SEVENTH FRAMEWORK PROGRAMME
THEME – ICT
[Information and Communication Technologies]

Contract Number:	223854
Project Title:	Hierarchical and Distributed Model Predictive Control of Large-Scale Systems
Project Acronym:	HD-MPC

Deliverable Number:	D7.1.3
Deliverable Type:	Report
Contractual Date of Delivery:	November 1, 2011
Actual Date of Delivery:	**November 1, 2011**
Title of Deliverable:	**Report that presents the closed-loop validation results for the combined cycle start-up**
Dissemination level:	PU
Workpackage contributing to the Deliverable:	WP7
WP Leader:	Damien Faille
Partners:	EDF, SUPELEC
Author(s):	A. Tica, H. Guéguen, D. Dumur, D. Faille, F. Davelaar

Table of Content

Project co-ordinator

Name: Bart De Schutter
Address: Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 Delft, The Netherlands
Phone Number: +31-15-2785113
Fax Number: +31-15-2786679
E-mail: b.deschutter@tudelft.nl

Project web site: http://www.ict-hd-mpc.eu

Executive Summary

This report presents different predictive control approaches based on the models developed for the Combined Cycle Plant Start-up Application. First, an open-loop optimization method is proposed in order to optimize the start-up sequence. Next, a centralized MPC method based on the open-loop approach is considered. The method proves significant improvements in terms of start-up performances but a high computational effort. In order to reduce the computational complexity, the predictive procedure is included in a hierarchical control structure. Finally, the applicability of a distributed control approach is also studied.

1 Synopsis of the report

It is worth to make a brief summary of the other two deliverables before to outline the content of this report. In the report D.7.1.1, the control specifications for the start-up of a CCPP, with a partitioning of the plant in subsystems, have been defined. For each subsystem the operational constraints and the existing control loops have been specified. The information from this report has been used to build a model of the process, adapted to the start-up sequence. Due to the complexity of such systems, we decided to focus on a process with a 1-1-1 configuration: a gas turbine (GT), a steam turbine (ST), and a heat recovery steam generator (HRSG) configuration with only a single level of pressure (high pressure circuit). In the report D.7.1.2, the development of the CCPP models adapted to the start-up sequence is described. A simulation model based on design considerations has been used to derive models for control purposes (Modelica, Matlab/Simulink). The elaboration and validation of these models, as well as the results of the open loop simulations have been presented.

In this report, in order to improve the CCPP start-up performance, a series of methods, based on developed models, in particular Modelica smooth model, are proposed. These methods have been applied on the last part of the start-up sequence (increasing load see report D.7.1.1), since in terms of lifetime consumption this part is the most critical. The applicability of these approaches can be easily extended to other start-up phases.

In Section 2 an open loop method for the optimization of control profiles is introduced. In this method, the control profiles are assumed to be described by a parameterized function, whose parameters are computed by solving a minimum-time optimal control problem, subject to the plant dynamics and to a number of constraints on the plants variables, in particular on stresses in the steam turbine rotor and the header of the high pressure superheater. Several types of parameterized functions have been considered in order to propose the best solution in terms of start-up performance. In general the optimization aims the GT Load profile, the objective being to minimize the start-up time, while keeping the constraints within their limits. The determination of the load profile is based on a black-box optimization using Modelica smooth model. The obtained results demonstrate that the method is able to significantly reduce the start-up time (more than 30%) compared to a classical procedure. Also other experiments show that the simultaneous optimization of several control profiles (GT Load and ST throttle) can be a solution to improve even further the start-up time, but with a high computational effort.

Model Predictive Control of the GT load profile, based on the same principle as open-loop optimization procedure, is implemented in Section 3. The objective in this predictive approach is formulated by means of a quadratic cost function, penalizing the sum of the squared deviations from the target value represented by the full load. The results have shown that the periodic computation of the profile leads to a reduction by 49% compared to the classical ramp profile, but with an increased computational complexity. In order to make a trade-off between the start-up performance and computational effort, a results analysis, in terms of the prediction horizon length as well as the type of the profile functions, is performed. The analysis has shown that even with an adequate selection of all these parameters, the computation times are inconsistent to an online applicability of the approach. Therefore, HD-MPC methods have to be addressed.

In order to reduce the computational effort a hierarchical approach has been proposed in Section 4. The hierarchical structure includes two layers with different time scales: a high layer, where at a long period a minimum time optimal control problem is periodically solved, and a low layer, where at a shorter period a quadratic optimization problem is solved, in order to reach as quickly as possible the target provided by the high level. The solution to the high level problem is used to update the set-point for the low level. The hierarchical approach leads to start-up times comparable to the centralized solution but with reduced computation times (approximately 39%).

Finally, in Section 5 the distribution of control has been studied. In a first phase, the potential decomposition of the system is studied. The system analysis demonstrates that is quite difficult to split the process because strong interactions among the subsystems exist. A solution in this direction, also adapted to the start-up phase is proposed. As the algorithms developed in the HD-MPC project are based on model structural information (e.g. gradients), or on a specific decomposition structure, their applicability in the case of the CCPP Modelica model is quite impossible at this moment. Therefore, a simple communication based algorithm is considered to solve the distributed optimal control problems. A series of issues generated for example by the feasibility of the interaction profiles, sensitivity of the simulations to the proposed profile leads to the algorithm failure, thus making quite difficult the applicability of a distributed approach.

2 Open loop trajectory optimization

2.1 Description

Combined Cycle Power Plants (CCPPs) are complex systems composed by several interacting subsystems. In terms of CCPP start-up, this procedure includes several phases and conditions that must be respected simultaneously.

In general the traditional CCPP start-up sequence is quite conservative since it uses limitations on the main control variables (e.g. Gas Turbine load rate), in order to guarantee the safety and the availability of the operations. This type of sequences limits naturally the stress in thick components. A new approach for the CCPP start-up is proposed, the idea is to eliminate as much as possible those limitations, without compromising the safe operation and keeping the life-time consumption of the most stressed components under control.

The optimization procedure is based on the model developed in the previous report. The approach aims at deriving an optimal profile of the plant control variables, by assuming that this profile can be described by a parameterized function. The parameters of this function are computed by solving a minimum-time optimal control problem, subject to the plant dynamics and to a number of constraints on the main plant variables, such as pressures, temperatures, stresses.

The optimization targets different control variables (gas turbine (GT) load, bypass, steam turbine (ST) throttle, desuperheating water flow rate, etc.). In the following sections, only two of the main variables are examined. These are considered by the experts in domain as the most influential variables for the start-up procedure, in particular in the last part of the sequence. Namely, the GT load, for a load ramp more rapid, in order to reduce the start-up time, and the ST throttle for a better control of the thermo-mechanical rotor stress (the most critical constraint for the start-up sequence).

2.2 Gas Turbine load profile optimization

In a first phase, the optimization procedure aims only the GT load profile, the objective is to reach the final point of the start-up (full load and stationary conditions) as quickly as possible, while keeping the thermal stresses within their allowable limits. As already pointed out, the proposed approach is based on the determination of a minimum time optimal control problem subject to constraints.

Considering that the CCPP model can be given into a dynamic explicit form:

$$\dot{x}(t) = f(x(t), u, L(t)) \tag{1}$$

where $x(t)$ is the vector of state variables (temperatures, pressures, etc.), u is the set of inputs which are considered as constant or predefined functions, in other words the inputs that are not concerned by the optimization procedure (e.g. feed flow to the desuperheater, bypass valve, etc.), and $L(t)$ represents the GT load.

Denote now by t_0 the initial time instant of the start-up procedure (conventionally let $t_0 = 0$) and by t_f the final time. Also considering the start-up conditions, for the load profile $L(t_0) = L_m$ and $L(t_f) = L_M$ is selected, where L_m and L_M are the initial and final (full) loads respectively.

It is assumed that the GT load is described by an increasing function $L(t,q)$, as in (7) for example, satisfying the boundary conditions stated above, where q is a vector of unknown parameters, which have to be selected through an optimization procedure (described below). The optimal load profile consists in finding the value of the parameter vector q together with the final time t_f as the solutions of the following optimization problem:

$$\min_{q,t_f}\left(J = \int_{t_0}^{t_f} dt \right) \tag{2}$$

subject to the constraints

$$\dot{x}(t) = f(x(t), u, L(t,q)) \quad (3)$$

$$L(t_f, q) \geq L_M - \varepsilon_1 \quad (4)$$

$$\left| f(x(t_f), u, L(t_f, q)) \right| \leq \varepsilon_2 \quad (5)$$

$$h(x(t)) \leq 0,\ t_0 \leq t \leq t_f \quad (6)$$

where ε_1 and ε_2 are tightening terms of the constraints (ideally equal to zero) and the corresponding constraints are included to guarantee that at time t_f the system has reached almost full load (4) and is almost in stationary conditions (5). The last constraint (6) includes, in general form, all the constraints to be imposed on the plant variables, in particular on the stresses, during the start-up procedure.

Based on the experience of the operators and the analysis of the GT load transient during the start-up procedure, in particular in the last stage (increasing load see report D.7.1.1), the GT load profile can be considered as a parameterized function (e.g. a sigmoid function). In fact the variety of functions that can be chosen is quite large. The selection of these functions is important, because, as it will be seen in the following sections, the performed choice can lead to more or less suboptimal solutions to the original problem.

In order to improve the start-up time and to allow a better management of the constraints, several types of functions have been considered, e.g. *Hill functions*, or a more general situation by using piecewise polynomial functions.

Hill function

A first choice is to describe the GT Load as a *Hill function:*

$$L(t,q) = L_m + (L_M - L_m)\frac{t^h}{t^h + k^h} \quad (7)$$

where $q = [h; k]$ is the parameter vector to be determined.

The GT load profile used in the plant can be represented by connecting two or more *Hill functions* (or equivalent) in cascade. The representation of the GT Load as a combination of two *Hill functions* is the following:

$$L(t,q) = L_m + (L_i - L_m)\frac{t^h}{t^h + k^h} + (L_M - L_i)\frac{t^p}{t^p + r^p} \quad (8)$$

where $q = [h; k; p; r; L_i]$ is the parameter vector to be determined through the optimization procedure. Considering an example with $L_m = 0.15$, $L_M = 1$, $t_0 = 0$, $t_f = 5000$ [s], the functions for different values of the parameters are shown in Figure 1.

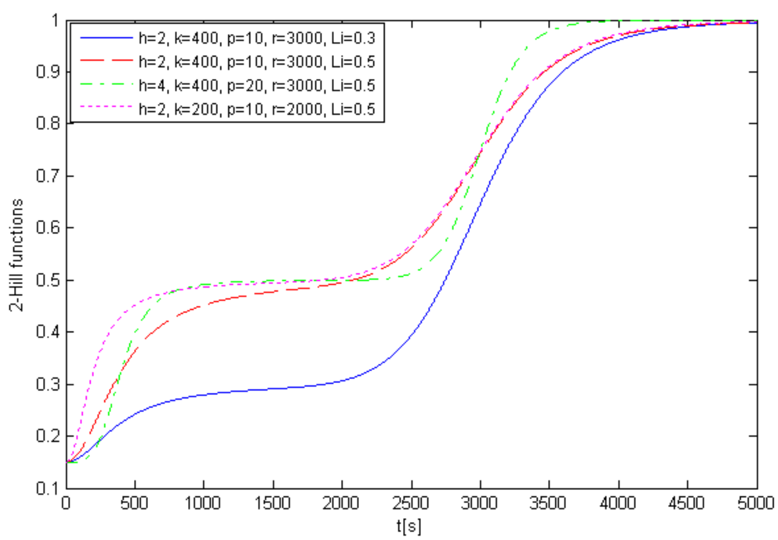

Figure 1: *2-Hill functions* with different values of the parameters

Piecewise polynomial functions

Another choice is to use one of the most popular parameterizations, namely piecewise polynomial functions (e.g. constant, linear, etc.). The range of these functions is also rich, in this work the attention is focused on the use of common spline functions (see Figure 2), where the polynomial pieces are joined together with explicit continuity conditions.

The principle of the proposed solution is to subdivide the optimization horizon $[t_0, t_f]$ in n partitions, $t_0 < t_1 < t_2 \ldots < t_n = t_f$, and for each data interval, to compute a corresponding spline function. The assumption is that the GT load profile is defined by a set of data points (t_i, L_i), where L_i is the GT load value at the time instant t_i. These points are then interpolated by means of spline functions.

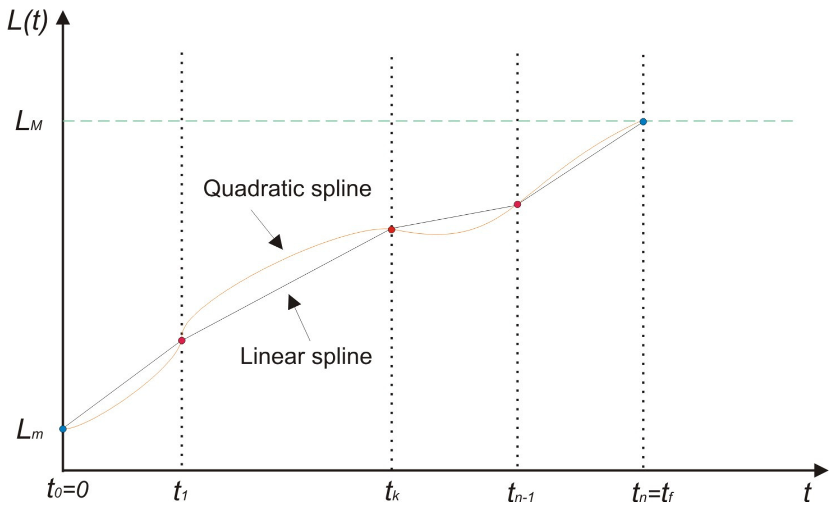

Figure 2: Approach based on spline functions

In general the selection of an appropriate spline function influences the solution optimality. Moreover, to assure a certain degree of smoothness, the splines of increasing order are necessary. In our case based on the results analysis a quadratic spline has been chosen. The corresponding representation of it is:

$$s_i(t) = a_i(t-t_i)^2 + b_i(t-t_i) + c_i\,, \quad t \in [t_i, t_{i+1}], \quad i = 1, 2, \cdots, n-1 \tag{9}$$

which is constrained to satisfy the C^0 and C^1 conditions. From these conditions the splines' parameters are determined (i.e. a_i, b_i, c_i).

Considering the same start-up conditions as in the previous case (i.e. $t_0 = 0$, $L(t_0) = L_m$, $L(t_f) = L_M$), the parameter vector to be optimized, in this case, is $q = [t_j, t_f, L_j]$, where $j=1,2,\ldots,n-1$. It should be noted that in order to maintain the feasibility of the solution, an additional number of linear constraints on the parameters bounds compared with the sigmoid function case are imposed (e.g. $L_m \leq L_j \leq L_M$, so that the GT load to remain between L_m and L_M.

2.3 GT load profile and ST throttle opening profile optimization

In the procedure outlined above, only the GT load is used as optimization signal, the other control inputs of the system being considered as constants (e.g. desuperheater water flow rate), or in the case of the ST throttle, its behavior is represented as a predefined function of the GT load. In order to improve the start-up performances even further, the procedure can also aim the optimization of the other inputs, for example the optimization of the ST throttle opening.

The ST throttle profile can be described by using a parameterized function $(O_t(t,v))$. The optimization procedure is mostly the same as in previous case; the optimal profiles of the GT load and ST throttle are determined by solving a minimum time problem:

$$\min_{q,v,t_f}\left(J = \int_{t_0}^{t_f} dt \right) \tag{10}$$

subject to the constraints (3-5) and the system dynamic

$$\dot{x}(t) = f(x(t), u, L(t,q), O_t(t,v)) \tag{11}$$

In this case the parameter vector to be optimized, q_{new} is composed by the parameters that describe the GT load profile (q) and the parameters of the ST throttle (v) respectively. Similar with the parameters vector q, v can contain the set of data points used by the spline functions or the parameters of the Hill functions.

2.4 Results

In order to show the advantages of the approach presented in the previous section, the method is applied to a hot part of the start-up sequence, when the system has the following initial conditions:

- the bypass valve is closed;
- the desuperheater water flow rate is close to its nominal value (0);
- the admission valve opening is at 30% (minimum load);
- the ST is connected to the grid;
- the GT load is set to 15% (minimum load).

The procedure has been applied to the final part of the start-up sequence because it represents the most critical phase of the procedure, in terms of thermal and mechanical stresses.

As already stated in D.7.1.1, during the start-up, the most stressed components are the rotor and the superheater outlet header, so the optimization procedure described in the previous section has been implemented by imposing a number of constraints on the main plant variables, in particular on the maximum value of these stresses in order to prevent any possible unsafe conditions and to preserve the life of the unit. The peak values for the header stress and the rotor stress have been fixed to 115 [MPa] and 440 [MPa], respectively. These limits are consistent with typical values estimated on a real plant during the start-up phase.

The new model-based approach has been implemented in Matlab, by using Dymola-Simulink interface. The CCPP Dymola/Modelica model has been transformed into a Simulink S-function that can be optimized and simulated as an input/output block. The optimization procedure is based on the Matlab nonlinear constrained optimization solver *fmincon.*

To see the improvement with the new approach, the results obtained by a classical procedure with a constant ramp rate of the GT load and the proposed optimal GT load profiles are compared. It must be noticed that for the classical start-up sequence, the GT load ramp has a slope of 2 MW/min throughout the start-up phase. This value corresponds to the constant GT load rate, which ensures that the constraints imposed on the stress are respected.

GT load profile optimization

The results obtained, with the GT load described as a *Hill function*, are shown in Figure 3. This type of function can guarantee a safe operation but with a very suboptimal solution in terms of start-up time (see Figure 3).

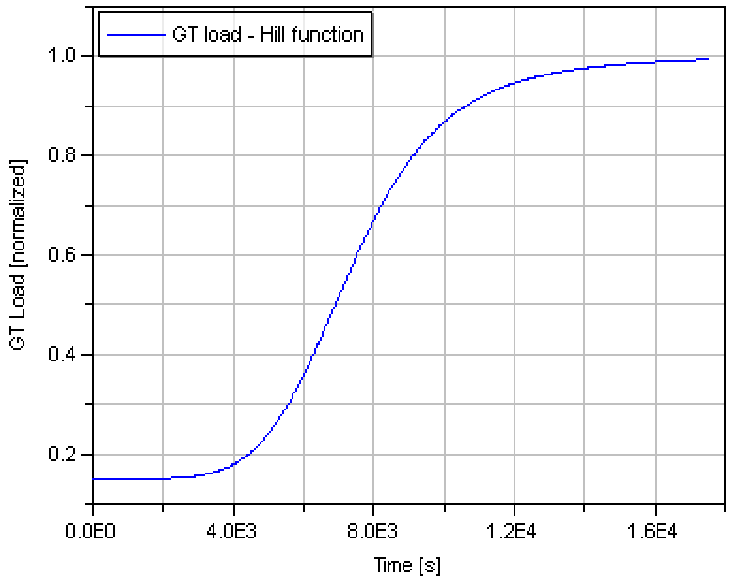

Figure 3: GT Load described as a Hill function

A faster start-up phase can be performed by assuming that the load profile can be described by a different input function. The results of the optimization obtained with the new types of functions chosen (e.g. 2-Hill functions, spline functions) to replicate the GT load profile are presented in Figure 4. An optimization procedure relied on these functions can guarantee a fast and safety start-up operation.

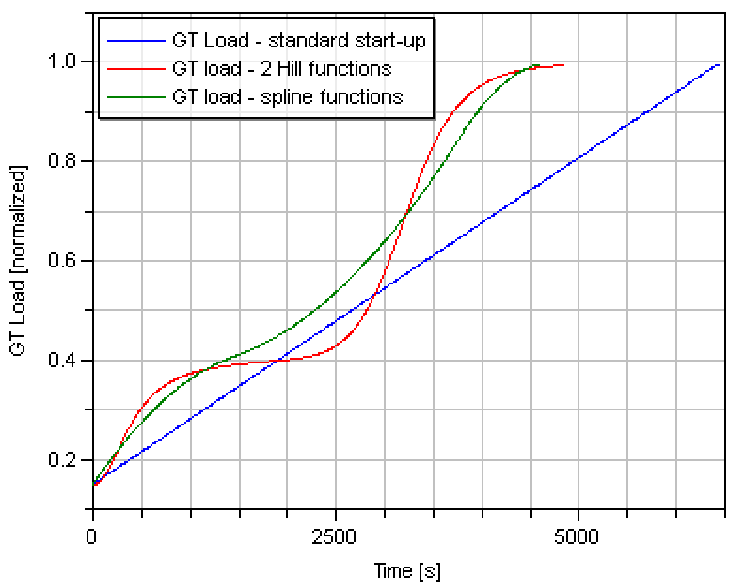

Figure 4: GT load profiles with standard and optimization procedure

The optimization results show that the new procedure is able to reduce the start-up time compared to the classical one by approximately 27 minutes for a GT load profile described as 2-Hill functions and by about 31 minutes when the GT load representation is performed with three spline functions (see Figure 4, the time is counted starting from the moment when the ST is synchronized to the grid), by fulfilling the imposed constraints.

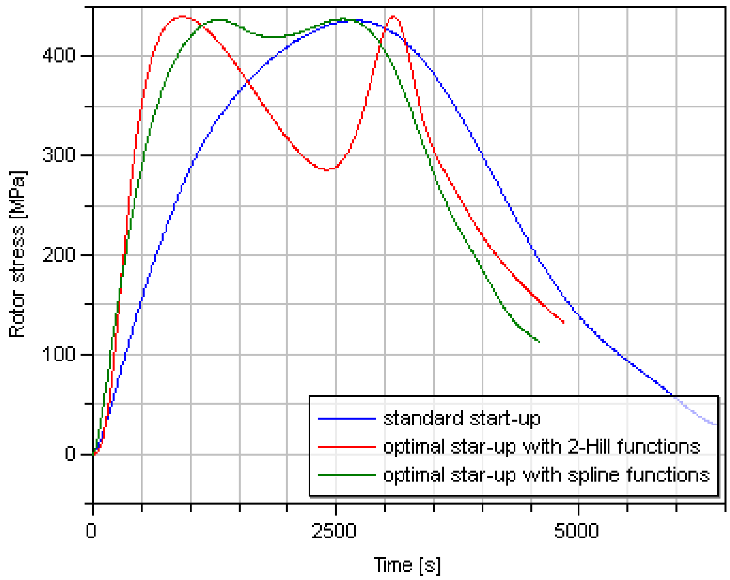

Figure 5: Rotor stress according to each GT load profile

The comparative analysis of the results obtained with the selected functions shows that during the first 1000 seconds, the GT load optimal solutions increase more rapidly compared to the classical one, causing a rapid growth of the rotor stress (Figure 5). After this first phase, in the case of the 2-Hill functions the GT load is very slowly increased, leading to a decrease of the rotor stress level, which allows thus a faster increase of the load later on. When the GT load reaches a value close to 60%, the gas exhaust temperature is no longer increased, causing only a small increase of the steam temperature. Therefore, the rotor stress decreases regardless of GT load profile. In the last part, only the Header stress (Figure 6) limits the load increase.

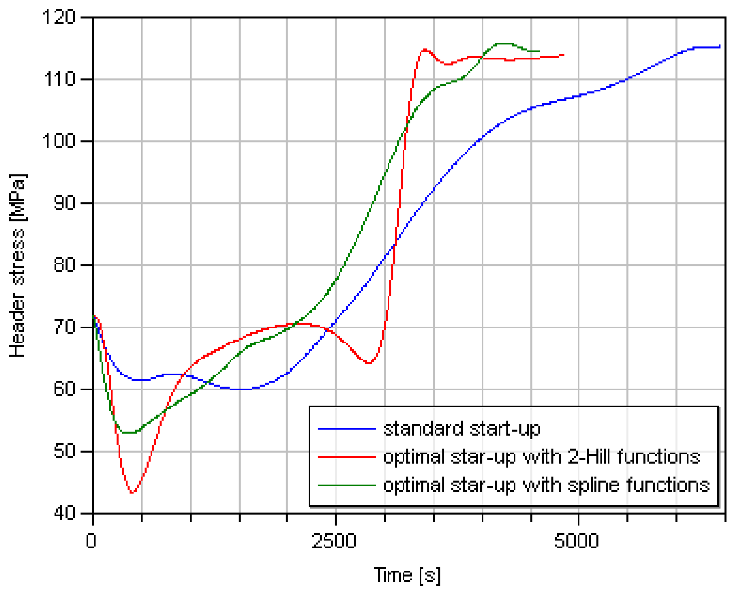

Figure 6: Header stress according to each GT load profile

For the piecewise polynomial functions (spline functions), the portion of slowing down observed in the 2-Hill functions case is eliminated through a better control of the GT load ramp slope. Thus these types of functions provide a higher flexibility, which leads to the performances improvement. Moreover, in the case of such functions, the flexibility can be even higher when the number of the data points used to describe the GT load profile is increased. For example in Figure 7, the GT load is described by using 4 and 5 points respectively. The results show that in the second case, the start-up time is reduced by 50 seconds compared to the first case.

A better approximation of the GT load profile involves a large number of points; on the other hand a large number of data points leads to the increasing of the computation time (see Table 1).

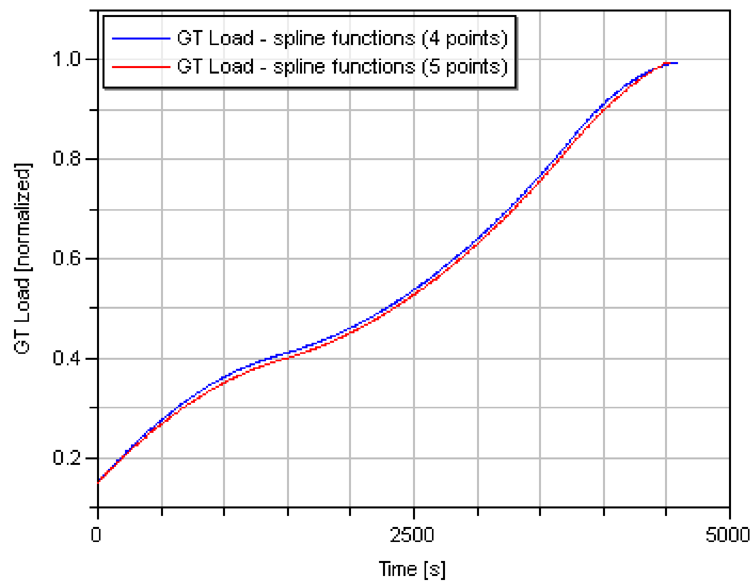

Figure 7: GT load profiles described using spline functions

Another important benefit, as a result of shortening the start-up time, is the reduction of the operating costs due to lower fuel consumption. A comparison between the fuel consumption for each procedure is illustrated in Figure 8.

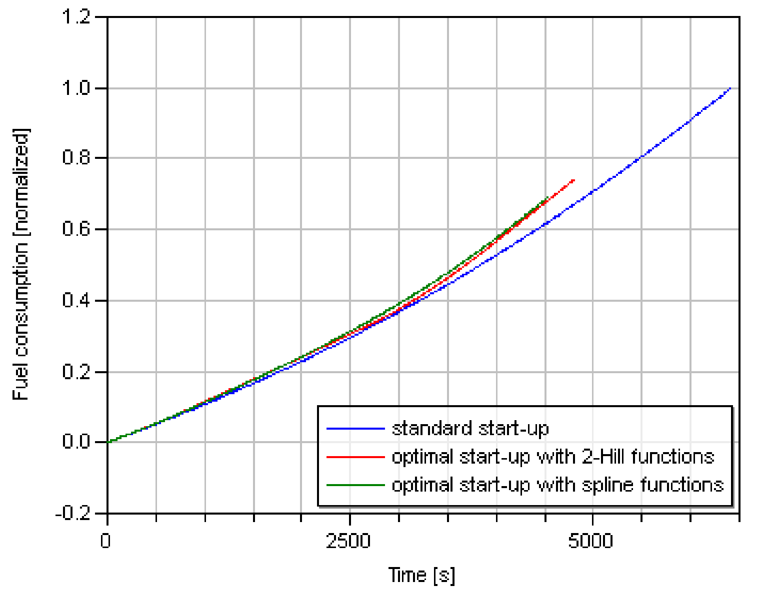

Figure 8: Fuel consumption comparison

The results show that the new start-up procedure consumes approximately 26% (in the 2-Hill functions case) and 31% (for spline functions case) less than the traditional start-up.

GT load profile and ST throttle opening profile optimization

As already stated, the procedure can target also the optimization of the ST valve opening profile. The result of the optimization procedure and the optimal solution found in the previous case (when only GT load profile is optimized) are shown in Figure 9a. In the both cases the control profiles are parameterized by using three spline functions.

The result illustrated in Figure 9a shows that by optimizing also the ST valve profile (2-controls optimization) a supplementary reduction of the start-up time (40 seconds) compared to single-control optimization with spline functions is performed (see Table 1 for a more complete comparison). This can be considered as a point of interest for the start-up phase, because in 1-control optimization the ST throttle behavior is described as a predefined function of the GT load level and by optimizing their profile a better control on the ST thermal stress can be performed (see Figure 10).

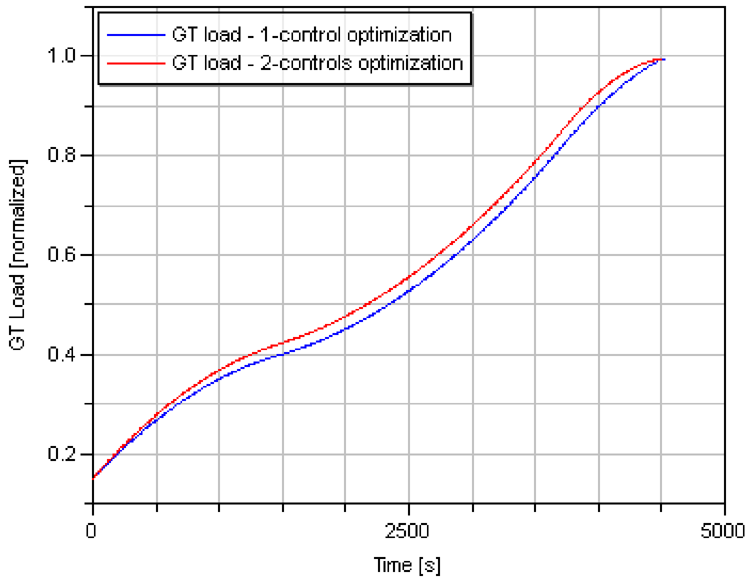

Figure 9a: Comparison between 1-control and 2-controls optimization

The optimized ST throttle profile and the ST valve opening as a predefined function of the GT load level are represented in Figure 9b.

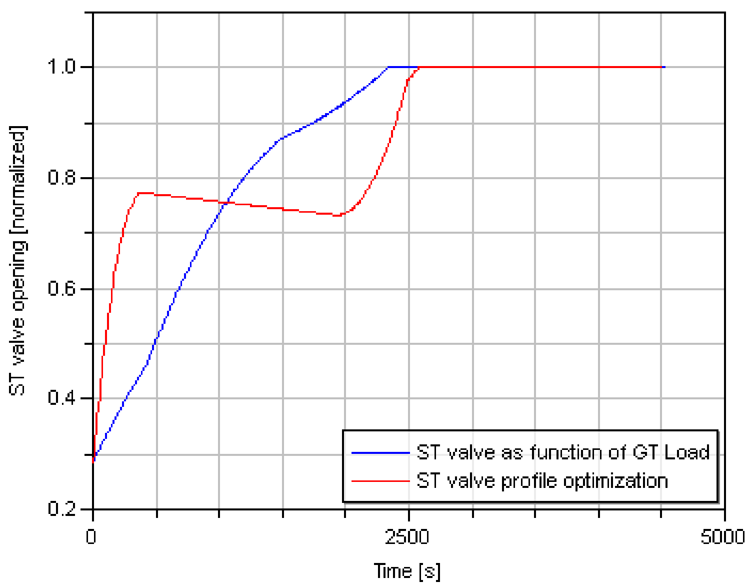

Figure 9b: ST throttle profile representation

It must be noticed that, in optimization procedure, only the peak values of the stress are considered to assess the life-time consumption during the start-up sequence. These values are maintained below the allowed limits, and no additional life-time consumption caused by the variations of the stress level is considered.

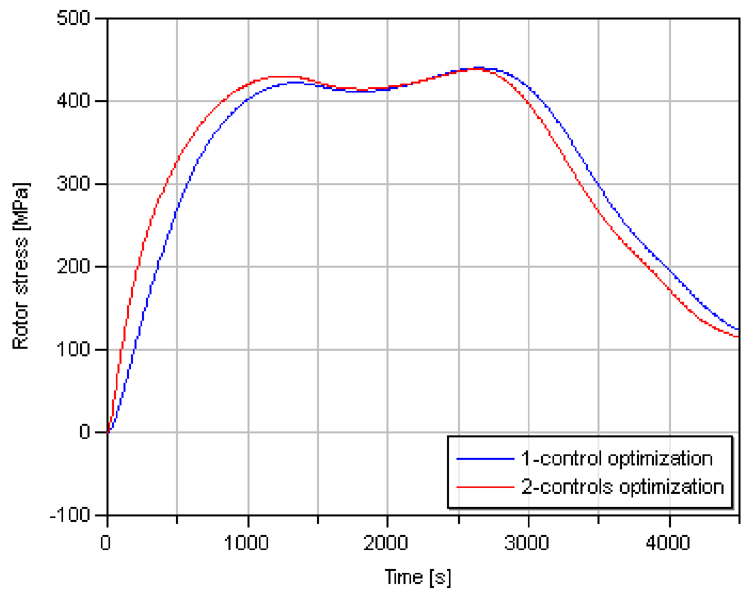

Figure 10: Rotor stress comparison between 1-control and 2-controls optimization

A comparison, in terms of benefits, computation time and number of optimization parameters, between all the functions considered in this work can be found in Table 1. Also in Table 1, a ratio between the time gain and complexity is presented:

$$r = \frac{g_t}{c}$$

with

$$g_t = \frac{t_o - t_{ref}}{t_{ref}} \qquad c = \frac{t_c - t_{rc}}{t_{rc}} \tag{12}$$

where t_o is the start-up time, t_{ref} is the reference time, t_c the computation time and t_{rc} denotes the computation reference time.

GT load profile optimization					
Functions	*Start-up time**	*Fuel saving**	*No. parameters*	*Computation time*	*Ratio*♣
Standard	6400 [s]				
2-Hill functions	4790 [s] (-25%)	~ 26%	5	705 [s]	
Spline (4 points)	4530 [s] (-30%)	~ 31%	6	1094 [s]	9.8%
Spline (5 points)	4480 [s] (-30%)	~ 33%	8	1414 [s]	6.4%
GT load and ST throttle profiles optimization					
Spline (4 points)	4440 [s] (-31%)	~ 35%	11	1578 [s]	5.9%

Table 1: Results comparison

As it can be observed from Table 1 the computational time increases with the number of optimization parameters. Moreover, it should be pointed out that the computational time and also the feasibility of

* The percentage values are reported to the standard start-up sequence

♣ The ratio is reported to the 2-Hill functions results the solution depend strongly on the initial point provided to the solver. The results have been obtained on a PC with a 3 GHz CPU.

2.5 Conclusions

Based on the models developed in the previous report D.7.1.2, an open-loop optimization procedure for the start-up of a CCPP is proposed. The optimization procedure shows an improvement in time reduction and fuel consumption saving, with respect to the traditional procedure.

The idea of solving the start-up problem by optimizing a parameterized function can be expanded in several ways, for example, by considering other types of functions to describe the control profiles, and/or by using the other control inputs (e.g. desuperheating water flow rate) as optimization signals.

In the following sections, the open-loop optimization procedure is included in a model predictive approach.

3 Centralized MPC

Concerning the MPC methods, it is well known that two key issues have to be considered. The first aspect is represented by the fact that a reliable model of the plant dynamics with accurate output prediction is required. The second one aims the algorithm used to solve the optimization problem, in other words the algorithm has to be able to solve the optimal control problem based on the dynamic model of the plant to be controlled.

The developed models, in particular the Modelica smooth model, fulfills the first requirement. The results presented in the previous report prove the model consistency and denote that it is accurate enough to be used as prediction model of a MPC controller.

Regarding the second aspect, the MPC methods require the repeated computation of the solution of an optimal control problem. For linear MPC, the solution can be efficiently calculated both offline as well as online, whereas in the case of NMPC strategies, a nonlinear programming problem (NLP) has to be solved. In general to solve such problems a significant computational effort is needed. However in the NMPC's case, a series of efficient numerical algorithms, which exploit the structure of the nonlinear program, exist, providing real-time feasible solutions to nonlinear optimization problems [1], [2].

In literature there are different approaches to solve optimal control problems (see for example [3], [4]). Traditionally to solve on-line NMPC problems, the so-called direct methods are used. In these approaches the inputs, constraints and/or states are finitely parameterized. By using a finite parameterization, the original infinite dimensional problem is approximated/ transcribed into a finite nonlinear programming problem. The resulted problem is solved by a finite dimensional optimization solver, e.g. Sequential Quadratic Programming (SQP).
The predictive approach proposed in this work is linked to the open-loop optimization procedure presented above, following the basic idea that is behind the direct methods, namely the control parameterization.

Mostly, for numerical solutions of the optimization problems, the traditional way to parameterize the control profiles is to use a finite number of basis functions e.g. piecewise constant over a sampling time T_s (see Figure 11).

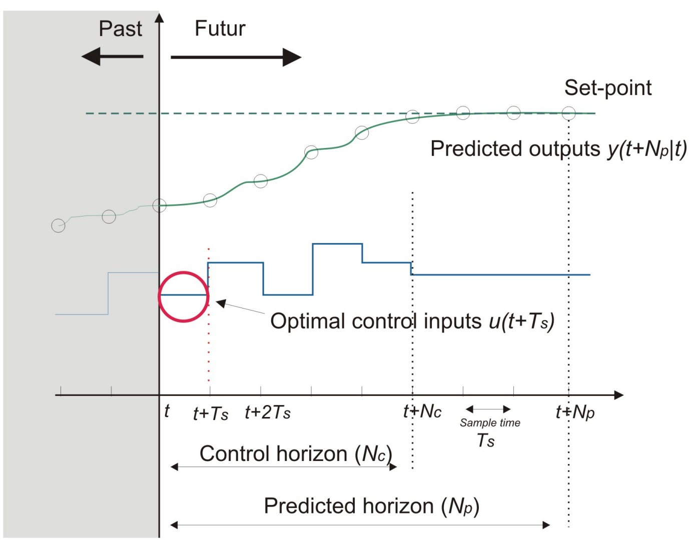

Figure 11: Classical MPC: piecewise constant control parameterization

Notwithstanding the fact that there are several optimization algorithms, their applicability to Modelica complex models is currently limited. To solve the open-loop optimal control problem, the same optimization procedure by considering the model as a black box without providing information about the model structure (e.g. gradients) to the solver, has been used. This procedure is in contrast to traditional direct methods in which certain solvers deliver sensitivities needed within the optimization. As a result, the performance in terms of convergence properties and execution times is relatively low.

The predictive method aims the determination of the GT load profile, but as could be seen previously in the open-loop optimization procedures, the optimization can target also the other control profiles.

3.1 GT load profile optimization

In order to be applied in a predictive method, the mathematical formulation of the finite horizon open-loop optimal control problem outlined above is slightly changed, namely:

$$\min_{q} J(x(t), L(t,q); N_p, N_c)$$

with
$$J = \int_{iT_s}^{iT_s+N_p} (L(t,q) - L_M)^2 dt \tag{13}$$

subject to:

$$\dot{x}(t) = f(x(t), u, L(t,q)) \tag{14}$$

$$h(x(t)) \le 0, \quad t \in [iT_s,\ iT_s + N_p] \tag{15}$$

where $i \in \mathrm{N}$ an index used to define the sampling time, N_p and N_c are the prediction and the control horizon with $N_c = N_p$ in this case. (15) includes all the constraints imposed on the plants variables.

The parameter vector q is computed by solving the above optimization problem (13-15), and the receding horizon paradigm is adopted. Thus, only the optimal value $L^*(t,q)$ computed for $t \in [iT_s, (i+1)T_s]$ is applied to the system and the overall procedure is repeated at the new sampling period T_s.

The objective in this approach is formulated by means of a quadratic cost function, penalizing the sum of the squared deviations from the target value. Actually, with this cost function, the computed solution is optimal in the sense that GT load is increased as fast as possible without violating the imposed constraints. Thus, by increasing the GT load with a maximum allowable loading rate, a minimum start-up time is assumed to be provided.

The NMPC problem solution is obtained based on the finite parameterization of the GT load (denoted by $L(t,q)$). In general in the predictive methods, the typical choice for controls parameterization is to use piecewise constant over sampling times. The control variables *u(t)* are piecewise constant on each predicted sampling interval: $u(t) = q_i, \quad t \in [t_i, t_{i+1}], \quad t_i = t + i * T_s, \quad i = 0 \ldots N-1$, with $N = \frac{N_p}{T_s}$. Thus the optimal control problem is reduced to the finding of the parameters vector $q = [q_0, q_1, ..q_i, ..q_{N-1}]$. In some cases a continuous control policy is preferred, rather than a policy that requires sudden switching from one level to another.

In the predictive methodology described in this paper, in order to increase the control performance, the control profiles are approximated by using piecewise polynomial functions over a predefined number of intervals (see Figures 12 and next paragraph).

3.2 MPC: piecewise polynomial control parameterization

In principle, this type of parameterization is similar to the one presented in the first part of this paper (see Section 2.1). In a first phase, the optimization horizon $[iT_s,\ iT_s + N_p]$ is subdivided in n control partitions ($N_p = nT_{col}$), and then in each subinterval, the control *u(t)* is approximated by means of the interpolation polynomials (16). The predictive approach is illustrated in Figure 12.

$$u(t) = \sum_{j=1}^{n} u_{i,j} P_j \left(\frac{t - t_{i-1}}{t_{i+1} - t_i}\right), \qquad t \in [t_{i-1}, t_i] \tag{16}$$

where P_j are interpolation polynomials.

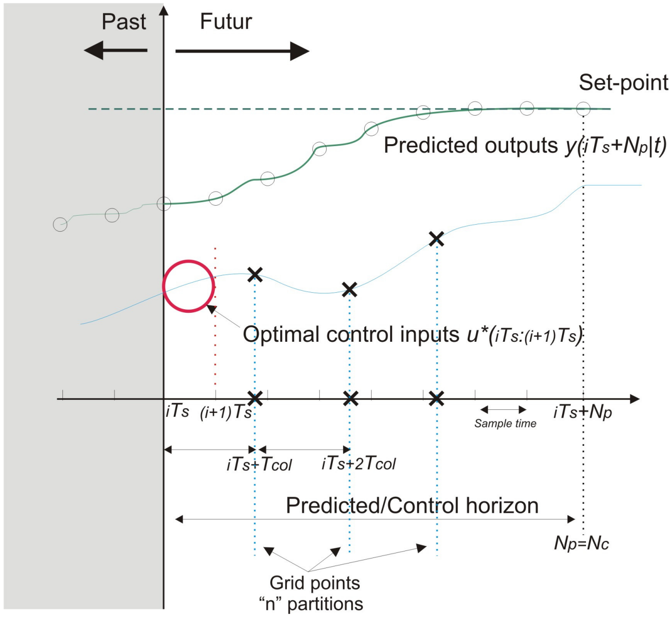

Figure 12: MPC: polynomial functions control parameterization

In our approach, in each subinterval of the optimization horizon, the GT load ($L(t)$) is approximated by using the Lagrange interpolation polynomials. Considering n points $\tau_1,...\tau_n \in [0,1]$, the corresponding Lagrange polynomials are:

$$\begin{cases} l_j^n(\tau) = 1, \quad for \quad n = 1 \\ l_j^n(\tau) = \prod_{k=1,k\neq j}^{n} \frac{\tau - \tau_k}{\tau_j - \tau_k}, \quad for \quad n \geq 2 \end{cases} \tag{17}$$

The property of Lagrange polynomials is that its value at any data point τ_k within the data set is either *1* or *0* (18).

$$l_j^n(\tau_k) = \begin{cases} 1, & for \quad j = k \\ 0, & for \quad j \neq k \end{cases} \tag{18}$$

Thus the Lagrange interpolation of degree n-1 used to describe GT Load ($L(t)$) is as follows:

$$L_{n-1}(t) = \sum_{i=1}^{n} L_i l_i^n(t) \tag{19}$$

3.3 Results

To illustrate the advantages and the limitations of the predictive approach, several case studies have been considered. In the sequel, only five of them are presented.

Case Study 1

In the first scenario:

- the prediction horizon $N_p = 600$ [s]
- sample time $T_s = 60$ [s]
- degree of the polynomial $\mathrm{n} = 1$

Case Study 2

In the second scenario:

- the prediction horizon $N_p = 180$ [s]
- sample time $T_s = 60$ [s]
- degree of the polynomial $n = 1$

Case Study 3

In the third scenario:

- the prediction horizon $N_p = 600$ [s]
- sample time $T_s = 60$ [s]
- degree of the polynomial $n = 3$

Case Study 4

In the fourth scenario:

- the prediction horizon $N_p = 300$ [s]
- sample time $T_s = 60$ [s]
- degree of the polynomial $n = 3$

Case Study 5

In the fifth scenario:

- the prediction horizon $N_p = 600$ [s]
- sample time $T_s = 60$ [s]
- degree of the polynomial $n = 5$

The results obtained with the predictive method are shown in Table 2. The GT load profiles obtained in the various cases are given in Figure 13. The computation times for each case refer to entire simulation.

GT load profile optimization/ Predictive approach				
Case study	*Start-up time**	*Fuel saving**	*Computation time*	*Ratio♣*
Standard start-up	6400 [s]			
Spline (5 point)	4480 [s] (-30%)	~ 33%	1414 [s]	
Case study 1	5280 [s] (-18%)	~ 22%	6333 [s]	
Case study 2	3660 [s] (-43%)	~ 44%	4762 [s]	123%
Case study 3	3540 [s] (-44%)	~ 46%	24437 [s]	11.5%
Case study 4	3420 [s] (-47%)	~ 48%	22134 [s]	14.1%
Case study 5	3300 [s] (-49%)	~ 50%	62545 [s]	4.2%

Table 2: Results comparison

As can be seen from Table 2, the predictive procedure reduces significantly the start-up time and ensures that the constraints imposed, in particular on the peak values of stress, to be respected (Figures 14 and 15).

The method requires an important computational effort that limits their on-line applicability. For an on-line implementation other solutions, like HD-MPC, must be addressed.

The results analysis shows that the quality of the results can be improved by increasing the number of basis functions (i.e. degree of the polynomial). Instead, by making an optimization without delivering to the solver the gradients (black-box approach), the current method is limited to a reduced number of optimization parameters.

* The percentage values are reported to the standard start-up sequence

♣ The ratio is reported to the results of the scenario 1

By using a quadratic cost function that penalizes the deviation from the target value (full load), the choice of the prediction horizon is quite important, in certain cases if the horizon length is too short, for example $N_p = 60$ [s], it leads to loss of performance, inadequate solution in terms of regularity, and often the solver impossibility to compute the optimization problem solution.

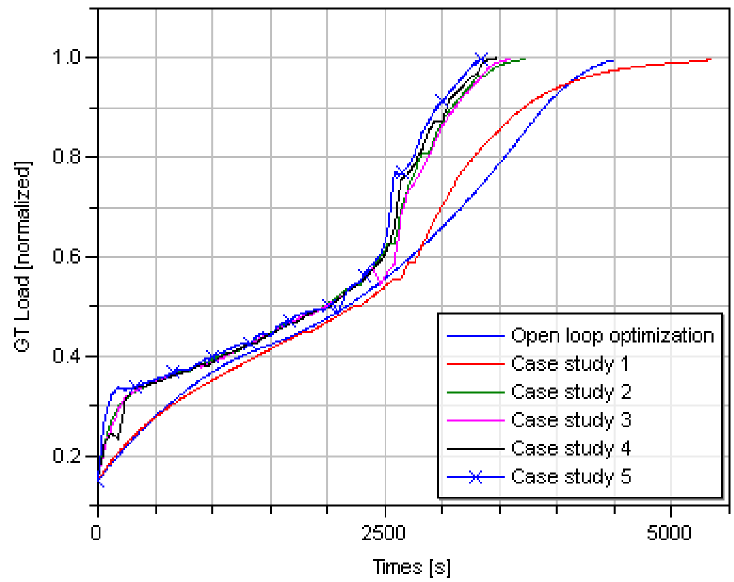

Figure 13: GT load profiles comparison

However, a simple and efficient solution, which is the trade-off between the performance and the computational complexity, is to use functions with a reduced number of parameters to describe the control profiles and a prediction horizon sufficiently long (see for example the case study 2 with linear polynomials). This solution enables the optimization method to provide quite good results, even if this method is not very efficient. Yet, the method is not suitable for on-line applicability.

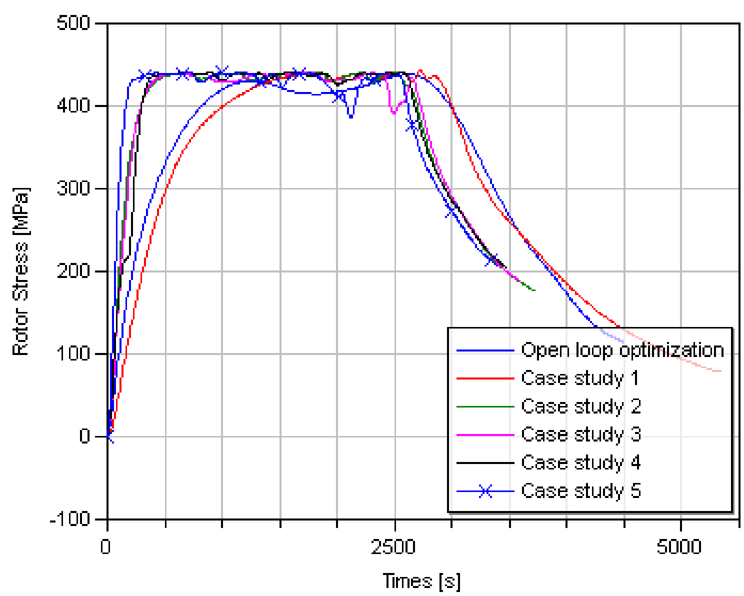

Figure 14: Rotor stress comparison

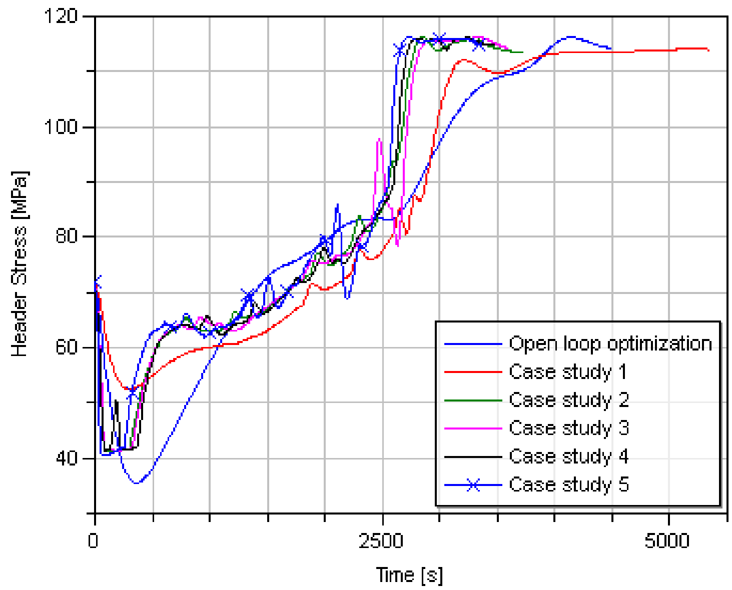

Figure 15: Header stress comparison

3.4 Conclusions

A predictive approach for optimization of the CCPP start-up has been presented in this section. The results obtained show that the new procedure is able to further reduce the start-up time compared to the open-loop optimization procedure. Also, in this section several studies with respect to the type of function chosen to parameterize the control profiles, the length of the prediction horizon, have been performed. The results have shown that a compromise between the performance and the complexity has to be made in order to achieve an acceptable level of computational effort.

To reduce the computational complexity at each sample time, another type of approach is needed. Several different attempts to reduce the computational complexity have been addressed during last decades (see for example the report D.3.1.1). In the following sections, a solution in this sense is proposed by implementing a hierarchical control structure.

4 Hierarchical control structure

Hierarchical control has received a significant attention during the last decades (see report D.3.2.2, for a detailed presentation). The interest is mainly motivated by the fact that an important number of systems can be better controlled with hierarchical structures than with traditional methods.

In this section, attention is focused on the design of hierarchical control systems with MPC. Relied on the solution presented in report D.2.2.2, a hierarchical control structure for the CCPP start-up is proposed. In our case the system under control is assumed to be structured into two layers, each having a different sampling time. For each layer a receding horizon control problem is formulated, considering at the high level an evolution over a longer horizon. The solution from the high layer is communicated to the low layer and used to determine its solution until the procedure is completed. It should be noticed that in this case the MPC problems for each layer are formulated based on the same model and not on the models with different dynamic behaviours as in most of the applications, including the approach proposed in D.2.2.2.

4.1 GT load profile optimization

Problem formulation

Consider that the CCPP model can be given in two dynamic explicit forms, corresponding to each layer:

- High level:

$$\dot{x}(t) = f(x(t), u, L_h(t,q)) \tag{20}$$

- Low level:

$$\dot{x}(t) = f(x(t), u, L_l(t,q)) \tag{21}$$

where $L_h(t)$ is the input variable associated with the high level, while $L_l(t)$ is the input variable corresponding to the low level. For each level, two time scales are used. A long horizon corresponds to the high level while for the low level a short horizon is considered.

Concerning the control variables ($L_h(t)$, $L_l(t)$) represent the GT load for each layer, assuming that the load is described by a parameterized function $L(t,q)$, which satisfies the start-up conditions (see Section 2.2).

In the multistage MPC algorithm presented above, for each layer a sampling time is defined. Thus T_h and T_l are the sampling times for the high level and low level respectively. Also, for simplicity, a relationship between T_h and T_l is fixed, by introducing a time index $k \in \mathrm{N}$, so that $T_h = k * T_l$, with $k \geq 1$.

MPC problem for the high level

At each high level sampling time T_h, the control variable (GT Load) $L_h(t,q)$ is computed by solving a minimum time problem as in the case of the optimization problem presented in Section 2.2.

$$\min_{q,t_f} \left(J = \int_{iT_h}^{t_f} dt \right) \tag{22}$$

subject to the system dynamic (20) and

$$L_h(t_f, q) \geq L_M - \varepsilon_1 \tag{23}$$

$$\left| f(x(t_f), u, L_h(t_f, q)) \right| \leq \varepsilon_2 \tag{24}$$

$$h(x(t)) \leq 0, \ iT_h \leq t \leq t_f \tag{25}$$

(23-25) with the same meaning as in Section 2.2.
The computed optimal solution at a predefined instant T_c, $L_h^*(iT_h + T_c, q)$, is then sent to the low level, and used as reference signal. The procedure is repeated at the new sampling period T_h, when a new optimization is performed, based on updated state, $x((i+1)T_h)$.

MPC problem for the low level

The goal at this level is to track the reference value (L_h^*) of the GT load provided by the controller from the high level. By adopting the receding horizon paradigm, the variable $L_l(t,q)$ is computed by solving the following optimization problem:

$$\min_q J = \int_{jT_l}^{jT_l+N_p} (L_l(t,q) - L_h^*(iT_h + T_c, q))^2 \, dt \tag{26}$$

subject to the system dynamic (21) and

$$h(x(t)) \le 0, \quad t \in [\, jT_l, jT_l + N_p\,] \tag{27}$$

where T_c is a predefined time constant with the condition $T_c \ge N_p$, $j \in \mathrm{N}$ an index used to define the sampling time and (27) denotes all the constraints to be imposed on plant variables.

Since the sampling times are considered different for the two layers, the MPC problem at this level is solved, under the assumption that $L_h^*(iT_h + T_c, q)$ is constant along the time interval $[\, jT_l, jT_l + N_p\,]$.

Then, according to the receding horizon paradigm, only the optimal value $L_l^*(t,q)$ computed for $t \in [iT_l, (i+1)T_l]$ is applied to the system and the overall procedure is repeated at the new sampling period T_l. The structure of the proposed hierarchical controller can be seen in Figure 16a. Also the temporal diagram of the approach is illustrated in Figure 16b.

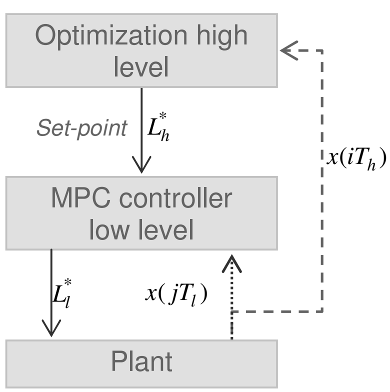

Figure 16a: Hierarchical structure adopted in this report

Remark

In the centralized MPC, the quadratic criterion aims at achieving a target point represented by the full load ($L_M = 1$), which actually enables to drive the system towards the full load as quickly as possible. The MPC problem at low level follows the same principle, the difference being made by the fact that the target points are defined by the solution to the high level optimization problem. In other words the target point $L_M = 1$ is replaced with a target that changes in time (intermediate points generated at each instant T_h by the high level optimized profile). In fact, the use of these intermediate points as targets in the quadratic low level optimization problem, involves the reduction of the computing time, since they “help” the solver to find a solution more quickly than the centralized optimization. This is quite natural because in the centralized case for a prediction horizon N_p, relatively short compared to the final time

t_f, a full load target is imposed. Furthermore, for $T_c = N_p$ the set-points given by the high level are always feasible for the low level optimization problem.

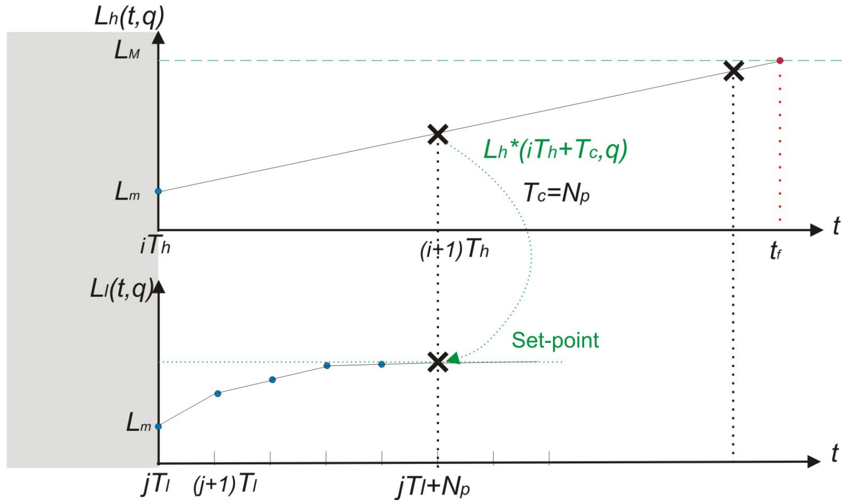

Figure 16b: H-MPC temporal diagram

4.2 Results

To demonstrate the advantages of the hierarchical approach compared to the centralized one, several case studies have been considered. In the sequel the comparison between these two approaches is made by means of the case study 2 presented above, using the linear polynomials to describe the control profiles for each level. The comparative results are shown in Table 3. Also the GT load, ST rotor stress and Header stress profiles obtained with both methods are illustrated in Figures 17, 18, 19.

Centralized/Hierarchical approach						
Case study	*Start-up time**		*Fuel saving**		*Computation time*	
	MPC	**H-MPC**	**MPC**	**H-MPC**	**MPC**	**H-MPC**
Case study 2 **MPC:** $N_p = 180$ [s] $T_s = 60$ [s] **H-MPC:** $T_l = 60$ [s] $N_p = 3T_l$ [s] $T_h = 2T_l$ [s] $T_c = 7T_h$ [s]	3660 [s]	3720 [s]	~ 44%	~ 43%	4762 [s]	2943 [s]

Table 3: Results comparison hierarchical vs. centralized control

It must be noted that the values of the parameters N_p, T_h and T_c, at the high level, have been chosen in order to ensure the best trade-off between the computation time and the start-up performances.

* The percentage values are reported to the standard start-up sequence

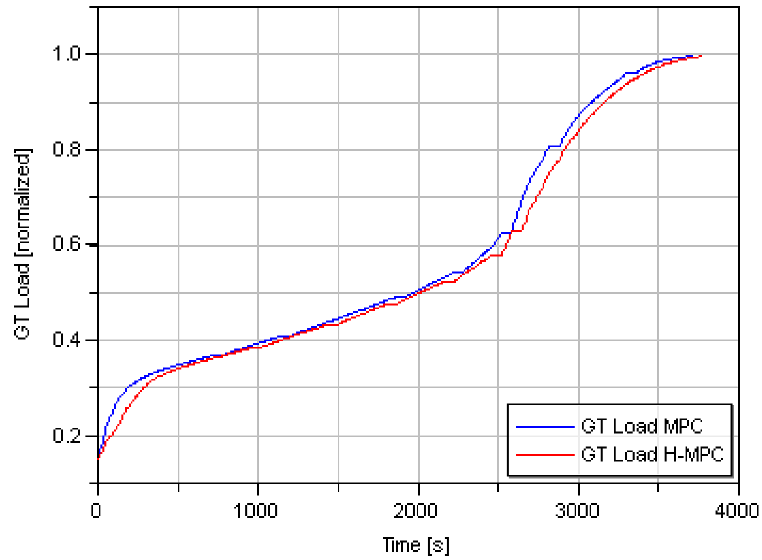

Figure 17: GT load profiles comparison hierarchical vs. centralized control

It can be observed that the hierarchical approach leads to a start-up time comparable to the centralized solution but with a significant reduction of the computation time (with over 39%).

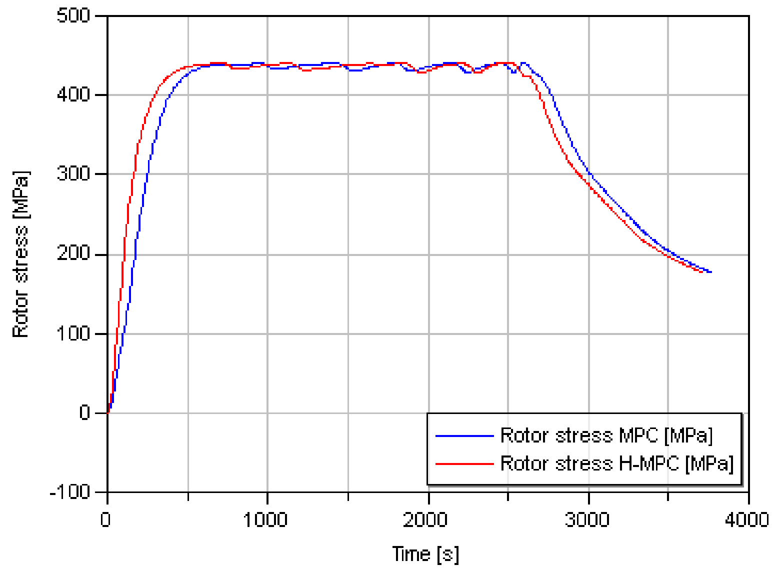

Figure 18: Rotor stress

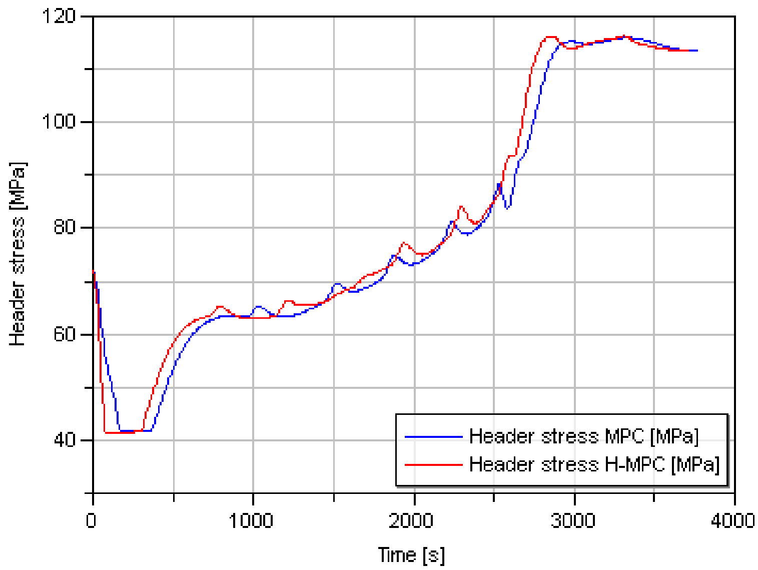

Figure 19: Header stress

4.3 Conclusions

In this section, in order to reduce the computational effort a hierarchical approach has been implemented. The hierarchical structure includes two layers. At the high level a minimum time optimal control problem is periodically solved at a long time period. The solution of this problem is used to update the set-point for the low level. At this level solved at a shorter period a quadratic optimization problem is solved in order to reach as quickly as possible the target provided by the high level.

The hierarchical approach leads to start-up times almost equivalent to the centralized solution but with reduced computation times.

5 Distributed control approach

As already seen in the case of a fully centralized MPC, to solve the optimization problem a significant computational effort is required. An alternative proposed in the HD-MPC project, to deal with the complexity of the control task, is to use distributed approaches.

In addition to the already existing solutions in the literature on distributed optimization (see report D.4.1.1), a series of strategies which deal with the computational complexity by exploiting the distributed and parallel computing, have been proposed in the HD-MPC project.

In general the applicability of these methods to a particular problem requires the system analysis and the assumption of specific separability properties or the reformulation of the original problem in an appropriate form to the required properties.

Concerning the large scale systems, in particular the CCPP start-up optimization, a solution in this direction is not quite obvious. In the following section, the distribution of control in the CCPP's case has been studied.

5.1 Distributed strategy applicability

The adopted steps in order to elaborate a distributed strategy for the CCPP application have been the following:

- analysis of the system and identification of the interaction variables;
- model and objective decomposition.

System analysis

The CCPPs are complex nonlinear systems where the dynamics of the subcomponents are strongly interacting. Accordingly, to identify the interaction variables as well as the possible decoupling of the subsystems, a model linearization is performed. The resulting state matrix is represented in Figure 20.

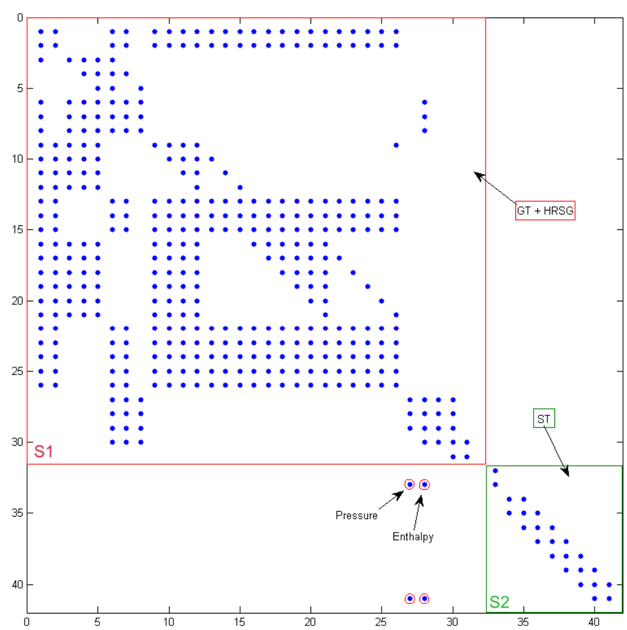

Figure 20: Model linearization: state matrix and subsystems interaction

Analysing the state matrix, the overall system can be decomposed in two subsystems. Apart from this decomposition, another partitioning of the system is quite difficult, due to the fact that strong interactions among the subsystems exist. The subsystems interact by means of the steam characteristics (pressure, enthalpy), which are highly dependent on each other.

Decomposition

The CCPP system has been decomposed in two subsystems S_1, S_2 (see Figure 20). This decomposition can be easily adapted to the start-up phase, because the subsystem S_1 contains GT and HRSG, subsystem that can be used in the first phases of the start-up sequence when the ST is stopped, while the subsystem S_2 contains the ST, unit which operates in the last start-up phases. Also, the physical decomposition of the system is shown in Figure 21.

To implement this decomposition, two Dymola models have been derived. In the first model the subsystem S_1 is connected to a flow rate sink and in the second one the S_2 is connected to a source of pressure. The control variables for the two subsystems are GT load for S_1 and ST throttle opening for S_2. The schematic representation of these models is illustrated in Figure 22.

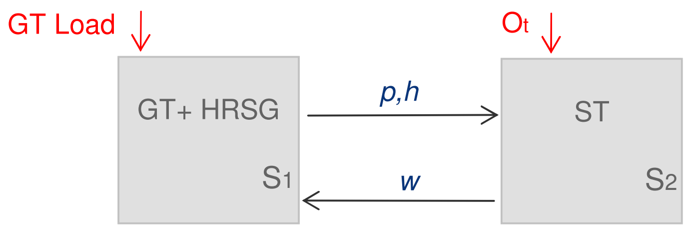

Figure 21: Model decomposition

Figure 22: Physical decomposition: Dymola models

5.2 Open loop profile optimization

The distributed approach is focused on the optimization of two control profiles, GT load and ST throttle.

Problem formulation

Consider the CCPP system consisting of two subsystems (S_1, S_2)

$$\begin{aligned} \dot{x}_i(t) &= f_i(x_i(t), u_i(t), m_i(t)), \qquad x_i(0) = x_{i,0} \\ y_i(t) &= g_i(x_i(t), u_i(t), m_i(t)), \qquad \mathrm{i} = 1,2 \end{aligned} \tag{28}$$

where $x_i(t)$ and $x_{i,0}$ are state vector and its initial condition, $u_i(t)$ is the input vector containing the GT Load and ST throttle ($L(t)$, $O_t(t)$) and $m_i(t)$ are the physical interaction variables of each subsystem (the flow rate - w in the case of the S_1 and the pressure - p and the enthalpy – h for the S_2).

The optimal control problem that has to be solved in a cooperative optimization method over a finite horizon $[t_0, t_f]$

$$\min_{u_{1,2}} J = J_1 + J_2 \tag{29}$$

with

$$J_1 = \int_{t_0}^{t_f} (P_{GT}(t) - P_{\max GT})^2 \, dt \tag{30}$$

$$J_2 = \int_{t_0}^{t_f} (P_{ST}(t) - P_{\max ST})^2 dt$$

subject to (28) and

$$h_1(x(t)) \le 0,\ t_0 \le t \le t_f \quad (31)$$

$$h_2(x(t)) \le 0,\ t_0 \le t \le t_f \quad (32)$$

where P_{GT} and $P_{\max GT}$ denote the GT power and GT maximal power, P_{ST} and $P_{\max ST}$ are the ST power and ST maximal power. (31) and (32) denote the constraints for each subsystem to be imposed on the plant variables. It should be noted, that the objective function and the constraints have been chosen to be separable.

Distributed strategy

To solve the open-loop optimal control problem, the same approach as in the previous cases has been adopted. Specifically, the control profiles are approximated by means of base functions and then the parameters vector are derived by using a nonlinear solver.

As in most distributed optimization methods, to solve the distributed optimal control problems an iterative algorithm is implemented. As the algorithms proposed in the project use structural/sensitivity information that are not available at the time being, the considered algorithm is based on simple communication principle. The algorithm is stated as follows:

Algorithm

1. *Initialization: feasible parameters vectors ($q_1^{[0]}, q_2^{[0]}$) for S_1 and S_2 are chosen and set k:=0.*
2. *Solve the local optimization problems.*

$$\min_{q_1} J_1 \qquad \min_{q_2} J_2 \quad (33)$$

s.t. (31-32)

3. *Communicate the generated interaction profiles:*
   → (w) from S_2 to S_1.
   → (p,h) from S_1 to S_2.

4. *Convergence test.*
   a. *Stop: if $q_1^{[k]}, q_2^{[k]}$ satisfy the convergence condition.*
   b. *Set k:=k+1 and go back to step 2.*

Here, *k* refers to iteration index.

Also, based on the same idea a sequential approach has been implemented:

1. *Initialization*
2. *Solve optimization problem for S_1*

$$\min_{q_1} J_1 \quad (34)$$

s.t. (31)

3. *Update interaction profiles (p,h)*
4. *Solve optimization problem for S_2*

$$\min_{q_2} J_2 \quad (35)$$

s.t. (32)

5. *Update interaction profile (w)*
6. *Convergence test*

Convergence issues

As it can be observed each subsystem does not know the dynamical models which derive the interaction variables trajectories. The main rationale behind the algorithm is to communicate between the subsystems the trajectories of the interaction variables until the convergence condition is ensured. The algorithm idea is inspired from the distributed non-cooperative algorithm introduced in D.3.3.3.

The method is slightly atypical compared to common iterative strategies, where in order to achieve a coordinated optimization the models or cost functions are adapted (e.g. dual optimization or partial goal coordination methods). A typical coordination mechanism (e.g. price-driven coordination), to take into account knowledge of the overall system, lacks from this algorithm, the coordination among subsystems is performed only through physical interactions.

The applicability of distributed methods requires the feasibility of the interaction profiles. In general, the distributed approaches assume that the solution generated by the subsystems is always feasible for their interacting subsystems.

Nevertheless, in many situations, such an infeasibility problem of the profiles can occur. What happens if the interaction profiles generated by the subsystem S_2 are not feasible for the S_1 and/or vice versa? Such an issue occurs in the case of the CCPP start-up optimization problem, when the profile (e.g. flow rate) computed by S_2 is not feasible for the subsystem S_1 simulation. In general the simulation of the CCPP model is quite sensitive to the input control profiles computed by the optimization procedure, leading to the algorithm failure. Moreover, an important aspect in the distributed mechanism for the CCPP start-up, which remains questionable, is the way to choose a coordination mechanism that takes into account the influence between the subsystems.

Although, a series of new optimization methods for distributed control have been proposed in the HD-MPC project, their applicability to the CCPP start-up problem is constrained by a particular model type, by the complexity, or in certain cases by the existence of powerful integration tools which, for example, have to be able to generate the gradients of the objective function and the constraints. Nowadays, such a tool support for complex physical models, in particular Modelica models, is quite weak, therefore makes it difficult to test and to use the developed algorithms.

5.3 Conclusions

In this section the applicability of the distributed approach to solve optimal control problem in the case of the CCPP start-up has been studied. As in the case of most distributed optimization strategies, an algorithm to compute the solution of the distributed control problems is implemented. In this case, a simple communication based algorithm has been considered.

The encountered issues underline once again that the applicability of a distributed approach for complex systems is quite difficult, involving on one side a good knowledge of the process and on the other side efficient algorithms. The latter exists but their applicability to complex Modelica models is limited.

6 General conclusions

During this report the control problem of combined cycle power plant start-up has been studied by means of physical models. The novel model-based methods proposed in this work have shown a significant improvement of the start-up performance. Nevertheless, the lack of optimization support able to handle large scale Modelica models and to provide required information in order to apply efficient optimization algorithms makes it inappropriate the use of such models for control purposes. Moreover, despite the fact that a smooth CCPP model has been developed the existing optimization tools are not mature enough to cope with its complexity.

The hierarchical and distributed model-based control approaches proposed in the HD-MPC project have proved attractive solutions with a series of real advantages. As previously pointed out in this report their applicability is currently limited in the CCPP case. The applicability of a distributed control approaches, where the global system is decomposed in several subsystems interacting with each other, remains questionable while the implementation of the hierarchical approaches such as the ones proposed in the project requires the consideration of a number of factors about the robustness and reliability of the control with respect to the lower level, considerations that are quite difficult to take into account at the moment.

One specific aspect of the combined cycle process and not only in their case, generally for large scale systems, is the sensitivity to simulation. Thus, if during optimization procedure, the control trajectories or model parameters are not well-posed the simulation fails. The implicit constraints defined by these limits on simulation are difficult to make explicit, but these play an important role in the optimization procedure.

Bibliography

[1] L. T. Biegler. An overview of simultaneous strategies for dynamic optimization. Chemical Engineering and Processing: Process Intensification, 46(11):1043–1053, November 2007.

[2] M. Diehl, H. J. Ferreau, and N. Haverbeke. Efficient Numerical Methods for Nonlinear MPC and Moving Horizon Estimation, in Chapter 32 of Nonlinear Model Predictive Control, 384:391–417. Springer, 2009.

[3] T. Binder, L. Blank, H.G. Bock, R. Burlisch, W. Dahmen, M. Diehl, T. Kronseder, W. Marquardt, J.P. Schloder and O. von Stryk. Introduction to model based optimization of chemical processes on moving horizons, Online Optimization of Large Scale Systems: State of the Art, 295–339. Springer, 2001.

[4] L. T. Biegler. Efficient solution of dynamic optimization and NMPC problems. Nonlinear Predictive Control, 219–244. Birkhauser, 2000.