

COMPUTATIONAL FLUID DYNAMICS MODELING  
OF ATMOSPHERIC FLOW  
APPLIED TO WIND ENERGY RESEARCH

by

Alan Russell

A thesis  
submitted in partial fulfillment  
of the requirements for the degree of  
Master of Science in Mechanical Engineering  
Boise State University

December 2009

BOISE STATE UNIVERSITY GRADUATE COLLEGE

**DEFENSE COMMITTEE AND FINAL READING APPROVALS**

of the thesis submitted by

Alan Russell

Thesis Title: Computational Fluid Dynamics Modeling of Atmospheric Flow Applied to Wind Energy Research

Date of Final Oral Examination: 15 October, 2009

The following individuals read and discussed the thesis submitted by student Alan Russell, and they also evaluated his presentation and response to questions during the final oral examination. They found that the student passed the final oral examination, and that the thesis was satisfactory for a master's degree and ready for any final modifications that they explicitly required.

Paul J. Dawson, Ph.D.

Chair, Supervisory Committee

Inanc Senocak, Ph.D.

Member, Supervisory Committee

John R. Gardner, Ph.D.

Member, Supervisory Committee

The final reading approval of the thesis was granted by Paul J. Dawson, Ph.D., Chair of the Supervisory Committee. The thesis was approved for the Graduate College by John R. Pelton, Ph.D., Dean of the Graduate College.

## ACKNOWLEDGMENTS

I have many people to thank for supporting me during my graduate studies at Boise State. First, thank you to my wife Roberta and daughter Adrienne who put up with many long hours of my working on a computer in the dining room. Now that I'm finally graduating, we won't have to schedule our lives around the academic calendar, classes, meetings, etc. Adrienne (age 4) likes to look at windmills because she knows I like them, and I'm sure she enjoyed our drive to the top of Cinder Cone Butte at age 2! Thanks to Dr. Paul Dawson for pointing me in the direction of wind energy, atmospheric science and computational fluid dynamics. Dr. Dawson started me on some great projects, helped me write technical papers for the AWEA WindPower conferences in 2008 and 2009, and made my research project part of the wind forecasting BPA grant. Dr. Inanc Senocak provided excellent direction and advice whenever I asked. Todd Haynes and Kevin Nuss are excellent project partners: we came up with much better projects as a team than we could have done individually.

I would like to thank my managers at HP for allowing me to take time from work to attend classes and work on school related projects. Thanks to the HP educational assistance program for funding most of my classes. Co-workers Alan Williamson and Brandon Chaffin were great study partners. Thanks to my good friend Chago Rodriguez

for interest in my projects, advice with statistics, encouragement, and for dragging me away from the computer and out on ski trips and bike rides when I really needed a break.

Finally, I would like to thank the faculty and staff at the Boise State College of Engineering for providing a great environment to learn, and a great undergraduate and graduate education. When I was an undergraduate student, the mechanical engineering faculty - Dr. Guarino, Dr. Tennyson, Dr. Dawson, Dr. Ferguson, Dr. Eggert, Dr. Parks and the Dean, Dr. Lynn Russell - worked hard to make sure we received the best education they could provide and that the college of engineering accreditation was complete before we graduated. The time and energy they spent teaching the Boise State mechanical engineering class of 2000, all eight of us, was something I will always remember. It was a great learning environment: Dr. Gardner and the other newer faculty members blended well into the environment and kept the energy.

Back in about 1994, I was working the night shift at Micron and decided to attend the college fair in the cafeteria. There were representatives from three or four colleges and universities in Utah, U of I and a few others, all looking like typical college faculty and professional staff. At the Boise State table was a guy with longish graying hair, an old BSU sweatshirt and jeans that might not pass the Micron employee dress code. I was working as an electronics technician, not particularly interested in Boise State, and wanted to talk about electrical engineering. He was very engaging and spent his time

telling me about mechanical engineering and all the great things that were happening at Boise State. Fifteen years later, I still claim the right to blame Dr. Joe Guarino for getting me into this mess - and I'm very glad he did.

## AUTOBIOGRAPHICAL SKETCH

A Masters of Science in Mechanical Engineering will probably be my last college degree. I currently have three degrees: BA in Music (Violin Performance) from the University of California Santa Barbara, 1981; MA in Music from Boston University, 1984 and a BS in Mechanical Engineering from Boise State University in 2000. I was a full time Junior College student at Palomar College (San Marcos, CA) before attending UCSB; I spent a year in graduate school at U.C. San Diego before moving on to Boston University, and have taken classes from Santa Monica College and the University of Idaho. Having been a college student for nearly 2/3 of my adult life, including the last eight years working on this degree, I think it's about time I considered doing something else with my spare time.

I'm married and have a 4 year old daughter. After getting my engineering degree, I worked for Jabil Circuit for a year, and have worked for the Hewlett-Packard Company for the last 8 ½ years. I am currently an engineer in the HP color LaserJet supplies R&D lab. I have enjoyed working on wind energy research projects at Boise State. I like the academic rigor and the expectation from everyone that the science and research will be done correctly. Academic standards are not always maintained in the work environment. My wind energy research projects have included two papers presented at the American Wind Energy Association WindPower conferences in 2008 with co-authors Dr. Paul

Dawson and Todd Haynes, and 2009 with co-authors Dr. Paul Dawson, Todd Haynes and Kevin Nuss (Russell et al., 2008; Russell et al., 2009).

I enjoy mountain biking, road biking, camping, and hiking in spring summer and fall, and alpine, backcountry and skate skiing in the winter. I am from southern California, but have enjoyed living in Idaho for the last 18 years.

## ABSTRACT

High resolution atmospheric flow modeling using computational fluid dynamics (CFD) has many applications in the wind energy industry. A well designed model can accurately calculate wind speed, direction, and turbulence at any point in a wind farm using data from a fixed location source. The model can extend point source data over an area of several square kilometers, or map terrain influenced microclimates using remote wind data. A local flow model is critical for wind resource site assessment, and for optimizing wind farm turbine layout for maximum power production. A CFD simulation of an operating wind farm, coupled with a local wind forecast, can increase accuracy of electric power generation forecasts, providing valuable information to electric grid managers and wind farm operators.

Computational Fluid Dynamics (CFD) models solve the governing equations of fluid dynamics, providing a mathematical solution that describes turbulent fluid flow. Atmospheric CFD models are ideal for flows over complex terrain and they can simulate both shear and convective turbulence. Reliable CFD solutions require knowledge of atmospheric science, fluid dynamics and numerical solutions in addition to the CFD software. Reliable CFD models also require validation with recorded wind data. Because of the complexity of CFD, a methodology is needed for generating consistent models for a variety of locations and climates. This methodology establishes processes

for importing surface map data, meshing, setting boundary conditions, running the model and analyzing results. In this thesis, mathematical theory is discussed along with methods for generating and meshing surfaces, handling wind data and validating results. This thesis describes model simulation over two locations and compares the results with published studies, and with available wind data. Parts of this research have been presented at the American Wind Energy Association WindPower conferences in 2008 and 2009. The CFD model project is also part of a Bonneville Power Administration sponsored Wind Energy Forecasting grant project under investigation by the Boise State University College of Engineering.

## TABLE OF CONTENTS

ACKNOWLEDGEMENTS.....	iv
AUTOBIOGRAPICAL SKETCH.....	vii
ABSTRACT.....	ix
LIST OF TABLES.....	xiv
LIST OF FIGURES.....	xv
LIST OF EQUATIONS.....	xvii
INTRODUCTION.....	1
CFD History and Definition.....	1
Wind Data and Resource Assessment.....	3
CFD for Wind Energy Forecasting.....	5
Context: The Scale of CFD Modeling.....	5
Objective.....	6
ATMOSPHERIC BOUNDARY LAYER.....	8
Surface Layer.....	9
Boundary Layer Stability.....	11
Surface Layer Modeling.....	12
TURBULENCE MODELING.....	15
Turbulence Closure Methods.....	17
Closure Order.....	17

Atmospheric Turbulence Modeling .....	19
Turbulence Modeling Terminology .....	21
BUOYANCY EFFECTS IN ATMOSPHERIC FLOW MODELING .....	23
Model Parameters .....	25
CFD MODEL EXPERIMENTS AND RESEARCH .....	26
Mesh Considerations.....	26
First Cell Height.....	27
Atmospheric Model First Cell Height.....	28
GAMBIT/FLUENT Y <sup>+</sup> Experiments .....	30
FLUENT Buoyancy Model Experiment.....	30
Atmospheric Roughness .....	33
PRACTICAL CONSIDERATIONS IN CFD MODELING .....	36
Surface Model and Meshing Process.....	36
Wind Data .....	43
Public Wind Data Sources .....	47
Processing Wind Data.....	47
Inlet Boundary Conditions.....	50
Inlet Turbulence .....	51
APPLICATIONS OF THE CFD MODEL .....	53
Cinder Cone Butte.....	53
Cinder Cone Butte Project Results .....	60

BPA Wind Forecasting Grant Project.....	61
CFD Model Changes.....	62
Completing the Forecasting Project.....	66
Future Study.....	67
Conclusions.....	68
REFERENCES .....	69
APPENDIX A.....	72
GAMBIT Mesh Process.....	72
APPENDIX B .....	81
Fluent Solution Steps .....	81
APPENDIX C .....	86
FLUENT Inlet Profile Examples .....	86
APPENDIX D.....	91
Life After FLUENT .....	91
APPENDIX E .....	95
AWEA WindPower Posters.....	95

## LIST OF TABLES

Table 1. k- $\epsilon$ turbulence model constants.....	21
Table 2. Roughness height and constant.....	34

LIST OF FIGURES

Figure 1. Components of the fair weather boundary layer ..... 8

Figure 2. Turbulence generation in the surface layer ..... 10

Figure 3. Laminar and turbulent velocity profiles in boundary layer flow..... 13

Figure 4. Rough wall and atmospheric boundary layer schematics..... 14

Figure 5. Experimental data showing diurnal heat flux patterns ..... 23

Figure 6. Seasonal changes in the boundary layer ..... 24

Figure 7. Law of the wall, velocity distribution plot ..... 27

Figure 8. Atmospheric surface layer with mesh cross section..... 29

Figure 9. Neutral flow velocity vectors in a vertical cross section..... 32

Figure 10. Buoyant flow velocity vectors in the same vertical cross section ..... 32

Figure 11. Sample velocity profiles across the CFD domain ..... 35

Figure 12. Cinder Cone Butte model ..... 38

Figure 13. MICRODEM map: SODAR location at the Mountain Home wind farm ..... 39

Figure 14. Google Earth view of the Maountain Home wind farm ..... 41

Figure 15. Mesh cross section showing the boundary layer ..... 42

Figure 16. Triton SODAR, Second Wind, Inc..... 44

Figure 17. SkyServe data example showing wind ramp event ..... 46

Figure 18. WAsP observed wind climate ..... 53

Figure 19. Comparison of modeled 60 meter wind speed at Cinder Cone Butte ..... 55

Figure 20. Wind velocity contours 60 meters above Cinder Cone Butte .....	56
Figure 21. Apsley wind velocity vectors .....	57
Figure 22. FLUENT 10 meter wind velocity vectors .....	57
Figure 23. Apsley model streamlines.....	59
Figure 24. Streamlines over Cinder Cone Butte .....	60
Figure 25. CFD model output compared to SODAR.....	64
Figure 26. Wind velocity forecast for SODAR site.....	65

## LIST OF EQUATIONS

Equation 1. Power in the wind.....	6
Equation 2. Reynolds Decomposition of velocity .....	15
Equation 3. RANS Continuity .....	16
Equation 4. RANS Conservation of Momentum.....	16
Equation 5. RANS Conservation of Heat .....	16
Equation 6. Log law (boundary layer profile.....	18
Equation 7. Power law (boundary layer profile).....	18
Equation 8. Atmospheric lapse rate .....	18
Equation 9. Generation of TKE .....	19
Equation 10. Dissipation of TKE.....	20
Equation 11. Eddy viscosity .....	20
Equation 12. First order forward difference equation.....	22
Equation 13. Second order forward difference equation .....	22
Equation 14. Friction velocity.....	28
Equation 15. Law of the wall equation .....	28
Equation 16. Boussinesq approximation for density .....	33
Equation 17. Reynolds decomposition of velocity .....	47
Equation 18. Standard deviation and wind speed turbulence .....	48
Equation 19. Turbulence intensity .....	48

Equation 20. TKE from wind speed turbulence.....	48
Equation 21. TKE from horizontal wind speed turbulence .....	49
Equation 22. Dissipation of TKE from meteorological data .....	49
Equation 23. Wind speed power law .....	50
Equation 24. Power law exponent from wind speed data.....	51

## INTRODUCTION

High resolution atmospheric flow modeling using computational fluid dynamics (CFD) has many applications in the wind energy industry. A well designed model can accurately calculate wind speed, direction, and turbulence at any point in a wind farm using data from a fixed location source. The model can extend point source data over an area of several square kilometers, or map terrain influenced microclimates using remote wind data. A local flow model is critical for wind resource site assessment: the process of determining the wind energy production potential of a proposed wind farm location. Models are also used to optimize turbine layout for a maximum power production. A CFD model of an operating wind farm, coupled with a local wind forecast, can increase accuracy of electric power generation forecasts, providing valuable information to electric grid managers and wind farm operators.

### **CFD History and Definition**

The study of fluid dynamics started in the eighteenth century. For the first three hundred years, fluid dynamics, like other branches of physics, was divided into theoretical and experimental disciplines (Anderson, 1995). With the development of digital computers, scientists adapted numerical solution methods to complex mathematical problems. This led to the third discipline: computational fluid dynamics. The CFD branch is not fully independent: while CFD research exists as a separate area, most CFD simulations connect theoretical and experimental fluid models.

In the United States, CFD evolved during the cold war, primarily driven by the aerospace engineering community and financed by the defense department. Other pioneering efforts, like Group T-3 at the Los Alamos National Laboratory, started with federal funding for atomic weapons programs and developed CFD solutions for a variety of applications, including atmospheric research. Aerospace engineers, starting with the Wright brothers, relied on wind tunnel results and theoretical models to develop aircraft prototypes. Early supersonic flight provided dramatic demonstrations of the limits of this design method. Computational fluid dynamics developed as a safe, cost effective and efficient way to test experimental aircraft before building prototypes. A new design could 'fly' in numerical experiments so that flight characteristics were reasonably well understood before prototypes were built. Today, CFD methods are used in the design process of everything from fighter jets to building ventilation systems.

CFD models fluid motion by solving the governing equations of fluid dynamics, commonly known as the Navier-Stokes equations. The Navier-Stokes equations are a set of second order non-linear differential equations that describe turbulent fluid motion. There are few closed form solutions for the Navier-Stokes equation set, so numerical methods are used to solve well specified complex problems. As the wind energy industry needs increasingly precise analysis, industrial and academic researchers are investigating and adopting CFD techniques to wind energy applications.

### **Wind Data and Resource Assessment**

Collecting data for wind analysis is expensive and time consuming. The current standard measurement system is a meteorological (MET) tower, 50 to 60 meters tall, instrumented at several levels with anemometers and wind direction vanes. The MET tower is typically installed in a location with good potential for wind energy: frequently at the best wind site in the local area. Considering the layout of a large wind farm, not every tower can be in the same ideal location as the MET tower. Typically at least a year's worth of data is collected and analyzed to assess the local wind climate. Before commercial wind resource models were commonly used, a wind farm designer would plot turbine locations and estimate annual power production based on data from one or two towers. This resulted in many underperforming wind farms and stimulated research in better resource assessment methods. With the development of commercial wind resource computer packages, the new standard is to collect tower data and process it through a software model. The models are a significant improvement over the single tower method, but most have limitations that reduce model accuracy when complex terrain or buoyancy effects are considered.

CFD modeling for wind resource assessment offers flexibility and accuracy that is not available in the commercial wind software. Of course, the added benefits have a cost: CFD solutions are computer resource and time intensive. Commercial CFD software packages are also expensive to purchase, frequently come with annual license fees, and must be adapted to atmospheric modeling. Scientific research oriented CFD packages are

often not as well documented and require more knowledge and training than the commercial codes. Scientists and engineers who run CFD simulations need sufficient theoretical understanding and modeling experience to obtain valid solutions. Obtaining a CFD solution does not mean the results are correct. A model must have good input data, a well designed mesh, and use appropriate turbulence models and solution parameters. In addition, the results should be tested and the model verified with measured data.

The atmospheric boundary layer is the part of the atmosphere directly affected by the earth's surface. Since we live in the boundary layer, people have a basic understanding of weather and how it interacts with their local terrain. We know that the wind blows harder on top of mountains or on treeless, open plains than it does in protected valleys. Farmers and ranchers have extensive knowledge of local weather trends from personal experience, and can easily point to the windiest spot on their land. Anecdotal climate information would tell someone to put a wind turbine on top of a ridge, but for commercial development of wind energy, a much more exact understanding is required. A high resolution of the wind resource is needed to decide if a site is appropriate for a wind farm, and to optimize turbine placement for maximum power production. Before financing wind farms, lenders need accurate resource assessments to understand potential power generation and to calculate the return on their investment. A CFD model provides data needed to assess and map wind resources, and predict power production.

### **CFD for Wind Energy Forecasting**

Like most forms of renewable energy, wind is an intermittent energy source. Electricity grid operators need to schedule generating sources to provide enough power to meet demand. Ideally, an accurate wind energy forecast would allow a grid operator to shut down fossil fuel power plants or hold water at hydroelectric plants when the wind farms are generating. Without a forecast, grid operators are forced to keep generators waiting in reserve in case the wind drops. Unused reserve generators are expensive, but power shortages are unacceptable in a modern society. As more wind farms are built, electric utilities and power grid operators are using wind farm production forecasts to efficiently integrate wind power. In the near future, forecasting will be the norm, the power grid will move from one hour to 10 minute forecast and generation cycles and an electric energy spot market will be adopted nationwide. A CFD model can map a large scale weather forecast into a specific wind farm, improving the accuracy of predicted power generation.

### **Context: The Scale of CFD Modeling**

Weather forecasts are generated from synoptic scale models (hundreds to thousands of kilometers) and mesoscale models (a few to hundreds of kilometers). The scale refers to the size of weather systems, or the area included in the model (the domain). These large models work well for weather forecasts because mesoscale weather events, like rain storms or heat waves, affect our daily lives. Small geographic features and accompanying microclimates make a critical difference when you are planning a wind

farm. Because the power in the wind varies with velocity cubed (1), small changes in wind speed can make significant changes in wind turbine power ( $\rho$  is air density,  $A$  is the swept area of the turbine rotor and  $V$  is the wind velocity).

$$P = \rho AV^3 \quad (1)$$

Advanced weather model research applications, like wind energy forecasting, are pushing the resolution of mesoscale models down to a kilometer or less. Atmospheric CFD models cover a domain of a few square kilometers with grid cells on the order of meters. Ten meter digital elevation maps are used in this project (horizontal resolution 10 meters, one meter vertical), and the model results show wind pattern changes caused by objects as small as a school bus. Because relatively small surface objects can change the airflow that drives a wind turbine, surface detail is important. Over the 30 year design life of a wind farm, a fraction of a meter per second in wind speed prediction can result in a difference of millions of dollars in energy production. These large financial implications are driving interest in high resolution CFD modeling for wind energy applications.

### **Objective**

The goal of this research is to establish a methodology for CFD modeling of boundary layer atmospheric flow. A process for gathering terrain and wind data and preparing data for the CFD experiments will be developed. Complex terrain, neutrally stable and buoyancy driven flow modeling methods will be described in detail. Applications of the CFD model, such as wind resource assessment, wind farm turbine placement and wind energy forecasting will be considered. There is enough detail in the text to set up, run

and analyze a validated CFD model of a wind energy resource in any type of terrain. The appendices list each step of the map conversion, meshing and modeling process. The theory behind the methodology will be discussed in each relevant area

## ATMOSPHERIC BOUNDARY LAYER

The atmospheric boundary layer represents the bottom 10% of the earth's atmosphere. A technical definition: the boundary layer is the part of the atmosphere that is directly affected by the earth's surface, and responds to surface forcings within an hour (Stull, 1988). Rapid vertical growth of cumulus clouds over warm land is an example of a surface forcing causing rapid change in the boundary layer. We live in the boundary layer, and much of the weather that directly affects us occurs in the boundary layer. After living in the same place for a while, most people have a basic understanding of local boundary layer characteristics. The hills above town generally get more wind and snow in the winter, but when the weather is clear and cold, the valley has colder temperatures and inversion fog while the hills are sunny and several degrees warmer.

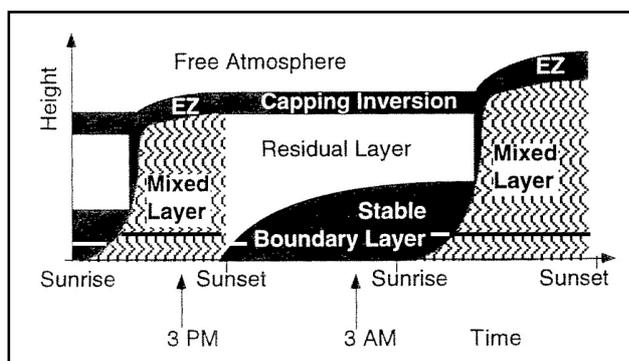


Figure 1. Components of the fair weather boundary layer (Stull, 1988)

The atmospheric boundary layer has several sub-layers as seen in Figure 1. Above the boundary layer, the free atmosphere responds very slowly to surface events. The

entrainment zone (EZ in Figure 1) is a buffer region between the free atmosphere and lower boundary layer. Air from the free atmosphere is dragged or entrained into the boundary layer in this region. Moisture, dust and pollutants are trapped below the entrainment zone by a strong temperature inversion. Air in the free atmosphere is much warmer than air at the top of the boundary layer. Buoyant air parcels that reach the entrainment zone spread out like the flat anvil top of a thunderstorm cloud (the anvil is actually at the top of the troposphere). The mixed layer has relatively uniform wind speeds and temperatures. Turbulence in the mixed layer is mostly driven by convection driven air parcels rising from the surface. Shortly before the sun sets, convection mixing slows down in the mixed layer. The residual layer is the remaining mixed layer with much less turbulence. Wind and temperature profiles are relatively constant through the night in the upper region of the residual layer. The stable boundary layer grows from the surface as daytime convection ends. Surface drag reduces turbulence, and near the surface wind speeds frequently drop at night. Low level jets, areas of high speed winds as low as 200 meters above the ground form in the stable layer. The black and white line near the bottom (white in the stable boundary layer, black in the mixed layer) indicates the top of the lowest part of the boundary layer; the surface layer (Stull, 1988).

### **Surface Layer**

The surface layer is our interface with the atmosphere. It's also where wind turbines operate, and thus the region of most interest in wind energy modeling. Surface drag slows upper level winds, creating vertical wind shear, or the change in wind speed with

height. Surface drag generates mechanical turbulence (turbulence from physical contact with the ground, vegetation, buildings, etc.) affecting the entire surface layer (Figure 2). Intermolecular forces cause air molecules in direct contact to stick to the surface. Intermolecular forces also generate drag between the air molecules stuck to the surface and the adjacent molecules above them. The forces acting between air molecules, or molecules of any fluid, are known as viscosity. Surface drag combined with viscosity slows upper level winds through the entire surface layer. The resulting wind velocity profile is a defining characteristic of the surface layer.

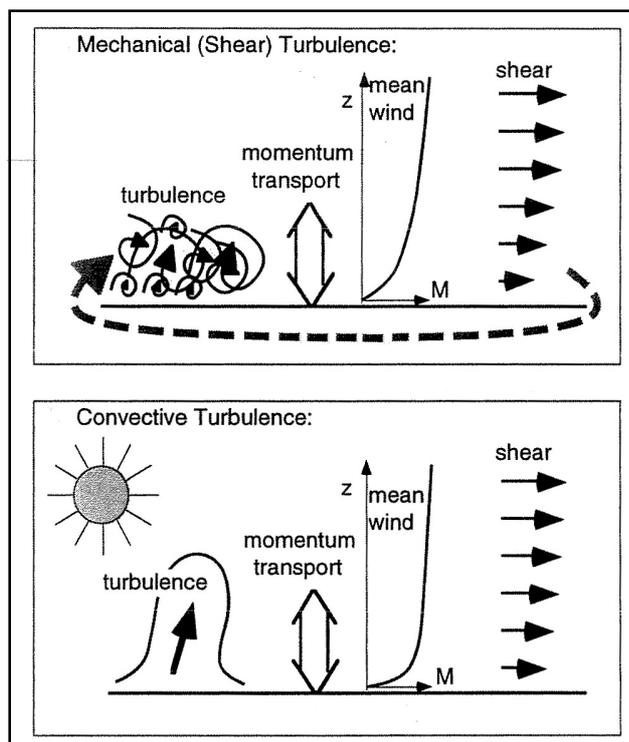


Figure 2. Turbulence generation in the surface layer (Stull, 2000)

At the bottom of the atmospheric surface layer is the microlayer, extending only a few centimeters above the surface. Surface drag and objects like uneven ground, rocks and plants limit almost all horizontal air motion. The primary mechanism for heat and mass transfer is molecular transport. In atmospheric terms, the top of the microlayer equals the roughness height or the height above the ground where wind speed is zero (Stull, 1988). The other source of surface layer turbulence is convective heat transfer (Figure 2). As the sun heats the ground, warm parcels of air (air parcels are bubbles of air with nearly constant temperature and humidity) rise from the surface. The rising parcels pass through the horizontal flow, generating turbulence. The surface layer only represents about 10% of the atmospheric boundary layer. During the day, convective turbulence pushes the surface layer to about 200 meters above the ground. At night, when convective turbulence dies, the top of the surface layer drops to about 100 meters. The variance of turbulence in the surface layer is relatively low, approximately 10% of the overall magnitude of daytime turbulence (Stull, 1988). Variance of turbulence is another defining surface layer characteristic.

### **Boundary Layer Stability**

A common term used to describe general conditions of the boundary layer is stability. Stability refers to the tendencies of air parcels that have been vertically displaced. If a displaced parcel tends to return to its original height, the local surrounding atmosphere is statically stable. If the parcel keeps moving, either up or down, conditions are statically unstable (Stull, 2000). The static part of the definition refers to the absence of horizontal

wind. Neutral stability describes conditions with very little convective turbulence.

Surface layer neutral stability is relatively rare: requirements for true neutral stability are overcast, windy conditions with little difference in temperature between the ground and air. Unstable surface layer conditions are common in fair, sunny weather, or when the ground is warmer than the air (Stull, 2000).

### **Surface Layer Modeling**

A basic fluid dynamics model for external flow is flow over a smooth, flat plate (Figure 3). The boundary layer forms downstream from the leading edge of the plate, and increases in height over the length of the plate. The height of the boundary layer is a function of the free stream velocity,  $u_\infty$ . The top of the boundary layer is the point where the boundary layer velocity is 99% of the free stream velocity; there is no physical change to the flow near the top edge of the boundary layer. Within the boundary layer, the no-slip surface and viscous forces generate shear turbulence. A velocity profile shows the increase in speed with height above the surface. In the free stream, the flow is considered to be inviscid.

A better model for atmospheric flow is the rough wall boundary layer. In rough wall flows, mechanical turbulence is generated by both smooth wall shear and fixed objects on the surface. Boundary layer turbulent structures are a function of the rough wall features and the ratio of the rough feature height to the boundary layer height.

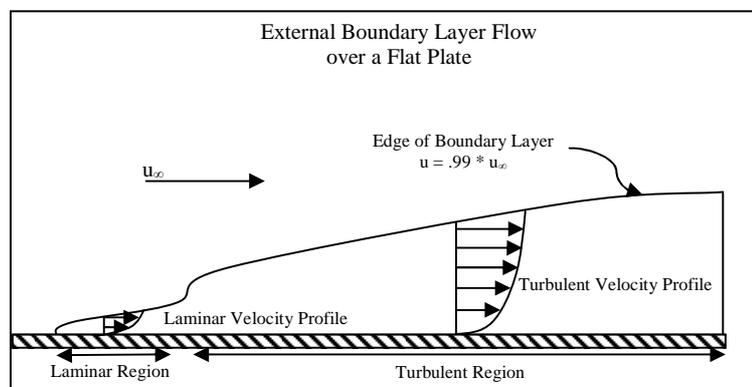


Figure 3. Boundary layer flow over a flat plate

Schematic drawings of experimental and atmospheric boundary layer structures are presented in Figure 4. Further investigation of rough wall turbulence would be of interest if a wind farm site had significant roughness features. Wind farms in southwest Idaho have relatively few surface obstacles. Using atmospheric roughness height of 2 cm for open rangeland, and the daytime surface layer depth of 200 m, the feature to boundary layer height ratio,  $k/\delta$  is 10,000. In Jimenez (2004), the critical  $k/\delta$  ratio for roughness to affect the entire boundary layer is 80. It appears that in open rangelands, surface roughness does not affect turbulence through the entire boundary layer. While there is certainly additional turbulence near the ground, it diminishes rapidly with height in CFD simulations.

The atmospheric surface layer has similar turbulence structure to the rough wall boundary layer. Surface layer depth is affected by surface topography and physical features like trees or buildings. In addition, the surface layer depth is changed by the upper part of the

boundary layer and weather systems. Because the earth is round and the atmosphere, weather and winds are continuous, the boundary layer is also continuous.

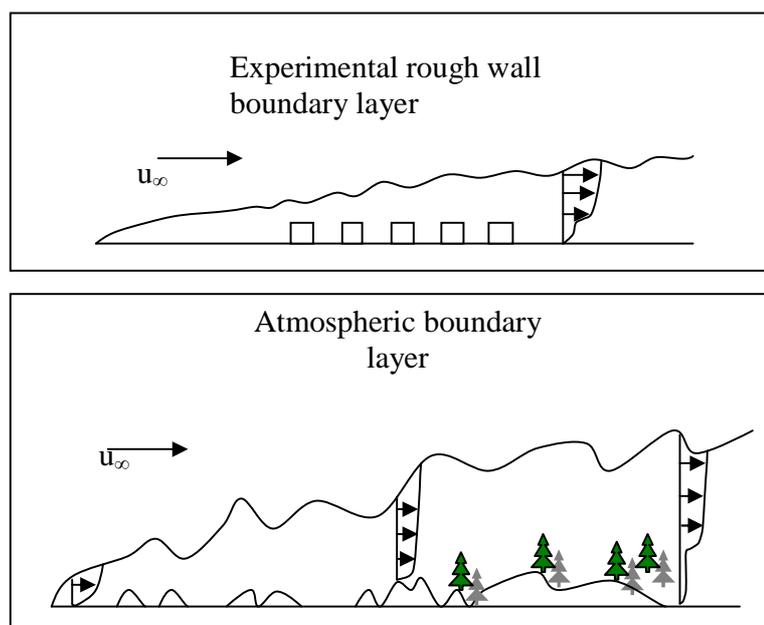


Figure 4. Rough wall and atmospheric boundary layer schematics

The atmospheric boundary layer has diurnal and seasonal changes in convective turbulence, as well as changes in shear turbulence corresponding to surface features. The surface layer over a lake is much more stable than the adjacent surface layer downwind of the lake shore, and surface drag over a forest is lower in the winter after leaves fall from deciduous trees. Wind speeds in the mixed layer are close to wind speeds in the free atmosphere, but the atmospheric boundary layer is typically flat at the top, capped by a strong temperature inversion in the entrainment zone (Figure 1). The warmer free atmosphere mostly stays above the mixed layer: only gradual downward mixing occurs within the entrainment zone.

## TURBULENCE MODELING

Atmospheric flow is almost always turbulent. The viscosity of air is low and air is almost always moving, making laminar atmospheric events relatively rare. Turbulent flow is random, chaotic, and three dimensional. Turbulent flows efficiently transfer heat and dissipate kinetic energy. Except for simplified, conceptual cases, atmospheric flow modeling requires the inclusion of turbulence.

The set of partial differential equations commonly known as the Reynolds averaged Navier-Stokes equations describe turbulent fluid motion. The Navier-Stokes equation set adapts the conservation of mass, momentum and energy equations to fluids. There are no known closed-form solutions to the complete Navier-Stokes equations, so they must be solved numerically. Computational fluid dynamics deals with a variety of numerical techniques to solve the governing equations. Anderson's *Computational Fluid Dynamics* provides an excellent, detailed derivation and explanation of the governing Navier-Stokes equations (Anderson, 1995).

Reynolds averaging or Reynolds decomposition is the process of dividing turbulent terms into average and instantaneous components. For example horizontal wind speed  $U$  is divided into  $\bar{U}$  average and  $u$  fluctuating.

$$U = \bar{U} + u' \quad (2)$$

Reynolds averaging is used by of the standard CFD turbulence models for any random flux variable like velocity. Computational models are typically referred to as RANS (Reynolds Averaged Navier-Stokes) models. The RANS equations set (below) includes average and fluctuating variables for velocity,  $u$ , pressure,  $p$ , density,  $\rho$ , viscosity,  $\mu$ , etc., using Einstein's notation and the Kroneker delta  $\delta_{ij}$ .

The continuity (conservation of mass) equation states that the net flow rate of mass through a control volume equals zero:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho \bar{u}_i)}{\partial x_i} = 0 \quad (3)$$

The conservation of momentum equation:

$$\frac{\partial \rho (\bar{u}_i \bar{u}_j)}{\partial x_i} = -\rho \delta_{i3} g - \frac{\partial \bar{p}}{\partial x_i} + \frac{\partial}{\partial x_j} \left[ \mu \frac{\partial \bar{u}_i}{\partial x_j} \right] - \frac{\partial (\overline{\rho u_i u_j})}{\partial x_j} \quad (4)$$

Terms in order are the advection of momentum, gravity force, pressure gradient force, viscous shear stress and Reynolds (turbulent) stress. And the conservation of energy equation in meteorological terms of heat as derived in Stull (1988):

$$\frac{\partial \bar{\theta}}{\partial t} + \frac{\bar{U}_j \partial \bar{\theta}}{\partial x_j} = \frac{v_\theta \partial^2 \bar{\theta}}{\partial x_j^2} - \frac{1}{\rho C_p} \left[ \frac{\partial \bar{Q}_j^*}{\partial x_j} \right] - \frac{L_p E}{\rho C_p} - \frac{\partial (\overline{u_j \theta'})}{\partial x_j} \quad (5)$$

In the energy equation,  $\theta$  is potential temperature,  $v_\theta$  is thermal diffusivity,  $L$  is latent heat associated with phase changes of water,  $C_p$  is the constant pressure specific heat of dry air and  $Q^*$  is net radiation. The terms in the energy equation (from left to right) are mean local energy storage and advection of heat on the left side and molecular

conduction, radiation divergence, latent release, and turbulent flux divergence of heat on the right side.

### **Turbulence Closure Methods**

The Navier-Stokes equations for turbulent flow have more unknown variables than equations, thus they are an open equation set: a set that cannot be solved without additional equations. This is commonly known as the turbulence closure problem. There are two basic options for solving the closure problem: simplify the model so that an algebraic solution is possible, or develop additional equations for turbulence and use numerical methods to solve the closed equation set. Closed form solutions exist only for simplified, controlled situations, like laminar flow between two infinite flat smooth plates (Couette flow). These solutions are academically interesting, but only apply to laboratory models designed to fit the equations, not any common flow. Most actual flows, including atmospheric flows require numeric solutions.

### **Closure Order**

Many closure methods exist, generally described by their order. The order of a solution refers to the level of approximation or modeling of statistical moment terms in a given solution. A first order solution, for example, solves the mean terms (first statistical moment), and approximates variances or standard deviations (second statistical moments). Definitions are provided for common solution order terms.

Zero order closure approximates turbulent flow with empirical relationships. A good example of zero order closure is the wind speed log law. In the next two equations,  $u$  is horizontal wind speed,  $u^*$  is the friction velocity (equation 12),  $k$  is the von Karman constant,  $z$  is elevation,  $u_0$  is the wind speed at a reference elevation, and  $z_0$  is the reference elevation.

$$u = \frac{u^*}{k} \ln \frac{z}{z_0} \quad (6)$$

Instead of solving the Navier-Stokes equations, log law profiles provides an accurate model to describe the variation of wind speed with height. Empirical relationships are very useful for approximation and estimates, but they have significant limitations. The log law, or the more simplified version, the power law (7), is only valid in the surface layer region of the atmosphere in situations where surface drag contributes most of the turbulence. Alpha,  $\alpha$ , is the shear exponent: 1/7 is the ‘standard’ value when measured wind shear data is not available.

$$u = u_0 \left( \frac{z}{z_0} \right)^\alpha \quad (7)$$

Half order solutions approximate turbulence by using a one equation model and basic assumptions. For example: using the assumption of a constant atmospheric lapse rate  $\lambda$ , temperature,  $T$ , in the boundary layer can be modeled using the lapse rate equation:

$$\lambda = -\frac{dT}{dz} \quad (8)$$

First order equations approximate the standard deviation of wind speed (the variance and covariance) by using mean values only. One and a half order equations use a

combination of mean values and selected second order variables like turbulent kinetic energy. Other second order terms are modeled. Second order closure methods use mean and instantaneous values for all variables, and only model third order variables. With the availability of more powerful computers and computer clusters, higher order solution methods are being used more frequently. LES, large eddy simulation, directly resolves turbulent flow down to the large eddy level and models energy dissipation in small eddies.

One of the most frequently used turbulence models is the k-epsilon (k- $\epsilon$ ) model. The k- $\epsilon$  model is really a one and a half order model (Stull, 1988). Turbulent kinetic energy, k, is used directly, while  $\epsilon$ , the dissipation turbulent kinetic energy, is modeled. Mean and standard deviations of velocity are used with mean values for density. The k-epsilon model, used for this project, will be explained in detail.

### **Atmospheric Turbulence Modeling**

The k-epsilon model is a good compromise of accuracy and computational efficiency, and is the most frequently used turbulence model for many situations, including atmospheric flow modeling. The two principal variables are the generation of turbulent kinetic energy, k, and the dissipation rate of turbulent kinetic energy,  $\epsilon$ . The two transport equations for k (9) and  $\epsilon$  (10) of turbulent kinetic energy close the set.

$$\frac{\partial(\rho k \bar{u}_i)}{\partial x_i} = \frac{\partial}{\partial x_j} \left[ \left( \mu + \frac{\mu_t}{\sigma_k} \right) \frac{\partial k}{\partial x_j} \right] - \overline{\rho u_i u_j} \frac{\partial \bar{u}_j}{\partial x_i} - \rho \epsilon \quad (9)$$

$$\frac{\partial(\rho \varepsilon \bar{u}_i)}{\partial x_i} = \frac{\partial}{\partial x_j} \left[ \left( \mu + \frac{\mu_t}{\sigma_\varepsilon} \right) \frac{\partial \varepsilon}{\partial x_j} \right] - C_{1\varepsilon} \left( \frac{\varepsilon}{k} \right) \overline{\rho u_i u_j} \frac{\partial \bar{u}_j}{\partial x_i} - C_{2\varepsilon} \rho \frac{\varepsilon^2}{k} \quad (10)$$

The k-ε model uses five constants in the transport equations,  $C_{1\varepsilon}$ ,  $C_{2\varepsilon}$ ,  $C_\mu$ ,  $\sigma_k$ , and  $\sigma_\varepsilon$ ; other variables are density  $\rho$ , and viscosity  $\mu$ . Constant,  $C_\mu$  is used to calculate eddy viscosity for the second term of the  $\varepsilon$  equation.

$$\mu_t = \rho * C_\mu \frac{k^2}{\varepsilon} \quad (11)$$

The standard values of these constants are the default values determined empirically when the k-ε model was first derived by Launder and Spalding (1974).

The k-ε model has been used in many atmospheric studies, including wind energy specific cases. Studies of particular interest were those using k-ε and FLUENT CFD, including studies of flow over complex terrain for dust control by Mandas et al. (2004) and flow in complex terrain to study pollution dispersion by Riddle et al. (2003). Other papers of interest involved atmospheric flow specifically for wind energy research. CENER, the national renewable energy center of Spain, has performed and published extensive research on a well instrumented complex terrain site called the Alaiz wind farm. Publications from CENER include comparisons of CFD models to recorded data and to other wind energy commercial models (Cabezón et al., 2005; Martí et al., 2004; Villanueva et al., 2004).

Because the k- $\epsilon$  model constants are empirically derived, constants modified to fit atmospheric data give better results for wind energy research than the standard constants. Alinot and Masson (2002), used wind farm data to optimize the k-  $\epsilon$  model for atmospheric flow. By extensive algebraic manipulation of the turbulence equations, Alinot and Masson derived a set of k-  $\epsilon$  constants that produced more accurate results. These modified constants, Table 1, were used for all CFD simulations in this project.

**Table 1. k-  $\epsilon$  turbulence model constants**

k- $\epsilon$ Constant	$C_{\epsilon 1}$	$C_{\epsilon 2}$	$C_{\mu}$	$\sigma_k$	$\sigma_{\epsilon}$
Standard	1.44	1.92	0.09	1.0	1.3
Alinot-Masson	1.176	1.92	0.03329	1.0	1.3

### **Turbulence Modeling Terminology**

Turbulence modeling uses a lot of confusing terminology. There is second order closure, second order discretization and two equation solutions. These are three very different things, and all could be used simultaneously in a perfectly reasonable CFD model.

Closure order, as explained earlier, ranges from zero order to third order and beyond.

The number refers to statistical moments. Zero order closure models a situation without using the governing equations, like the log law wind velocity profile. A first order closure scheme uses the first statistical moment; the mean. First order data (mean flow variables) is the most precise data included in the equation set; higher order terms are

modeled. A second order closure uses second statistical moment, or standard deviation data. The standard deviation is the sum of squared distances from the mean, so the squared term makes it the second moment. In wind speed measurement, the standard deviation is the instantaneous term in the Reynolds decomposition. A third order scheme is analogous to the third moment – the cubed distance from the mean.

Discretization is the process of transforming a continuous (temporal or spatial) mathematical equation into a stepwise equation. In atmospheric CFD, the spatially continuous Navier Stokes equations are transformed into finite difference equations for numeric solution. A second order upwind solution refers to the form of the difference equation. Second order solutions are generally more accurate than first order. Second order discretization is more complex, requiring more computer resources, and second order solutions can be mathematically unstable. First order forward difference equation and second order central difference equations are shown in general form (the final term means other first order ( $\Delta x$ ) terms):

$$\frac{\partial u}{\partial x} \approx \frac{u_{i+1,j} - u_{i,j}}{\Delta x} + O(\Delta x) \quad (12)$$

$$\frac{\partial u}{\partial y} = \frac{u_{i,j+1} - u_{i,j-1}}{2\Delta y} + O(\Delta y^2) \quad (13)$$

Finally, a one equation model uses one additional equation to close the Navier-Stokes equations for turbulence. The Spalart-Allmaras one equation model (the additional equation is for  $k$ ) is frequently used for airfoil modeling.

## BUOYANCY EFFECTS IN ATMOSPHERIC FLOW MODELING

Boundary layer flow is significantly affected by surface heat flux during the day (Figure 5). As the sun heats the ground, thermal energy is transferred into the air near the surface.

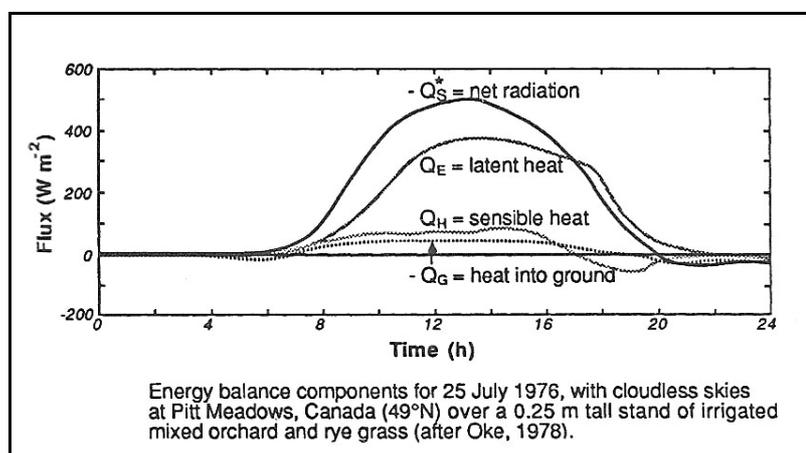


Figure 5. Experimental data showing diurnal heat flux patterns (Stull, 1988)

Air near the surface becomes more buoyant and begins to rise, adding significant turbulence and changing the wind profile in the surface layer (Figure 2). Wind patterns shift between day and night because of local surface heating (Figure 1). Heat flux variation has both seasonal and diurnal variation. Low surface heating in the winter completely changes the makeup of the boundary layer (Figure 6).

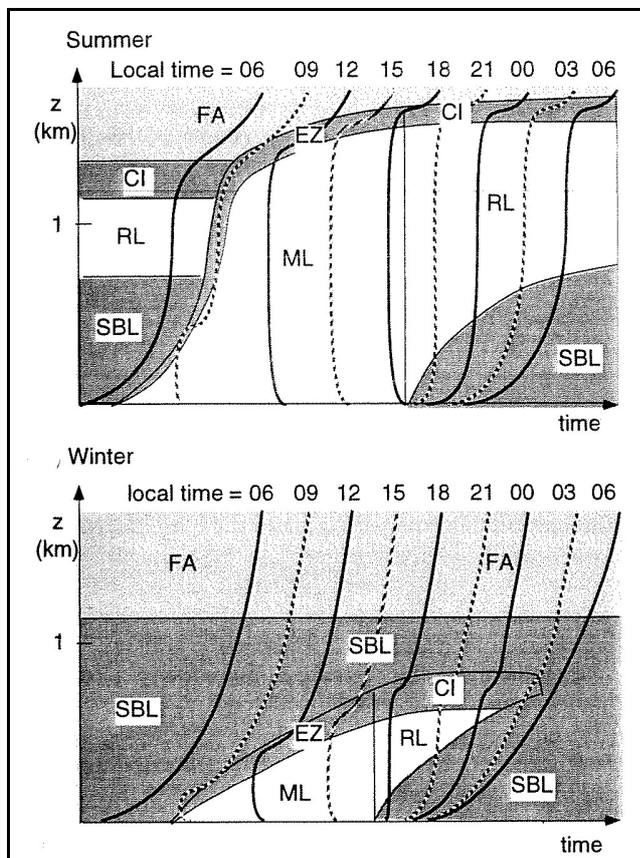


Figure 6. Seasonal changes in the boundary layer (Stull, (2000)

Many CFD models of atmospheric flow model neutral stability to simplify calculations (Apsley, 1995; Apsley & Castro, 1996; Cabezon et al., 2005; Mandas et al., 2004; Marti et al., 2004; Riddle et al., 2003; Villanueva et al., 2004). The neutral model solves the continuity and momentum equations. While the neutral model gives a reasonable overall assessment of the local scale terrain effects, it leaves out buoyancy: the largest contributor to daytime turbulence. Alinot and Masson (2005; 2002) include buoyancy effects in their work. Experimental work on simple geometry showed that the buoyancy

model makes significant changes to the flow without a major impact on computation time.

### **Model Parameters**

The design of the CFD model was based on project goals, scientific literature research, and experimentation. The final CFD model for this project is a RANS model, including buoyancy, using k- $\epsilon$  turbulence closure with modified constants for the atmosphere and the Boussinesq approximation for density. The next section includes discussion of CFD experiments to determine mesh cell size, parameters of the buoyancy model (including the Boussinesq approximation), surface roughness modeling and experiments with various inlet velocity profiles.

## CFD MODEL EXPERIMENTS AND RESEARCH

With the theoretical aspects of the model set, the next part of the project is working out details. Mesh type, cell size, buoyancy model inputs, and surface roughness details are all important pieces of the model that are not explained in the theory. This part of the project required both research and experimentation with GAMBIT (the ANSYS meshing package) and FLUENT (the ANSYS CFD package). Usually, after running an experiment, there are more questions and research leading to more experiments. Spending time and effort in this phase of a project is where real learning takes place. Experimental results on simple geometry lead to better overall results.

### **Mesh Considerations**

Mesh design is critical to obtaining a valid and accurate CFD solution. Factors to consider are mesh element shape, surface grid resolution, boundary layer resolution and the overall number of elements. Mesh cells are made up of connected flat faces. Three dimensional cells can be an unstructured mix of tetragonal and hexagonal shapes, or a structured group of hexagonal cells. While unstructured meshes conform well to irregular surfaces, structured meshes have other advantages. When CFD solution algorithms are considered, a structured mesh, with consistent dimensions in the x, y and z directions allow equal step sizes across the model surface. Consistent step sizes makes the calculations run and converge faster than the same size model with an unstructured mesh.

### First Cell Height

The height of the first mesh cell and vertical resolution is another critical aspect of mesh design. For high speed flows over smooth surfaces, the boundary layer is very thin, making it hard to measure. To model surface drag forces, it's important to have mesh cells in the boundary layer. The law of the wall (Figure 7) is useful to help find the height of the boundary layer from measurable data such as wall shear stress.

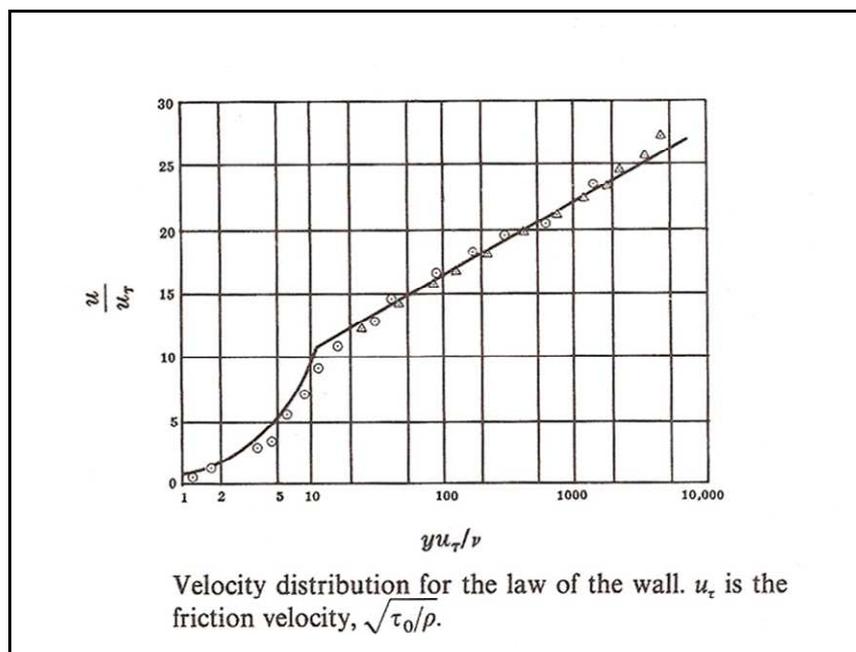


Figure 7. The Law of the Wall. The horizontal axis is dimensionless height (also called  $y^+$ ), the vertical axis is dimensionless velocity ( $u^+$ ), (Hughes et al., 1991).

The law of the wall comes from dimensional analysis. The friction velocity,  $u_\tau$  ( $u_*$  in the equations is the same variable, friction velocity), is a shear stress term with units of velocity.

$$u_* = \sqrt{\frac{\tau_w}{\rho}} \quad (14)$$

The other variables for calculating the law of the wall (15) are the fluid kinematic viscosity  $\nu$ , fluid density  $\rho$ , horizontal velocity  $u$ , and height above the surface  $y$ .

$$\frac{u}{u_*} = \frac{yu_*}{\nu} \quad (15)$$

The non-linear area near the origin represents the microlayer, or the thin layer closest to the surface where  $u^+$  is a linear function of  $y^+$ . The linear part of the graph is where the log law is valid, and where fluid velocity forms a log law profile (equation 6) as seen in Figure 7. Above the log law region near the upper limit of the boundary layer, the graph is again non-linear. For smooth surface external flow, the first mesh cell should have a  $y^+$  value between 20 and 300. If there are additional cells in the linear region of the graph, the mesh generally has sufficient resolution in the boundary layer. Another requirement of first cell mesh height is that roughness elements can't be higher than the top of the cell. The no-slip surface boundary in the CFD model will set the velocity of the first cell to zero. Everything above the first cell is calculated as part of the interior fluid. If there are roughness elements penetrating into the second cell, the model is invalid.

### **Atmospheric Model First Cell Height**

For atmospheric modeling, the approximate height of the surface layer can be measured, and the law of the wall- $y^+$  method is not compatible with atmospheric roughness.

Because atmospheric roughness elements are objects like shrubs, trees or buildings, the

first cell height must be relatively tall. When the first cell height is on the order of one or two meters tall, the  $y^+$  values are on the orders of magnitude above the recommended range. The goal is to have good mesh resolution in the surface layer. The atmospheric surface layer is easy to measure using a standard MET tower or a remote sensing instrument like SODAR. First cell mesh height can be set to account for roughness elements, and higher cells set for good surface layer resolution. Figure 8 shows atmospheric surface layer flow, a velocity profile and an example of the mesh cross section.

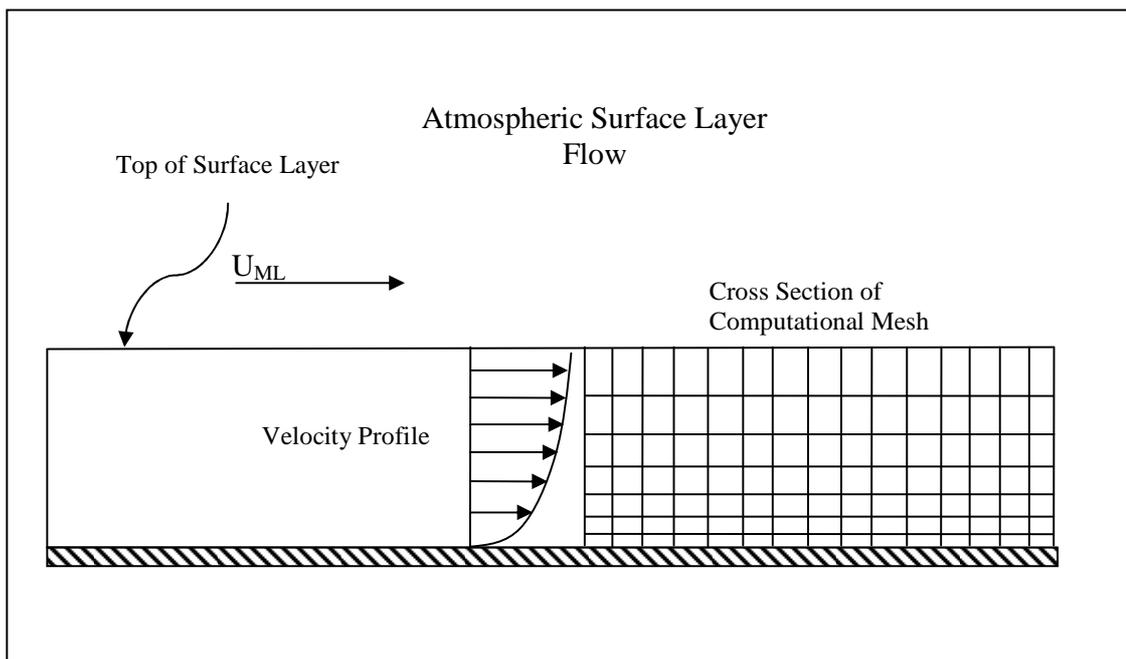


Figure 8. Atmospheric surface layer with mesh cross section

For this project, the first cell height (the top of the cell) is set at two meters above the surface. Several additional cells in the surface layer provide good resolution for the velocity profile. The method of choosing a first cell height to cover the roughness

elements at the surface was used by Alinot and Masson (2005, 2002) and Mandas et al., (2004).

### **GAMBIT/FLUENT $Y^+$ Experiments**

CFD experiments were performed to determine a first mesh cell size that would meet the standard  $y^+$  guidelines ( $20 < y^+ < 300$ ). The experimental domain was 100 meters on each side and 32 meters tall, with a smooth surface (zero roughness height). FLUENT  $y^+$  contour plots were used to assess the effectiveness of the grid size changes. With a one meter square surface grid and a one meter tall (to top) first cell, FLUENT calculated an average  $y^+$  of 2100. Cutting all cell dimensions in half (0.5 m.) only reduced the  $y^+$  to around 900. This kind of resolution for the wind farm domain would generate over 10 million cells. The workstation running FLUENT ran well with models up to about 1.7 million cells, so the required number of cells became another reason to abandon the smooth wall  $y^+$  guidelines. Grid dimensions for the project have surface cells approximately 8 meters square, first cell height of 2 meters, seven or eight cells in the first 200 meters above the ground and approximately 1.4 million total cells in a domain of approximately 6 square kilometers.

### **FLUENT Buoyancy Model Experiment**

To find the best method for buoyancy driven flow modeling in FLUENT, initial experiments were designed on a small, model with simple geometry. A three-dimensional smooth surface with a sine curve shaped hill was used to set up the

buoyancy model and examine the effects of changing boundary conditions. The model was 6.1 m long, 2 m. wide and 2 m. tall with a surface cell size approximately 1 mm square with a total of over 570,000 mesh cells. With a reasonable set of common boundary conditions, experiments were conducted to compare neutral and buoyancy driven flows.

The two figures below show a cross section of velocity vectors through the center of the model. Air flow in the cross section images below is from left to right. The inlet velocity profile is constant with height (2 m/s inlet velocity at 277K). The high velocities in the domain were observed near the summit of the hill. In the neutral flow model (Figure 9), there was a low velocity recirculation zone at the downwind base of the hill.

In the buoyancy model (Figure 10), the high velocity zone was very similar to the neutral model. The flow downwind of the hill was quite different. The model had  $1000 \text{ w/m}^2$  of constant heat flux from the surface.  $1000 \text{ w/m}^2$  is a standard value for clear day solar insolation on a surface normal to incoming solar radiation. The recirculation and low velocity region was clearly separated from the surface. The buoyancy model also had a larger region of low velocity, and stronger vertical flows near the ground. The velocity in the top half of the domain, downwind of the hill, was higher than the neutral flow model. A complete list of settings to run the buoyancy model is given in Appendix B.

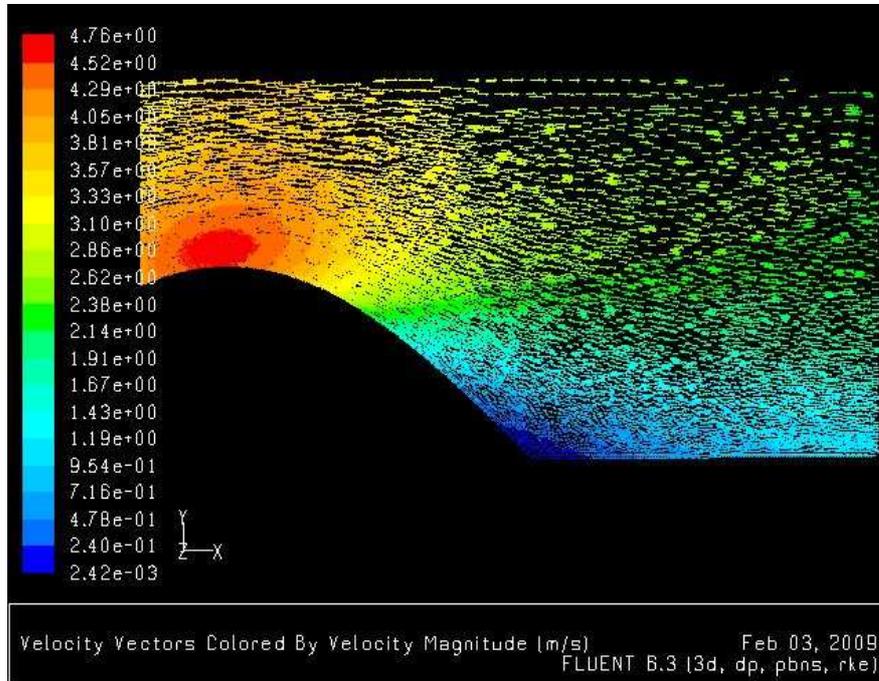


Figure 9. Neutral flow velocity vectors in a vertical cross section

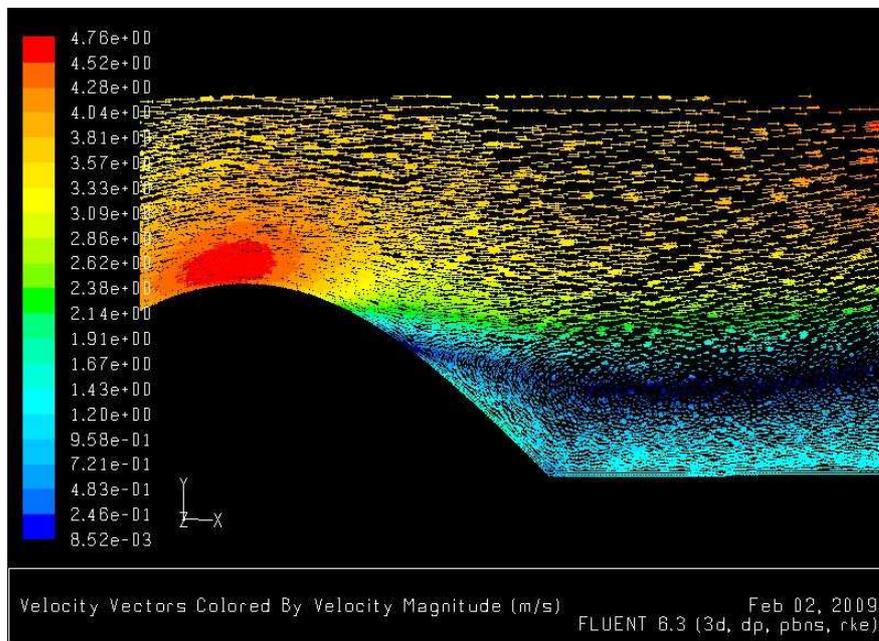


Figure 10. Buoyant flow velocity vectors in the same vertical cross section

The buoyancy model setup followed the natural convection modeling method recommended in the FLUENT User Guide (ANSYS, 2003). The energy equation and gravity effects were used, along with boundary specifications for the operating temperature, inlet temperature and surface heat flux. Properties of air were adjusted, setting the coefficient of thermal expansion, and using the Boussinesq approximation for density. The Boussinesq approximation makes density a function of height only, holding it constant in horizontal dimensions. The Boussinesq approximation simplifies the momentum equation by keeping density constant everywhere except the buoyancy term in the momentum equation (16). Variable density in buoyancy term becomes a function of temperature:  $\rho_0$  is the constant average density of the flow,  $T_0$  is the operating temperature and  $\beta$  is the coefficient of thermal expansion (of air at the operating temperature).

$$(\rho - \rho_0)g \approx -\rho_0\beta(T - T_0)g \quad (16)$$

### **Atmospheric Roughness**

The surface roughness model in FLUENT was designed for pipe flow and external flow over man made surfaces. Typical roughness elements are sand grains in pipes and rivets on airfoil surfaces. Using the default FLUENT surface roughness model for atmospheric flow generates very low surface drag. To produce reasonable surface drag, the actual atmospheric roughness height (the height where wind velocity is zero) must be adjusted. This problem was addressed in the CENER studies by iterating a set of FLUENT roughness inputs to generate results that matched their measured data (Villanueva et al.,

2004). In Riddle et al., (2003), a relationship between FLUENT roughness values and roughness lengths over 0.3 meters for urban studies was determined by experiment. A slightly different approach was found in a paper on FLUENT modeling of flow over beach dune vegetation in New Zealand (Pattanapol et al., 2007). This project developed a CFD model for wind erosion of beach dunes. Careful wind speed measurements were taken with fine vertical resolution. A set of CFD experiments was used to determine the best combination of the dimensionless roughness constant and roughness height multiplier (FLUENT uses these two values to set surface roughness). The resulting roughness to atmospheric roughness conversion factors are presented in Table 2.

**Table 2. Roughness height and constant**

Model Parameter	FLUENT Default	Pattanapol et al.
Roughness Height	$z$ (m)	$30*z$ (m)
Roughness Constant	0.5	0.327

To verify the modified roughness parameters, a small domain with a smooth surface was used. The domain surface was 100 meters on each side and 32 meters high. The surface cells were one meter square by one meter tall and the model had approximately 100,000 total grid cells. FLUENT runs using an inlet velocity profile with the default and modified roughness parameters were compared using XY plots of horizontal velocity (Figure 11). The inlet profile became nearly flat half-way across the domain using the standard roughness model (Figure 11, right side profile). The modified roughness height

and constant in the table maintained the inlet velocity profile across both the small model and the Mountain Home wind farm models.

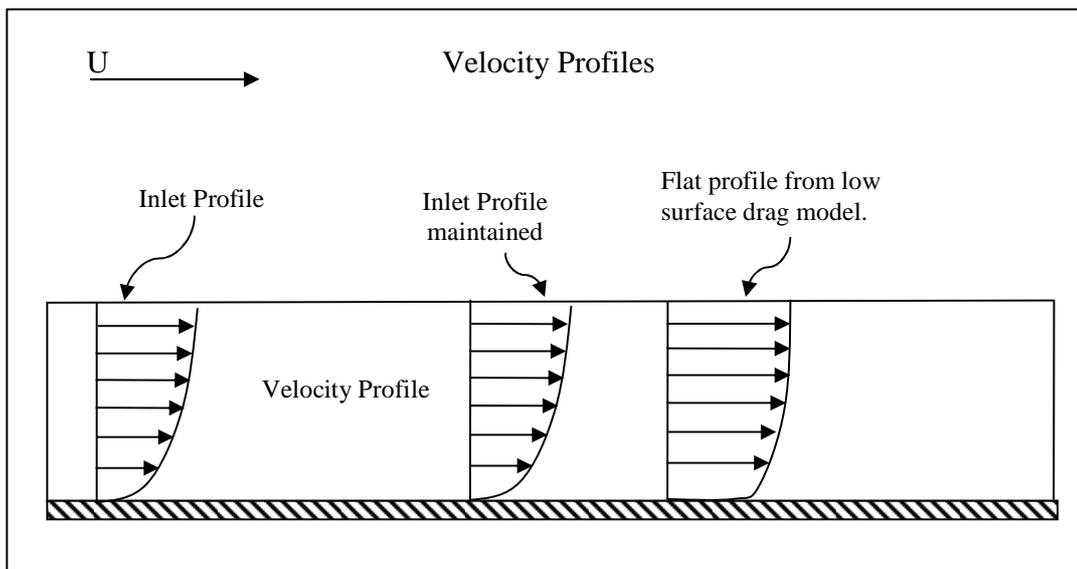


Figure 11. Sample velocity profiles across the CFD domain

## PRACTICAL CONSIDERATIONS IN CFD MODELING

Most meteorologists and mechanical engineers remember at least the overall concept of the governing equations of fluid dynamics. Reading through a fluids textbook explanation of the governing equations to remember a few more details is a straightforward task. Frequently, the practical considerations of applying the theory, finding and processing data, and running the software present more of a challenge. This section discusses practical aspects of making the CFD model work. The goal is to provide enough information so that someone with a basic understanding of fluid dynamics and some background in atmospheric science can make a surface map, mesh it, and gather and process wind data, and get all of it to run in a FLUENT CFD model.

### **Surface Model and Meshing Process**

Modeling atmospheric flow over surface terrain for multiple sites requires a reasonable process for transforming a topographical map into a surface mesh. Generating a grid and establishing a method for transforming a map into a CFD mesh is a complex project on its own. Meshing is frequently the most challenging part of CFD modeling. It can take many hours to generate a mesh that is small enough to run but has high resolution in areas of interest. Refining the mapping and meshing process required many hours of study and experimentation.

The goal of the first CFD project was to model flow over Cinder Cone Butte. Cinder Cone Butte is an isolated hill about 30 miles south of Boise. The U. S. Environmental Protection Agency (EPA) ran a series of gas dispersion experiments at Cinder Cone Butte in the early 1980s. For a range of wind speeds and directions, tracer gases were released from sources upwind of the hill, and gas concentrations were measured by a fixed array of detectors. With measured data available to the public (Snyder et al., 1980), Cinder Cone Butte has been the subject of several CFD studies involving dispersion models (Apsley, 1995; Apsley & Castro, 1996).

Working backwards from the GAMBIT list of acceptable input file formats, and matching the list with available CAD software, a process emerged from many hours of investigation. SolidWorks solid modeling CAD (computer aided design) software was used to make a three dimensional solid surface. To construct the terrain surface, USGS DEM (digital elevation map) data was manually converted into a set of data matrix files with a MATLAB program. Each file contained one column of the elevation data matrix, in a format that could be imported into SolidWorks. Files were individually added to the model to build a set of two dimensional spline curves. A surface was lofted over all of the spline curves and the volume between the bottom of the model and surface was filled and converted into an ACIS format solid. This process required creation of over 100 data files to build the splines from elevation data, and loft the surface. The lofted surface was extruded down to a flat plane to make a solid model (Figure 12). The solid model was converted to ACIS format in SolidWorks and imported into GAMBIT.



Figure 12. Cinder Cone Butte solid model

After many more hours of experimenting and consulting GAMBIT technical support, a faster and more accurate method evolved. In the new method, DEM data was converted from the standard USGS package to XYZ format using a freeware mapping software package called MICRODEM (Guth, 2007). MICRODEM, written by a professor at the US Naval Academy, opens SDTS DEM files and loads them into a graphical display.

MICRODEM's extensive tool set was used to reduce the map size to a CFD domain and save the new map (Figure 13) in XYZ format.

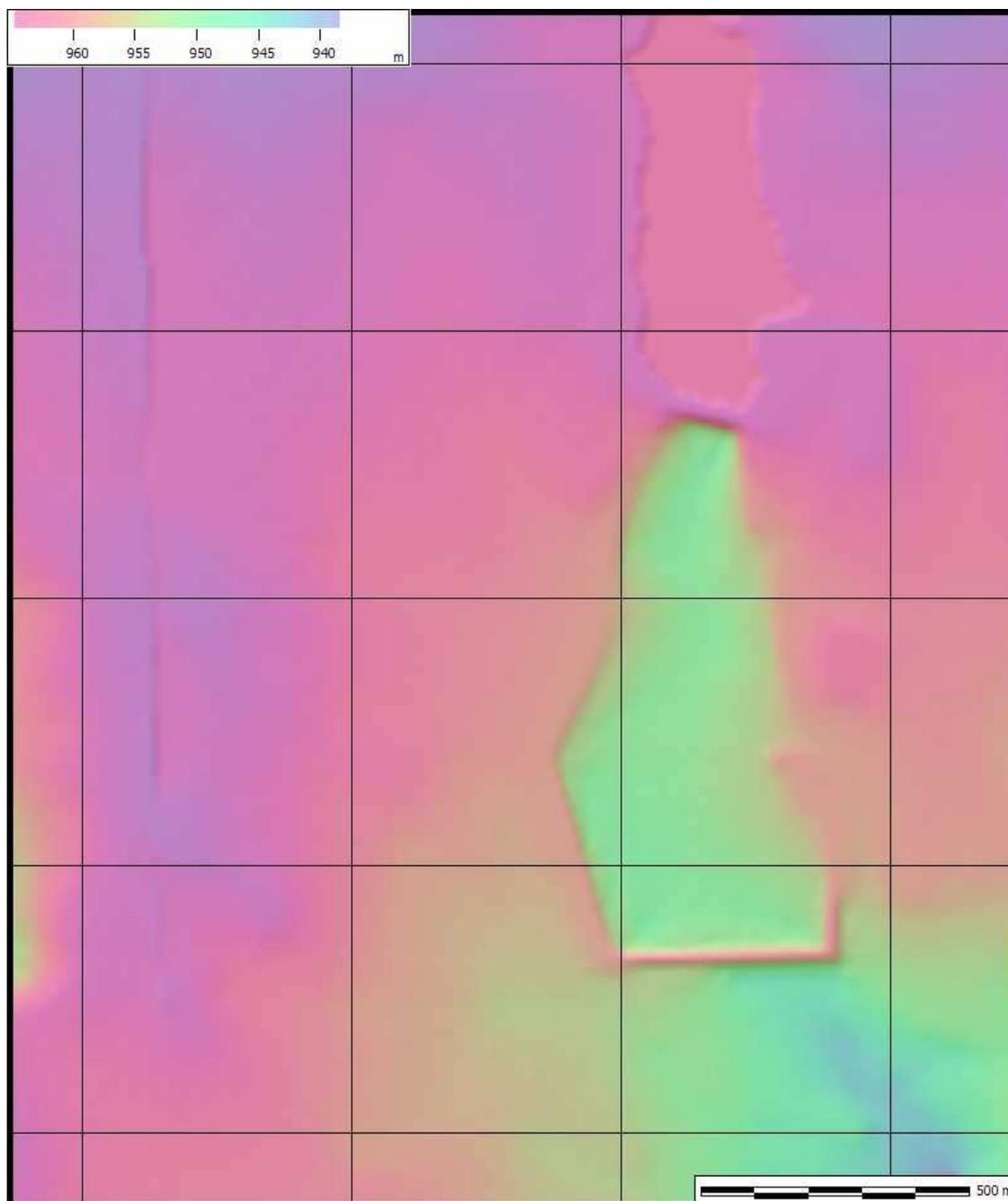


Figure 13. MICRODEM map: SODAR location at the Mountain Home wind farm

As a visual example of the map resolution, the vertical line on the left side is a fence and jeep track at the edge of a pasture. The height difference between the purple and pink areas ranges from one to two meters. The large green area in the lower right is an irrigation reservoir with a 5 meter tall earth fill dam. For comparison to the SolidWorks model (Figure 11), the Cinder Cone Butte summit road with cuts up to 2 meters deep on the north side of the hill is barely visible.

To import the data into GAMBIT, the XYZ map must be square ( $X_{max} = Y_{max}$ ). The XYZ elevation data is manipulated in a Microsoft Excel spreadsheet so that Y is the vertical axis to simplify working with FLUENT. The Y and Z data columns must be re-indexed so that the northwest corner of the new map is the origin. This makes a right-handed co-ordinate system that matches the co-ordinate system used by FLUENT user defined function macros. After manipulating the XYZ data in Excel, the file is saved as a text file.

GAMBIT imports XYZ format DEM data files as a vertex points using the IECM import function. GAMBIT face geometry functions convert the vertex points into a surface. With a surface made, the vertex points are deleted to reduce the mesh file size and make it easier to see points above the surface in the GAMBIT interface. A three dimensional computational domain with flat sides and top is constructed above the surface using GAMBIT geometry functions. If the area of interest is large, like the Mountain Home wind farm, it can be divided to reduce the file size and memory requirements (as shown

in Figure 14). The Mountain Home wind farm was modeled with two domains, one for each row of turbines. A third domain, with the SODAR at the center, was used for forecast model validation (Figure 14).

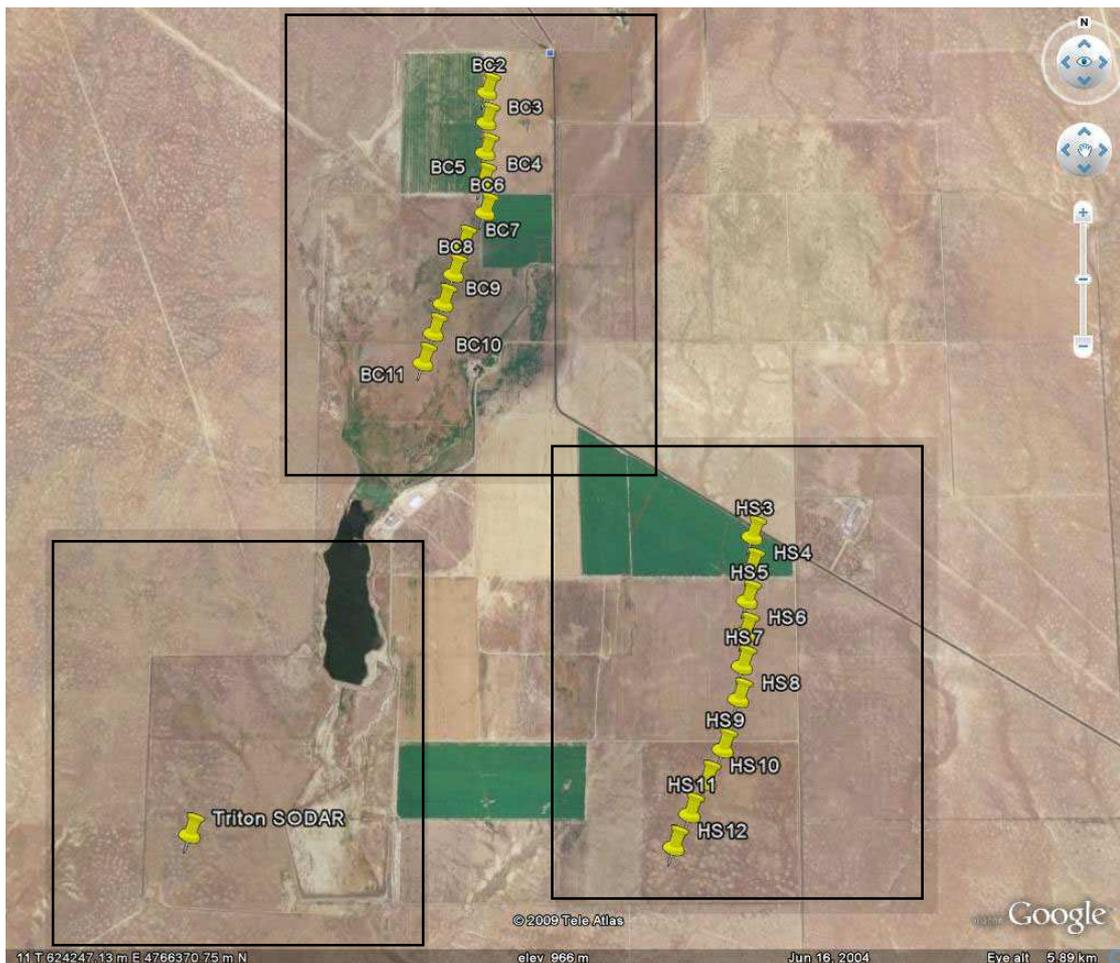


Figure 14. Google Earth view of the Mountain Home wind farm with turbine towers, SODAR and mesh domains are highlighted in gray boxes

The boundary layer mesh tool is used to control the height of the first four vertical rows of the mesh. The mesh resolution is an eight to ten meter square surface grid with a first

cell height of two meters. A size function controls the vertical growth of the rest of the mesh. Typical domains contain approximately 1.5 million cells.

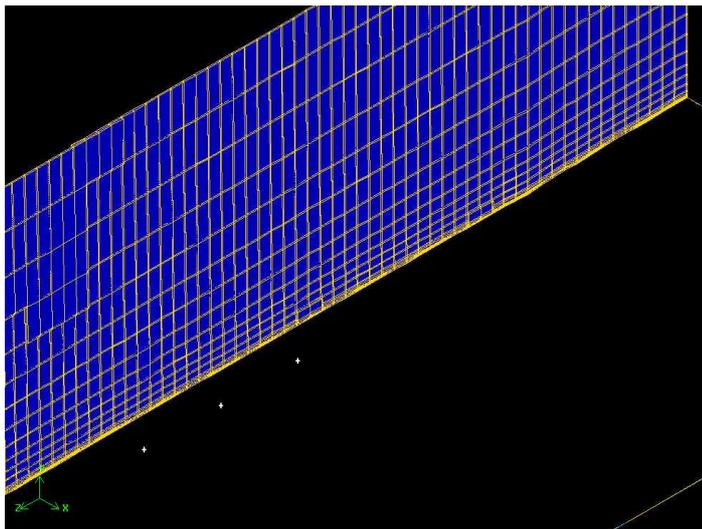


Figure 15. Hexagonal structured mesh cross section showing the boundary layer over the Mountain Home wind farm

A structured, hexagonal mesh was used for this study (Figure 15). The structured mesh had a square surface grid, so  $\Delta x$  and  $\Delta z$  were always equal. Hexagonal meshes are less adaptable to complex surfaces than unstructured mesh, but terrain surfaces are large and the best available data is on a ten meter grid, so structured mesh methods are sufficiently flexible. In more complex terrain, mesh volume adaption in FLUENT can improve iteration speed and convergence. Volume adaption equalizes the cell volume size across the mesh, using an iterative scheme. Early versions of the Cinder Cone Butte model used unstructured grids, but they were slower to run, and slow to converge. This is one of

several situations where FLUENT will generate a solution, but understanding the relationship between the mesh and numerical methods leads to better results.

With a square domain (or reasonably close to square), each side boundary can be used as an inlet or outlet. Wind from any direction can be modeled using the same mesh by setting the side boundaries and the vector components of the wind at the inlet. A single inlet and outlet can be used for winds aligned with the axes. Using a single mesh is a tremendous time savings over rotating the surface model to align the wind direction with a single inlet side, and making a new mesh for each wind direction. A step-by-step set of instructions for mesh generation is included in appendix A.

### **Wind Data**

The first area to consider is wind data, including instruments and data formats. The standard instrument for measuring wind speed is the cup anemometer. Calibrated anemometers from established manufacturers (NRG Systems is the best known) are accurate and reliable. To measure wind, anemometers and wind direction vanes are installed on either a portable or permanent tower. A standard portable wind tower (usually called a MET tower) will have anemometers at several heights, ranging from 10 meters above the ground to 50 or 60 meters. Wind direction vanes will be installed at the top and a lower height, with a temperature sensor at the base. Portable is a relative term: it takes two or three experienced people a few days to assemble and install a 50 meter tower. In Idaho, the Idaho National Laboratory operates a network of MET towers and

makes the data available to the public online at [www.inl.gov/wind/idaho](http://www.inl.gov/wind/idaho). Towers have data loggers in a weatherproof box at the base where data is stored on a memory card. Data loggers are available with cell phone and internet links for remote data access. Typical data includes wind speed in 10 minute averages and wind speed standard deviation for each instrument, with wind direction from each vane and temperature. Another wind measurement instrument that was available for our project is a Second Wind Triton SODAR (Figure 16).



Figure 16. Triton SODAR, Second Wind, Inc.

SODAR (sonic detection and ranging) is a remote sensor that measures wind speed using sound waves. The SODAR has a set of speakers that emit chirps at regular intervals. The SODAR detects the frequency shift of the reflected sound, and converts this to wind speed. SODAR is analogous to RADAR, using sound instead of radio waves. Another remote sensor called LIDAR uses laser pulses instead of sound for remote sensing.

LIDAR provides very accurate wind speeds, and some units have more range than SODAR, but LIDAR is also much more expensive.

The Triton SODAR has been a very useful instrument. It can be moved with a flatbed truck, and installed in minutes. It has a solar panel and battery for power, and has been very reliable. Data is available online, about 10 minutes after the actual measurement. The website has a nice graphical interface and spreadsheet format data can be downloaded for any time interval (Figure 17). Data provided includes wind speed and direction from 40 to 200 meters, temperature, vertical wind speed, and turbulence intensity. The SODAR measures volumes of air, so each wind speed reading is a volume average. The range of the SODAR covers the entire surface layer, so it can easily capture a velocity profile and detect a nocturnal jet. The SODAR range covers the entire rotor plane of a large wind turbine. At times, the signal to noise ratio is low, so data from the higher range of the SODAR has low quality. Triton manufacturer Second Wind recommends setting the quality factor filter to 90%, and not using data with lower quality. One of the best features of the SODAR is the instant graphical view of low shear, high shear and ramp events. Figure 17 shows a wind ramp event recorded on May 12, 2009 at 10:00 GMT, by the Triton SODAR at the Mountain Home wind farm. Notice the increase in wind velocity and direction shear from 12:30 to 13:30, GMT.

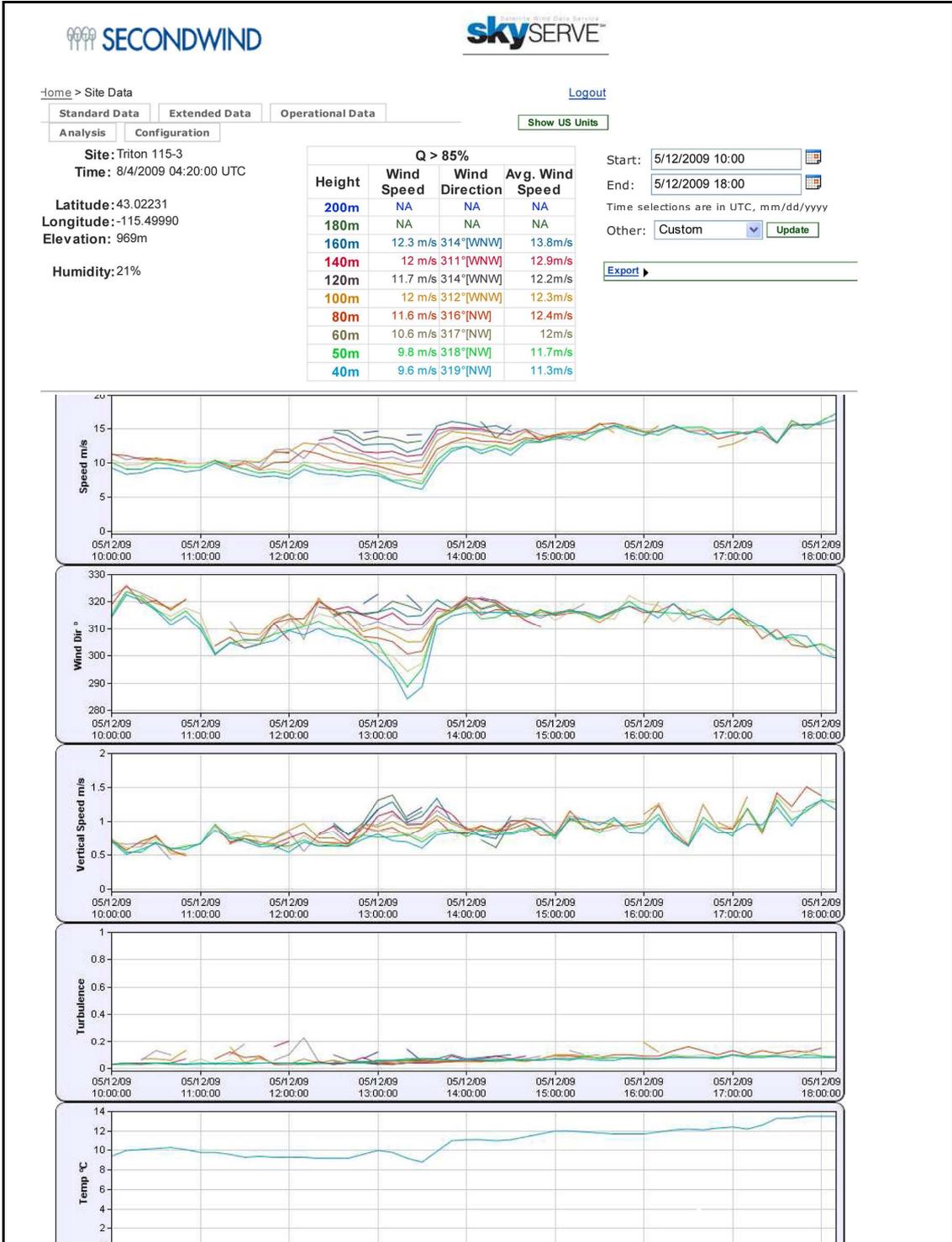


Figure 17. SkyServe SODAR data with wind ramp, May 12, 2009, 10:00 to 18:00 GMT

### **Public Wind Data Sources**

Wind speed data is available from the National Weather Service and automated networks like SNOTEL and MESOWEST. Most of the instruments in these networks are at ten meters above ground or less. A few years of ten meter data provides a general understanding of the wind climate, but it is too close to the ground for wind energy resource assessment. Utility scale wind turbines have hub heights of 80 meters or more, and operate in a very different part of surface layer than a 10 meter anemometer. The INL (Idaho National Labs) network provides data from 30 to 50 meter towers. Many other states have similar public wind energy data. Another important consideration when using public data is surface topography. Surface terrain can significantly alter wind direction, speed and turbulence, so it's important to investigate the terrain around the MET towers and the turbine site. Public data sites that are a significant distance from the area of interest can be used if the terrain and wind resource are similar, but data from the turbine site is preferred. Data from a distant site with different terrain should be used with caution.

### **Processing Wind Data**

Most commercial wind resource software packages, like WAsP, process data taken directly from standard instruments. CFD modeling and other theoretical work requires more understanding of the data and processing methods. Reynolds decomposition separates the average and turbulent parts of the wind speed.

$$U = \bar{U} + u' \quad (17)$$

Ten minute average wind speed data is recorded by all wind speed instruments and most will provide the standard deviation of the wind speed for each recording interval. The standard deviation data is directly related to the turbulent part of the Reynolds decomposed wind speed. Standard deviation data can be used to calculate turbulence with the following equation:

$$\sigma_u^2 = \overline{u'^2} \quad (18)$$

The SODAR provides average wind speed and turbulence intensity. Standard deviation can be calculated from the formula for turbulence intensity:

$$I = \frac{\sigma_m}{M} \quad (19)$$

I is the dimensionless turbulence intensity and M is the mean wind speed. Turbulence intensity measures the relative significance of turbulence, and is frequently used to determine the suitability of a site for wind energy production. Highly turbulent sites stress wind turbine rotor blades and hub bearings, so most turbine manufacturers set a maximum limit of turbulence intensity.

The k-ε model considers the generation and dissipation of turbulent kinetic energy (TKE or k both refer to turbulent kinetic energy). The equation for TKE per unit mass is

$$\frac{TKE}{m} = \sqrt{\overline{u'^2 + v'^2 + w'^2}} \quad (20)$$

Most wind data includes only horizontal and vertical wind velocity (instead of three dimensional velocity vectors). The TKE equation for horizontal wind speed u is:

$$\frac{TKE}{m} = \sqrt{\overline{u'^2} + \overline{w'^2}} \quad (21)$$

Since vertical wind velocity is usually much lower than horizontal and vertical wind speeds are often not available,  $u'^2$  can be used alone or with an estimate of  $\overline{w'^2}$ .

Dissipation of TKE,  $\varepsilon$ , is more complicated. TKE is transferred from larger to smaller eddies, eventually dissipating into heat. There is an equation for calculating  $\varepsilon$  from meteorological data but it requires virtual potential temperature which is not available from MET towers.

$$\varepsilon = \frac{g}{\theta_v} \overline{w' \theta_v'} - \overline{u' w'} \frac{\partial U}{\partial z} \quad (22)$$

Eddy viscosity,  $\mu_t$ , be used to calculate  $\varepsilon$  using standard and k-  $\varepsilon$  constants (equation 11).

In equation 22,  $\rho$  is density,  $k$  is the von Karman constant (0.4),  $\theta_v$  is virtual potential temperature and  $C_\mu$  is a k-  $\varepsilon$  model constant (Table 1).

Because it is not measured, CFD packages often default to a preset value of  $\varepsilon$  for inlet conditions. The value of TKE can be used as an estimate since generation and dissipation of turbulent kinetic energy is approximately balanced (in a relatively large volume of the boundary layer). Advection of TKE makes it difficult to calculate an exact energy balance for a finite control volume.

### **Inlet Boundary Conditions**

The goal for setting inlet boundary conditions is to initialize the CFD model with the same conditions that exist in the atmosphere. There are several methods to initialize the inlet boundary. The simple method is to use a constant average wind speed. FLUENT applies single value inlet boundary conditions as a uniform velocity for the entire inlet. A better approach is to use a log law profile (23) to model wind speed at the inlet. The 1/7 power law equation can be substituted for the log law, is easy to program and gives an accurate profile. The power law equation can be solved for alpha (using equation 24) if wind speeds for two heights are known, making the velocity profile match measured conditions. FLUENT user defined functions (UDF) allow custom modifications of inlet (and many other) parameters. FLUENT UDFs use C language syntax, and have many time saving macros. There is good documentation and many examples in the FLUENT User's Guide (ANSYS, 2003).

For point source data, the recommended method is the power law equation in a FLUENT user defined function (UDF).

$$u = u_r * \left( \frac{z}{z_o} \right)^\alpha \quad (23)$$

In the power law equation,  $o$  refers to reference quantities and  $z$  is the height above the surface. Alpha was calculated by taking wind speeds, averaged over one year, from anemometers at two different heights (30 m and 82 m) on the met tower.

$$\alpha = \frac{\ln u/u_o}{\ln z/z_o} \quad (24)$$

UDF profiles could also be used to set boundary conditions for TKE or  $\epsilon$ . The UDF profile has limitations: the curve in this case is built on two data points and the UDF can only be used at one boundary.

For the forecasting project instead of a power law profile, direct output from the WRF model set the initial boundary conditions. WRF was programmed to output wind speed vectors at fixed heights along its innermost grid cells. The output data was formatted for FLUENT and saved as a boundary profile file. The boundary profile was applied to two sides of the domain. The direction of the flow was set with the velocity vectors in the profile. By adjusting the domain boundaries to inlet or outlet, setting the appropriate boundary profiles and calculating direction vectors, the same mesh can model flow from any direction.

### **Inlet Turbulence**

To more closely model the continuous flow, WRF calculations of turbulent kinetic energy production (TKE or  $k$ ) are included in the inlet profile. FLUENT sets constant values of  $k$  and epsilon by default; or calculated constant values can be used. TKE is easy to calculate from raw anemometer data using equation (19) or (20). A UDF could be used to generate a TKE profile for single inlets. Coupling WRF and FLUENT, using numerical output from WRF and a model with two inlet sides, works very well with

FLUENT inlet profiles. The profile file format and method are well documented in the FLUENT User Guide (ANSYS, 2003). After establishing and testing the method on a few cases, WRF was customized to output profile data files in the correct format for the forecasting project. Appendix D has a sample inlet boundary profile file.

## APPLICATIONS OF THE CFD MODEL

### Cinder Cone Butte

The FLUENT modeling project started at the suggestion of Dr. Paul Dawson. He had modeled flow over Cinder Cone Butte using FLUENT with a relatively low resolution surface map. A goal of the project was to make a better surface map, model the flow in FLUENT and compare the results to the work of Dr. David D. Apsley, who made detailed CFD models of Cinder Cone Butte and published results in his 1995 PhD thesis (Apsley, 1995). The other part of the project was to model the flow using WAsP, a wind industry standard commercial wind resource modeling software package, and compare results with the CFD model (Russell et al., 2008). WAsP solves a linearized flow model based on the RANS equations and the theory of Jackson and Hunt (Corbett et al., 2007). WAsP is widely used and its capabilities are well known: numerous papers cover WAsP performance in a variety of settings (VanLuvanee et al., 2009). Wind speed data was collected from the Bryan's run 82 meter anemometer tower managed by the INL and processed in WAsP. WAsP produces a wind climate summary report, along with a wind rose and annual wind speed distribution plot (Figure 18).

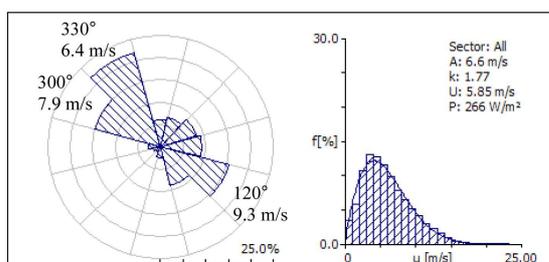


Figure 18. WAsP observed wind climate

The first challenge of the project was to make a high resolution surface that could be meshed in FLUENT. Simultaneously, a method to create a WASP map was also needed. WASP maps are usually created using commercial mapping software that was not in the project budget. WASP input file types included MATLAB files, so a side investigation resulted in MATLAB code to process map data into the correct format for WASP. For the FLUENT model, a combination of MATLAB code, Microsoft Excel, Microsoft Notepad and SolidWorks was used to generate a surface model. The resulting solid surface model, previously described and shown in Figure 11, had significantly more resolution than the surface in the Apsley study (Figure 22).

With a surface model and mesh work complete, considerable research on FLUENT, including reading many papers on atmospheric modeling, was needed to develop and refine a CFD model. Each step of the process included experiments with simple CFD models. The Cinder Cone Butte CFD model was a turbulent  $k-\epsilon$ , neutral stability atmospheric model running on an unstructured mesh. The domain size was initially set using the height of the hill as the length scale. Results from this large model (3 hill heights upstream, 5 downstream and 1 on each side) were post-processed to look at static pressure contours. New domain boundaries were set to include only the extent of the static pressure field generated by flow over the hill. The final domain was approximately 2 km from west to east, 1.5 km from north to south and 750 meters high with about 1.5 million mesh cells. The inlet boundary was initialized using velocity profiles predicted using the power law and with alpha calculated from anemometer tower data.

The biggest problem in proving the WAsP and CFD models of atmospheric flow over Cinder Cone Butte was the lack of tall tower data from the summit. The EPA studies in the 1980s emphasized the dispersion of pollutants around an isolated hill (Snyder et al., 1980). Since the EPA was only interested in monitoring low level wind speeds to control dispersion experiments, the only wind speed data collected was from low level (10 and 15 meter) towers. Apsley was more interested in modeling flow patterns than wind speeds, so the best comparisons to his work involve visual comparison of flow plots (Apsley, 1995). In that regard, the model was successful. Without tower data, wind speed predictions at the summit could not be verified. To compare models, both were initialized with the wind data shown in Figure 18, to predict wind speed at a theoretical 60 meter tower at the summit of Cinder Cone Butte. The WAsP model did not show effects of the hill, while the CFD model predicted a significant increase (Figure 19).

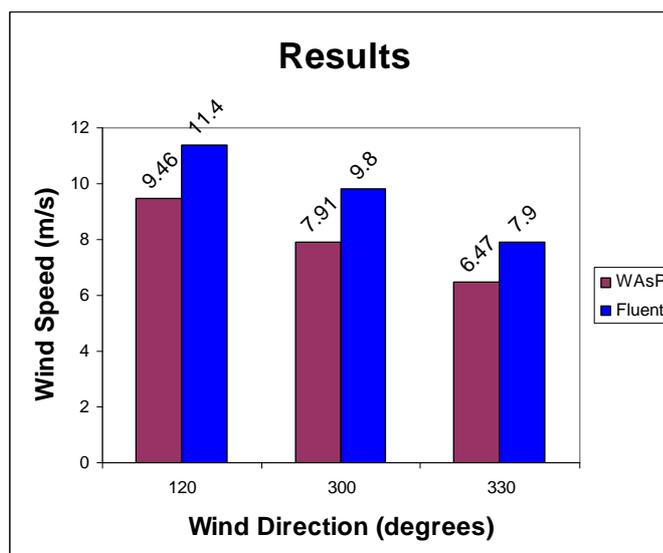


Figure 19. Comparison of modeled 60 meter wind speeds at Cinder Cone Butte summit

WAsP was designed for level terrain in northern Europe and has significant limitations in complex terrain. Experimenting with RIX (ruggedness index, a WAsP complex terrain adjustment factor) may have improved the results. The CFD model indicated a 23 to 24% increase in wind speed at the summit calculated at 10 meters above the ground, which is comparable to measured EPA data (Snyder et al., 1980). FLUENT showed wind speed increasing nearly 30% at 60 meters above the ground (Figures 21 and 22).

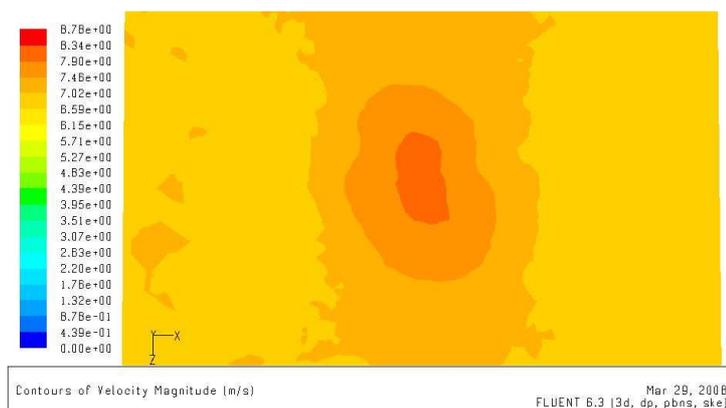


Figure 20. Wind velocity contours, 60 meters above Cinder Cone Butte. The dark orange area of highest wind speeds is over the summit.

One of Apsley's more interesting findings was that low level winds diverge laterally as they pass over the hill. This pattern was evident in the FLUENT model. Figures 23 and 24 below compare wind speed vectors 10 meters above ground with wind from 127°.

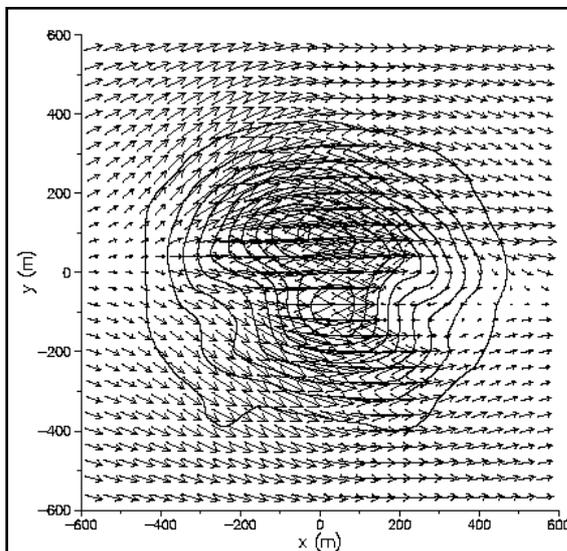


Figure 21. Apsley 10 meter wind velocity vectors

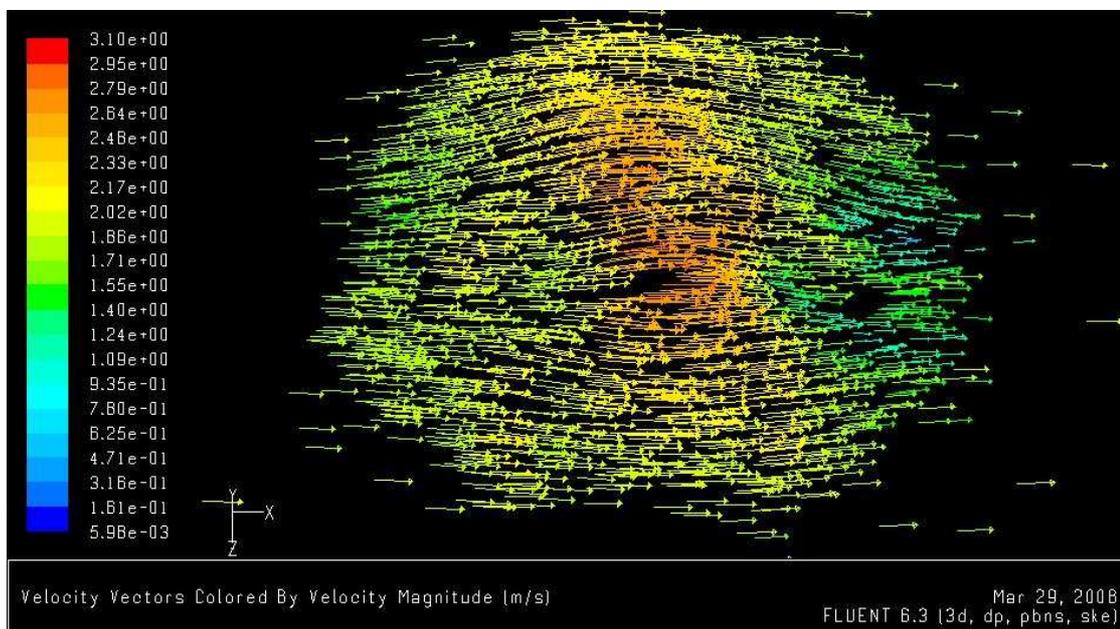


Figure 22. FLUENT 10 meter wind velocity vectors

Apsley also looked at the lateral divergence of streamlines at various heights above the ground. His CFD model duplicated results from measurements of tracer gas concentrations. Apsley's streamlines start from the same point as the tracer gas release. Streamlines at three different heights are shown in Figure 23 below.

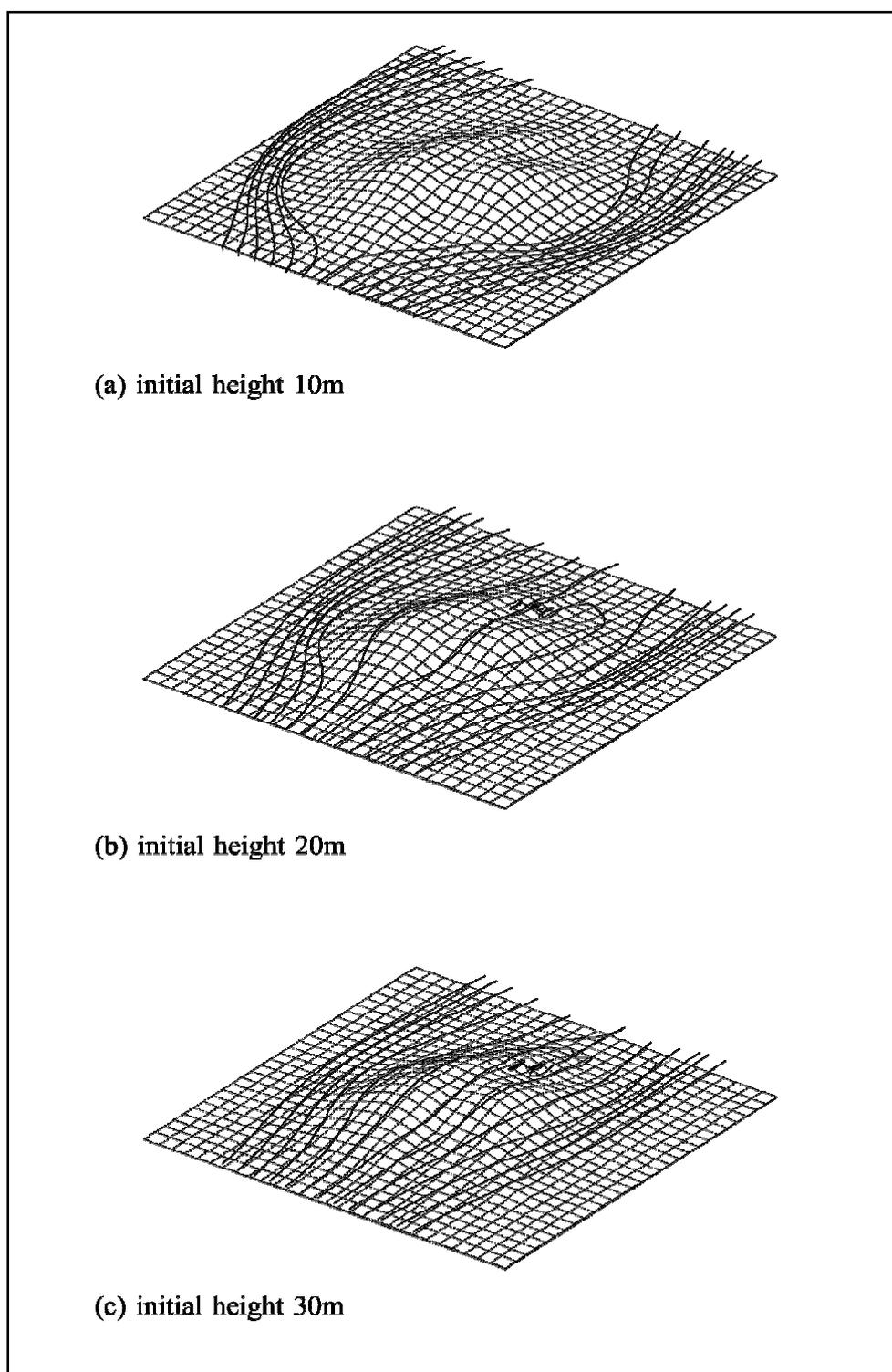


Figure 23. Apsley model streamlines

Streamlines in the FLUENT model show the same pattern. In Figure 24 shows streamlines with 10 meter vertical separation. The lowest three streamlines are at the same height as Apsley's plots in Figure 23. The divergence near the surface is evident in this view looking in the downwind direction. Convergence of the streamlines is also seen downwind of the summit.

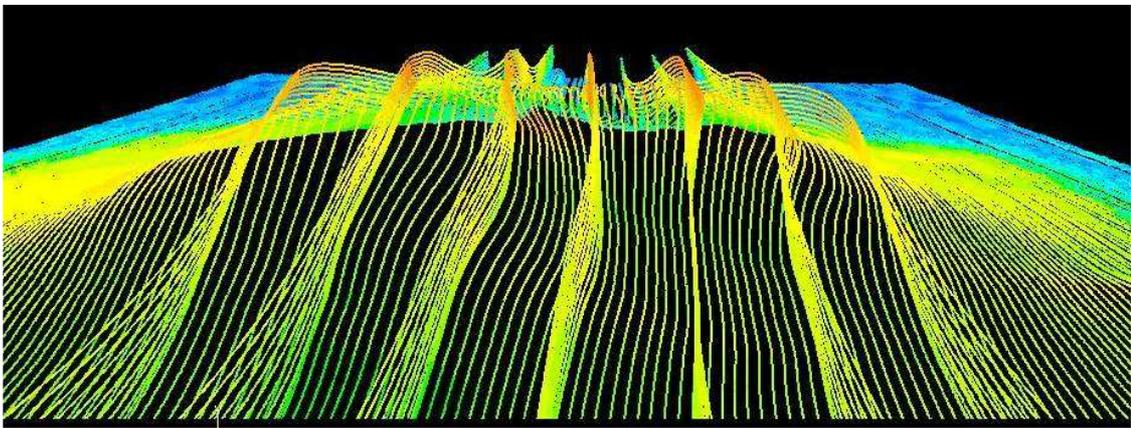


Figure 24. Streamlines over Cinder Cone Butte

### **Cinder Cone Butte Project Results**

The Cinder Cone Butte project showed that a CFD model with a high resolution surface map could be made using standard software. An atmospheric flow domain of significant size will run on a standard Linux workstation with reasonable speed. The FLUENT CFD model successfully duplicated significant results from Apsley's studies. The CFD model shows expected and reasonable wind speed increases over Cinder Cone Butte. The project goal, running accurate CFD simulations of flow over complex terrain was

achieved. Results were presented in a poster and technical paper at the AWEA WindPower 2008 conference in Houston, Texas (Russell et al., 2008).

### **BPA Wind Forecasting Grant Project**

The Bonneville Power Administration has fifteen wind forecasting grant projects this year. The BPA is the largest generator of electricity in the Northwest US. The primary generating source is hydroelectric dams, but the recent construction of over 2000 MW of wind power in the BPA system has driven a significant research on wind power grid integration and forecasting. Most wind energy forecasting uses mesoscale meteorological models and statistical processes to produce a wind speed forecast. The Boise State proposal was to use WRF for a mesoscale wind forecasting and couple WRF to a FLUENT CFD model to precisely forecast wind speeds for each turbine in a wind farm (Dawson, 2008). The results of the forecast will be converted to wind farm power output, the principal objective of the grant project.

The grant proposal included a description of the CFD model and results from Cinder Cone Butte. As with any research project, there were many aspects of the Cinder Cone Butte model that needed improvement. The new project also provided the opportunity to directly measure the output of the CFD model with measured data from the Triton SODAR.

### CFD Model Changes

First, the DEM to SolidWorks surface modeling method was extremely slow and the resulting solid model had less resolution than the original map. The DEM to GAMBIT method, developed for the forecasting project, generates a surface with the original DEM resolution. The same data used to render the MICRODEM image (Figure 12) makes the surface in GAMBIT. The forecasting model uses a structured mesh which improves the results and runs faster than an unstructured mesh. Additional improvements include accurately translating atmospheric roughness to FLUENT terms, and the addition of turbulence ( $k$  and  $\epsilon$ ) data in the inlet boundary profile.

The most significant change between the Cinder Cone Butte and forecasting CFD models is the inclusion of buoyancy forces. The buoyancy method models heat flux from the surface to the atmosphere. For early CFD experiments and presentation at the AWEA conference, heat flux average values were taken from literature, accounting for seasonal and diurnal patterns (Stull, 1988). The forecasting model has since been improved and now uses WRF forecast values for surface heat flux as a surface boundary condition. Another achievement was finding a method to initialize the inlet boundary of the FLUENT model with WRF output data. This involved experiments with FLUENT profile files including generating profile data from the profile formula used in the old model. Once the inlet profile file format for FLUENT was set (example provided in Appendix C), WRF was customized to produce data in the correct format. Another round of WRF and FLUENT experiments proved the method.

The forecasting project was presented on a technical poster (Appendix E) at WindPower 2009 in Chicago (Russell et al., 2009). The focus of the presentation was the coupling of WRF and FLUENT. We found one related paper where a forecast model was refined using a FLUENT CFD model from CENER (Marti et al., 2004). Our project represents a significant amount of new research, and generated interest at WindPower. The poster and paper included details of the CFD model, and a description of the process for coupling WRF output to FLUENT input using inlet profiles. A comparison of wind speeds between FLUENT and the SODAR was presented (Figure 25). FLUENT was initialized with WRF forecast data for the wind speed comparison. At this stage of the project, the inlet profile did not include turbulence data so FLUENT default values for  $k$  and  $\epsilon$  were used as initial boundary conditions. The 10-Feb winds were more turbulent than the 4-Jan, which partially explains why the 4-Jan simulation was more accurate: the actual turbulence was close to the FLUENT initial default values. The 4-Jan experiment was for a local forecast at 1 PM with  $400 \text{ w/m}^2$  of surface heat flux. The 10-Feb forecast time was 6 AM, so heat flux was set to zero. The domain volume used for these simulations was 2.3 km by 2 km by 650 meters tall. Surface cell size was 8 meters square, with a first cell height of 2 meters, and an overall mesh size of 1.4 million cells.

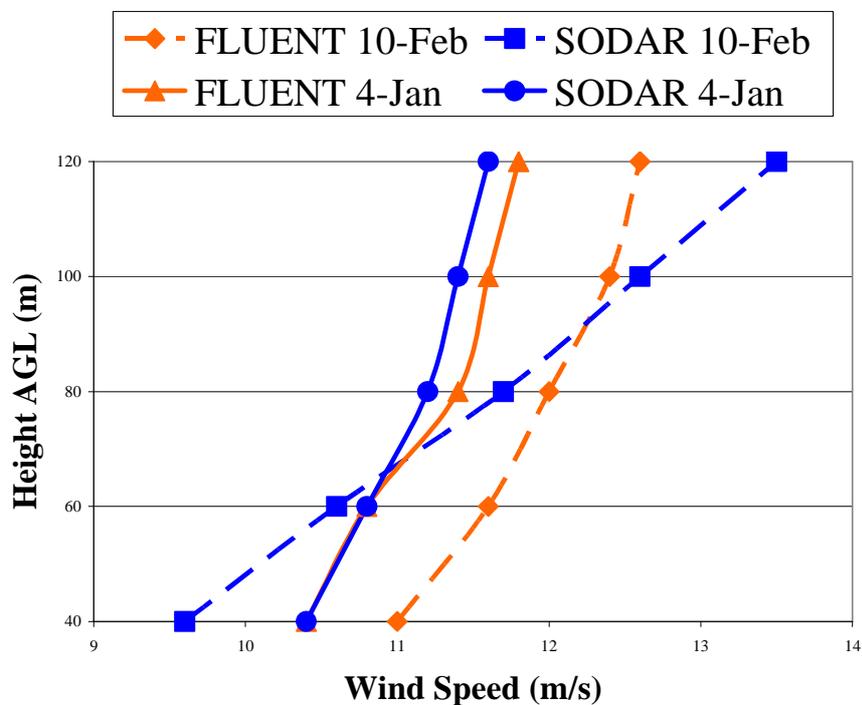


Figure 25. CFD model output compared to SODAR

After the AWEA presentation, WRF was programmed to calculate turbulence data. An automated process was developed to initialize and run FLUENT with the latest WRF forecast data including wind speed, temperature, density, pressure, surface heat flux, and K and epsilon.

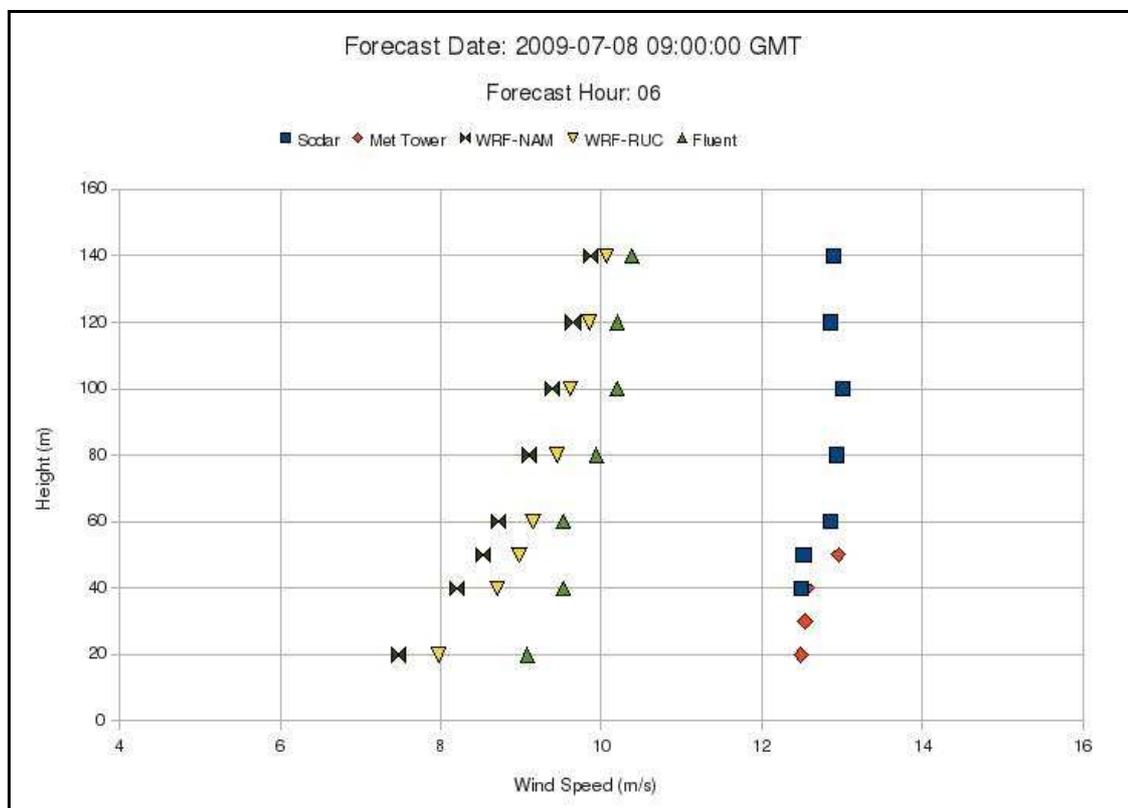


Figure 26. Wind velocity forecast for SODAR site

In Figure 26 above, the six hour WRF wind speed forecast, initialized with either the RUC or NAM synoptic models, generated a typical log shaped wind velocity profile. The SODAR data (blue squares) made an irregular profile, and the FLUENT data mirrored the profile shape of the SODAR (Dawson, 2009). In this example, the FLUENT model, incorporating local terrain and the atmospheric k- $\epsilon$  model with buoyancy effects, improves the WRF forecast. In other forecast simulations, the improvement is less evident. The entire process, WRF and FLUENT was run in automated mode on the Beowulf cluster at Boise State University. In automated mode with 16 processors, the 1.4 million cell FLUENT model ran in less than two hours. The

output of the model generated wind speed data at specified locations including the individual turbine towers and the SODAR.

### **Completing the Forecasting Project**

The grant project will end in December 2009. At this stage, we are comparing forecast wind speeds to the SODAR at the wind farm using the area shown in Figure 14. In general, WRF forecast wind speeds are lower than the SODAR data. In discussions with project partners, we found that bias in WRF forecast wind speeds is relatively common. The team will study methods for adjusting the WRF forecast to correct the wind speeds. We will also continue to work on model improvements and refinements to increase the overall accuracy of the wind speed forecast. The grant requirements are to forecast wind farm power production, so the final step will be to model wind speeds at each of the turbine towers from the WRF forecast and convert the wind speed forecast to turbine power output. Forecast results will be compared to wind farm power output data (SCADA system data) to validate results.

Although the project is not complete, Figure 26 and other results showed that microscale CFD model generally improved the forecast for a specific location such as a wind turbine tower. Microscale and buoyancy effects modeled in FLUENT adjusted the mesoscale model wind profile to a profile similar to what was measured by the SODAR.

### **Future Study**

There are several areas we have considered for future study:

- Improve boundary layer mesh with a lower first cell height and more cells between the surface and the top of the turbine rotor plane
- Include the wind turbine towers in the mesh
- Model the wind turbine wakes, either with the actuator disk method or a dynamic mesh of the turbine blades
- Use the lookup table approach to couple the WRF forecast to FLUENT model data
- Run forecast simulations with other turbulence models like LES and compare results to the k- $\epsilon$  model. Atmospheric research using LES has produced good results in other studies. It would also be interesting to run one equation model and compare accuracy and speed
- Investigate other CFD packages like OpenFOAM in place of FLUENT.

## **Conclusions**

The overall objective of this project was to establish a methodology for CFD modeling of atmospheric flow. A successful method was established through research and experiments for the Cinder Cone Butte project. The forecasting project provided the opportunity to make significant changes. The improved surface modeling process rendered more precise surfaces in simple or complex terrain. The CFD model treated turbulence in the atmospheric surface layer with more accuracy, particularly in the areas of surface roughness and buoyancy driven turbulence. Examples and data validating the model in two projects were discussed. Adaption of the model to resource assessment and forecasting was described in detail. The text and appendices provide theory and instructions so that a CFD student or someone with some experience with FLUENT can recreate the model. The final model is capable of predicting accurate wind speeds and other surface layer data. The method has solid theoretical background, with many references to peer-reviewed research. Finally the CFD methodology and model is flexible and adaptable for any location for which there is a digital elevation map and wind data.

## REFERENCES

- Alinot, Cedric and Christian Masson, (2002). Aerodynamic Simulations of Wind Turbines Operating in Atmospheric Boundary Layer with Various Thermal Stratifications. AIAA 2002-0042.
- Alinot, Cedric and Christian Masson, (2005). k- $\epsilon$  Model for the Atmospheric Boundary Layer Under Various Thermal Stratifications. *Journal of Solar Engineering*, Vol. 127, November 2005, pp 438-443.
- Anderson, John D., (1995). *Computational Fluid Dynamics: the basics with applications*. McGraw-Hill: New York, NY.
- ANSYS, Inc., (2003). FLUENT User's Manual.
- Apsley, David D., (1995). Numerical Modeling of Neutral and Stably Stratified Flow and Dispersion In Complex Terrain. PhD Thesis, University of Surrey, Department of Mechanical Engineering.
- Apsley, David D., and Ian P. Castro, (1996). Numerical Modeling of Flow and Dispersion Around Cinder Cone Butte. *Atmospheric Environment*, Vol. 31, No. 7, pp 1059-1071.
- Cabezon, D., A. Iniesta, E. Ferrer, I. Marti, (2005). Comparing linear and non-linear wind flow models. Wind Energy Department, National Renewable Energy Centre (CENER), Navarra Spain.
- Corbett, J.F., S. Ott and L. Landberg (2007). The new WAsP flow model: a fast, linearized Mixed Spectral-Integration model applicable to complex terrain. *European Wind Energy Conference Proceedings (EWEC)*.
- Dawson, Paul, (2008). Forecasting for Wind Energy Grid Integration. Technical Proposal for Evaluation Purposes by or on behalf of the Bonneville Power Administration, Contract No. 00039902. Boise State University.
- Dawson, Paul, (2009). Forecasting for Wind Energy Grid Integration. Progress Report, September 15, 2009. Bonneville Power Administration Contract No. 00039902.
- Guth, Peter (2007). MICRODEM. Retrieved January 2009, from <http://www.usna.edu/Users/oceano/pguth/website/microdem.htm>.

Hughes, William F., and John A. Brighton, (1991). *Fluid Dynamics, Second Edition*, (Shaum's Outline Series). McGraw-Hill, New York.

Jimenez, Javier, (2004). Turbulent Flows over Rough Walls. *Annual Review Fluid Mechanics*, 36:173-96.

Launder, B. E., and D. B. Spalding, (1974). The Numerical Computation of Turbulent Flows. *Computer Methods in Applied Mechanics and Engineering*, 3:269-289.

Mandas, N., F. Cambuli, G. Castro, G., Cau, (2004). Numerical Simulation of the Atmospheric Boundary Layer (ABL) over complex terrains. *Universita degli Studi di Cagliari, Dipartimento di Ingegneria Meccanica*.

Marti, I., M. J. San Isidro, D. Cabezón, Y Loureiro, J. Villanueva, E. Cantero, I Perez, (2004). Wind power prediction in complex terrain: from the synoptic scale to the local scale. Wind Energy Department, National Renewable Energy Centre (CENER), Navarra Spain.

Pattanapol, Wichai, Sarah J. Wakes, Michael J. Hilton and Katharine J. M. Dickinson, (2007). Modeling of Surface Roughness for Flow Over a Complex Vegetated Surface. *Proceedings of World Academy of Science, Engineering and Technology*, Vol., 26, December 2007, ISSN 2070 3740, 2007 WASET.ORG.

Riddle, Andrew, David Carruthers, Alan Sharpe, Christine McHugh, and Jennifer Stocker, (2003). Comparisons between FLUENT and DDMS for atmospheric dispersion modeling. Brixham Environmental Laboratory, AstraZeneca UK Limited, and Cambridge Environmental Research Consultants.

Russell, Alan, Paul Dawson and Todd Haynes, (2008). Application of FLUENT for Atmospheric Flow, Comparison to WASP in Complex Terrain. American Wind Energy Association, WindPower Conference Proceedings.

Russell, Alan, Paul Dawson, Kevin Nuss and Todd Haynes, (2009). Wind Energy Forecasting Using WRF and Fluent. American Wind Energy Association, WindPower Conference Proceedings.

Snyder, William H., Robert E. Lawson Jr., Roger S. Thompson, and George C. Holzworth, (1980). Observations of Flow Around Cinder Cone Butte, Idaho. Environmental Sciences Research Laboratory, US Environmental Protection Agency.

Stull, Roland B., (1988). *An Introduction to Boundary Layer Meteorology*. Kluwer Academic Publishers: Dordrecht, The Netherlands.

Stull, Roland B., (2000). *Meteorology for Scientists and Engineers, Second Edition*. Brooks/Cole, Pacific Grove CA.

VanLuvanee, David, Tony Rogers, Gordon Randall, Alex Williamson and Todd Miller, (2009). Comparison of WASP, CFD, NWP and Analytical Methods for Estimating Site Wide Wind Speeds. American Wind Energy Association, WindPower Conference Proceedings.

Villanueva, M. P, M. P. Curtet, D. Cabezón, E. Cantero, I. Martí, (2004). Wind resource assessment in complex terrain using a CFD model. Wind Energy Department, National Renewable Energy Centre (CENER), Navarra Spain.

APPENDIX A  
**GAMBIT Mesh Process**

## Part 1. Download DEM Data and Make GAMBIT Input Data File

Mesh geometry should be made in a right handed coordinate system with y as the vertical axis. Other right handed systems work in GAMBIT, but eventually cause problems in FLUENT. FLUENT user defined functions and other advanced features that assume a y-vertical axis system. The standard coordinate system in meteorology, atmospheric science and wind energy applications is to have z as the vertical axis, so data conversion to fit the FLUENT engineering coordinate system is required.

Terrain surface map data are easily managed in GAMBIT, although the method is not well documented. The basic input is a USGS DEM (digital elevation map). Idaho DEMs and maps for other states are available for free download at [www.geocomm.com](http://www.geocomm.com). You need to open a user account and use the slow downloads to get free DEM data. Users can pay a small fee for access faster downloads and more map types. The free download of a standard DEM only takes a minute or two.

To import map data into GAMBIT, the data must be converted to the XYZ DEM format and saved in an ASCII text file. The only data restriction for GAMBIT is that the map area must be square ( $X_{max} = Z_{max}$ ). There are several free DEM to XYZ conversion utilities available on the web. MICRODEM, a freeware mapping software package makes the process easy. With MICRODEM, you can load SDTS DEM (SDTS is the USGS standard format) directly into a 3d surface map window. Adjacent maps can be

combined if the area of interest is near a map border. It's easy to crop or resize maps.

MICRODEM has many export format options, including XYZ files.

This process outline assumes that the user has some familiarity with GAMBIT:

1. Download free SDTS DEM (10 or 30 meter data) map from [geocomm.com](http://geocomm.com).
2. Open the map in MICRODEM (browse to the tar.gz download file. MICRODEM will open the zipped file)
3. Resize the map to the area of the CFD domain. Getting the optimal size takes some experience or a few iterations. Start with an area 2km square or less. Save the resized map with a new file name in DEM format.
4. Re-open the resized map file and save it in xyz format
5. Open the file in Excel for easy data manipulation (the axis swapping process could be automated in Matlab, but Excel is easy and fast).
6. Standard xyz files have the elevation in the z column. Swap the y and z columns using Excel highlight, copy, paste, insert and delete functions.
7. To convert the data back into a right-handed coordinate system, the new z (old y) column must be re-indexed. The origin of the new system will be the northwest corner.
  - a. Highlight the y and z columns. Do an Excel sort of the two columns based on the (new) z column in descending order (Data-Sort-By Column-Descending)

- b. Make a formula to re-index the z data starting with 1 in ascending order.

Example:  $-1*(-(z_{\max}+1)+z)$ .

8. To make a square area data file for GAMBIT
  - a. Find the maximum x value in the file ( $X_{\max}$ )
  - b. Sort the three columns by column z ascending
  - c. If  $Z_{\max}$  is larger than  $X_{\max}$ , cut off all data (rows) with  $Z_{\max}$  greater than  $X_{\max}$ . If  $X_{\max}$  is larger, undo the sort and cut off the excess  $X_{\max}$  data.
  - d. Scroll to the bottom and find the number of rows (the Excel line number of the last data point).
  - e. Scroll to the top. Insert a row. In column 1, row 1, enter the number of rows. In column 2 row 1, enter the number 1
  - f. This is the header information GAMBIT needs to interpret the data file. The header information tells GAMBIT the number of data points in the file, and the number of points in each row.
  - g. Calculate the number of rows (and columns) in the map. This is the square root of the number of rows. Write this down somewhere, you will need this to tell GAMBIT how the size of the data grid.
9. Save the data as a .txt file. Open it in Notepad and check to see if it looks like the three Excel columns.

## Part 2. Create Mesh in GAMBIT

1. Open GAMBIT
  - a. On Linux/Unix systems, open GAMBIT with the following command line to connect a file name with your GAMBIT session
    - i. `GAMBIT -id filename`
2. File-Import-IECM input
  - a. Deselect face and select vertices
  - b. GAMBIT will import and display the vertex points. Right click and drag the mouse to rotate the image.
  - c. Select Isometric view, then fit to window (Global Control tools)
3. Face commands
  - a. In Face commands, select make face from vertex rows
  - b. Select all vertices, enter the face name and the number of rows (calculated in step 7g above) in the dialog box and click apply.
4. Vertex commands
  - a. Delete vertex, select all, delete.
  - b. Four will remain, one for each corner of the face.
  - c. Make four vertices for the top of the domain. A reasonable height is 600 to 750 meters above the highest corner of the surface. The top of the domain should be flat (i.e. the same elevation for each top corner).
5. Import additional vertex data (optional)

- a. Data files for the turbine locations are imported as IECM vertices. This can be useful as a visual reference when examining the geometry before meshing. It's also useful to help determine the relative size of surface mesh cells. If the distance between turbines is known, the number of face mesh cells between towers is easy to calculate.

#### 6. Split Face (optional)

- a. If the square surface area is too large, it can be split with vertices using face-split. Note: If the upper left corner of the map is removed, it is difficult to keep track of the location of the domain in geographic terms. This is not recommended. It's OK to reduce the size of the map by splitting and deleting small areas from the right and/or bottom edges.
- b. Delete face to delete the smaller faces made with the split face command.

#### 7. Edge commands

- a. Make edges using the bottom and top corner vertices. There should be four top edges and four corner/side edges to make a closed box. To select a group of vertex points, edges, etc. with the mouse, hold down a shift key and select by right clicking.

#### 8. Face commands

- a. Make a faces from the edges.
- b. There will be six faces – top, bottom and four sides.

#### 9. Volume commands

- a. Make a single volume from the six faces.

## 10. Mesh commands

- a. Mesh the surface face (Mesh Face) using Quad elements and the Pave option. Set the spacing (interval size) to set the number of mesh elements.
- b. Use the known distance between turbine vertex locations to determine the size of the surface mesh.
- c. Adjust mesh size as needed. Optimize between overall mesh size and boundary layer model accuracy.

## 11. Boundary Layer

- a. Make a boundary layer to control the height of the first three to five mesh cells. The first mesh cell should be high enough to be clear of the roughness height, but small enough to model the surface layer. Two meters is a reasonable height for the first cell if the terrain is smooth and the vegetation is relatively low (like sage brush desert or most agricultural land).
- b. A reasonable boundary layer: First row =2, Growth factor = 2, Depth = 4 (Depth = 30, calculated by GAMBIT). Attachment – Faces, surface face. The boundary layer appears on the sides of the domain in white if it worked correctly.

## 12. Mesh Volume

- a. Mesh the volume above the boundary layer. Select the volume, hex elements, map type. If you use a size function, the spacing is ignored.
- b. Getting a reasonable number of cells takes some iteration.

- c. Use a size function for vertical cell height control
  - i. Source – surface, Startsize – choose a cell height that matches up with the boundary layer, Growthrate and Maximum – choose to optimize cell volume. Growthrate 1.7 and max 75 is reasonable.
- d. On a dual-core processor Linux workstation, models with less than 1 million cell meshes are easy to open and run fast in GAMBIT and FLUENT. Up to 2 million cell meshes are OK, around 1.8 million cells is where GAMBIT starts getting slow. Over 2 million cells often don't run well. The system speed and memory makes a big difference with large meshes. FLUENT is much better than GAMBIT with large files. GAMBIT is crash and lockup prone so save files often and know how to kill the process in Linux.
- e. Note: If you are setting up a process, generate a small (less than a million cells) mesh to run experiments in FLUENT. This makes it much easier to test inlet profiles, buoyancy effects, etc.

### 13. Size Function

- a. Note: the size function controls the cell growth rate above the boundary layer. It can also be used instead of a boundary layer, but the combination of a boundary layer and a size function allow more cell growth control.
- b. Make a size function to start with the meshed boundary and mesh the remainder of the volume
- c. Size Function – meshed – set growth rate and maximum cell volume

- d. This takes some iteration – try 1.75 for a growth rate, 75 for a maximum, mesh the volume, look at the number of cells and adjust as needed to optimize the total number of cells.

#### 14. Set Boundary types

- a. Note: this step can be done after step 9. Without the mesh, filling up memory, it goes faster.
- b. Surface = wall
- c. Sides = Velocity Inlet, outflow or symmetry as determined by the model
- d. Top is symmetry (can be outflow in complex terrain if there is only one outflow side boundary).
- e. The boundary type can easily be changed in FLUENT. Make each face separate boundary.

#### 15. Export mesh

- a. Use the Export-mesh command to generate a .msh file for FLUENT.
- b. Save the GAMBIT file (save it often while working in GAMBIT)
- c. Open the mesh file in FLUENT to make sure it works.

To check the mesh in FLUENT, use Read-case and select the mesh file. If it opens, displays data about the mesh and gets to a prompt, the mesh is probably good. Another good check is Grid-check and/or Display mesh.

APPENDIX B

**Fluent Solution Steps**

Start Fluent in 3ddp mode (three dimension, dual precision) with parallel processors (if available). Steps with an asterisk apply specifically to the buoyancy model (energy equation on). To run without the energy equation, skip these entries. Several options will not be presented without turning the energy equation on in step 3.

1. Start FLUENT

- a. Command syntax 'fluent 3ddp -t2' Main window opens.
- b. If running command line Fluent, 'fluent 3ddp -t2 -g'. FLUENT starts and is ready when the > prompt appears.

2. Read Case or Read Case and Data

- a. Select mesh file to start a new case.
- b. Grid-check. Fast error check for new mesh files.

3. \* Define-Model-Energy – Energy On

4. Define Operating Conditions

- a. Define-Operating Conditions:
  - i. Operating pressure: input a point near the inlet within the domain
- b. \* Gravity: On  $y = -9.81 \text{ m/s}^2$
- c. \* Boussinesq Parameters: Op Temp 275K

5. Define Turbulence Model

- a. Define-model-viscous: k- $\epsilon$ , standard, standard wall functions
  - i. Constants for atmospheric flow are  $C_\mu$ : 0.03329,  $C_{1\epsilon}$ : 1.176, and default values for the other constants

- ii. \* Check Full Buoyancy Effects
6. Solve-control-solution: 2<sup>nd</sup> order upwind
    - a. SIMPLEC pressure-velocity coupling
    - b. PRESTO pressure interpolation
    - c. Under-relaxation factors – adjust as needed (reduce to help convergence).  
To start, reduce Density, Body Forces TKE, Turbulent Viscosity and Energy (subtract 0.1 from defaults)
  7. Solve-Monitors-Residual
    - a. Select Plot, convergence criteria 1E-5
  8. Define-Profiles
    - a. Read velocity profile data file
  9. Define Boundary
    - a. Outlet – Outflow with percentage if more than one outflow boundary
    - b. Surface/Wall-Momentum
      - i. Roughness Height = 30 \* actual roughness height (0.02, 0.03...).  
Value is 0.6, 0.9, etc.
      - ii. Roughness Constant 0.327 ()
    - c. Surface
      - i. Thermal Tab: Heat Flux 700 W/m<sup>2</sup> or appropriate flux
    - d. Velocity Inlet
      - i. Momentum: Velocity Specification Method-Components: Set X, Y and Z velocity for each inlet

1. With Profile data file

- a. X velocity: u component u
- b. Y velocity: v component v
- c. Z velocity: w component w
- d. K and epsilon if part of profile
- ii. Turbulence: TKE = X if not using K in profile
- e. \* Inlet
  - i. Thermal Tab: Temp = 277 (match Operating Conditions)
- f. Fluid (optional)
  - i. Click Source Terms
  - ii. Under Source Terms Tab, set mass, momentum TKE (k), Turb Dis. Rate ( $\epsilon$ ) and Energy

10. \* Define Materials: Air

- a. Properties:
  - i. Density: Boussinesq in drop down
  - ii. Density = 1.205 (match Op Temp)
  - iii. Thermal Exp Coeff. .00343 1/K, click Change/Create

11. Solve-Initialize-Initialize

- a. Compute from Inlet or All zones
- b. Set Velocity Components

12. Post Processing Surfaces

- a. Surface-Iso-Surface-Surface of constant grid

- i. X-Ctr, Z-Ctr, Y-average hub height, etc. Choose slices through the domain for contour plots, vector plots.
- b. Surface-Transform
  - i.  $Y = 80$  creates a surface 80 meters above the ground, parallel to the ground at all points
- c. Rake
  - i. Choose turbine or instrument location, with x,y,z co-ordinates. Make x and z constant and y from the surface to 200 meters, with 21 rake points. An xy plot of the rake will give velocity at 10 meter intervals from the ground to 200 meters.

### 13. Solve-Iterate-Iterate

APPENDIX C

**FLUENT Inlet Profile Examples**

## FLUENT User Defined Function

This is the code for FLUENT user defined function (UDF) that sets a 1/7 power law wind speed boundary profile at the inlet. The #define constants set the wind speed to 12 meters per second at 50 meters above the surface. The wind speed and height can be changed by changing these constants. The power law calculation is in the line that begins with 'F\_PROFILE(f,t,i)'. All statements in caps are FLUENT macros (built in functions) designed to make programming UDFs easy. This code is linked and compiled after loading a mesh, using the define drop down menu. The menu path is: Define-User Defined-Interpreted. Browse to the file containing the following code, then use the FLUENT command interpreter to make the UDF ready for use.

```
#include "udf.h"

/* Constant Ur - wind velocity at ref height */
/* Constant Yr - reference height */

#define Ur 12.0
#define Yr 50.0

DEFINE_PROFILE(x_velocity, t, i)
{
  real x[ND_ND];
  real y;
  face_t f;

  begin_f_loop(f,t) /*loops over all faces in thread passed in Define Profile*/
  {
    F_CENTROID(x,f,t);
    y = x[1];
    F_PROFILE(f,t,i) = Ur*pow((y/Yr),(1./7.));
  }
  end_f_loop(f,t)
}
```

### FLUENT Inlet Boundary Profile File

This sample data file contains inlet boundary data formatted from a WRF forecast. The first part (variable name `u_WRF_NE mesh 1 49`) profile sets the boundary across the west side of the domain, starting from the origin. The second part (`w_WRF_NE mesh 1.49`) sets the profile across the north. There are seven data points at each specified location along the z axis, 990 meters apart for even spacing across the side of the domain. In this example, only velocity components are specified. Data for k and epsilon can be used in the profile along with the wind speeds. To add data for k (TKE), just add it below the w velocity component, with one value of k for each specified data point in the profile.

The co-ordinate system for the profile is right handed with y as the vertical axis. The length scale is meters, y axis elevation data is in meters above sea level, and wind speeds are in meters per second. The y axis data is a set of seven elevation points 30 meters apart for each x, z point in the profile. In the velocity data, u is the x direction wind speed component, y is the vertical component and w is the z direction component. The origin is in the upper right or northwest corner of the domain.

((u\_WRF\_NW mesh 1 49)

```

(x
1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    )
(y
1014 1044 1074 1104 1134 1164 1194 1005 1035 1065 1095 1125
    1155 1185 999 1029 1059 1089 1119 1149 1179 999 1029
    1059 1089 1119 1149 1179 990 1020 1050 1080 1110 1140
    1170 991 1021 1051 1081 1111 1141 1171 987 1017 1047
    1077 1107 1137 1167 )
(z
1    1    1    1    1    1    1    991  991  991  991  991
    991  991 1981 1981 1981 1981 1981 1981 1981 1981 2971 2971
    2971 2971 2971 2971 2971 3991 3991 3991 3991 3991 3991
    3991 4981 4981 4981 4981 4981 4981 4981 4981 5551 5551 5551
    5551 5551 5551 5551 )
(u
9.80 10.48 10.78 10.96 11.09 11.22 11.35 9.80 10.48 10.78 10.96 11.09
    11.22 11.35 9.80 10.48 10.78 10.96 11.09 11.22 11.35 9.80 10.48
    10.78 10.96 11.09 11.22 11.35 9.80 10.48 10.78 10.96 11.09 11.22
    11.35 9.80 10.48 10.78 10.96 11.09 11.22 11.35 9.80 10.48 10.78
    10.96 11.09 11.22 11.35 )
(v
-0.03 -0.04 -0.05 -0.06 -0.06 -0.06 -0.07 -0.03 -0.04 -0.05 -0.06 -0.06
    -0.06 -0.07 -0.03 -0.04 -0.05 -0.06 -0.06 -0.06 -0.07 -0.03 -0.04
    -0.05 -0.06 -0.06 -0.06 -0.07 -0.03 -0.04 -0.05 -0.06 -0.06 -0.06
    -0.07 -0.03 -0.04 -0.05 -0.06 -0.06 -0.06 -0.07 -0.03 -0.04 -0.05
    -0.06 -0.06 -0.06 -0.07 )
(w
5.75 6.30 6.64 6.92 7.18 7.48 7.80 5.75 6.30 6.64 6.92 7.18
    7.48 7.80 5.75 6.30 6.64 6.92 7.18 7.48 7.80 5.75 6.30
    6.64 6.92 7.18 7.48 7.80 5.75 6.30 6.64 6.92 7.18 7.48
    7.80 5.75 6.30 6.64 6.92 7.18 7.48 7.80 5.75 6.30 6.64
    6.92 7.18 7.48 7.80 )
)

```

((w\_WRF\_NW mesh 1 49)

```

(x
1    1    1    1    1    1    1    1021 1021 1021 1021 1021
    1021 1021 2011 2011 2011 2011 2011 2011 2011 2011 3001 3001
    3001 3001 3001 3001 3001 4021 4021 4021 4021 4021 4021 4021
    4021 5011 5011 5011 5011 5011 5011 5011 5011 5551 5551 5551
    5551 5551 5551 5551 )
(y
1014 1044 1074 1104 1134 1164 1194 1005 1035 1065 1095 1125
    1155 1185 1018 1048 1078 1108 1138 1168 1198 1018 1048
    1078 1108 1138 1168 1198 1022 1052 1082 1112 1142 1172
    1202 1022 1052 1082 1112 1142 1172 1202 1027 1057 1087
    1117 1147 1177 1207 )
(z
1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    )
(u
9.80 10.48 10.78 10.96 11.09 11.22 11.35 9.80 10.48 10.78 10.96 11.09
    11.22 11.35 9.80 10.48 10.78 10.96 11.09 11.22 11.35 9.80 10.48
    10.78 10.96 11.09 11.22 11.35 9.80 10.48 10.78 10.96 11.09 11.22
    11.35 9.80 10.48 10.78 10.96 11.09 11.22 11.35 9.80 10.48 10.78
    10.96 11.09 11.22 11.35 )
(v
-0.03 -0.04 -0.05 -0.06 -0.06 -0.06 -0.07 -0.03 -0.04 -0.05 -0.06 -0.06
    -0.06 -0.07 -0.03 -0.04 -0.05 -0.06 -0.06 -0.06 -0.07 -0.03 -0.04
    -0.05 -0.06 -0.06 -0.06 -0.07 -0.03 -0.04 -0.05 -0.06 -0.06 -0.06
    -0.07 -0.03 -0.04 -0.05 -0.06 -0.06 -0.06 -0.07 -0.03 -0.04 -0.05
    -0.06 -0.06 -0.06 -0.07 )
(w
5.75 6.30 6.64 6.92 7.18 7.48 7.80 5.75 6.30 6.64 6.92 7.18
    7.48 7.80 5.75 6.30 6.64 6.92 7.18 7.48 7.80 5.75 6.30
    6.64 6.92 7.18 7.48 7.80 5.75 6.30 6.64 6.92 7.18 7.48
    7.80 5.75 6.30 6.64 6.92 7.18 7.48 7.80 5.75 6.30 6.64
    6.92 7.18 7.48 7.80 )
)

```

APPENDIX D

**Life After FLUENT**

GAMBIT and FLUENT are part of the ANSYS software product family. ANSYS products are industry leading, top quality, well documented, supported and proven. FLUENT is widely used and well accepted in the CFD community as the industry standard. Most competing CFD software aims to be as good as FLUENT, or to provide some specific benefit to the user compared to FLUENT. Unfortunately, ANSYS products are among the most expensive on the market. Many large corporations with a specific need, like airplane manufacturers, are willing to pay for the best software available. For small companies, like wind energy consultants, or wind farm developers, a single seat license for GAMBIT and FLUENT is a prohibitive expense.

Fortunately, there is an alternative. OpenFOAM is an open source CFD product that can compete with FLUENT and other commercial CFD packages. Open source software is provided for free download and use. The open source model allows widespread peer review, and open source code developers use the knowledge of the user base to enhance and improve the product. OpenCFD Ltd. offers free CFD software that runs on Linux operating systems. OpenFOAM source code can be downloaded from the company website: [www.opencfd.co.uk/openfoam](http://www.opencfd.co.uk/openfoam). It has official documentation, including a user guide with tutorials. Many papers, case studies and peer reviewed journal articles about OpenFOAM are available online. OpenCFD generates income by selling support and consulting services. OpenFOAM training is regularly offered in several locations around the world for a reasonable fee.

Running OpenFOAM is very different than FLUENT. OpenFOAM is a set of C++ libraries that run specific CFD models. OpenFOAM meshing tools are part of the same environment and run just like the solvers. There are many utilities for mesh format conversion, post-processing and other operations in the libraries. The current version of OpenFOAM does not have a GUI interface. All OpenFOAM operations are run from Linux terminal windows and editors. OpenFOAM dropped GUI support because it was draining their limited resources and a customer survey indicated that very few customers were using it. Fortunately, the software is designed so that a GUI is not necessary. The third party post processor ParaView has a GUI interface, and is part of the OpenFOAM installation package. The ParaView website has a user guide and tutorials available for download. ParaView offers a full range of post-processing tools, views and options.

After taking the two day OpenFOAM introductory course, it's clear that OpenFOAM can fully replace FLUENT for atmospheric flow modeling. A methodology, similar to the one outlined can be developed in OpenFOAM. OpenFOAM has all of the turbulence models available in FLUENT. The User Guide has excellent documentation of basic operations. OpenFOAM source code is written in a simple, consistent style, and is considered as part of the documentation. OpenFOAM expects users to open source code files and reading the formulas to understand how things work. Source code is easy to modify, and the developers expect advanced users to modify source code. The second day of the introductory course mostly covered how to make simple, application specific modifications to source code. OpenFOAM operates in parallel, three dimensional mode

by default, so it's designed to run on a multiple microprocessor computer or a computer cluster.

After a few days of investigation, a method of importing DEM surfaces to mesh looks reasonable. USGS DEM data processed in MICRODEM can be saved in .obj format, which can be handled by OpenFOAM. The standard k-epsilon turbulence model is available in OpenFOAM and model constants are easy to change. Post processing tools like contour plots, vector plots and streamlines are part of ParaView.

OpenFOAM is not designed like many commercial software products, so it's not the easiest package to learn. After the introductory course, the advantages to the design of OpenFOAM are clear. This CFD product is very capable of producing a high quality atmospheric flow model, accurate results, and an efficient modeling process. It will take some time to learn the software, and develop a method to duplicate the FLUENT model. I'm looking forward to it.

APPENDIX E

**AWEA WindPower Posters**

# Application of Fluent for Atmospheric Flow: Comparison to WAsP in Complex Terrain

Alan Ruseell, Paul Dawson and Todd Haynes, Boise State University

## Introduction

Wind power forecasting in Idaho and the Rocky Mountains is difficult. Popular software tools like WAsP are designed for relatively level terrain. Computational Fluid Dynamics (CFD) can model atmospheric flow over any terrain. This objective of this project is to design a process for modeling wind flow using the engineering CFD code, Fluent. After designing the CFD process, WAsP and Fluent results will be compared for accuracy. The ease of use, time and computer requirements for each process are also compared.

The location of this study is Cinder Cone Butte, near Boise, Idaho. Besides being close to home, Cinder Cone Butte is an isolated hill on level terrain. Since it was the location of EPA tracer gas experiments in the early 1980s, measured data is available for wind speeds at the base and summit, in addition to the tracer gas dispersion data. With the availability of the EPA data, Cinder Cone Butte has been the location for several studies of CFD methods for modeling atmospheric flow. EPA data and subsequent research are extensively used in this research project to test the validity of the WAsP and Fluent models.

## WAsP Observed Wind Climate

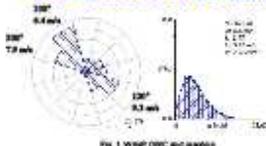


Fig. 1 WAsP OWC and graphs

## WAsP Process

- Wind Data from 82m research tower managed by Idaho National Lab (INL).
  - Tower location: 16.5 km NINE of Cinder Cone Butte
  - Data processed with WAsP OWC Wizard
- WAsP Map – free download DEM 10 meter data processed with MATLAB (available at College of Engineering)
  - Surfer software is easier
- WAsP Accuracy – adjusted roughness rose to wind farm
  - Data from Lewandowski wind farm 12.5 km N of Cinder Cone Butte
  - WAsP output within 4% of wind farm tower data
- WAsP Output – Wind rose, Weibull wind speed distribution and wind power density data for any turbine site placed on the map.
  - Cinder Cone Butte turbine site on summit 60 m. hub height.



Fig. 2 Cinder Cone Butte map

## Cinder Cone Butte

- Current MET data available, additional data from nearby wind farm, close to Boise (Fig. 2)
- Extensive EPA studies in the 1980s, including anemometer tower data, tracer gas studies, etc.

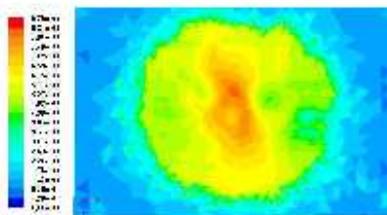


Fig. 3 Fluent velocity contours in a horizontal cross-section

## CFD Process

- Generate elevation data files from DEM using MATLAB
- Import data files into SolidWorks to make spline curves
- Loft single surface over all curves, form solid model
- Export ACIS model. Rotate model to match wind direction
- Import ACIS solid into Gambit
- Make volume around solid model large enough to contain pressure change distribution caused by the hill
- Mesh surface using 2.5 meter quad cells
- Use size function starting from surface to capture boundary layer
- Mesh size ~ 2 million cells (Fig. 4)



Fig. 4 Mesh Cross Section

- Open mesh file in Fluent
- K-s turbulence model with modified constants for atmospheric flow
- Define inlet wind velocity profile using 1/7 power law in Fluent user defined function
- 2<sup>nd</sup> order upwind solution
- Convergence in approximately 1900 iterations
- Post-processing includes contour, vector and pathline images and velocity plots
- Compare velocity contour plots on transform surface parallel to ground at hub height to determine optimal turbine placement
- Fluent results can be used to calculate wind shear, turbulence, flow divergence (Fig. 7), location of highest wind speed (Fig. 3), and wind speed change over terrain (Fig. 5).

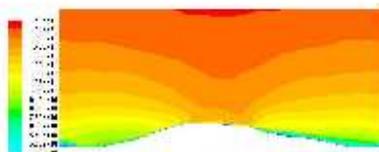


Fig. 5 Fluent velocity contours in a vertical cross-section

Acknowledgements:  
The authors thank Dr. Inanc Senocak, Boise State University, College of Engineering and Kurt Myers, Idaho National Lab for their help with this project.

## Cinder Cone Butte Results

- Fluent CFD model more accurate
- Fluent under-predicts EPA Data by ~11%
  - More work needed on model
- CFD model compares well with research papers for flow pattern and speed at several elevations (Fig. 7).
- WAsP misses the speed increase over Cinder Cone Butte
  - WAsP forecasts wind farm data within 4% over level terrain

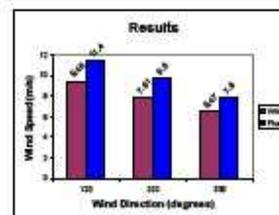


Fig. 6 Wind speed positions at Cinder Cone Butte summit

## Compare WAsP and Fluent

### WAsP

- Easy to learn with tutorial and program help
- Wind data input is easy with wizard
- Map is easy with Surfer software, more knowledge and time required with MATLAB Maps in WAsP format are easily adjusted
- Results are easy to generate
- Recalculating results (new input) is very fast (mouse-click)
- Some program tuning is available
- Runs well on laptop PC
- WAsP results are easy to understand, good graphics output
- WAsP is widely used in the industry and its capabilities and limitations are well known. Many research papers available on the web.



Fig. 7 Horizontal direction of flowline patterns

### Fluent

- Scientific literature survey to determine the most appropriate turbulence model and model inputs – k-s constants, etc.
- Fluent is used in atmospheric research, but not extensively. Some research papers available on web.
- CENER research group in Spain uses Fluent for wind resource assessment for complex terrain.
- Fluent requires extensive knowledge of CFD to generate reasonable results
- Ran on dedicated dual processor Linux workstation.
- Approximately 12 hours of computer time to run model for each wind direction.
- Project is part of masters thesis research

## Application of WRF and Fluent for Wind Energy Forecasting

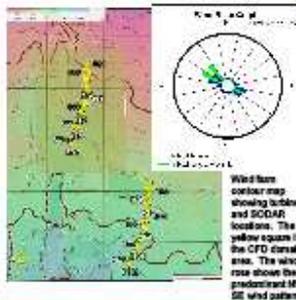
Alan Russell, Kevin Nuss, Paul Dawson and Todd Haynes, Boise State University

### Project Objectives

#### Forecasting for Wind Energy Grid Integration

- Develop forecast methodology
- Reliably forecast power production in MW
- Produce 5 minute, 1 hour and 24 hour forecasts
- Weather forecast using WRF, Weather Research and Forecasting, a mesoscale meteorological model
- Couple WRF with FLUENT computational fluid dynamics (CFD) model
  - CFD model for individual wind farms
  - Complex terrain effects
  - Buoyancy effects
  - Wind fields and turbulence
- Develop met tower network and SODAR field data to improve accuracy and forecast ramp events
- Evaluate forecast model accuracy

### Wind Farm



### Forecast Network



Top instrument height, distance, direction and ramp forecast time from wind farm

- 80 m tower 69 km NW ~ 3 hours upwind
- 30 m tower 51 km NW ~ 2 hours upwind
- 50 m tower 5 km NW
- SODAR at the wind farm
- 50 m tower 56 km SE ~ 2.5 hours upwind

### CFD Model

#### Mesh inputs

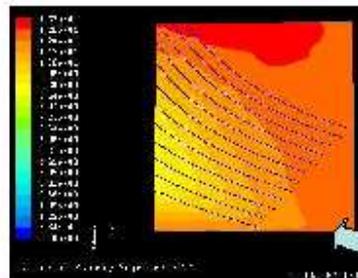
- 30 meter USGS DEM (digital elevation map) data
- DEM turned into Gambit (ANSYS mesh software) face
- Mesh domain built above surface
- Hexagonal mesh with surface boundary layer
- 2 meter resolution in surface layer
- 2 million cells model a 2.4 km by 2.6 km by 600 meter volume domain.

#### CFD Model

- FLUENT: ANSYS commercial CFD code for science and engineering
- k- $\epsilon$  turbulence closure
- Good compromise between accuracy and speed
- Second order upwind scheme
- Numerical solution convergence requires up to 12 hours run time on a 2 processor Linux workstation

#### FLUENT inputs

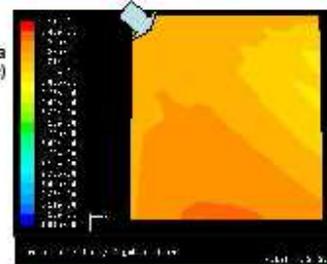
- WRF results for vertical wind speed profiles, wind direction,  $k$  and  $\epsilon$  along 1 km inner nest boundary
- WRF output converted into FLUENT inlet velocity profile data grids
- Set domain boundaries for wind direction
- Non-standard  $k$ - $\epsilon$  model constants for atmospheric flow
- Surface roughness height
- Surface heat flux for buoyancy model



FLUENT buoyancy model output: wind speed contours at 80 m. on 1/4/09 at 1 PM. WRF input wind was 100° (blue arrow) at 11.4 m/s. SODAR recorded 11.2 m/s wind at 100° at 80 m during the same time period

Thank you for assistance and advice:  
 Dr. Inacio Senocak and Marty Lukes, Boise State University College of Engineering  
 Dr. John Gardner, Boise State University Dept. of Energy Research, Policy and Campus Sustainability  
 Kurt Myers, Idaho National Lab  
 Doug Taylor, John Deere Wind Energy  
 Second Wind Triton SODAR  
 Bonneville Power Administration (BPA)  
 Research Funding: BPA Contract 00039902

### Results

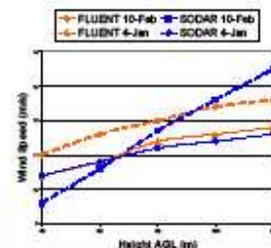


FLUENT neutral model output: wind speed contours at 80 m. on 2/10/09 at 6 AM. WRF input wind was 318° (blue arrow) at 12.51 m/s. At 80 m, SODAR recorded 13.2 m/s at 310°

#### CFD Output

- CFD model outputs velocity at hub height
- Velocity and turbulence across rotor plane of each turbine
- CFD process is too slow for real time forecasting
- Use CFD to model
  - Neutral and buoyancy driven flow
  - Diurnal and seasonal effects
  - Wind ramp events
- Project goal: Generate look-up tables using CFD simulations to match the wind rose

### CFD & SODAR



### Next Steps

- Complete data network
  - Install 30 and 50 m towers
  - Update existing towers
- Refine FLUENT model
- Generate CFD look-up tables
- Compare model power forecast to wind farm SCADA
- Run actual forecasts
- Complete final BPA grant report

