NUMERICAL OPTIMISATION FOR WIND TURBINE DESIGN, BASED ON AERO-ELASTIC ANALYSIS

B.H. Bulder, J.G. Schepers
Netherlands Energy Research Foundation ECN
Westerduinweg 3, 1755 LE || post office box 1, NL-1755 ZG, Petten
tel. (+31)(0)224 56 4102, fax. (+31)(0)224 56 3214, email bulder@ecn.nl

ABSTRACT This paper describes the development of the HATOPT code which has been made during the SITEOPT project. The HATOPT code is based on the BLADOPT [1] code which did not make use of a sophisticated aero-elastic code but a simplified quasi static rotor performance code. The HATOPT code does not make use of the cost model of BLADOPT but a cost model made by the University of Sunderland.
Keywords: Wind turbine, optimisation, design tool.

1 INTRODUCTION

For the project “Site Specific Design Optimization of Wind Turbines” [2, 3] in short SITEOPT, a new design tool has been developed. This project, performed by RISØ (DK), University of Sunderland (U.K.), Bonus (DK), Lagerwey the WindMaster (NL) and ECN (NL), within the JOULE programme of the European Community. The objective of the project was to show that site specific optimisation is worthwhile. To perform these optimisations both, RISØ and ECN, upgraded their numerical optimisation design tool making use of the same cost evaluation model. This cost model which uses state of the art aero-elastic code load predictions, was developed by the University of Sunderland, see [4], in the SITEOPT project.

In the SiteOpt project the optimisation tool is used to show that site specific design optimisation can be worthwhile but this tool is useful in a much broader sense of course.

This paper is about the on going developmnent of the design optimisation tool HATOPT at ECN.

The objective or target function which was minimised in all optimizations performed was the Levelised Production Cost of Energy, **CoE**, determined on the basis of the procedure described in the IEA Recommended Procedures for the Cost evalulation.

The CoE is determined as follows

$$\mathrm{CoE} = \mathrm{I}/\left(\mathrm{a}\cdot\mathrm{AUE}\right) + \mathrm{TOM}/\mathrm{AUE}$$

in which

a	Annuity factor	-
AUE	Annual Utilized Energy	kWh
I	Investment including possible interest during construction	€
TOM	Total (levelized) annual “downline cost”	€

These parameters are calculated according to the following procedures:

Annual Utilized Energy yield: The energy yield is determined from the aero - elastic response predictions performed with PHATAS- SWIFT, [5, 6]. Resulting stand alone energy yield is reduced for site specific conditions, like wind farm efficiency, availability etc.

Investment cost of the wind energy system, including installation: with the cost model of University of Sunderland “feeding” it with the response predictions of the PHATAS– SWIFT aero-elastic wind turbine simulation package.

Down line cost: assumed to be the operational cost of the wind turbine which is proportional to the investment cost, the percentage is site dependend.

2 CODE DEVELOPMENT

The code developed is called HATOPT. The code is based on the previously developed code BLADOPT which can optimize rotor design parameters to minimize the CoE. However in the BLADOPT code, the load prediction and performance prediction code are simplified version of the industry standard of today. An advantage of the simplification is that it is very fast. The main disadvantage is of course that the quality of the wind turbine model is rather poor.

In the HATOPT tool more design variables are taken into account and to use a more sophisticated cost model a redesign of the optimisation tool is necessary. The main changes are

- a state of the art aero-elastic code is used to predict the wind turbine performance and load spectra;
- the cost model of the University of Sunderland is implemented using the relative changes with respect to the zero design to tune the results.

The HATOPT code consist of two optimisation modules, a zeroth order algorithm, [7] and a second order feasible quadratic programming algorithm, [8]. The 0th order algorithm is used because it is rather efficient for problems where derivative information of the objective function is costly, with respect to computer time, to determine. Zeroth order means that no derivatives of the cost function need to be determined to search for the optimum. The second order alghorithm means that derivatives up to second order of the cost function are used to determine the optimum. This second order algorithm will not make use of the true objective function but an approximation of this objective function, to prevent that the analysis takes to much (computational) time. This approximation of the cost function will be an algebraic function that can be evaluated very fast and derivatives can be obtained easily by determining the derivatives analytical or numerical. Therefor it is fast to evaluate the second order derivative information.

The approximation of the cost function consists of a multinomial fitted through data obtained from true cost function evaluations. The multinomial, a polynomial in the design parameters including the cross terms, is described in [9].

The main difference between the HATOPT code and the BLADOPT code is the use of a sophisticated aero-elastic response prediction code. In stead of the quasi static rotor code used in BLADOPT the PHATAS– SWIFT code combination is used. Initially this looks rather straight forward but it is rather complex to implement. Other items that were changed is the type and number of parameters that can be optimised, not only rotor blade related parameters but also tower and hub design parameters. In conjuction it was also necessary to implement all kind of *response* parameter constraints. A response parameter is e.g. a design load on the hub or tower top or the tip deflection of the rotor blade. One typical example of the SITEOPT project is to determine the optimum length of a hub extender keeping all other design parameters equal. The constraints were on the hub loading and tower top loading and the deflection of the tip times the safety factor should be less than the tip cleareance.

The main steps to be performed during the optimisation are:

1. determine the CoE for the start design F ($\underline{X}_0$), i.e.
   (a) predict aero-elastic performance;
   (b) determine cost of all components;
   (c) determine Cost of Energy;
   (d) determine ratio between masses and stiffnesses of aero-elastic model and the results of the cost model.
2. check constraints, on design parameters $\underline{X}$ and response parameters G ($\underline{X}$).
3. write data to database;
4. Optimisation module I determines a variation of the design vector $\underline{x}$. If optimum is found than goto 8;
5. update aero-elastic model
6. perform cost function evaluation:
   (a) predict aero-elastic performance;
   (b) determine cost of all components;
   (c) determine Cost of Energy;
   (d) multiply the results of the cost model, cost, stiffness and masses with the ratio's determined for the zero design.
7. goto 2
8. generate approximation functions of the objective function $P(\underline{X})$ and of the response constraint functions G ($\underline{X}$). Approximation of the objective Function $F(\underline{X}_0)$ is a multinomial fit of the $\underline{X}_i$ and the corresponding $\mathrm{Co}E_i$. The approximation of the constraint functions is a multinomial fit of the parameter in question as function of the design parameters $\underline{X}$;
9. Optimisation module II varies the design parameters $\underline{X}$ to minimise the $\mathrm{Co}E$;
10. evaluate the *cost function* $F(\underline{X})$ and the constraint function(s) $G(\underline{X})$
11. perform 6 and put results in database;
12. continue with 8 unless optimum has been reached.

The dataflow of the HATOPT code are shown in figure 1. As already mentioned using PHATAS– SWIFT for the re-

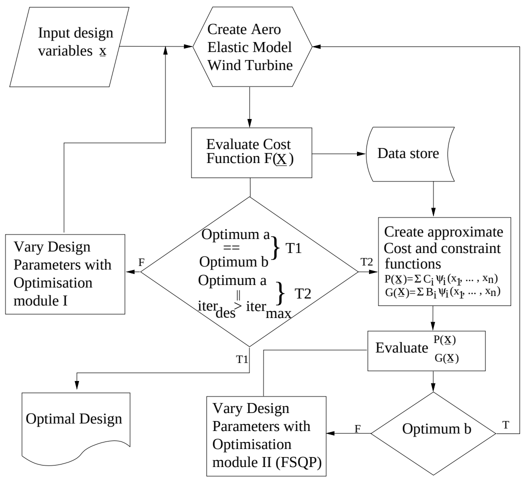

Figure 1: The flow chart for HATOPT code

sponse prediction means that one evaluation of the CoE takes several hours, on a state of the art computer, in stead of less than one second for the BLADOPT code. In conjuction with that, when optimising e.g. the rotor chord and twist distribution it is necessary to make an estimate for the mass and stiffness distribution on the basis of a new chord distribution to get the same dynamic properties as a result from the blade cost model as was used for the aero-elastic response prediction tool.

The command interface to the aero-elastic prediction tools is ProgSeq a shell tool able to produce and read replay files. Once a set of calculations has been described these calculations can be repeated for a different design with just one command with argument. A complete description of the dataflow with ProgSeq starting with the aero-elastic model and finishing with the input file for the cost model is in [10].

As indicated before the cost module determines the cost of the major components and does so on the basis of a design made of these components. Resulting in the mass and other properties.

The development of the HATOPT code needed a complete new implementation of the objective function based on aero-elastic response prediction, with the PHATAS-SWIFT code combination. Also the cost prediction code will make use of the response parameters, like the rotor and tower load spectra, to predict the cost of each major component. Next to the cost the cost model will also be able to predict the aero-elastic properties of the tower, drive train and rotor. To make sure that the dynamics of the cost model and the dynamics used in the aero-elastic simulations do not differ to much, without creating an extra iteration loop, some rules have been implemented predicting the effects of changes of the design parameters on the mass and stiffness distribution of the rotor blades and tower.

3 Cases

In the SITEOPT project a number of cases have been analysed of which some typical items to be tackled are listed below.

Towerheight and hub extender This case was about the optimisation of the a wind turbine for a number of specific site conditions. The freedom to change design parameters were only the length of a hub extender and the height of the tower. The constraints were that

- the fatigue and extreme loading on the connections between the tower top and nacelle and between the hub and the extender can not be higher than in the original design;
- the fatigue loading generated by the rotor transmitted through the gearbox should be less than the design loading on the gearbox;
- the deflection of the blade tip should not be so high that the maximum deflection times 1.5 would mean that the blad would hit the tower.

For the optimisation this is a reasonable simple case of only two parameters to vary and in total 16 constraints. The constraint on the tip deflection is satisfied implicitly by the cost module. The blade design should be optimised for cost taking into account the maximum allowable deflection.

For each new design the dynamic properties of the tower had to be predicted. The actual dynamic properties were a result of the cost model which redesigned the tower using the load spectra determined by the wind turbine simulation codes. For this case the cost module was tricked by letting it design a new tower using the load spectra of the zero design which were adapted to the wind speed distribution which was corrected for the tower height. The dynamics of the rotor blade were not changed only a relative stiff hub extender was introduced which did not change the rotor eigenfrequencies and modes to much.

Rotor blade length, chord, twist and thickness A new rotor blade should be designed taking into account the maximum loading which is allowed on the hub, drive train and tower.

To reduce the number of design parameters it was chosen to use the relative chord, twist and thickness distribution of the original blade and vary only two parameters for the chord, twist and thickness each. One parameter that influences the overall distribution and one that influences the taper linearly from root to tip.

The constraints are dealt with in a similar way as in the previous case.

This time the dynamics of the rotor blade will be influenced thus it is necessary to predict the change in mass and stiffness of the rotor blades when changing e.g. the length of the rotor blade.

Using the BLADOPT code many sensitivity analysis have been made changing the design parameters between the upper and lower boundary and the resulting blade mass/stiffness distribution is fitted with exponential functions. The resulting exponents are used in the module *generate aero-elastic model* to predict the changes in mass and stiffness of the rotor blade.

ACKNOWLEDGEMENT

This work performed within the scope of the SiteOpt project has been partially funded by the European Commission, JOR3-CT98-0273, and Netherlands Agency for Energy and the Environment NOVEM

REFERENCES

[1] Bulder B.H. and F. Hagg. "BLADOPT - A numerical optimization tool for rotor blades using cost of energy as the target function". In *Proceedings of the EWEC conference held at Dublin*, 06 – 10 October 1997.

[2] Fuglsang, P., C. Bak, J.G. Schepers, B.H. Bulder, T.T. Cockerill, P. Claiden, A. Olesen, and R. Rossen. "Site Specific Design Optimisation of Wind Turbines based on Numerical Optimisation". In *Proceedings of the European Wind Energy Conference, Copenhage, Denmark*, 2– 5 July 2001.

[3] Thomsen, K., P. Fuglsang, and J.G. Schepers. Potential for site-specific design of mw sized wind turbines. *Journal of Solar Engineering*, submitted, 2001.

[4] Cockerill, T.T., P. Claiden, P. Fuglsang, C. Bak, B.H. Bulder, and J.G. Schepers. Development of a new engineering cost model for horizontal axis wind turbines. *Wind Energy Journal*, submitted, 2001.

[5] Lindenburg C. and T. Hegberg T. "PHATAS – IV, Program for Horizontal Axis wind Turbine Analysis and Simulation version IV, USER'S MANUAL". ECN–C- 99-093, ECN, Feb 2000.

[6] Winkelaar, D. "Fast three dimensional wind simulation and the prediction of stochastic blade loads". In *Proceedings of the tenth ASME Wind Energy Symposium*, pages 5–14, January 1991.

[7] Powell M.J.D. A direct optimisation mehtod that models the objective and constraint functions by linear interpolation. In Gomez S. and Hennaart J-P., editors, *"Advances in Optimisation and Numerical analysis"*, pages 61–67. Kluwer Academic, Dordrecht, 1994.

[8] Zhou Jian L., André L. Tits, and Craig. T. Lawrence. "User's Guide for FFSQP Version 3.7: A FORTRAN code for solving constrained non lineair (miniMax) Optimization problems, generating iterates Satisfying all all inequality and Linear Constraints", September 1997. www.isr.umd.edu/Labs/CACSE/FSQP/fsqp.html.

[9] Richard H. Bartels and John J. Jezioranski. Algorithm 634: CONSTR and EVAL: Routines for fitting multinomials in a least-squares sense. *ACM Transactions on Mathematical Software*, 11(3):218–228, September 1985.

[10] Schepers J.G. "HatOpt: Modules and data flow to calculate the design loads and resulting cost of energy". CX 01-016, ECN, May 2001.