered here. This feature can be useful if you do not want a turbine to encounter coherent structures during the startup transient of a simulation.
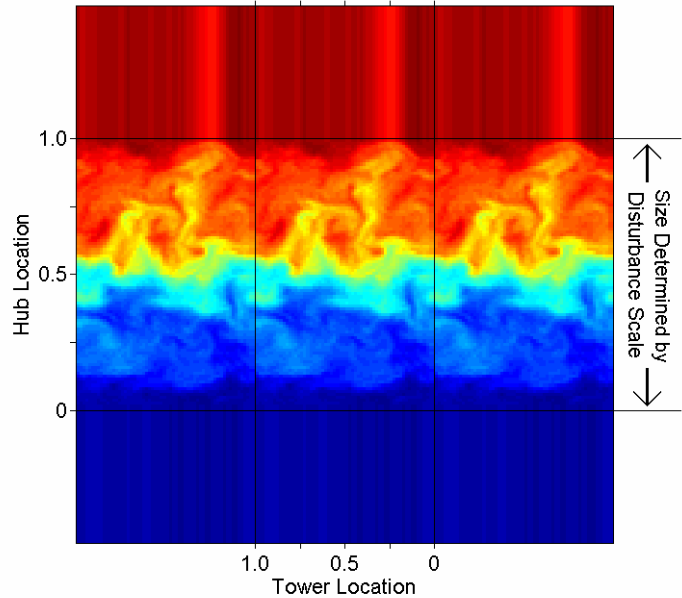
Figure 2. Scaling the coherent structures dataset.

Spectral Models

TurbSim uses a modified version of the Sandia method to generate time series based on spectral representation. Several different spectral models are available.

The IEC Spectral Models

TurbSim offers the Kaimal (IECKAI) and von Karman (IECVKM) spectral models as defined in the IEC standard. Please refer to "IEC 61400-1, Ed. 2: Wind Turbine Generator Systems - Part 1: Safety Requirements" [24] and "IEC 61400-1, Ed. 3: Wind Turbines - Part 1: Design Requirements" [21] for details on these models. Both of these models assume neutral atmospheric conditions.

The Risø Smooth-Terrain Model

TurbSim also offers the Risø smooth-terrain model (SMOOTH), based on work by Højstrup et al [25], [26]. This spectral model is more flexible in the atmospheric conditions it can model than the IEC spectral models are.

The NREL Wind Farm Models

The three NREL wind farm spectral models are based on measurements taken at a large wind farm in San Gorgonio Pass, California. A model based on measurements upwind of the wind park (WF_UPW), a model based on measurements taken at 7 rotor-diameter row-to-row spacing within the wind park (WF_07D), and a model based on measurements taken at 14 rotor-diameter row-to-row spacing within the wind park (WF_14D) are available. Please refer to work by Neil Kelley [6] for details on these wind farm models.

The NWTC Model

The NWTC model (NWTCUP) is a modified version of the NREL wind farm upwind model. The basic spectra in the two models are identical, but we have made some modifications to the cross-correlation-coefficients, the correlation matrix, the M-O z/L parameter, and u_* in order to reflect more accurately the atmospheric conditions observed at the NWTC [27]. We recommend that you always generate coherent structures when using the NWTCUP model.

The Great Plains Model

The Great Plains model (GP_LLJ) is based on measurements from the Lamar Low-Level Jet Project in southeastern Colorado. Please refer to Neil Kelley et al [28] for details of the LLLJP.

This is the only model that allows the user to specify a "JET" wind profile (see the section "Wind Profiles" in this report for more details on the jet wind profile). The user must then specify either a requested jet height or enter "default" to allow TurbSim to randomly pick a height, based on the Richardson number, M-O z/L parameter, and u_*. The reference height and wind speed inputs are used to specify the jet peak wind speed (intensity). For the "JET" model, the user may also enter "default" as the reference wind speed to allow TurbSim to randomly pick a wind speed, based on the Richardson number, M-O z/L parameter, and height of the jet.

If a "JET" wind profile is specified, a direction profile is also generated to simulate wind direction changes with height. This profile is a function of the jet height and maximum wind speed, the gradient Richardson number, and the friction velocity (u_*). Figure 3 gives an example of direction profiles for three different jets up to 300 m.

The coherent structure scaling for the Great Plains model is also different than the scaling for the other models. The GP_LLJ model incorporates scaling based on analysis of data from the LLLJP.

Output files

TurbSim can generate eight different sets of output files. They take the root name of the TurbSim input file and add extensions that indicate what type of files they are. See the "Runtime Options" section above to learn how to tell TurbSim which sets to output. TurbSim always generates a summary file.

Summary Files

TurbSim generates a summary file for all runs. It has a ".sum" extension. The first part of the file tells you what you specified in the input file. After that,
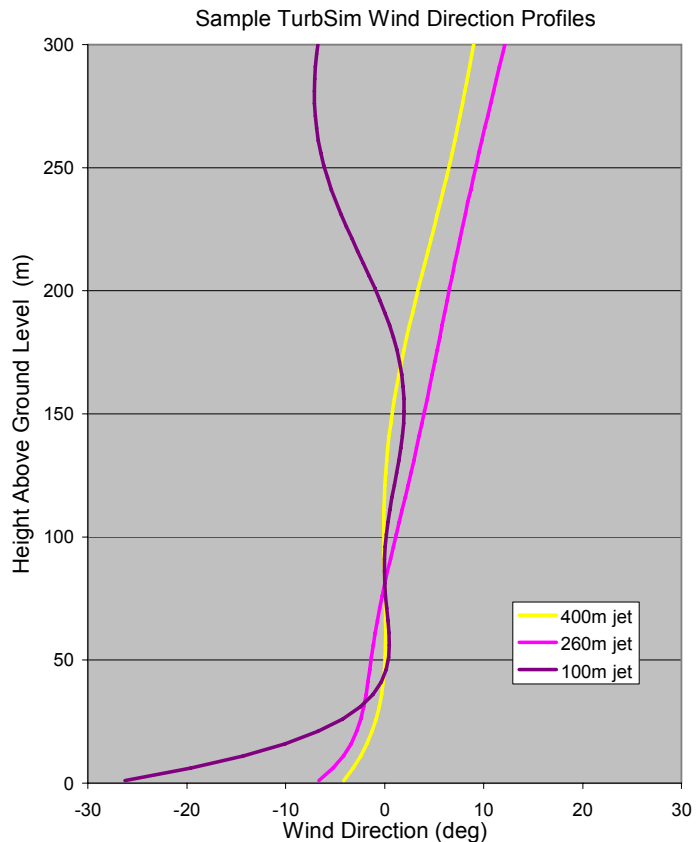

Figure 3. Example wind direction profiles with the GP_LLJ model.

TurbSim prints out many statistics for the run. These statistics are calculated using the entire analysis time so if you have requested a shorter usable time, the statistics of the output time series may be different than what is displayed in the summary file. Also keep in mind that the turbulence statistics are only for the background turbulence. They do not include effects of any coherent structures generated in coherent event time step files. If a coherent event time step file is generated, TurbSim prints the number of events and the total length of those events in the summary file. If you request *BLADED*-style FF files or separate tower output, TurbSim will add another section that tells AeroDyn how to convert the normalized data to floating-point form.

Hub Height Binary Files

These files are readable by GenPro, a postprocessor from the National Center for Atmospheric Research (NCAR), and are in machine-readable form. TurbSim gives them a ".bin" extension. At each time step, TurbSim writes the values of a series of parameters in the binary file. Each value is stored as a 4-byte floating-point (real) number. See Table 2 for the parameters stored in these files. A MATLAB® script for reading these files is included in the "Test" folder of the TurbSim archive; it is named "readHHbin.m".

Hub-Height Formatted Files

These files contain essentially the same information as the hub height binary files, but the parameters are written in columns in human-readable form. See Table 2 for the list of parameters. These files carry a ".dat" extension.

Hub-Height AeroDyn Formatted Files

These human-readable files are in a format compatible with AeroDyn. They carry the ".hh" extension. See Table 3 for the file format. TurbSim always sets the horizontal wind-shear, vertical wind-shear, and gust speed parameters to zero in the files. The vertical power-law parameter is the power law exponent. If input wind profile type is "JET" or "LOG," the power law exponent is calculated based on the mean wind speeds at the top and bottom of the rotor disk. Otherwise, it is the value of the input power law exponent.

Table 2. Format of Hub Height Binary Files.

Parameter	Description
Time	time
u	downwind wind speed
u_h	horizontal wind speed (u+v)
u_t	total wind speed (u+v+w)
v	sideways wind speed
w	vertical wind speed
u'	u fluctuating-velocity component
v'	v fluctuating-velocity component
w'	w fluctuating-velocity component
u'w'	u-w Reynolds stress component
u'v'	u-v Reynolds stress component
v'w'	v-w Reynolds stress component
TKE	Turbulent Kenetic Energy
CTKE	Coherent TKE

Full-Field TurbSim Binary Files

These are binary files designed to be read by AeroDyn. They have a ".bts" extension. TurbSim normalizes the data and encodes them in two-byte integers. The first part of the file is a header that provides information about the grid and tells AeroDyn how to convert the integers to floating-point values. The wind speeds for the grids and tower points (if specified) follow that. See "Appendix B: TurbSim Binary Full-Field File Format" in this report for the file format. A MATLAB® script for reading these files is included in the "Test" folder of the TurbSim archive; it is named "readTSgrid.m".

This binary format has been designed so that AeroDyn does not need to read any other file,

including the summary file, to properly convert the data to floating-point form. This format also provides the maximum resolution possible with this normalization method. *Please note that at the time of this writing, AeroDyn has not yet been updated to read these files; continue to use the FF BLADED-style binary ".wnd" files until AeroDyn v12.60, which will be able to read these files, is released.*

Table 3. Format of Hub-Height AeroDyn Files.

Time
Horizontal wind speed
Wind direction
Vertical wind speed
Horizontal wind-shear, linear parameter
Vertical power-law, wind-shear exponent
Vertical wind-shear, linear parameter
Gust speed (not sheared by AeroDyn)

Full-Field BLADED-Style Binary Files

These are binary files designed to be read by AeroDyn and *BLADED*. They have a ".wnd" extension. TurbSim normalizes the data and encodes them in two-byte integers. The first part of the file is a header that provides information about the grid and tells AeroDyn and *BLADED* how to convert the integers to floating-point values. The wind speeds for the grids follow that.

When generating these files, TurbSim adds a section to the end of the summary file that tells AeroDyn how to convert the data to floating-point form. AeroDyn needs to read both the summary file and the binary FF in order to decode the data. TurbSim uses a newer AeroDyn and *BLADED* binary file format than the format SNwind used. In general, this updated format keeps more resolution in the normalized two-byte integers than the previous encoding method did.

Tower Data Binary Files

These files are similar to the full-field *BLADED*-style binary files, except they contain data for points in a single line at the grid center, going from the bottom of the grid to the ground. These files have a ".twr" extension. TurbSim normalizes the data and encodes them in two-byte integers. The first part of the file is a header that provides information about the location of the tower points and size of the file followed by wind speeds. When generating these files, TurbSim adds a section to the end of the summary file that indicates how to convert the data to floating-point form. See "Appendix C: Tower Data Binary File Format" in this report for a more complete description of this binary format.

If you have requested FF binary files in TurbSim format, the tower points will be normalized and stored as two-byte integers along with the grid data in the FF binary file with a ".bts" extension. A separate file with the ".twr" extension will not be generated in that case.

Full-Field Formatted Files

These files are the traditional format of SNLWIND-3D FF output. These three files are human readable, but consume five times more storage than the binary files. They are no longer readable by AeroDyn. There is one file for each component. They have ".u," ".v," and ".w" extensions.

The tops of the files have headers with some basic information about the simulation. Blocks of data follow. The first line in each block has the time and the hub-height wind speed. Following that line is a table with the number of rows and columns being the number of grid points specified in the input file. The tables contain the wind speeds for the different grid points. Their orientation appears as if you are looking upwind (i.e. y increases from left to right and z increases from bottom to top). A MATLAB® script for reading these files is included in the "Test" folder of the TurbSim archive; it is named "loadFFtxt.m".

Coherent Turbulence Time Step Files

One of the unique features of TurbSim is its ability to add coherent turbulence events based on data obtained from numerical simulations of a Kelvin-Helmholtz billow. The data comes from two sources: a Large-Eddy Simulation (LES) from NCAR, and a Direct Numerical Simulation (DNS) from Colorado Research Associates (CoRA), both of Boulder, Colorado. Because the grid size of the coherent events is very large (roughly 92 x 92 points), we do not add these events directly to the background turbulence in TurbSim. Instead, we create a file with a ".cts" extension that contains the times and file numbers of the subset of LES or DNS data that define the coherent events. AeroDyn then reads this file along with the background wind file and adds the two wind fields together. As a result, this feature can be used only in programs that use AeroDyn v12.57 or later.

Wind Profiles

TurbSim offers the user a choice of profiles for the mean wind speed. Figure 4 shows five different mean wind profiles that were generated using default boundary conditions with the GP_LLJ model and a Richardson number of 0.076. The low-level jet wind profile is available only with the GP_LLJ spectral model. The "IEC" wind profile uses the power law profile over the (assumed) rotor disk and the logarithmic wind profile above and below the rotor disk.

Power Law Wind Profile

The power law wind profile is defined as

$$V(z) = V(z_{ref}) \left(\frac{z}{z_{ref}} \right)^{\alpha}$$

where z is the height above ground, $V(z)$ is the wind speed at height z, z_{ref} is a reference height above ground where the wind speed $V(z_{ref})$ is known, and α is the power law exponent.

Logarithmic Wind Profile

The diabatic (logarithmic) wind profile is defined as

$$V(z) = V(z_{ref}) \frac{\ln\left(z/z_0 \right) - \psi_m}{\ln\left(z_{ref}/z_0 \right) - \psi_m}$$

where z is the height above ground, $V(z)$ is the wind speed at height z, z_{ref} is a reference height above ground where the wind speed $V(z_{ref})$ is known, z_0 is the surface roughness length, and ψ_m is a stability function that is zero in neutral
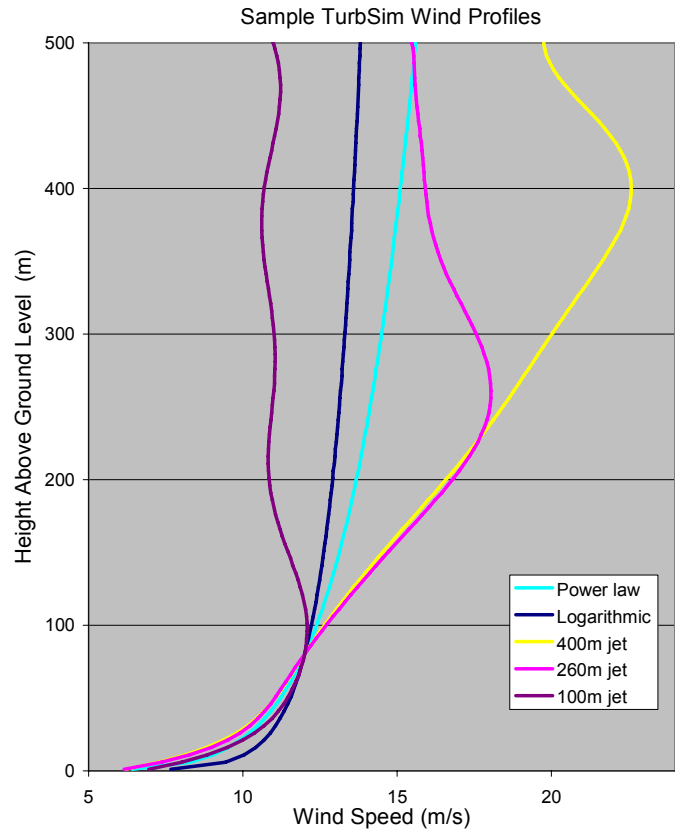

Figure 4. Example wind speed profiles generated in TurbSim with a 12 m/s wind speed at 80 m.

conditions (i.e. when the Richardson number is zero).

Low-Level Jet Wind Profile

The low-level jet wind speed profile is defined as

$$V(z) = \sum_{n=0}^{10} c_n \cdot T_n(z)$$

where z is the height above ground, $V(z)$ is the wind speed at height z, $T_n(z)$ is the nth order Chebyshev polynomial, and c_n is a Chebyshev coefficient. The Chebyshev coefficients are derived from LLLJP data and are based on the jet height and maximum wind speed, the gradient Richardson number, and the friction velocity.

A wind direction profile is also generated with the low-level jet wind speed profile (all of the other wind speed profiles have constant wind direction over the entire grid). The wind direction profile, like the wind speed profile, is a Chebyshev polynomial with coefficients derived from the same parameters in the LLLJP data. See the Great Plains Model (GP_LLJ) in the "Spectral Models" section of this report for more information.

Spatial Coherence Models

The spatial coherence functions that are implemented in TurbSim are defined below.

Coherence for IEC Spectral Models

The root coherence function for the u-component of the IEC spectral models is defined as

$$Coh_{j,k} = \exp\left(-b \sqrt{\left(\frac{f\,r}{U_{Hub}} \right)^2 + \left(0.12 \frac{r}{L_c} \right)^2} \right)$$

where f is the frequency, r is the distance between points j and k, b is the coherence decrement (either 12 or 8.8, depending on the IEC 61400-1 edition number), U_{Hub} is the hub-height wind speed, and L_c is the IEC 61400-1 length scale.

The coherence function for the v- and w-components of the IEC spectral models is

$$Coh_{j,k} = \begin{cases} 1 & j = k \\ 0 & j \neq k \end{cases}$$

Coherence for Non-IEC Spectral Models

The root coherence function for all three of the wind components for non-IEC spectral models is defined as

$$Coh_{j,k} = \exp\left(-0.5 b \left(\frac{r}{z_m} \right)^\alpha \frac{f\,r}{U_m} \right)$$

where b is the input coherence decrement (it may be different for each wind component), r is the distance between points j and k, z_m is the mean height of the two points, α is the input coherence exponent, f is the frequency, and U_m is the mean wind speed of the two points. Please note that the 0.5 multiplier is a result of the input coherence decrements being based on the *coherence-squared* function.

Using Coherent Turbulence Time Step Files with AeroDyn

To use the coherent time step files that TurbSim generates, you must create a coherent turbulence parameter input file for AeroDyn (or modify the example file included in the archive). This file must have a ".ctp" extension. Enter the name of this file on the WindFile parameter line in AeroDyn v12.57 or later. As with TurbSim, do not add or delete lines from the sample input file because AeroDyn assumes parameters are on specific lines. See "Appendix D: Sample AeroDyn Coherent Turbulence Parameter Input File" in this report for an example of one of these input files.

Parameters That Can Vary From One Simulation to Another

The first parameter is the name of the path that contains the binary data files, which you must get from the coherent structure archive on the TurbSim website (in folder x90_i16). This directory must contain files called "Scales.les" and "Scales.dns" which contain scaling parameters for the two event types, and are used to read and convert 16-bit integer binary data to normalized real numbers. There should also be three folders in this directory, named "*u*", "*v*", and "*w*" respectively, containing data for the three wind components. Each of these three directories contains files named something like "u_i16_*xxxxx*.les".

The parameter CTTSfile should be the name of the coherent time step file generated by TurbSim. It will have a ".cts" extension.

The parameter CTbackgr is the name of the background turbulence file. This should be the FF wind file with the ".wnd" or ".bts" extension that was generated at the same time the ".cts" file was created. AeroDyn will also look for the summary file that goes with a ".wnd" wind file.

The last two parameters in this section are used for decimating the coherent turbulence data. Enter the decimation factors in the y and z directions. Enter "1" on both lines to use the entire grid. *Since this feature has not been sufficiently tested yet, we recommend that you always use the entire grid.*

Suggestions for Generating Coherent Turbulent Structures

Every effort has been made in TurbSim to randomize the occurrence and scaling of coherent event structures that occur in natural, nocturnal boundary layer flows. Simulations that generate coherent turbulence time step files have up to twelve degrees of stochastic freedom and are designed to give some feel of the expected variability in the atmosphere. Because of the degree of variability, we recommend using a minimum of 31 different random seeds for a specific set of boundary conditions.

If you wish to test the effects of a coherent structure (KH billow), we recommend using the "KHTEST" option in the turbulence characteristic input parameter with the NWTCUP spectral model. This test function superimposes one intense coherent event in the middle of the output time series, reducing the number of stochastic degrees of freedom to no more than four. The gradient Richardson number and wind shear of the background flow will be overwritten, and TurbSim will use fixed values to scale the LES-type event. This test function is designed to generate intense turbulence, and does not necessarily reflect the variability for given boundary conditions.

The choice of the gradient Richardson number and hub wind speed largely control the impact of coherent structures on turbine response. We recommend making at least one series of runs at rated wind speed and a Richardson number between 0.02 and 0.05. Further discussion on the impact of coherent turbulent structures on wind turbines is found in [29].

Warnings

- You must use AeroDyn v12.57 or later in order to read TurbSim files correctly.

- If you compile AeroDyn, you must use the compiler option "/assume:byterecl" to read the TurbSim coherent structures binary files correctly. If you use ADAMS2AD [30], be sure to use v12.17 or later so that this compiler option is set.

- HH wind files and FF wind files do not have events happening at the same time because AeroDyn shifts the FF files.

- Because of the way the FFT routine works, we must add extra time to the analysis time to get the FFT to run efficiently. Because of this and the fact that the output time may be less than the analysis time, the mean wind speed for the portion of the run you will actually use may be different from what you specified in the input file.

- The statistics calculated in the summary file are based on the full time series generated (the analysis time plus any extra time added for the FFT calculations). Because the output time may be less than the analysis time, these statistics may be different than what can be calculated from the output files.

- TurbSim occasionally cannot perform the spectral matrix factorizations. If this occurs and you believe all of the input values are correct, try modifying the coherence matrix by changing the grid point locations, wind speed, coherence decrements, and/or coherence exponent.

- You should use caution when using mean flow angle inputs with full-field grids for AeroDyn. AeroDyn marches FF grids through the turbine in the x-direction at the mean hub-height wind speed, regardless of the flow angles. This can possibly give strange results if the mean flow angles are not small.

Limitations

- The GP_LLJ spectral model is estimated to be applicable up to a height of 230 meters.

- The SMOOTH spectral model and the coherent turbulence time step files are both estimated to be applicable up to a height of 120 meters at this time.

- The NWTCUP spectral model is estimated to be applicable up to heights of 100-120 m.

- The wind farm spectral models (WF_UPW, WF_07D, and WF_14D) and the NWTCUP model are valid up to heights of only about 50 meters.

- We have not been able to generate a grid larger than 19 x 20 points for a 10-minute run of 20 Hz data on a 32-bit machine. A larger grid requires more than 2GB of virtual memory, which Windows NT does not permit. Larger grids can be generated on a 64-bit machine.

Possible Future Enhancements

- Update the NTWCUP spectral model with spectra derived from LIST data.

- Add the ability to scale the IEC models so that the hub-height statistics of the output portion of the time series are equivalent to the requested statistics.

Known Bugs

- None.

Caveats

NREL makes no promises about the usability or accuracy of TurbSim, which is essentially a beta code. NREL does not have the resources to provide full support for this program.

Acknowledgements

Feedback

If you have problems with TurbSim, please contact Bonnie Jonkman, Neil Kelley, or Marshall Buhl. If we have time to respond to your needs, we will do so, but please do not expect an immediate response. Send your comments or bug reports to:

Bonnie J. Jonkman
NWTC/3811
National Renewable Energy Laboratory
1617 Cole Blvd
Golden, CO 80401-3393
United States of America
E-mail: bonnie_jonkman@nrel.gov
Phone: +1 (303) 384-6907
Fax: +1 (303) 384-6901

Marshall L. Buhl Jr.
NWTC/3811
National Renewable Energy Laboratory
1617 Cole Blvd
Golden, CO 80401-3393
United States of America
Web Site: wind.nrel.gov/designcodes/
E-mail: marshall_buhl@nrel.gov
Phone: +1 (303) 384-6914
Fax: +1 (303) 384-6901

References

[1] Laino, David J. and Hansen, A.C. (December 2002) "User's Guide to the Wind Turbine Dynamics Aerodynamics Computer Software AeroDyn." Windward Engineering, LC, Salt Lake City, UT. Available from <http://wind.nrel.gov/designcodes/simulators/aerodyn/>.

[2] Jonkman, J. M. and Buhl Jr., M. L. (August 2005) "FAST User's Guide." NREL/EL-500-29798. National Renewable Energy Laboratory, Golden, CO. Available from <http://wind.nrel.gov/designcodes/simulators/fast/>.

[3] Laino, David J. and Hansen, A.C. (January 2003) "User's Guide to the Wind Turbine Dynamics Computer Program YawDyn." Windward Engineering, LC, Salt Lake City, UT. Available from <http://wind.nrel.gov/designcodes/simulators/yawdyn/>.

[4] (2005) *MSC.ADAMS®* . MSC.Software Corporation, Santa Ana, CA. <http://www.mscsoftware.com/>

[5] Veers, P.S. (March 1988) "Three-Dimensional Wind Simulation." SAND88-0152. Sandia National Laboratories, Albuquerque, NM.

[6] Kelley, N.D. (November 1992) "Full Vector (3-D) Inflow Simulation in Natural and Wind Farm Environments Using an Expanded Version of the SNLWIND (VEERS) Turbulence Code." NREL/TP-442-5225. National Renewable Energy Laboratory, Golden, CO.

[7] Bossanyi, E.A. (2003) "GH Bladed Version 3.6 User Manual." Document 282/BR/010 Issue 12. Garrad Hassan and Partners Limited.

[8] Buhl, M.L. Jr. (October 2001) "SNwind User's Guide." NREL/EL-500-30121. National Renewable Energy Laboratory, Golden, CO.

[9] Kelley, N.D. and Jonkman, B.J. (April 2006) "Overview of the TurbSim Stochastic Inflow Turbulence Simulator." NREL/EL-500-39796. National Renewable Energy Laboratory, Golden, CO.

[10] Buhl, Jr. M.L. (December 2000) "Installing NWTC Design Codes on PCs Running Windows NT®." NREL/EL-500-29384. National Renewable Energy Laboratory, Golden, CO.

[11] (2006) *MATLAB®.* The MathWorks, Inc., Natick, MA. <http://www.mathworks.com/>

[12] Swarztrauber, P.N. (November 1988) "FFTPACK version 4.1." National Center for Atmospheric Research, Boulder, CO. Available from <http://www.scd.ucar.edu/softlib/FFTPACK.html>.

[13] Anderson, E.; Bai, Z.; Bischof, C.; Blackford, S.; Demmel, J.; Dongarra, J.; Du Croz, J.; Greenbaum, A.; Hammarling, S.; McKenney, A.; and Sorensen, D. (August 1999) "LAPACK User's Guide, Third Edition." Society for Industrial and Applied Mathematics, Philadelphia, PA. Available from <http://www.netlib.org/lapack>.

[14] Lawson, C.L.; Hanson, R.J.; Kincaid, D.; and Krogh, F.T. (September 1979) "Basic Linear Algebra Subprograms for FORTRAN usage." *ACM Trans. Math. Soft.*, Vol. 5, pp. 308-323.

[15] Dongarra, J.J.; Du Croz, J.; Hammarling, S.; and Hanson, R.J. (March 1988) "An extended set of FORTRAN Basic Linear Algebra Subprograms." *ACM Trans. Math. Soft.*, Vol. 14, pp. 1-17.

[16] Meissner, Loren P. (August 1995) "ranlux.f90." Available from <http://flash.uchicago.edu/~tomek/htmls/refs/ranlux.f90.html>.

[17] Buhl, M.L., Jr. (February 2004) "NWTC Design Codes (NWTC Subroutine Library)." Available from <http://wind.nrel.gov/designcodes/miscellaneous/nwtc_subs>. Last modified 22-Feb-2006; accessed 22-Feb-2006.

[18] L'ecuyer, P. (June 1988) "Efficient and Portable Combined Random Number Generators." *Communications of the ACM*, Vol. 31.

[19] Lüscher, M. (February 1994) "A portable high-quality random number generator for lattice field theory simulations." *Computer Physics Communications,* Vol. 79.

[20] James, F. (February 1994) "RANLUX: A Fortran implementation of the high-quality pseudorandom number generator of Lüscher." *Computer Physics Communications,* Vol. 79.

[21] IEC 61400-1 (August 2005) "Wind turbines-Part 1: Design requirements." 3rd edition. International Electrotechnical Commission.

[22] IEC 61400-2 (March 2006) "Wind turbines-Part 2: Design requirements for small wind turbines." 2nd edition. International Electrotechnical Commission.

[23] IEC 61400-3 (Draft January 2006) "Wind turbines-Part 3: Design requirements for offshore wind turbines." 1st edition. International Electrotechnical Commission.

[24] IEC 61400-1 (1999) "Wind turbine generator systems-Part 1: Safety requirements." 2nd edition. International Electrotechnical Commission.

[25] Højstrup, J. (October 1982) "Velocity Spectra in the Unstable Planetary Boundary Layer." *Journal of the Atmospheric Sciences*, Vol. 39.

[26] Olesen, H.R.; Larsen, S.E.; and Højstrup, J. (July 1984) "Modeling Velocity Spectra in the Lower Part of the Planetary Boundary Layer." *Boundary-Layer Meteorology,* Vol. 29.

[27] Kelley, N.; Hand, M.; Larwood, S.; and McKenna, E. (January 2002) "The NREL Large-Scale Turbine Inflow and Response Experiment – Preliminary Results." NREL/CP-500-30917. National Renewable Energy Laboratory, Golden, CO.

[28] Kelley, N.D.; Shirazi, M.; Jager, D.; Wilde, S.; Adams, J.; Buhl, M.; Sullivan, P.; and Patton, E. (January 2004) "Lamar Low-Level Jet Project Interim Report." NREL/TP-500-34593. National Renewable Energy Laboratory, Golden, CO.

[29] Kelley, N.D.; Jonkman, B.J.; Scott, G.N.; Bialasiewicz, J.T.; and Redmond, L.S. (August 2005) "The Impact of Coherent Turbulence on Wind Turbine Aeroelastic Response and Its Simulation." NREL/CP-500-38074, National Renewable Energy Laboratory, Golden, CO.

[30] Laino, D.J.; and Hansen, A.C. (September 2001) "User's Guide to the Computer Software Routines AeroDyn Interface for ADAMS®." Windward Engineering, LC, Salt Lake City, Utah. Available at <http://wind.nrel.gov/designcodes/simulators/adams2ad/>.

Appendix A: Sample TurbSim Input File

```
TurbSim Input File. Valid for TurbSim v1.10, 10-Apr-2006


---------Runtime Options-----------------------------------
2318573    First random seed (-2147483648 to 2147483647)
RANLUX     Second random seed for intrinsic pRNG, or other pRNG: "RanLux" or "RNSNLW"
False      Output HH turbulence parameters in GenPro-binary form?  (Generates RootName.bin)
False      Output HH turbulence parameters in formatted form?  (Generates RootName.dat)
False      Output hub-height time-series data in AeroDyn form?  (Generates RootName.hh)
False      Output FF time-series data in TurbSim/AeroDyn form? (Generates Rootname.bts)
True       Output FF time-series data in BLADED/AeroDyn form?  (Generates RootName.wnd)
False      Output tower time-series data? (Generates RootName.twr)
False      Output FF time-series data in formatted (readable) form?  (RootName.u, .v, .w)
True       Output coherent turbulence time steps in AeroDyn form? (Generates RootName.cts)
True       Clockwise rotation looking downwind? (Used only for FF binary files w/ BLADED)


--------Turbine/Model Specifications----------------------
13         Vertical grid-point matrix dimension
13         Horizontal grid-point matrix dimension
0.05       Time step [seconds]
600        Length of analysis time series [seconds]
40         Usable length of output time series [seconds] (program adds GridWidth/MeanHHWS sec)
84.30      Hub height [m] (should be > 0.5*GridHeight)
80.00      Grid height [m]
80.00      Grid width [m] (should be >= 2*(RotorRadius+ShaftLength))
0          Vertical mean flow (uptilt) angle [degrees]
0          Horizontal mean flow (skew) angle [degrees]


--------Meteorological Boundary Conditions-------------------
"SMOOTH"   Turbulence model (IECKAI, IECVKM, GP_LLJ, NWTCUP, SMOOTH, WF_UPW, WF_07D, WF_14D)
"1-ED2"    Number of the IEC standard (61400-x, x=1,2,3) with optional 61400-1 edition number
"A"        IEC turbulence characteristic ("A", "B", "C" or TI in %) or KHTEST
"NTM"      IEC turbulence type ("NTM", "xETM", "xEWM1", or "xEWM50" for x=class 1, 2, or 3)
"IEC"      Wind profile type ("JET"=Low-level jet, "LOG", "PL"=power law, "IEC", or "default")
84.30      Height of the reference wind speed [m]
18.2       Mean wind speed at the reference height [m/s]
450        Height of the low-level jet [m] (70-490 m or "default", used only for "JET" profile)
default    Power law exponent (or "default")
default    Surface roughness length [m] (or "default")


--------Non-IEC Meteorological Boundary Conditions------------
default    Site latitude [degrees] (or "default")
0.05       Gradient Richardson number
default    Friction or shear velocity [m/s] (or "default")
default    Mixing layer depth [m] (or "default")
default    U'W' cross-correlation coefficient (or "default")
default    U'V' cross-correlation coefficient (or "default")
default    V'W' cross-correlation coefficient (or "default")
default    U-component coherence decrement (or "default")
default    V-component coherence decrement (or "default")
default    W-component coherence decrement (or "default")
default    Coherence exponent (or "default")


--------Coherent Turbulence Scaling Parameters-------------------
"M:\coh_events\eventdata" Name of the path where event data files are located
"Random"   Type of event files ("LES", "DNS", or "RANDOM")
True       Randomize the disturbance scale and locations? (true/false)
 1.0       Disturbance scale (ratio of wave height to rotor disk).
 0.5       Fractional location of tower center from right to left of dataset looking downwind
 0.5       Fractional location of hub height from the bottom of the dataset.
30.0       Minimum start time for coherent structures in RootName.cts [seconds]
```

Figure A-1. Sample TurbSim input file

Appendix B: TurbSim Binary Full-Field File Format

Table B-1. TurbSim FF binary file header format

Type (Bytes)	Parameter	Description
Integer (2)	ID	Identifies the file as a TurbSim binary file. ID should have the value 7.
Integer (4)	n_z	The number of grid points in the vertical direction.
Integer (4)	n_y	The number of grid points in the horizontal direction.
Integer (4)	n_{tower}	The number of tower points below the grid.
Integer (4)	nt	The number of time steps.
Real (4)	dz	The distance in meters between two adjacent points in the vertical direction.
Real (4)	dy	The distance in meters between two adjacent points in the horizontal direction.
Real (4)	dt	The time in seconds between consecutive grids.
Real (4)	U_{hub}	The mean wind speed in m/s at hub height.
Real (4)	Z_{hub}	The height in meters of the hub.
Real (4)	Z_{bottom}	The height in meters of the bottom of the grid.
i = 1, 2, 3 Real (4)	$V_{slope}(i)$	The slope used to scale the ith-component[1] from 4-byte reals into 2-byte integers.
Real (4)	$V_{intercept}(i)$	The intercept used to scale the ith-component[1] from 4-byte reals into 2-byte integers.
Integer (4)	$n_{characters}$	The number of characters in the ASCII string that gives the TurbSim version number, date, and time the file was generated. This number is no larger than 200.
i = 1, 2, … $n_{characters}$ Integer (1)	$Character_i$	The ASCII integer representation of the ith character of the string that gives the TurbSim version number, date, and time the file was generated. ACHAR($Character_i$) gives the character.

---

[1] The three wind components are defined as *u = 1, v = 2,* and *w = 3.*

Table B-2. TurbSim FF binary file grid format

	Type (Bytes)	Parameter	Description
it = 1, 2, … n_t *iz = 1, 2, … n_z* *iy = 1, 2, … n_y* *i = 1, 2, 3*	Integer (2)	$V_{grid\_norm}(i,iy,iz,it)$	The normalized i^{th}-component[2] of the wind speed at time step *it* and grid location *(y(iy), z(iz))*. To convert to m/s, use $$V_{grid}(i,iy,iz,it)=\frac{V_{grid\_norm}-V_{intercept}(i)}{V_{slope}(i)}$$ $$Y_{grid}(iy)=-0.5(ny-1)dy+(iy-1)dy$$ $$Z_{grid}(iz)=z_{bottom}+(iz-1)dz$$.
iz = 1, 2, … n_{tower} *i = 1, 2, 3*	Integer (2)	$V_{tower\_norm}(i,iz,it)$	The normalized i^{th}-component[2] of the wind speed at time step *it* and tower height $z_{tower}(iz)$. To convert to m/s, use $$V_{tower}(i,iz,it)=\frac{V_{tower\_norm}-V_{intercept}(i)}{V_{slope}(i)}$$ $$Y_{tower}=0$$ $$Z_{tower}(iz)=z_{bottom}-(iz-1)dz$$.

---

[2] The three wind components are defined as *u = 1, v = 2,* and *w = 3.*

Appendix C: Tower Data Binary File Format

TurbSim tower files have a ".twr" extension. Each file contains a header of 4-byte real and integer values, followed by 2-byte integer time series of the three wind components at each point on the tower grid. The wind components at the tower points are normalized and stored in 2-byte binary integers, exactly the same way AeroDyn/*BLADED* full-field wind files are written. The tower files have the same vertical resolution as the full-field grid, with points going downward from the bottom of the full grid in a single line at the tower centerline.

Table C-1: TurbSim tower data binary file header format

Type (Bytes)	Parameter	Description
Real (4)	GridRes_Z	Vertical grid resolution, in m
Real (4)	UHub *TimeStep	Longitudinal grid resolution, in m
Real (4)	Zmax	The height of the highest tower point, in m
Real (4)	NumOutSteps	The number of points in the longitudinal direction
Real (4)	NumZ	The number of vertical grid points
Real (4)	U_{hub}	The mean wind speed in m/s
Real (4)	100 * TI(U)	The turbulence intensity of the *u*-component, in percent
Real (4)	100 * TI(V)	The turbulence intensity of the *v*-component, in percent
Real (4)	100 * TI(W)	The turbulence intensity of the *w*-component, in percent

Table C-2: TurbSim tower data binary file grid format

For each increasing time step (NumOutSteps points),starting at the top of the grid, going downward (NumZ points) the data are stored as:

Type (Bytes)	Parameter	Description
Integer (2)	$\frac{1000}{TI(U)}\left(\frac{U}{U_{hub}}-1\right)$	Normalized *u*-component of the wind speed
Integer (2)	$\frac{1000}{TI(V)}\left(\frac{V}{U_{hub}}\right)$	Normalized *v*-component of the wind speed
Integer (2)	$\frac{1000}{TI(W)}\left(\frac{W}{U_{hub}}\right)$	Normalized *w*-component of the wind speed

Appendix D: Sample AeroDyn Coherent Turbulence Parameter Input File

```
Example Coherent Turbulence Parameter input file (TurbSim_AD.ctp).  Valid with AeroDyn 12.57.
# Parameters that can vary from one turbine simulation to the next:
 "H:\x90_i16"  | CTSpath  - Path to coherent turbulence data files
 "TurbSim.cts" | CTTSfile - File containing time steps of the coherent turbulence event files
 "TurbSim.wnd" | CTbackgr - Name of file containing background wind data (quoted string)
   1           | CT_DF_Y  - Decimation factor for wind data in the y direction
   1           | CT_DF_Z  - Decimation factor for wind data in the z direction

==================================================
NOTE: Do not add or remove any lines in this file!
==================================================
For decimation factors, 1 = use every point, 2 = use every other point, etc.
```

Figure D-1. Sample AeroDyn coherent turbulence parameter input file

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
September 2006	Technical Report	

4. TITLE AND SUBTITLE
TurbSim User's Guide

5a. CONTRACT NUMBER
DE-AC36-99-GO10337

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
B.J. Jonkman and M.L. Buhl, Jr.

5d. PROJECT NUMBER
NREL/TP-500-39797

5e. TASK NUMBER
WER6.2104

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
National Renewable Energy Laboratory
1617 Cole Blvd.
Golden, CO 80401-3393

8. PERFORMING ORGANIZATION REPORT NUMBER
NREL/TP-500-39797

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)
NREL

11. SPONSORING/MONITORING AGENCY REPORT NUMBER

12. DISTRIBUTION AVAILABILITY STATEMENT
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161

13. SUPPLEMENTARY NOTES

14. ABSTRACT *(Maximum 200 Words)*
The TurbSim stochastic inflow turbulence code was developed to provide a numerical simulation of a full-field flow that contains coherent turbulence structures that reflect the proper spatiotemporal turbulent velocity field relationships seen in instabilities associated with nocturnal boundary layer flows. This report provides a user's guide for the TurbSim code.

15. SUBJECT TERMS
wind turbine design code simulations; inflow turbulence code; TurbSim

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE	UL		
Unclassified	Unclassified	Unclassified			19b. TELEPONE NUMBER *(Include area code)*

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18