AeroDyn Overhaul Kick-Off Meeting

National Wind Technology Center (NWTC)
National Renewable Energy Laboratory (NREL)

Wednesday – Thursday, February 13 – 14, 2008

Agenda

February 13 — AeroDyn Theory and Structure

9:00 Welcome / Introductions
- Introductions (All)
- Agenda and Purpose of Meeting (JJ/SB)

9:30 Overview of Existing Code
- What is It (PM)
- Inputs (PM)
- Outputs (PM)
- AeroDyn Flowchart (PM)
- Wake Modeling – Blade / Element Momentum (BEM) (PM)
- Wake Modeling – Generalized Dynamic Wake (GDW) (PM)
- BEM Correction Models – Hub loss, Tip loss, Skewed flow (PM)
- Dynamic Stall (PM)
- Rotational Augmentation (PM)
- Tower Shadow (PM)

10:15 Overview of Existing Problems and Proposed Enhancements
- Reasons for Overhaul (JJ)
- Existing Problems (JJ)
- Proposed Modifications (JJ)

10:30 Break

10:45 Detailed Description of Problems, Proposed Enhancements, and Discussion
- AeroDyn Coordinate Systems (JJ)
- Binary Wind Files (BJ/NK)
- AeroDyn Wind Outputs (BJ/NK)
- Rotational Augmentation (PM)
- Interpolation of airfoil Data (MB)
- BEM Solution Iteration (MB)
- Influence of Drag on Induction (MB)

- Hub and Tip Loss Models (PM)
- Coned Rotor Corrections (PM)

12:30 Lunch

13:30 Detailed Description of Problems, Proposed Enhancements, and Discussion (cont)
- Initialization of GDW (MB)
- GDW Flow States (PM)
- GDW Stability – Turbulent Wake State (PM)
- GDW Stability – Inflow Velocity (PM)
- Dynamic Stall Models (Galbraith/et al, CENER, ONERA) (RG/XM/GB)
- Gaussian Quadrature (GB)
- Time Integration Issues (GB)
- Aerodynamic Model Interaction (PM)

15:00 Break

15:15 Detailed Description of Problems, Proposed Enhancements, and Discussion (cont)
- Impact of Turbine Motions and Turbulence on Wake (JJ)
- Rotor Motion Effects (PM)
- Direction Shear and its Influence on Induction (AS)
- Non-Rotor Aerodynamic Loads (MB/PM)
- Wake Tracking (PM)
- Tail Fin Aerodynamics (JJ)
- Apparent Aerodynamic Mass Effects (GB)
- Considerations for Water Turbines (JJ)
- Linearization of Wake and Dynamic Stall Models (JJ/GB)

17:00 Adjourn

18:30 Dinner at the Red Lion Inn [38472 Boulder Canyon Drive]
- Going "Dutch"
- Reservations made for 25 – 30 people with the open menu and open bar option
- East Room of the Main Dining Room requested

February 14 — Interfacing of AeroDyn with Structural-Dynamic Codes

9:00 Summary of Structural-Dynamic Codes that Interface with AeroDyn
- Coupled Aero-Hydro-Servo-Elastics (JJ)
- YawDyn (MB)
- SymDyn (KS)
- FAST (JJ)
- MSC.ADAMS (JJ)
- MBDyn (JJ)

- SIMPACK (SH)

10:00 Overview of Existing Interface
- Overview of Existing Interface (JJ)

10:15 Overview of Existing Problems and Proposed Enhancements
- Reasons for Overhaul (JJ)
- Existing Problems (JJ)
- Proposed Modifications (JJ)

10:30 Break

10:45 Detailed Description of Problems, Proposed Enhancements, and Discussion
- Coupling Aerodynamic and Structural Modules – Issues (GB)
- Co-Simulation (AE)
- Modularization of AeroDyn (JJ/MB)
- Development of a “Standardized” Interface (JJ/MB/SB)

12:30 Lunch

13:30 Detailed Description of Problems, Proposed Enhancements, and Discussion (cont)
- Requirements for the Introduction of Advanced Wake Theories (SG)
- Passing Data Back and Forth (JJ)
- Static or Dynamic Library (JJ)
- NWTC Subroutine Library (MB)
- Stand-alone Aeroacoustic Module (PM)
- Input-File Format (MB)
- AeroDyn Output Files (MB)
- Other? (All)

15:00 Break

15:15 Review and Wrap-Up
- Plan Forward (JJ)
- Review of Proposed Changes (JJ)
- Discussion Forum (MB)

17:00 Adjourn

Attendees

Name	Initials	Company / Institution	Country
Mark Balas	MB2	University of Wyoming	USA
Gunjit Bir	GB	NREL	USA
Jim Bleeg	JB	Garrad Hassan	USA
Marshall Buhl	MB	NREL	USA
Sandy Butterfield	SB	NREL	USA
Ken Cheney	KC	Independent Consultant	USA
Jason Cotrell	JC	NREL	USA
Jeff Darrow	JD	NREL	USA
Stephane Eisen	SE	DONG Energy	Denmark
Andy Elliott	AE	MSC Software	USA
Lee Jay Fingersh	LJF	NREL	USA
Roderick Galbraith	RG	University of Glasgow	Scotland
Brandon Gerber	BG	General Electric	USA
Sandeep Gupta	SG	Clipper	USA / India
Stefan Hauptman	SH	University of Stuttgart	Germany
Bonnie Jonkman	BJ	NREL	USA
Jason Jonkman	JJ	NREL	USA
Neil Kelley	NK	NREL	USA
David Laino	DL	Endurance Wind Power	USA
Mikel Lasa	ML	CENER	Spain
Mauricio Leal	ML2	University of California at Davis	USA
Tim McCoy	TM	Global Energy Concepts	USA
Paul Migliore	PM2	Independent Consultant	USA
Pat Moriarty	PM	NREL	USA
Xabier Munduate	XM	CENER	Spain
Walt Musial	WM	NREL	USA
Francisco Oyague	FO	NREL	USA
Kirk Pierce	KP	General Electric	USA
Mike Robinson	MR	NREL	USA
Dave Simms	DS	NREL	USA
Scott Schreck	SS	NREL	USA
Akihiro Suzuki	AS2	Wind Energy Institute of Tokyo	Japan
Andy Swift	AS	Texas Tech.	USA
Kyle Wetzel	KW	Wetzel Engineering	USA
Bob Wilson	BW	Oregon State University	USA
Alan Wright	AW	NREL	USA

Attendees via Net-Meeting / Conference Call

Name	Initials	Company / Institution	Country
Thanasis Barlas	TB	Technical University of Delft	Netherlands
Hugh Currin	HC	Oregon Institute of Technology	USA
Earl Duque	ED	Northern Arizona University	USA
Andreas Knauer	AK	Institute for Energy Technology	Norway
Fanzhong Meng	FM	Technical University of Delft	Netherlands
Jeff Minnema	JM	Windward Engineering	USA
Khanh Nguyen	KN	Clipper	USA
Mark Potsdam	MP	NASA Ames	USA
Robert Raulinson-Smith	RRS	Garrad Hassan	UK
Nilay Sezer Uzol	NSU	TOBB University of Economics and Technology	Turkey
Karl Stol	KS	University of Auckland	New Zealand
Ahmed Zobaa	AZ	University of Exeter	UK
Delong Zuo	DZ	Texas Tech.	USA / China

Minutes

NOTE: Please see the MS Power Point slides for the information that was formally presented. That information is not repeated here. The following minutes contain the discussion materials.

February 13 — AeroDyn Theory and Structure

9:00 Welcome / Introductions
JJ – We've been planning this for two years, but other, higher, priorities have postponed this project until now. The main goals are to develop an improved code and to create a better / standard interface.
SB – Is one of the UpWind projects working on a similar goal?
JJ – Garrad Hassan has created a standard DLL interface for controls. We want to develop a similar interface for aerodynamics codes.
SB – This is not about commercialization and sales, so let us please keep topics to the technical aspects of the project.

9:30 Overview of Existing Code

10:00 Overview of Existing Problems and Proposed Enhancements
AS – Can you discuss previous AeroDyn validation efforts?

JJ – A portion of my M.S. thesis involved the validation of FAST/AeroDyn. I also helped with validation in the Small Wind Research Turbine (SWRT) test. These reports are available in the NWTC Publications database.
KC – RANN compared AeroDyn's GDW and BEM models to the NASA/Ames data. The GDW model seemed to agree "pretty well;" BEM didn't work quite as well.
NK – I ran the LIST experiment. BEM seemed to match loads okay for the few cases we've tried.
TM – Our experience is that AeroDyn produces "fair" results and is doing OK.

10:45 Detailed Description of Problems, Proposed Enhancements, and Discussion

AeroDyn Coordinate Systems (JJ)
TM – Is the aerodynamic pitch/twist angle definition going to be changed? Is the right-hand rule preferred?
JJ – We have no plans to change the aerodynamic or structural pitch definitions. Positive pitch/twist is to feather, which is about the negative z-axis of the blade coordinate system (the x-axis is nominally downwind along the shaft and the y-axis points to the trailing edge). Notation is the same whether the turbine is upwind or downwind.
KC – What about the issues related to rotor rotation?
RRS – Bladed uses the GL-defined coordinate system, where positive pitch = positive twist angle. We have been wrestling with this for 20 years. There is no consensus on what is the best approach. Just make sure that the documentation is very clear as to the definition. All turbines rotate the same way these days so issues related to rotor rotation are not such a big deal.
TM – If there is a difference between coordinate systems in AeroDyn and the structural code, the transformations should be done in one place (not like currently where it is done multiple places)
JJ – Agrees that AeroDyn will use the GL / IEC coordinate systems
All – Agree that this is the best approach
AE – AeroDyn should not do the structural transformation computations. Don't blur the lines between structural and aerodynamic routines. The structural codes should compute the relative velocity of the wind relative to the blade element, which should then be passed to AeroDyn.
RG – We need the codes to be fully coupled.
KC – AeroDyn is already dealing with the wind, so why should the structural code have to know about the wind as well?
GB – All codes have their own coordinate systems. The structural codes should know the wind velocities in space. The relative wind velocities and the aerodynamic forces should be the one and only line of communication. AeroDyn should not need any knowledge of the wind.
AE – Every other aerodynamics code will have to do the wind computations if they're done in AeroDyn; mistakes are probable.
JJ – But if the wind transformations are computed by the structural code, then this logic must be in every structural code. The wind transformations are best done in a module within AeroDyn.
BW – The best approach depends on where the aerodynamic loads/forces are computed. Are they computed in the deflected or undeflected position and orientation?
JJ – AeroDyn and the structural codes currently share information in the deflected position and orientation of the blade elements.

KW – It appears to me that we are talking about three pieces of information: Structural / Aerodynamic / Wind. Where should the wind inflow go? Perhaps it should be in a third module?
RG – The aerodynamics routines often involve iteration; and the interface should not be developed so that the structural code calls AeroDyn and vice-versa. The aerodynamics and structural parts should be completely separate.
KC – AeroDyn needs more than just the relative wind velocities at the blade elements; some theories, such as vortex methods, depend on where the element is in space. We need to also pass the structural positions in the ground-based coordinate system to AeroDyn.
MB – Another issue is how the structural codes separate the high and low frequency components. The structural codes know this information and AeroDyn may have to as well.
JJ – Suggests we hold off continued discussion of these topics till we discuss modularization and the standardized interface.

Binary Wind Files (BJ/NK)
AE – Should the precision in the wind files be increased?
JJ – The precision is fine.
NK – Three decimal digits of precision is all that's needed.
MB – The current scheme gives five digits.
PM – What about changes in the wind direction?
BJ – Wind direction is included within the wind files, but AeroDyn still assumes that the turbulent planes march downwind in one direction.

AeroDyn Wind Outputs (BJ/NK)
MB/PM – We need to know the influence from induction when measuring the wind with anemometers downstream of the rotor.
MB – The inflow should be read by AeroDyn because AeroDyn knows the induction with which to influence the wind to be passed back from AeroDyn.
All – Agree that more wind information is needed than just the hub-height wind at the undeflected tower centerline. All of the proposed wind outputs are needed.

Rotational Augmentation (PM)
?? – Has rotational augmentation been an undocumented feature in previous versions?
MB – No, it has always been done within a preprocessor (AirfoilPrep or FoilCheck).
KC – Post-stall augmentation shouldn't have a whole lot to do with induction so it needn't be part of the BEM iteration loop.
PM – One of the rotational augmentation models does have something dependent on induction.
KW – The rotational augmentation models seem to blow up on occasion. When you do the calculations by hand, you can do a manual reality check. Can there by "reality" limits incorporated into AeroDyn if the rotational augmentation calculations are buried inside the code?
MB – We also plan to add an option to output the corrected airfoil data to tables so users can see how the airfoil data has been augmented. These can be output at all radial locations and at a variety of conditions (i.e., rotor speeds).
PM – Reality checks are certainly possible.
KW – What should be used in the reality check calculation? What about if drag rises before lift?

KC – The Eggers rotational correction model can't currently be implemented correctly within AirfoilPrep because some of the model parameters depend on some parameters that AirfoilPrep doesn't know about. The Eggers model must be implemented within AeroDyn. Strange results are thus not surprising.
KW – There ought to be an option to turn drag correction on/off.
KC – These lift and drag corrections are separate in the Eggers model.
MB – Mixing bits and pieces of different models is asking for trouble.
XM – Doing the calculation outside of AeroDyn can allow you to verify the correction and use for other codes.
AS – Using the method in the AirfoilPrep instructions sometimes gives strange results. What's the best way to do this?
KW – Having these corrections built in is nice, but is there consensus on whether it is worth doing or whether these are correct predictions?
AS – What is the current state of this research? It is not an issue in the helicopter industry; it is only an issue in the wind community. Therefore, money for research is not there.
RG – The CFD guys could work on it a bit.
SS – There are many assumptions in the models. Pretty much all of them are debunked, but is there anything better? No, it's simply a hard problem.
RRS – Hiding details in a black box (within AeroDyn) makes me nervous. There is no consensus in the models. 3-D effects are ignored in most turbines being built. The issue of rotational augmentation is academic with respect to modern turbine design because most turbines are pitch regulated.
SG – And because the rotational augmentation models are invalid at high yaw errors. One must also be careful that they don't double book the effects.
MB – However, not all machines are pitch regulated. We have to make AeroDyn work with all types and sizes of turbines, including stall regulated machines. On pitch-regulated turbines, we also have to deal with what happens when one blade goes haywire (i.e., fails to pitch).
KW – It would be nice to have some automation in this laborious process. Also, in most (even pitch-regulated) turbines the inboard stations' design point is near stall. A sudden gust condition can push the point past stall so rotational augmentation effects may be important even in modern turbine design.
All – Agree that is worth putting rotational augmentation in the code if there is an option to disable it.

Interpolation of airfoil Data (MB)
All – Agree that it is worth putting this feature into AeroDyn. It can always be disabled by supplying individual airfoil data at each node. As before, there will be an option to output modified airfoil data to a table.
BG – Can there be different options with regards to how to interpolate? By radial location? Thickness to chord ratio? User-defined interpolator?
MB – One problem is that the Beddoes-Leishman parameters are defined within the airfoil tables.
KP – These can also be computed within AeroDyn.
AE – Besides, users can still input airfoil data at each node if they want to.
KW – What happens if you don't have matching data at each airfoil?

BEM Solution Iteration (MB)
MB – BEM includes correction for tip loss, whereas the GDW model only includes tip loss if one retains a large number of induction modes
AE – In the GDW model, the Legendre polynomials don't have to be chosen in sequence, so one could feasibly look at the shapes to determine what modes are necessary to capture the tip loss correctly.
TM – Are the modes a programmers' choice or user defined?
AE – The modes are Legendre polynomials and are well defined.
MB – Are the modes affected by wind speed or structural characteristics?
AE – A little bit of up-front work can determine which modes are necessary.
MB – Would a MATLAB script be useful for examining the modes?
TM – It would be tedious if one had to change the selection of modes for each of the load cases; it would be easier just to use more modes than necessary.
?? – Is it a problem that there are no tangential modes?
PM – Tangential induction is included by applying the BEM correction model.
AE – Can tangential modes be added to the GDW model?
RG – Don't eliminate the BEM model. It's useful for comparison and basic calculations.
TM – BEM is also necessary when GDW is unstable. Should BEM and GDW be separate modules?
PM – The GDW instabilities are "fixed" in an alpha version of AeroDyn; but they still occasionally blow up.
?? – The BEM solution iteration problems are probably not part of the normal design space. Can error checks be put in?
All – Agree that the updated BEM algorithm is worthwhile

Influence of Drag on Induction (MB)
All – Agree to leave the option in there, but make the option documented instead of undocumented.

Hub and Tip Loss Models (PM)
XM – Is the Shen model valid for only one blade?
PM – Not sure; thinks it is valid.
JJ – Is there a consensus on which model to use? Ervin Bosanyi's book uses F*(1 – a*F), but Prandtl's original correction just had F*(1 – a). Should this be an option to the user?
MB – During a code comparison between FAST and Bladed that I performed 10 years ago, I thought Garrad Hassan admitted that the F*(1 – a*F) method was incorrect. NREL changed AeroDyn to match. I thought that the F^2 approach was double booking?
BW – Originally thought F^2 was appropriate; but now thinks that a single F is correct.
AE – Is there an article about this in Wind Energy?
JB – I checked with RRS and we agree that F*(1 – a) is correct. Bladed apparently does this now, too.
All – Agree to continue with F*(1 – a).
LJF – Are these models accurate? A real 3-bladed rotor has measured the tip effect at twice the distance predicted by the model.
MB – Do the hub and tip loss models need to be calculated by the structural code as is currently done?

LJF – Is an iterative process required? Can this actually be done, especially if the codes will be separated?
AE – Put the calculations within modules. One can then wrap an iterative loop around both codes. We do this in ADAMS all the time.

Coned Rotor Corrections (PM)
MB – With flexible blades and dynamic coning, do you use cone angles local to each element? Does Currin Crawford's model account for this?
PM – Is unsure.
AE – Neither BEM nor GDW work for large cone angles. Need some better wake models for this, such as vortex methods.
LJF – Are these corrections equally valid for upwind and downwind coning?
BW – Are the loads calculated in the deflected location or projected?

13:30 Detailed Description of Problems, Proposed Enhancements, and Discussion (cont)

Initialization of GDW (MB)
DL/AS2 – Neither recall why the 1-second initialization of GDW with BEM was chosen.
AE – What about modeling of start-up procedures? GDW will crash with unsteady conditions. Startup, shut down, etc are cases people will need to study.
MB – BEM doesn't work when the rotor is stopped either.
All – Agree that initializing GDW with a regular BEM solution during the first time step is the best approach.

GDW Flow States (PM)
SG – The tangential induction factor was incorrectly implemented per MB at some point. Has it been corrected?
PM – The tangential induction in BEM (not GDW) has been totally revamped in the latest alpha version of AeroDyn. In GDW, the tangential induction is calculated based on axial induction. No tip loss calculation currently in GDW; the roll-off of polynomials is not fast enough at the tips.
SG – Did we validate this with the NASA/AMES data?
AE – You should add an option to choose which states to keep, not just the number of states.

GDW Stability – Turbulent Wake State (PM)
SG – Should the turbine ever reach the turbulent wake state?
KP – This has been a problem at 8 m/s and below.
LJF – In the real world, there is no problem at low wind speeds or high TSRs; but the model can't handle certain regions.
KC – However, the model (as is) may run without crashing and generate loads that don't truly exist.
KW – Is the model completely incapable of modeling runaway conditions?
ML2 – How was the 8 m/s transition speed chosen? I've seen problems with 9 and 10 m/s, too.
DL – This was based on testing with a variety of models; of course the actual transition is model dependent.

JJ – The model only switches to BEM (based on the mean wind speed of 8 m/s and lower) when using turbulent wind. What happens when using hub-height wind files?
AE – The GDW model is only applicable to a subset of operational conditions. It can't be applied to all cases with low wind speed, high TSR, etc. One may have to go to simpler models in these cases.
KC – Fundamentally, the problem is that the model breaks down when induction is excessive. Maybe the check should be on induction instead of the wind speed? The induction factor might be the way to catch a lot of these issues. AeroDyn didn't generate errors when this was problem I had once.
LJF – Some of the load cases are pathological, and these models are invalid for these cases. We can't just "trap" these cases, we need better models. IEC has whacky cases and we need to get plausible results.
JJ – Suggests that we keep the current proposal. We will look into the instability when we reimplement the model.
XM – Are you going to validate the model with helicopter data? Can CFD models be used to validate these cases, too?

GDW Stability – Inflow Velocity (PM)
KC – What is the time constant?
PM – 32/(15*pi).
KC – Is this based on the propagation distance of the wake?
JJ – Suggests that we keep the current proposal. We will look into the instability when we reimplement the model.

Dynamic Stall Models (Galbraith/et al, CENER, ONERA) (RG/XM/GB)
JJ – Are there any technical issues associated with implementation of the new Glasgow model?
RG – No.
SS – NREL already has a copy of the code.
SG – The most critical part of the model is the prediction of the hysteresis loop. That's what results in reduced aerodynamic damping.
XM – We made no changes to the classical Beddoes-Leishman model. We simply used different time constants and other parameters.
AE – Is the curve fitting algorithm some sort of optimization method?
XM – It is a genetic algorithm.
XM – We would like to do more validation with more data.
RG – We could also optimize Glasgow's new model with this optimization method.
AE – If the model is implemented in state-space form, then one wouldn't need two separate integrators if you can use the same time steps.

Gaussian Quadrature (GB)

Time Integration Issues (GB)
GB – One can run into stability problems if the wrong iteration schemes are used with a given model.
AE – ADAMS equations are DAEs. The architecture of these modules is critical in developing a suitable integrator. There will be aliasing and synchronization issues and you can get completely

wrong answers if you don't use the right integrator. Users need to understand the way it's implemented so they don't use the code in ways that don't make sense.

Aerodynamic Model Interaction (PM)
SS – How does AeroDyn handle rotational augmentation verses dynamic stall?
DL – Looked at implementing rotational augmentation within AeroDyn, but some parameters needed to implement the rotational augmentation models weren't available. There may also be problems in the time marching scheme. Will the airfoil tables need to be recalculated at each time step? This may become a big problem to implement.
SG – One should be careful if they put rotational augmentation into AeroDyn. Logic is needed so that rotational augmentation and dynamic stall don't happen at the same time.
MB – Years ago when we compared the codes, Bladed said that GDW and dynamic stall were incompatible.
JJ – These are no longer mutually exclusive models in Bladed.
RG – Implicit in the Beddoes-Leishman dynamic-stall model is the shed vorticity. Vortex wake models determine the effect of the wake on the induced velocity. The wake and dynamic stall models should be applied at the same time.
JJ – Should the skewed wake correction be part of the iteration scheme?
MB – It is in WT_Perf.
XM – Dynamic stall may act over half of a rotor revolution. Therefore, it should affect the induction. The iteration scheme should be wrapped around everything.
KC – There was confusion years back on how the skewed wake should effect the induction. It should be applied using the disk-averaged induction instead of to the individual annulus.
JJ – We agree with KC; in fact, we were going to talk about this in the agenda but decided against it after thinking about it some more.
All – Agree that the iteration loop should be wrapped around all (wake, correction, DS) calculations.

15:15 Detailed Description of Problems, Proposed Enhancements, and Discussion (cont)

Impact of Turbine Motions and Turbulence on Wake (JJ)
LJF – If you leave in the high frequencies does it screw up the wake?
KC – Yes. There will be big swings in the induction with BEM. GDW should already have the filter, so it is only a problem in BEM. If you include high frequencies in the BEM calculations, the velocities can get large.
RG – Are there numerical problems associated with modeling both low and high frequencies?
AE – What range of scales are we talking about?
NK – We are talking about scales of 1 meter or so for the turbulence.
JJ – For the structure, the frequencies drop with larger turbines.
RG – Floating platforms, such as the barge, will also have very low frequencies; the wind will be of higher frequency.
AE – We don't know a priori what is a high or low frequency.
KC – Can we normalize by the chord length instead of the rotor radius?
TM – Previous versions of AeroDyn assumed that the turbines would not move much. With larger turbines this assumption may no longer be valid.
KC – It is probably important that the frequencies be split.

AE – Splitting the structural velocities is not reasonable.
KC – Filtering seems like the better approach.
All – Agree that filtering would be better than trying to define a transition frequency.
?? – Has anyone proposed how best to handle this issue?
AS – If it is up to the user, will it be sophisticated to determine what defines high and low frequencies?
JJ – Guidance should be given to the user.
LJF – Default options will be necessary with all these new options.
KC – During sharp edge gust events, the loads will change instantly, but the induction will not. We don't know how well the GDW model really handles this. The BEM model is still needed for a sanity check.
KW – The transition frequency ought to be calculated internally. There are gray areas – we should have some method for determining what high and low frequency is.

Rotor Motion Effects (PM)
All – Agree that nacelle-pitch motion should be treated like nacelle-yaw motion.

Direction Shear and Its Influence on Induction (AS)
AS – With larger rotors, direction shear is a problem. As in skewed wake, there are problems with the wake going in multiple directions. What does this do with loads?

Non-Rotor Aerodynamic Loads (MB/PM)
PM – Proposal is to add Bak model for the tower influence on flow and noninduced table look-up for drag.
SG – These would be very nice features to add.
KC – For the nacelle, a simple potential flow source/sink model would give a pretty good effect for influence of the nacelle on the flow.
MB – Is it worth including the influence of induction on the wind speed at the tower (by calculating the mean induction without iteration)? Also, should the tower influence effect the rotor induction?
SG – With stalled flow around the tower, there will be no induction.
JJ – He doesn't mean the tower induction, but the rotor induction.
KC – I am not sold on the tower wake having a big factor on the rotor induction. Induction on the rotor is a macroscopic rotor property. You're not doing the physics justice. It will be more affected by global wake. Think more macroscopically.
SG – Yes, keep it simple.
KC – The problem cancels out because the deficit is proportional to the wind speed.
RG – The reward isn't very great for some of this effort.
XM – Upwind and downwind machines should be treated differently.
All – Agree that we should add Bak potential flow model for the influence model and a Cd drag (or table look-up) force for the tower load.

Wake Tracking (PM)
PM – The GDW/BEM calculates induction only at the rotor plane. Should we try to estimate the wake in these models away from the rotor? This slide refers to the GDW/BEM models as opposed to the vortex wake model.

RG – Both prescribed and free wake models should be included in AeroDyn. A simple model should also be an option.
MB – Yes, that is the question; do we want both?
KC – Anemometers and tail fins tend to be in the core of the wake. Inboard elements are important here, but there is little induction inboard.
KP – We can't predict what the wind speed will be at an anemometer downwind of the rotor (say, on the nacelle)—that's a CFD problem. Simple models are of little value.

Tail Fin Aerodynamics (JJ)
LJF – The current method assumes that the induction at the tail is the same as it is at the plane of the rotor, which is invalid.
KC – The induced velocity at the tail using the current (simple) model is going to be incorrect. Many experiments have shown that the wind velocity is much different. This is a very messy problem. What we have done in the past is to calculate the tail fin loads both with and without the rotor-plane-averaged induced velocity. These calculations can be used to bound the problem in order to identify what range is expected.
RG – Improvement to the tail fin aerodynamics model should be of low priority. Only look into this after the major stuff is done.
LJF – Is this even worth addressing since it's going to be wrong anyway?

Apparent Aerodynamic Mass Effects (GB)
SS – The apparent mass effect is already included in the Beddoes-Leishman dynamic stall model. Are you asking if we should also include it when dynamic stall is disabled?
GB – Yes.
XM – One could check if the terms are important by removing features from the Beddoes-Leishman model.
GB – That is true for the torsion mode, but not for the flap motions.
SS – Are these additional effects important?
MB – Added mss is very importation for water turbines even if it's not important for a wind turbine.
SS – But the stiffness of the rotor also increases in a water turbine, so the mass effect may not be as important then either.

Considerations for Water Turbines (JJ)
RG – There should be no need to model the wave kinematics across the rotor. The rotors should be placed deep enough where this is not important or should be slowed down if they aren't deep enough.
KC – Could the buoyancy effect be captured by adjusting gravity?
KW – The Navy uses a variety of cavitation models. Can these be incorporated?
JJ – We should not try to implement a cavitation model into AeroDyn; just indicate through a warning or error message when the potential for cavitation exists.

Linearization of Wake and Dynamic Stall Models (JJ/GB)
GB – The MBC doesn't apply to GDW because the induction states are in the fixed frame.
AE – My dissertation looked at the Beddoes-Leishman model with fewer states than people use today. How useful this would be for controls work?

MB2 – These features are very important for controls design.

February 14 — Interfacing of AeroDyn with Structural-Dynamic Codes

9:00 Summary of Structural-Dynamic Codes that Interface with AeroDyn
JJ – What features does SymDyn have that FAST does not?
KS – Floquet theory and flapping (hinged) blades.
SG – Has AeroDyn (and the other codes) been certified by GL for offshore use or are there plans to do so?
JJ – The IEA Annex XXIII OC3 project is an offshore code-to-code comparison effort. We are hoping to use these results in place of the need to have GL certification.
SB – And GL hasn't participated in the day-to-day activities of the OC3 project.

10:00 Overview of Existing Interface (JJ)
AS – How is the disk-averaged induction calculated?
JJ – AeroDyn stores values during the current calculation of each element to be used on the next calculation. Thus, the disk-averaged induction averages the induction across all elements during the previous time step.
XM – What about backwards time steps?
JJ – AeroDyn ensures that a new time step is reached (DTAero past the previous time step) before recalculating the aerodynamic forces. If the structural code goes backward in time, then the aerodynamic forces are not recomputed. Also, AeroDyn will always use the predictor's values (never the corrector's values) of the integrator.
AS – How does the structural code know what the wind is?
JJ – The TurbSim wind is read into AeroDyn and passed from AeroDyn to the structural code.
AE – How is blade flexibility handled?
JJ – The relative speed between the blade motions and the undistributed wind velocities is sent from the structural code to AeroDyn in a coordinate system oriented with the deflected element. AeroDyn does its induction calculations, however, assuming that the velocities are relative to the undeflected disk. The element forces resulting from this calculation are assumed to be relative to the deflected blade element and then passed back to the structural code.
DL – We're really stretching the assumptions in all of the models here.
KW – Is the area of annulus reduced by the rotor deflections?
JJ – No, AeroDyn is essentially assuming small deflections.
KP – To summarize, the velocities and forces are calculated perpendicular to the blade element, but the induction calculation assumes that the calculations are being done relative to an undeflected disk.

10:15 Overview of Existing Problems and Proposed Enhancements

10:45 Detailed Description of Problems, Proposed Enhancements, and Discussion

Coupling Aerodynamic and Structural Modules – Issues (GB)
MB – I think we should share data through modules.
AE – It is dangerous to use modules to share data because the data can be easily overwritten. If you don't have tight control of when things are overwritten, you can get instabilities and lots of other problems. I'll explain this in my presentation.

Co-Simulation (AE)
AE – MSC works a lot with getting different codes to talk to each other. We know what kinds of architectures work and what kinds don't work. In this method, the difference in the time steps can't generally be more than two orders of magnitude. ADAMS always moves first so it's always 0-1 time steps ahead of the linked code, but it could work where another code is the driver. One code always has to lead with the other codes following.
AW – What is the difference between continuous time and discrete time integration? Aren't all codes numerical?
AE – Most structural codes operate in continuous time. The solvers use temporal shape functions in their predictors/correctors to account for what is happening within any particular time step. Discrete methods no nothing about what happens between time steps. ADAMS always assumes the process is continuous. If the new time solution doesn't match, it goes back in time and takes a smaller time step. In ADAMS' gear-stiff integrator, the order will go up to 6th order (but that's not a good thing)…. You should give it a smaller time step or figure out what's going on in your model. 2nd and 3rd order solutions are usually optimal.
MB – What portions of the interpolator/extrapolator do we have to codify?
AE – You'd have to write the sampler and the current force/control boxes. The two middle boxes are "freebies" and are provided with ADAMS.
AE – If you stick to Windows operating systems, pipes are recommended over DLLs. ADAMS is not written to handle multiple threads. But the Windows OS will automatically offload all the I/O onto one processor and the other will do all the calculations.
RG – It is great when each of the individual components integrates at its optimal rate, but couldn't you force them to run them all at the same rate to eliminate the problems?
AE – This is not always possible with control systems, for example.
GB – What happens with a near-singular Jacobian?
AE – ADAMS will warn before crashing.
All – Agree that the co-simulation approach where the structural codes and AeroDyn are combined through "glue" is the best approach.

Modularization of AeroDyn (JJ/MB)
NOTE: After much debate, we all agree on the following approach, illustrated in the image below:

- AeroDyn itself will be modular as proposed by GB's scenario #3. That is, there will be modules for processing the wind, modules for rotational augmentation, modules for induction, modules for dynamic stall, etc.
- AeroDyn will interface to the structural codes through AE's proposed "glue" using the co-simulation approach. Each structural code will have an interface to the glue (and be the leader in the solution). AeroDyn will also have an interface to the glue, and the main logic of AeroDyn will be in choosing which of the aerodynamic modules to use. As

different modules within AeroDyn may need their own time steps (such as time steps for the induction calculation being different than time steps for the dynamic stall calculations), the separate modules of AeroDyn may also have to be tied through more glue.

- Other (non-aerodynamic) models, such as the hydrodynamics module (HydroDyn) and the controls routines will also interface to the structural code through glue.

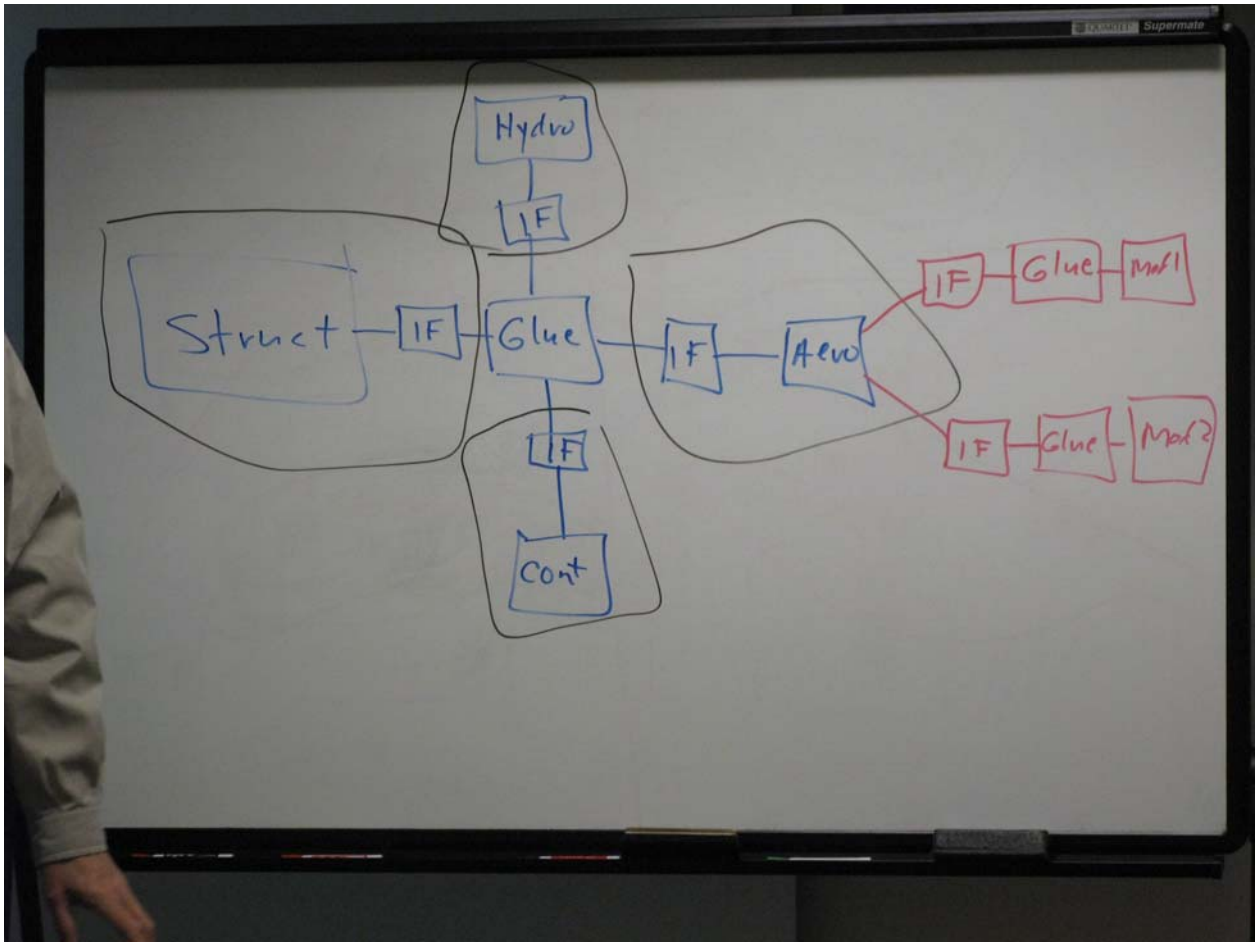

Figure. Agreed-Upon Interface Scheme Using Co-Simulation ("IF" in the figure indicates an interface)

AE –Who's going to be responsible for maintaining the interface?
RG – NREL should maintain the interface. The initial interface should be established based on consensus, then periodically updated as necessary.
LJF – What about extensibility and backwards compatibility? The best approach would be to pass data between the codes through structures (that is, structures are passed through the glue instead of variables). This way, new properties (i.e., new information) can be added to the structure without affecting anything else.
KS – I had originally thought that the integration of all states in the system (structure, aero, ...) could be handled by one integrator. The system is currently small enough to do this but future models would require far more states. For the future, I see the benefit of the co-simulation approach presented by AE where each component may have its own integrator. I think this

approach is analogous to digital control of physical plants - the interpolators are like A/D and D/A converters of a truly continuous process.

13:30 Detailed Description of Problems, Proposed Enhancements, and Discussion (cont)

Requirements for the Introduction of Advanced Wake Theories (SG)
?? – Have you tried different diffusion models?
SG – No.
AE – How does this model compare to CAMRAD?
SG – The CAMRAD model is the same except for the time integration scheme.
SG – Vortex methods are the next step before going to CFD. Vortex element length is discretized at 10 degrees (other steps were tried, but were not better). Calculation time increases by N^3 number of elements.
JJ – Are there any numerical problems with interacting wakes in highly skewed flow?
SG – No; there is no problem. There is enough diffusion in the model to handle it.
PM – Any problem with inflow turbulence?
SH – Has turbulent wind been tried before?
SG – The model was not been implemented with turbulence, but it should not be a problem to add. Each blade gets separate information. CAMRAD doesn't care about inflow turbulence because helicopters create their own wind.
JJ – What load cases most need these methods?
SG – The method works better than the BEM / GDW methods in yawed flow. For example, this method might be the best candidate for handling extreme direction changes (the IEC ECD and EDC events). Vortex methods would be good at solving this problem.
NK – With turbulence, the diffusion model is going to make a significant difference. It should be tied to the stability of the atmosphere. A field project to asses these effects would be difficult.
SG – Numerical diffusion makes it difficult to test this numerically. CFD models can't track more than one or two revolutions. LES has the same problem.
RG – The effect of the diffusion model is quite strong. Vorticity tracking methods help this problem a bit. You have to know what the limitations are, however. You can't try to solve the entire problem or you won't succeed.
LJF – Who will set this up to work with our models? It's different than AeroDyn, but modular.
XM – Does the method need look-up tables for the aerodynamic data?
SG – Yes, it uses look-up tables. We have not implemented a stall-delay model. The method has similar structure to BEM, except the way the induction is determined.
LJF – Is the method parallizable for multi-threaded computers?
SG – Yes.
LJF – Then we should consider paralyzing the code wherever possible. This is because the speed of individual processors is not rapidly increasing as it has in the past, but multi-threaded computers will become more and more popular in the future (next year, 16-thread PCs, and 8-thread laptops will be available). MB should be tasked with examining what it would take to develop explicit and/or implicit multi-threaded algorithms within AeroDyn.

Development of a "Standardized" Interface (JJ/MB/SB)
JJ – Proposes that the aerodynamic nodes be distinct from the structural nodes.

AE – They can't be completely separate. How can Gaussian Quadrature work when the aerodynamic forces need to be applied to parts in the multi-body codes?
GB – This is only a problem in ADAMS.
JJ – FAST is modal-based, so it doesn't matter. The distribution of aerodynamic forces within ADAMS (or other multi-body codes) needs more thought.
SH – SIMPACK also uses the modal form to model the blades in its time-marching schemes.
AE – How is the centrifugal stiffening accounted for in SIMPACK then?
SH – Through augmentation of the stiffness matrices.

Passing Data Back and Forth (JJ)
AE – Does FAST operate in the rotating or fixed frame?
JJ – Some states are in the rotating frame, others are in the fixed frame. The global solution is solved in the fixed (inertial) frame.
KC – AeroDyn will want the nodal discretization to be fixed across all time steps. Are there cases where—for example, extendable blades—this can be adjusted?
KW – Can any of the aerodynamic models/theories actually accommodate extendable blades?
ML2 – My M.S. thesis deals with the extendable blade concept, but we never tried to model the dynamics of the extension since the motion is slow. We simply have different models for the extended and retracted cases.
TM – We've simulated the extension and retraction of extendable blades. To do this, we hacked the current AeroDyn by changing the DeltaR on the fly (using it as a multiplier between full and zero).
All – Agree that we should not worry about the implementation of this now, but keep it in mind for future development.

Static or Dynamic Library (JJ)
Skipped.

NWTC Subroutine Library (MB)
AE – A nice feature would be to add a set of routines to read in XML-style input files.
LJF – Should XML be the standard format for the input file?
AE – It wouldn't be a big deal to implement. It is beneficial because there's no requirement for sensitivity of order. You can add stuff without changing your old inputs. Units (etc.) can be documented using comments.
DL – It's easy to compare files and debug problems with the current system. This will become harder with XML.
JJ – Users hardly ever need to work with the input files. Loads analysis scripts use a single "master" input file and create copies to run the individual load case simulations based on the load case prescriptions. Why bother changing what's not broken?
MB – We will put this on our "To-Investigate-List".
XM – I have had issues with some compiler-specific code in routines that are not compiler-specific.
MB – E-mail me to let me know so we can fix it.

Stand-alone Aeroacoustic Module (PM)
?? – What propagation model is used?

PM – Simple spherical spreading (some directional).
ML2 – Could Xfoil be used to generate the airfoil characteristics?
PM – Xfoil is not good past stall.

Input-File Format (MB)
GB – I would like to see the "research inputs" in a separate file.
KP – Or if it's there, read it; if it's not, use default settings.
JJ – As I said before, it doesn't really matter how long the input file is because users should hardly ever be messing with them. The load case scripts do that. I think that all of the inputs should be in one file to minimize confusion.
NK – Make sure to make all inputs self-documenting by sending the default selections to a summary file.
KW – In the summary file you should put the input selections in standard order, even if the XML format is used whereby the inputs are placed in random orders.
KW – Would it help to develop a GUI interface or Excel file for control of the program?
KP – We developed a GUI for YawDyn but people didn't use it. They'd rather use the input file. Besides, it's expensive to maintain. Whenever changes are necessary, the GUI has to be changed as well.
ML2 – I like the current method with one file.
HC – We need to add some scheme where new input parameters could be added easily. New inputs will be needed when adding new aerodynamic modules, like vortex wake models. We need a standardized way to add new inputs.
XM – Does XML work in other operating systems?
MB – Yes. XML is really just a text file.
GB – How does XML handle input arrays?
AE – It handles arrays just fine.

AeroDyn Output Files (MB)
KP – Writing out data in WT_Perf-style blocks would be too much information. These files would be huge and hard to plot.
NK – You could have users specify a start and stop time for this so you don't have so much information.
RG – This feature would be incredibly useful. The helicopter industry does this all the time.
?? – Will you make this feature available for all the blades?
MB – How do people analyze this type of data?
Most – MATLAB and Excel are used for plotting. This format may be easy to plot in MATLAB.
XM – GenPlot and/or TechPlot would work well also.
Most – Agree that this would be useful as an option.
KW – Can I ask that you don't bury the channel names within a text string? Instead, add it as an extra column so that it is easily identifiable in Excel. Don't say things like, "the following data is for wind speed," just output "wind speed" in a separate column.
KC – Can I also ask that you don't output headers? It makes reading the data in Matlab harder with the extra text.
MB – Should this become an option in the input file?
BJ – Reading and plotting these files could be an option in the new MCrunch.

All – Agree that it is not a big deal to have to specify the output column names for AeroDyn outputs within the structural codes.
AE – I am concerned that we will be passing too much data between AeroDyn and the structural codes. The amount of passed data should be kept to a minimum.
MB – Is it OK just to have separate output files for the aerodynamic and structural files?
Most – Agree that it would be nice to have both options—that is, both a single file and separate files.
KP – Isn't it a problem in Excel to have too many columns?
MB – Old versions of Excel could not process very many columns, but the newer version can.

Other? (All)
TM – What are the benefits of DLLs versus compiling separately?
JJ – Check the DLL slides we skipped.
PM – How does the debug stuff work with DLLs?
AE – If the DLL is compiled in debug, then you can debug through the DLL from the calling program.
SG – How are you going to set priorities and incorporate feedback from the users?
AE – How is the division of labor to be sorted out?
RG – You need to set goals for the project. What do you want to accomplish? What is the mission statement? What is the policy associated with implementation of AeroDyn? You should create a detailed structure (flow chart) before programming. How are you going to compare the results with previous results (debugging)? After you fix what is already implemented then you should start with the new features we'd like to see.
SB – Do we want animation capability?
AE – Would there be an advantage to tie together the structural codes and AeroDyn by recasting the unsteady aerodynamics into state-space form? We can't linearize with co-sim method.
JJ – Can't a "linearization flag" be passed through the "glue" to control linearization features?
AE – Perhaps. Also, the glue code may contain all of the derivatives you may need during linearization.
RG – How can stationary blades be modeled in the code, especially during hurricane-type winds?
MB – What is the loading of a blade pointing directly into the wind (or at an angle)?
RG – You may need CFD to solve this problem. This is not a trivial problem.

15:15 Review and Wrap-Up

Plan Forward (JJ)
MB – Should we change the name from "AeroDyn" to something else?
All – Agree that the name should not be changed unless there is a good reason to.
KW – What will the coupled modes do to the speed of FAST?
JJ – The coupled modes will slightly increase computation time due to the new off-diagonal terms, but this isn't a big deal.
AE – This schedule is very aggressive. More time should be devoted to AeroDyn's development.
RG – People could begin to develop the physics models once the interface has been defined.
AE – One shouldn't establish the architecture until you know if the ideas are going to work.

Review of Proposed Changes (JJ)
JJ – The top priorities are to redo the interface, restructure the code, reimplement what is already in AeroDyn (BEM, GDW, dynamic stall, etc.), and to add some of the features that are needed, such as tower influence and loading.

Discussion Forum (MB)
MB – Our design codes forum is at: http://wind.nrel.gov/forum/wind. Click on the AeroDyn development forum (or use this link: http://wind.nrel.gov/forum/wind/viewforum.php?f=19). Everyone should subscribe to "new topics" and to each specific topic they want to contribute to (to do this, go to the page for the topic and click on "watch this topic for replies" on the bottom left of the page).
All – Agree that the forum is better than the mailing list for ongoing feedback.
DS – Is it possible to do this work under a CRADA? If the wind industry contributes to a CRADA, it helps get work done and DOE will get the picture that it's important work. I have no specific request as to the amount of money that should be contributed to the cause, but whatever the amount (even if no funds are exchanged), it shows DOE we're leveraging with industry.
All – Participants had mixed reactions to the suggestion by DS. Many would have to check with management.
SB – I will work with JJ and MB to come up with options. I'd rather not spend the time making a CRADA (which is too much overhead time and hassle) if people wouldn't collaborate.
SB/JJ – Should we establish an annual user's meeting for our codes? Each year we would focus the meeting (such as this one) on the code(s) that we are putting our main development efforts into.
Most – Agree that this is a great idea and that they would come.
NK – We will be hosting a TurbSim workshop in September. At the Office of Science meeting in January, people expressed interest in doing site-specific design. This is possible with TurbSim. Is there a level of interest?
Most – Say yes.
KC – Can a repository for user-developed code be set up to share with others (like is available for MathWorks users)?
SG – Should a system such as CVS be used to document changes and revisions to the code? This would make it easy to roll back to previous versions?
JJ – This has not been a big problem so far as few people are working on any one code at any particular time.