

Steam Turbine Start-up Optimization Tool based on ABAQUS and Python Scripting

Andreas Ehresam¹, Thomas Schreier¹, Philipp Brunner¹, Abhishek Mitra²

¹Alstom Power Switzerland Ltd., ²Infosys Technologies Ltd. India

Abstract: One key aspect for the design of fast and flexible steam turbine operation is thermal stresses arising during transient operation. If the stresses exceed the fatigue limits of the material, the lifetime of the steam turbine is shortened. Detailed finite element analysis is applied during design phase to assess the effect of transient temperature and stress profiles on the complex geometries. A significant amount of design effort is invested to determine the optimal process parameters for start-up (e.g. steam temperatures, run-up and loading gradients), in order to achieve the fastest possible starts without exceeding allowable material stress limits. The first step of the current practice is to derive the transient thermal boundary conditions for the whole start-up simulation, based on pre-defined process parameters. In a second step a finite element analysis is performed to verify these thermal boundary conditions. Using this sequential approach, a high number of iterations are required to arrive at the optimal process parameters. An automation process in the form of a design tool was developed to determine the optimal process parameters, by means of a feedback control algorithm. Using Python scripting, the tool interlinks the finite element package ABAQUS/CAE (Version 6.7 and above) with an Alstom in-house thermodynamic program, determining optimal transient thermal boundary conditions based on real time thermal stresses. Use of this tool eliminates the need for the high number of manual iterations previously required. This paper presents the concept of the optimization tool and how it interlinks the interdependent programs.

Keywords: Power Generation, Steam Turbine, Start-up, Thermal Stress, Optimization

1. INTRODUCTION

The world wide liberalization of the electricity markets led to the changes in the way power plant owners react to the market. Because of changes in demand and competition within the power generation market, many steam power plants are today subjected to unscheduled operation like, double shifts or load following operations. Especially for Combined Cycle Power Plants (CCPP) high operational flexibility is required. The ability for fast start-up and to react to the load changes in a quick and predictable way are key technological challenges for modern power plants.

A steam turbine start-up begins with the acceleration of the rotor shaft from turning gear operation to nominal speed, i.e. 3000 rpm in the 50Hz market and 3600 rpm in the 60 Hz market. Once the nominal speed is reached, the generator that sits on the same shaft as the turbine is connected to the power grid (synchronization). After the turboset is synchronized, the load (electrical output) is increased up to the nominal load value gradually by increasing the steam mass flow through the turbine and the steam temperature allowed inside the turbine.

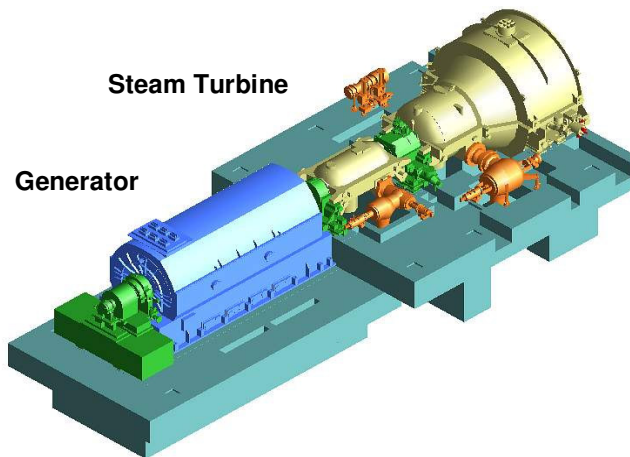


Figure 1: Example of Alstom Reheat Steam Turbine STF15c for CCPP

Steam turbines are operated at high pressures and temperatures. At base load operation with high temperatures the hot turbine components are exposed to 'creep loading'. Additionally, high fatigue loading occurs because of thermal stresses during transient events such as start-up, shut down and load changes. Thermal stresses occur especially in thick-walled components, such as rotors, valves, casings etc. Generally the faster the transient event is the higher the fatigue loading becomes. The combination of creep and fatigue leads to a creep-fatigue loading of the hot steam turbine components. Extensive creep-fatigue loading eventually leads to crack initiation and crack growth, thereby limiting the life of the main steam turbine components.

Goal of the steam turbine start-up optimization according to the Low Cycle Fatigue theory (LCF) is to reach nominal load as fast as possible without exceeding the allowable fatigue stress limits for the different turbine components. For this purpose the different time-dependent process parameters relevant for start-up are optimized. The key process parameters are the speed gradients

for run-up, the loading gradients as well as the admission steam temperatures to the high pressure and intermediate pressure turbines.

A significant design effort is invested to determine the optimal process parameters for start-up (e.g. steam temperatures, run-up and loading gradients). The first step of the current practice is to derive the transient thermal boundary conditions for the whole start-up simulation, based on pre-defined process parameters. In a second step a finite element analysis is performed to verify these thermal boundary conditions. Using this sequential approach, a high number of iterations are required to arrive at the optimal process parameters.

2. AUTOMATED START-UP OPTIMIZATION

A design tool was developed to automatically determine the optimal process parameters, by means of a feedback control algorithm. Based on Python scripting, the tool interlinks ABAQUS with an Alstom in-house thermodynamic program, determining optimal transient thermal boundary conditions based on real time thermal stresses. Use of this tool eliminates the need for the high number of manual iterations previously required. The implemented process leverages the automation capabilities offered inside ABAQUS/CAE in the form of ABAQUS Scripting Interface and Python Scripting.

The tool iterates through a thermo-mechanical start-up simulation with ABAQUS, determining at every time step the allowable thermal boundary conditions based on the actual stress utilization at the critical locations. The thermal boundary conditions are the local steam temperatures seen by the components and the corresponding heat transfer coefficients. The speed, load (mass flow) and the admission steam temperatures influence these thermal boundary conditions. The tool determines the optimal mass flow for every time step and associated parameter values.

To assure reasonable optimization results, practical limitations have to be taken into account, like maximum speed gradients, critical speeds (where speed holds are not allowed), maximum possible loading gradients, etc.

The interface between ABAQUS and the Alstom in-house thermodynamic program was developed using the ABAQUS kernel scripting language Python. This allows direct and easy communication with ABAQUS/CAE. In the following paragraphs the principle of the start-up optimization is explained, a process overview is shown in Figure 2.

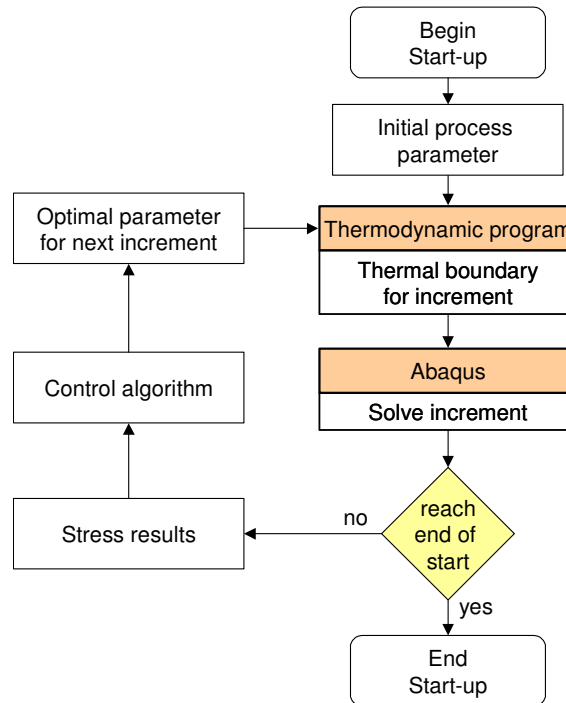


Figure 2: Overview start-up optimization tool

3. PROCESS OVERVIEW

Two inter-dependent processes of creating thermal boundary conditions and ABAQUS analysis are integrated within the scope of the automation tool.

The Thermodynamic Program is a standard program for thermodynamic data computation inside Alstom. By calling the officially released executable from the standard network drive it can be assured that the optimization tool always uses the latest released version of the Thermodynamic Program.

The two inter-dependent processes of creating thermal boundary conditions and ABAQUS analysis are controlled by means of two Signal Files. The ABAQUS SFILM-subroutine has read-write access on one of them and read only on the other. Similarly the Main Program has read-write access on one Signal File and read only on the other.

1. When the tool is started, the ABAQUS job is submitted. ABAQUS calls the SFILM-subroutine, which applies the computed thermal boundary conditions on the component model. But the subroutine is immediately set on hold, as the thermal boundary conditions needs to be computed first; for analysis.
2. The Main Program calls the Thermodynamics Program for computation of the thermal boundary conditions for the first time step. The thermal load defined by the film

coefficient (heat transfer coefficient) and sink temperature (local steam temperature) is written to an ASCII file as per pre-defined template. Then the Main Program changes the status value in the Signal File indicating that the thermal boundary condition computation is complete for the current time step, thus allowing the ABAQUS job to continue.

3. The SFILM-subroutine reads the ASCII file containing the thermal boundary conditions. With this input ABAQUS completes the thermo-mechanical analysis for the first time step. Once the analysis is completed the subroutine sets the status flag value in the Signal File having write access indicating the analysis is over.
4. At the beginning of the next time step ABAQUS again enters the SFILM-subroutine for required thermal boundary conditions. Again it is set on hold at this point. The Main Program now extracts the actual stresses at the critical locations from the ABAQUS Output Database (ODB-file) and calls the Control Algorithm for determination of the optimal mass flow for the next time step. Based on this mass flow information, the Thermodynamic Program is called for calculation of the corresponding thermal boundary conditions. As soon as the thermal boundary conditions are updated, ABAQUS performs the thermo-mechanical analysis for the next time step as per the Signal File status value.

This process of controlled computation is continued for the entire start-up period. The optimization tool repeats the pre-defined process until the simulation is completed for the whole start-up time. Finally the optimized loading history is obtained along with other important data.

At every time step the Stress Utilization Factors at the life-limiting locations of the component is computed by comparing the stresses extracted from the ODB with the material stress limits.

The mass flow for the next time step is chosen such that the stress limit is utilized at the maximum level possible (fast start-up) but not exceeded. The change of relative mass flow is determined applying an algorithm based on a PD controller:

$$u(t) = K_p e(t) + K_d \frac{de}{dt}$$

with the error: $e(t) = 1 - SUF(t)$

SUF(t) is the Stress Utilization Factor at the life-limiting location of the component, t stands for time. The tuning parameters for the PD control algorithm are the gain factors K_p (proportional) and K_d (derivative).

The relative mass flow for the next time step ($t+\Delta t$) is calculated as:

$$\frac{m}{m_N}(t + \Delta t) = \frac{m}{m_N}(t) + f(t) \cdot u(t)$$

The relative mass flow (m/m_N) is the fraction of the nominal mass flow (m_N), which passes through the turbine. $f(t)$ is a tuning function, based on Alstom experience.

Typical start-up times range from around 30 minutes to several hours, depending on the application. It was found that a time step for controlling time steps defined between 10 seconds to 60 seconds lead to good results, depending on the start-up.

For the thermo-mechanical analysis with ABAQUS a coupled temperature-displacement procedure is used to solve simultaneously for the stress/displacement and the temperature fields (fully coupled thermal-stress analysis). For example the rotor is modeled with axi-symmetric elements CAX8RT. The analysis type was validated against the previously used sequentially coupled thermal-stress analysis, and successful agreement of the results was found.

4. CALCULATION EXAMPLE

In this section a calculation example is given for a steam turbine rotor. The ABAQUS model of steam turbine rotor is shown in Figure 3, with the steady state temperature profile at base load.

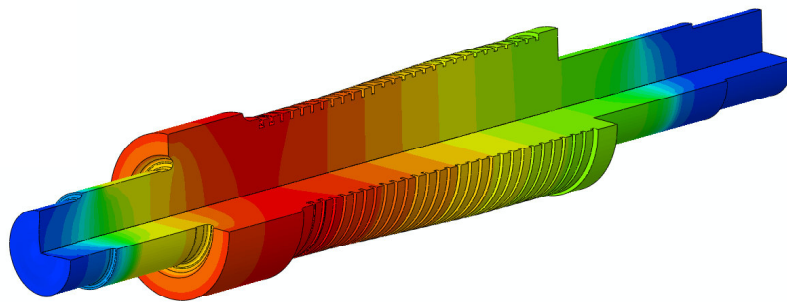


Figure 3: Rotor model – steady state temperature profile at base load

The loading curve of the steam turbine was automatically optimized applying the optimization tool described above. The resulting loading is shown in Figure 6. After reaching nominal speed (Speed) and grid synchronization the loading gradient is optimized such that the maximum stress (Stress) in the hot section of the rotor is running along the material stress limit (Limit). Thus the fastest possible loading without exceeding allowable stress limits is achieved. As an example Figure 4 and Figure 5 show the thermal and stress profile 60 minute after beginning of start-up.

The finite element model contains 2494 number of CAX8RT elements with 18864 variables. The time step for mass flow controlling was set to 60 seconds. The calculation time for this optimization run is about 16 minutes on a standard engineering PC (Intel Xeon Processor, Dual Core, 2.99 GHz, 2GB RAM). For comparison, a standard thermo-mechanical calculation of this rotor model with pre-defined thermal boundary conditions takes around 5 minutes on the same PC. The optimization algorithm, especially the multiple interactions between the thermodynamic program and ABAQUS increase the calculation time by a factor of 3. This increase in computation time is outweighed by the time saving and quality improvement achieved for the whole optimization process.

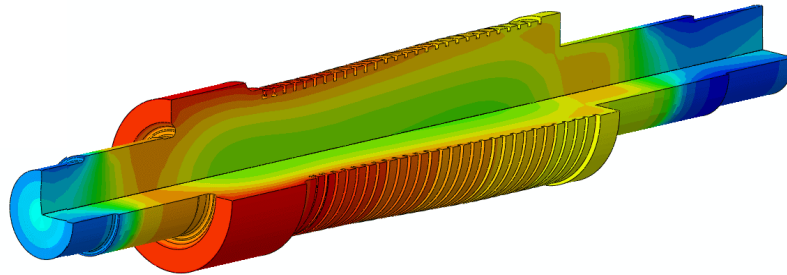


Figure 4: Rotor model – non-stationary temperature profile during start-up

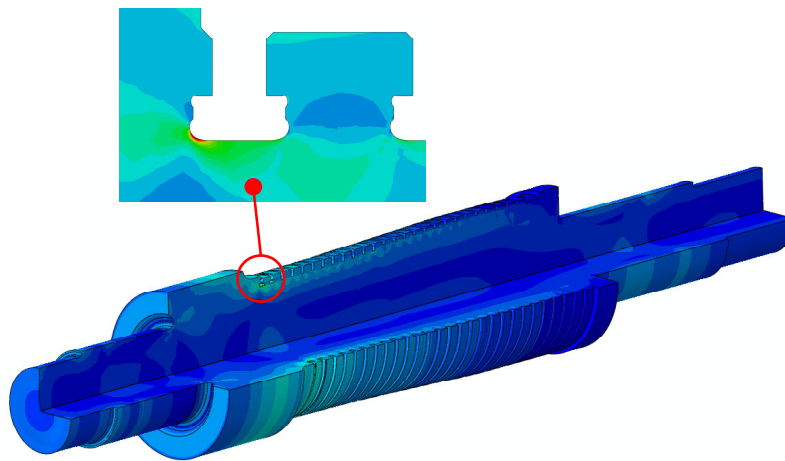


Figure 5: Rotor model – stress profile, with detail of stresses near 1st blade groove

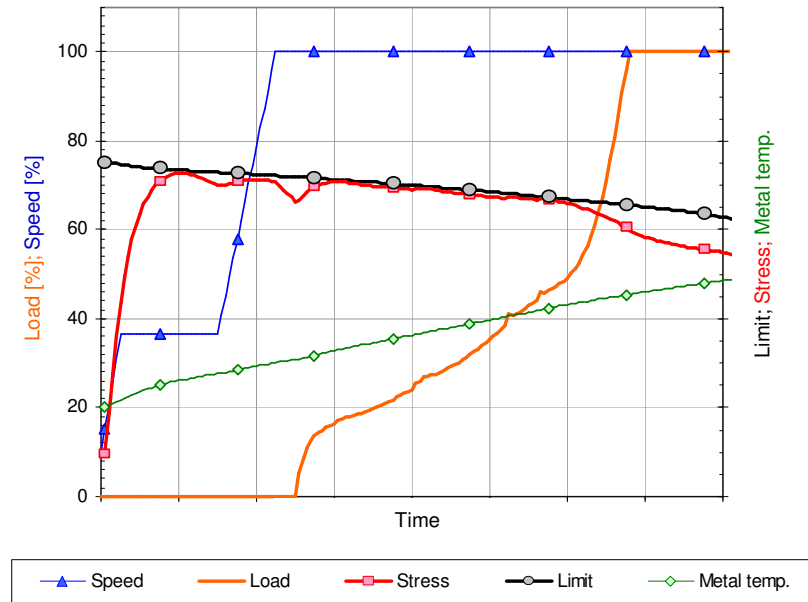


Figure 6: Start-up diagram

5. CONCLUSIONS

A tool was developed that automatically determines the optimal loading of steam turbines in order to achieve the fastest possible start-up, without exceeding the allowable material stress limits. Use of this tool eliminates the need for the high number of manual iterations previously required for start-up optimization.

The tool interlinks the finite element package ABAQUS/CAE with an Alstom in-house thermodynamic program, utilizing the automation capabilities of ABAQUS in terms of APIs (Application Programming Interfaces) within the ABAQUS Scripting Interface. Python as a scripting language proved to be an easy and effective medium for the integration of multiple program components inside the start-up optimization tool. The flexibility of the described approach allows it to be adopted for applications of a similar nature. Additionally, the ABAQUS GUI Toolkit is being used to develop an interactive Graphical User Interface for the optimization tool.

6. REFERENCES

1. Abaqus 2007. User's Manual. Version 6.7.
2. www.python.org