Design, optimization and validation of start-up sequences of energy production systems.

Adrian Tica

N° d'ordre : 2012-09-TH

THÈSE DE DOCTORAT

DOMAINE : STIC

SPECIALITE : AUTOMATIQUE

Ecole Doctorale « Mathématiques, Télécommunications, Informatique, Signal, Systèmes Electroniques »

Présentée par :

Adrian TICĂ

Sujet :

Conception, optimisation et validation des séquences de démarrage des systèmes de production d'énergie

Soutenue le 1 juin 2012 devant les membres du jury :

M. Damien FAILLE	EDF R&D	Examinateur
M. Didier DUMUR	SUPELEC	Co-directeur de thèse
M. Eric BIDEAUX	INSA de Lyon	Rapporteur
M. Frédéric KRATZ	ENSI de Bourges	Président
M. Hervé GUEGUEN	SUPELEC	Directeur de thèse
M. Mohammed M'SAAD	ENSI de Caen	Rapporteur

Pentru familia mea...

"...the people who are crazy
enough to think they can change
the world are the ones who do."

STEVE JOBS

Acknowledgments

The work reported in this thesis has been conducted in the Hybrid System Control Team (ASH) of the Supelec Rennes Campus. This research has been carried out in collaboration with EDF (Electricité de France) and supported by the European 7th framework STREP project "Hierarchical and Distributed Model Predictive Control (HD-MPC)", contract number INFSO-ICT-223854.

At Supelec, I have been fortunate to meet and work with wonderful people. Firstly, I would like to express my gratitude to my supervisor, Prof. Hervé Guéguen, for giving me the opportunity to integrate the research project, and for the unequalled support over the last three years. I can not put into words what I have learned from him. His intelligence and attitude to research have been an important source of inspiration. I sincerely thank my co-supervisor, Prof. Didier Dumur. His ability to synthesize, comments and advice, based on long experience in process control, have been very helpful. I am grateful for his attention to detail and professionalism shown towards my work. It has been a huge honor to work and learn from them.

The participation in an international project was very interesting and challenging. I have had the opportunity to meet brilliant people, whose experience and knowledge helped me to enrich my own theoretical and practical skills. Also, the collaboration and the support provided by EDF's experts have been a significant factor in the development of this research. I would like to express my gratitude to Dr.-Ing. Damien Faille, from EDF R&D, for his confidence in my work and results. His constant support and experience in the modeling and the control of power plants have made an agreeable and rewarding collaboration. I would also like to thank Ing. Frans Davelaar for his availability and assistance during my research. It was a pleasure to work with both of them.

I would like to express my most sincere gratitude to the members of my thesis committee: Mr. Frédéric Kratz, Professor at ENSI Bourges, who did me the honor of chairing the examination committee, Mr. Eric Bideaux, Professor at INSA Lyon, and Mr. Mohammed M'Saad, Professor at ENSI Caen, who agreed to review this dissertation. I am very grateful for their efforts to accomplish this work and for their constructive comments and suggestions.

It was a very pleasant experience to work in the ASH Team. I would like to thank all team members: Jean Buisson, Hervé Cormerais, Marie-Anne Lefebvre, Nabil Sadou, Pierre-Yves Richard and Romain Bourdais. All my colleagues: Antoine, Eunice, Ilham, Khang, Maxime, Mihai, Meriem, Soma, ... are acknowledged for creating a friendly atmosphere and for all those scientific/non-scientific discussions during these years. Daniel Moroşan deserves my warmest thanks for his friendship, support and many unforgettable moments experienced during this time.

I also wish to thank everybody from Supelec Rennes Campus who helped me throughout this thesis, and especially, to Catherine Pilet, Clairette Place, Grégory Cadeau, Ophélie Morvan and Myriam Andrieux for their administrative support and their kindness.

Working away from home and from loved ones is always a challenge. I have been lucky to be surrounded by great friends here in Rennes, and I thank to: Alexandra, Alina, Anthony, Dali, Diana, Eliza, Rado, Remus, Roxana, ... I could not forget those who have supported me from back home. They are too many to be mentioned here.

A special thanks to my wonderful girlfriend, Raluca, for her love and support during this time.

Finally, and above all, I want to express all my gratitude and love to my parents and to my brother. Thank you for your unconditional support and for giving me the opportunity to fulfill my dreams!

Adrian

Contents

Résumé en français

0.1 Contexte

De nos jours, les producteurs d'électricité doivent répondre à des enjeux économiques et environnementaux de plus en plus élevés et s'adapter aux nouveaux modes de fonctionnement des systèmes de production d'énergies.

Actuellement, plus de la moitié des besoins mondiaux en énergie électrique sont couverts par des centrales thermiques à combustibles fossiles. Parmis celles-ci, les centrales à cycles combinés utilisent une technologie qui permet d'obtenir un rendement nettement supérieur à celui des centrales classiques à combustible fossile, et d'assurer ainsi une diminution considérable des émissions polluantes [1]. De plus, les installations à cycles combinés offrent plusieurs avantages économiques, avec entre autre, un coût d'investissement réduit et un temps de construction relativement court surtout vis-à-vis des centrales nucléaires.

C'est pourquoi ces dernières années, les centrales à cycles combinés ont rencontré un fort succès et se sont développées [2]. Elles représentent le lien entre les anciennes technologies de production basées sur la combustion des combustibles fossiles (destinées à jouer un rôle dominant dans le scénario de l'énergie au moins jusqu'en 2030) et l'évolution vers des sources novatrices et renouvelables d'énergie.

Dans le contexte français de l'énergie électrique, ce type de centrale thermique est destiné à compléter la production de base des centrales nucléaires en particulier lors des pics de consommation. Elles sont donc démarrées et arrêtées relativement souvent pour offrir ce service d'appoint. De plus, la derégulation du marché de l'électricité a introduit de nouveaux producteurs sur le réseau, et des variations importantes du prix de l'électricité peuvent être constatées pendant une journée. Par conséquent, les centrales sont soumises à un nombre important des séquences démarrage/arrêt.

Ces transitoires, en particulier les démarrages, sont cependant délicats car ils conduisent à un vieillissement accéléré des équipements. Il est donc intéressant de fournir une nouvelle stratégie de gestion plus efficace qui permet d'optimiser le temps de démarrage de la centrale, tout en respectant les contraintes de vieillissement des matériels.

L'objectif de cette thèse est de proposer une méthode de conception, d'optimisation et de validation des séquences de démarrages qui permette de diminuer les temps et les coûts de démarrage tout en préservant la durée de vie, en respectant des contraintes qui limitent le vieillissement des composants physiques.

La solution proposée consiste à effectuer l'optimisation du démarrage par une approche basée sur un modèle physique. En outre, les travaux ont porté sur l'application des approches de commande prédictive pour optimiser la séquence de démarrages des centrales à cycles combinés.

Afin d'obtenir un fonctionnement sûr et efficace, et en même temps de pouvoir exploiter pleinement le potentiel du processus, les méthodes suggérées dans ce manuscrit sont basées sur un modèle physique précis de la centrale lors la phase de démarrage. Pour cela, une bibliothèque spécialisée, basée sur un langage de modélisation physique a été utilisée.

0.2 Centrale à cycles combinés, la problématique du processus de démarrage

0.2.1 Description générale

Les centrales à cycles combinés sont des systèmes thermodynamiques qui comportent deux ou plusieurs cycles de puissance, dont chacun utilise un fluide de travail différent (d'où l'appellation cycles "combinés") [3], [4], [5].

En général, cette technologie combine deux cycles thermodynamiques : elle associe le fonctionnement d'une turbine à combustion (cycle de Brayton) à celui d'une chaudière de récupération et d'une turbine à vapeur (cycle de Rankine). Les cycles mixtes vapeur/air (fluides de travail les plus communément utilisés) fournissent un rendement élevé du fait que les deux cycles sont complémentaires du point de vue thermodynamique : l'énergie calorifique des fumées issues de la turbine à combustion (TAC) est récupérée par la chaudière, et constitue la source d'énergie pour générer de la vapeur pour la turbine à vapeur (TAV).

L'utilisation du deuxième cycle (eau-vapeur) peut avoir différentes finalités :

- production d'électricité ;
- production de vapeur pour un réseau de chauffage ou à des fins industrielles ;
- production de vapeur et d'électricité : cogénération [6].

0.2.2 Centrale à cycles combinés : structure, fonctionnement

Les installations à cycles combinés présentent une large gamme d'options de design et de configurations en ce qui concerne le nombre d'unités, le type de chaudière, le type de TAV, etc. La structure de base d'une centrale électrique à cycles combinés contient les composants principaux suivants :

- Turbine à combustible (TAC) ;
- Chaudière de récupération ;
- Turbine à vapeur (TAV) ;

- Ligne vapeur ;
- Condenseur ;
- Générateur.

Dans ce document, le type de centrale étudié a une configuration typique 1-1-1 (une TAC, une chaudière, une TAV). La représentation schématique d'une telle centrale est illustrée dans la Figure 1. Aussi pour l'objectif proposé, une chaudière et une turbine à vapeur avec un seul niveau de pression (haute pression) ont été considérées.

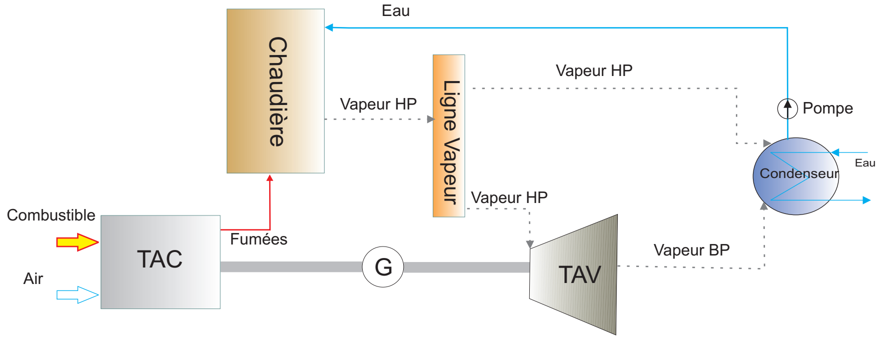


Figure 1: Schéma d'une centrale à cycles combinés de type 1-1-1

Le fonctionnement normal d'un tel circuit se présente comme suit : de l'air ambiant est comprimé, puis mélangé avec du combustible et brûlé à une pression constante. Le gaz à haute température et haute pression créé par la combustion est détendu dans la TAC pour générer une puissance mécanique. Les fumées issues de la TAC sont utilisées pour produire de la vapeur dans la chaudière de récupération. La vapeur produite est détendue dans la TAV pour fournir une puissance supplémentaire. La vapeur issue de la TAV (BP-basse pression) est collectée par un condenseur. Le cycle eau/vapeur est fermé, en alimentant la chaudière avec de l'eau provenant du condenseur.

0.2.3 Le processus de démarrage

Le démarrage des cycles combinés est une tâche complexe, comportant plusieurs phases et des conditions qui doivent être respectées simultanément. La procédure de démarrage dépend de chaque technologie utilisée, mais en général elle peut être décomposée en quatre étapes principales :

1. *Phase de préparation* : cette première étape prépare l'allumage de la TAC ;
2. *Phase de lancement de la chaudière* : cette phase correspond au conditionnement thermique, et son objectif est de démarrer la chaudière afin d'obtenir les caractéristiques de la vapeur (température, pression, qualité) adaptées pour l'admission dans la TAV ;
3. *Phase de lancement de la TAV* : au cours de cet étape, la TAV est démarrée par l'admission de vapeur ;

4. *Phase de prise de charge* : pendant cette dernière étape, la charge de la centrale est augmentée jusqu'à la puissance nominale.

En général, les techniques classiques de démarrage sont assez conservatrices, en se concentrant principalement sur la sécurité et la disponibilité plutôt que sur l'efficacité de l'opération. Ces procédures sont conçues de manière à préserver naturellement la durée de vie des composants de l'installation. Toutefois, afin de répondre aux exigences du contexte concurrentiel actuel, sans compromettre la sécurité du fonctionnement, de nouvelles stratégies efficaces de démarrage sont nécessaires. Les nouvelles stratégies doivent intégrer la fiabilité et la flexibilité du fonctionnement, et fournir la puissance attribuée par le dispatching économique dans un temps minimal, tout en satisfaisant les contraintes d'exploitation et en minimisant les coûts.

Dans cette thèse le processus de démarrage de la centrale est traité comme un problème d'optimisation dynamique, consistant à trouver un profil optimal pour les variables de contrôle qui minimise une fonction objectif et remplit un ensemble de contraintes imposées par la dynamique de l'installation et par les principales variables du processus (en particulier les stress thermiques et mécaniques).

L'optimisation de la procédure dans son ensemble exige de prendre en compte un système hybride contenant des variables discrètes et continues qui interagissent. Toutefois, afin de réduire la complexité de l'optimisation, le problème a été décomposé en sous-problèmes plus simples correspondant à chaque phase de démarrage. Ce document est centré sur la dernière partie du processus, lorsque les deux turbines sont synchronisées et connectées au réseau (*Prise de charge*). Cette phase est considérée comme la plus critique, car l'impact du stress sur les composants est important, ce qui affecte fortement leur durée de vie.

0.3 Modèle physique de centrale à cycles combinés pour le démarrage

L'optimisation du démarrage repose sur la détermination d'une trajectoire sous contraintes d'un système dynamique correspondant à la centrale. Pour réaliser cette optimisation il est donc nécessaire de disposer d'un modèle de l'installation.

Actuellement, les processus doivent fonctionner sous des spécifications économiques et environnementales de plus en plus strictes. En général, ces exigences peuvent être satisfaites lorsque les systèmes fonctionnent sur une gamme large de conditions d'exploitation et, en particulier à proximité des limites de sa région d'exploitation admissible. Dans ce contexte, pour décrire adéquatement le comportement dynamique du processus, des modèles non-linéaires sont nécessaires. Ceci est d'autant plus vrai pour les centrales électriques à cycles combinés qui ont des comportements dynamiques fortement non-linéaires lors de leur démarrage.

Pour mener les études, nous avons construit un modèle non-linéaire de centrale avec la configuration spécifiée dans le chapitre précédent, en utilisant pour la modélisation le langage Modelica [7] et l'environnement de simulation Dymola [8]. Modelica est un langage orienté objet pour la modélisation des systèmes physiques de grande dimension, complexes et hétérogènes. Ce choix permet en effet de construire un modèle fiable en assemblant et configurant des composants à partir des bibliothèques dans une approche hiérarchisée et structurée. Le modèle développé utilise la bibliothèque libre pour la modélisation de centrales thermiques (ThermoPower [9]) et a été paramétré avec des données de conception et de fonctionnement d'une unité typique [10].

Dans ce document, nous sommes intéressés à la réalisation d'un modèle physique de la centrale pouvant être utilisé pour l'optimisation de la séquence de démarrage. Ainsi, le modèle développé est adapté à l'étude du démarrage, c'est-à-dire qu'il est en mesure de représenter la séquence de démarrage. La représentation mathématique du système considère des modèles simplifiés pour les éléments non critiques pour l'optimisation du démarrage (turbine à gaz, condenseur, etc). Pour les autres éléments, il comprend des composants dont le comportement est valable sur tout le domaine de fonctionnement et modélise des caractéristiques telles que les stress thermiques et mécaniques lorsqu'ils sont les plus limitants.

Les équations du modèle proviennent principalement du premier principe de la thermodynamique (conservation d'énergie, conservation de masse).

0.4 Modèle physique de centrale adapté pour l'optimisation

0.4.1 Optimisation des modèles Modelica

Avec le langage Modelica, un grand nombre de modèles (linéaires, non linéaires, hybrides, etc.) peut être décrit, et de nombreux outils performants permettent de les simuler facilement. Par contre, l'optimisation de ces modèles est plus difficile. En effet, les outils pour l'optimisation statique et dynamique des modèles Modelica sont généralement peu puissants.

En raison de l'hétérogénéité et de la complexité des modèles Modelica, l'optimisation est en général faite en traitant le modèle comme une boîte noire : l'optimisation utilise les résultats de simulations sans que la structure particulière du modèle ne soit explorée selon des méthodes heuristiques. L'objectif a donc été d'étudier comment on pouvait utiliser de tels modèles, en particulier le modèle de la centrale développé, pour appliquer des algorithmes d'optimisation efficaces.

La classe d'algorithmes envisagée pour l'optimisation appartient aux méthodes fondées sur les gradients. Cette classe peut traiter des systèmes de grande échelle, en étant capable de résoudre des problèmes de contrôle optimal sous conditions de temps réel. En général, l'application des méthodes de gradient impose une série de conditions sur la formulation du modèle, en particulier, celle de l'absence de sources de discontinuité (par exemple, les structures *si-équations*).

En ce sens, une méthodologie a été proposée. L'approche vise à obtenir des modèles Modelica appropriés pour l'optimisation avec des méthodes basées sur le gradient. Le modèle du chapitre précédent est considéré comme modèle de référence. Celui-ci est aussi utilisé pour réaliser et valider des modèles adaptés pour l'optimisation.

0.4.2 Méthodologie pour créer des modèles Modelica appropriés pour l'optimisation

0.4.2.1 Élimination des sources de discontinuité

Le principe de l'approche consiste à associer à chaque composant du modèle initial, une version dans laquelle les sources de discontinuité sont approchées par le biais des représentations continues, en gardant ainsi la même structure et une bonne précision du modèle.

En bref, les étapes accomplies dans l'approche sont les suivantes :

a) l'analyse des composants du modèle de simulation et l'identification des sources potentielles de discontinuité ;

b) l'élaboration de composants du modèle sans sources de discontinuités, en introduisant les représentations continues ;

c) la validation des composants du modèle.

Dans la phase de construction, la représentation continue est basée sur l'utilisation d'une approximation de la fonction de Heaviside :

$$\forall x \in \mathbb{R}, \quad H(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \tag{1}$$

remplacée par l'approximation lisse :

$$\forall x \in \mathbb{R}, \quad H_k(x) = \frac{1}{1 + e^{-kx}} \tag{2}$$

où k est le coefficient de lissage (voir la Figure 2).

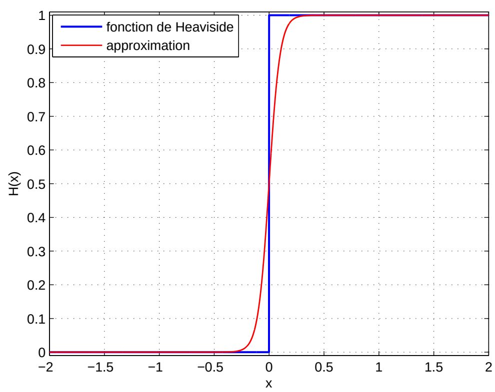


Figure 2: La fonction de Heaviside et son approximation pour k=*20*

L'utilisation d'une approximation lisse sert à rendre continues les discontinuités des structures conditionnelles. Par exemple, pour un modèle conditionnel Modelica comme présenté ci-dessous :

$$h_m = \textbf{if} \quad h \leq h_l \quad \textbf{then} \quad h_l \quad \textbf{else} \quad h_v \tag{3}$$

la représentation continue est ainsi formulée :

$$f(h) = \frac{1}{1 + e^{-k*(h-h_l)}} \tag{4a}$$

$$h_m = h_v * f(h) + (1 - f(h)) * h_l \tag{4b}$$

Un problème spécifique a été la représentation des tables de vapeur/eau. Ces modèles ont été estimés sur différents domaines thermodynamiques (régions), et ensuite raccordés de manière lisse pour obtenir une représentation continue.

0.4.2.2 Bibliothèque ThermoOpt

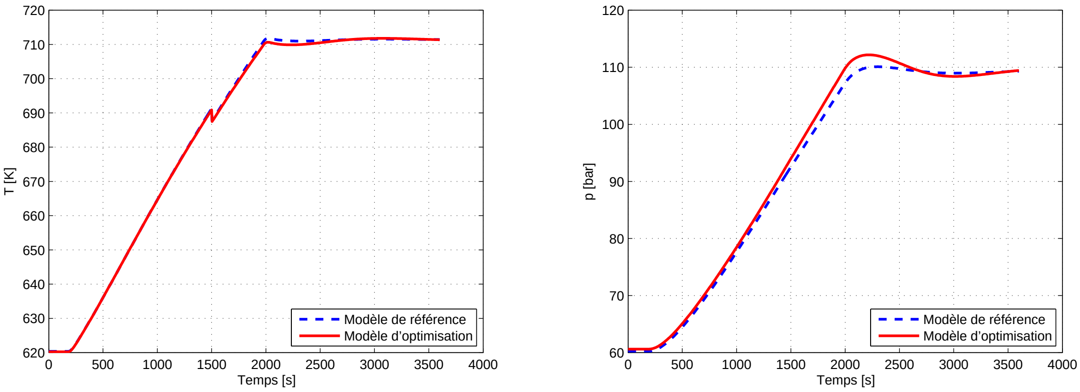
La méthodologie proposée a permis de construire une bibliothèque de composants Modelica adaptés à l'optimisation à partir d'une bibliothèque standard Modelica conçue pour la simulation. Un modèle de la centrale, approprié pour l'optimisation, a été construit en utilisant des composants provenant de la bibliothèque ThermoOpt. Le modèle a les mêmes caractéristiques et la même configuration que la représentation à cycle combiné de référence. Ainsi, le modèle pour l'optimisation, peut être déduit du modèle de simulation par simple substitution de la bibliothèque.

0.4.2.3 Validation des modèles de composants

L'ensemble des nouvelles versions des composants et le modèle global de la centrale a été validé par comparaison en simulation avec les modèles originaux. Pour démontrer la cohérence des modèles de composants, plusieurs scénarios de simulation ont été étudiés. En général, la validation des éléments de la bibliothèque d'optimisation a été effectuée en comparant leurs réponses, aux variations des signaux d'entrée, avec les réponses obtenues par les modèles de référence.

Quelques résultats de simulation pour une variation de type échelon du débit du désurchauffeur et pour un profil de la charge de la TAC qui suit une rampe avec une pente d'environ 0,65%/min, appliquée après 180 secondes, sont illustrés dans la Figure 3.

Comme on peut le constater, les réponses du modèle d'optimisation sont assez proches de celles de référence. Par rapport au modèle original, le comportement de la température et de la pression de la vapeur dans la chaudière se caractérise par une très légère différence (erreur relative inférieure à 2.5%).



(a) La température de la vapeur

(b) La pression de la vapeur

Figure 3: Réponses comparatives

0.5 Procédure d'optimisation du démarrage

0.5.1 Optimisation des trajectoires en boucle-ouverte

En utilisant le modèle élaboré dans le chapitre précédent (celui approprié à l'optimisation), une approche nouvelle est proposée dans le but d'optimiser le profil des variables de commande pendant la phase de démarrage. Le modèle de la centrale est utilisé pour déduire le profil optimal, en supposant que ce profil peut être décrit par une fonction paramétrée, dont les paramètres sont calculés en résolvant un problème de commande optimale en temps minimal, sous réserve d'un certain nombre de contraintes sur les variables principales du système. En étendant cette idée, le processus de démarrage a été posé comme un problème d'optimisation "juste-à-temps".

L'entrée qui induit la plupart des caractéristiques dynamiques pendant le démarrage étant la charge de la TAC, en particulier pour la phase considérée, celle-ci a été choisie comme variable d'optimisation.

Néanmoins, la procédure d'optimisation peut viser d'autres profils de contrôle et différents objectifs, comme par exemple, la minimisation d'usure des matériaux, ou encore la minimisation des coûts d'exploitation.

La proposition d'effectuer l'optimisation en temps continu en recherchant des solutions dans des ensembles de fonctions paramétrées particulières permet de transformer le problème original de dimension infinie en un problème de programmation non-linéaire de dimension finie. Les travaux ont permis d'étudier diverses familles de fonctions (fonctions Hill, fonctions spline) et de proposer un compromis entre l'efficacité du démarrage et la complexité du problème d'optimisation.

Les principaux facteurs limitant pour la réduction du temps de démarrage proviennent des gradients de température qui apparaissent dans les parties métalliques les plus épaisses des composants, en particulier dans le rotor de la TAV et dans le collecteur de la sortie du surchauffeur, l'échangeur situé dans le flux de gaz le plus chaud. Par conséquent, dans le problème d'optimisation, des contraintes sur les valeurs maximales des contraintes thermiques et mécaniques dans le rotor et dans le collecteur ont été imposées. En effet, ces pics déterminent la consommation de la durée de vie pendant toute la séquence de démarrage, sans tenir compte des effets additionnels causés par la variation du niveau de stress (fatigue).

0.5.2 Résultats

Afin de montrer les avantages de l'approche proposée, une comparaison entre la performance du démarrage lorsque la procédure optimale est appliquée, et une séquence classique à prise de charge constante a été effectuée. Le tableau 1 résume les résultats obtenus.

Optimisation de la charge TAC						
Fonction/ sous-intervalles (N)		Temps de démarrage [s]	Gain de temps[1][%]	Économie de carburant[1][%]	Opt. paramètres	Temps calc. [s]
linéaire par morceaux	N=1[*]	*6038*	-	-	1	20
	N=10	5540	≈8	≈8	10	561
	N=20	4980	≈17	≈15	20	855
2-Hill		4466	≈26	≈30	6	169
spline quadratique	N=3	4777	≈21	≈23	3	58
	N=3[v]	4397	≈27	≈30	5	154
	N=4[v]	4278	≈30	≈34	7	253
Optimisation de la charge TAC et de l'ouverture de la valve TAV						
spline quadratique	N_{TAC}=4[v] N_{TAV}=3[v]	4120	≈32	≈36	12	663
Problème du démarrage "juste-à-temps"						
2-Hill		4510	≈25	≈29	5	321
spline quadratique	N=4[v]	4321	≈28	≈31	6	388

[1] *relativement à la procédure standard*
[*] *correspondant à la procédure standard avec une rampe de la charge TAC de 2 MW/min*
[v] *paramétrage avec sous-intervalles variables*

Table 1: Résultats de simulation/optimisation

Les résultats montrent que la nouvelle procédure optimale réduit considérablement le temps de démarrage, en consommant jusqu'à 36% de moins qu'un démarrage traditionnel à prise de charge constante. En raison du fait que les outils pour l'optimisation statique et dynamique des modèles physiques, en particulier pour les modèles développés dans le langage Modelica, sont généralement peu puissants, les solutions proposées sont basées sur un compromis entre l'efficacité du démarrage (par exemple, le temps de démarrage), d'une part, et les performances de calcul (le temps de calcul, les propriétés de convergence), d'autre part.

0.6 Approche de commande prédictive appliquée au processus de démarrage

0.6.1 MPC approche pour l'optimisation du démarrage

La dernière partie des travaux a été centrée sur l'utilisation de la démarche d'optimisation à temps continu précédemment présentée dans une stratégie de commande à horizon glissant de type Model Predictive Control (MPC). Cette approche permet de corriger les dérives liées aux erreurs de modèle et aux perturbations, d'une part, et de considérer pour l'optimisation à chaque instant d'échantillonnage une fonction plus simple conduisant à un meilleur compromis complexité/temps de démarrage, d'autre part. Cette approche de commande conduit cependant à des temps de calcul importants.

0.6.2 Approche de commande prédictive hiérarchisée

Afin de réduire le temps de calcul, une structure de commande prédictive hiérarchisée travaillant à des échelles de temps et sur des horizons différents a été proposée.

La structure comporte deux niveaux : haut (NH) et bas (NB). Au niveau haut un problème de contrôle optimal en temps minimal est résolu sur une longue période de temps. La solution de ce problème est utilisée pour mettre à jour la référence pour le niveau bas, où un problème de suivi de consigne sous contrainte est résolu sur une courte période (T_p).

L'idée derrière l'approche hiérarchique a été de créer une structure de contrôle avec des problèmes d'optimisation plus appropriés en terme de temps de calcul.

0.6.3 Résultats

L'application d'une stratégie de commande prédictive permet d'améliorer d'une manière significative les performances du démarrage. Dans le Tableau 2 les résultats obtenus pour deux scénarios sont présentés en faisant varier les paramètres de contrôle (l'horizon de prédiction T_p, les périodes d'échantillonnage pour chaque contrôleur T_s pour MPC, $\{T_l, T_h\}$ pour H-MPC, la constante T_k qui définit l'instant auquel le signal de référence est envoyé au niveau bas et le type de paramétrisation pour la charge TAC).

Les profils de la charge TAC issues de la stratégie de MPC sont adaptés à l'état actuel du système et sont générés avec une capacité de prédiction spécifique. Ainsi, avec une estimation précise de l'évolution future du comportement de l'installation, en particulier sur l'évolution du stress dans le rotor et dans le collecteur, le profile de la charge peut être plus agressif, en réduisant le temps de démarrage sans dépasser les limites admissibles (voir la Figure 4).

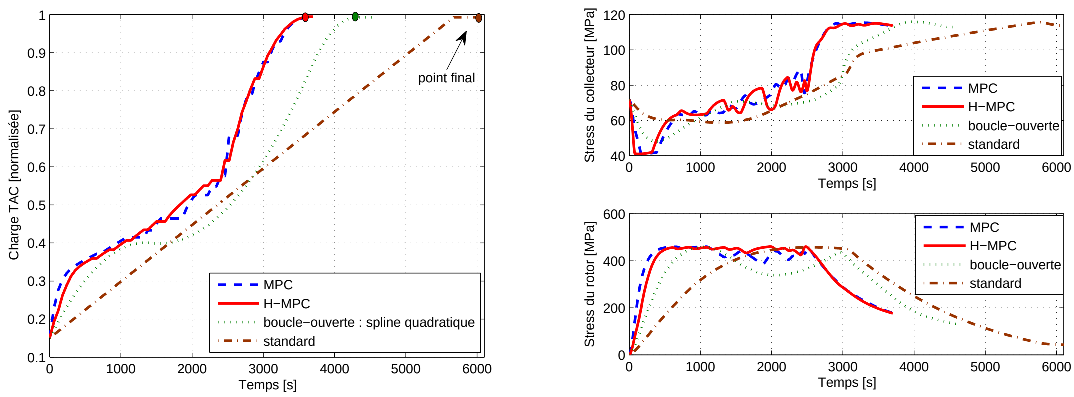
Cependant en raison de la complexité du problème d'optimisation, la résolution en ligne à chaque pas d'échantillonnage est assez difficile. A l'exception de quelques cas particuliers, le solveur n'est pas en mesure de fournir une solution dans un délai défini inférieur à la période d'échantillonnage du processus.

L'algorithme hiérarchisé (H-MPC) fournit des solutions compatibles avec une implantation en temps réel. En outre, les performances du démarrage sont proches de celles données par le régulateur MPC avec un seul niveau. Néanmoins, l'application efficace de l'algorithme en temps réel conditionne la précision avec laquelle le profil de charge TAC est approximé.

Optimisation de la charge TAC								
Approche	Paramètres de contrôle [s]		Temps de démarrage [s]	Gain de temps [%]	Économie de carburant [%]	Temps calc.[s] total	Temps calc.[s] moyen
standard	-	-	*6038*	-	-	-	-
spline quad.	-		*4278*	-	-	-	-
MPC S_1	T_s=60	T_p=3T_s	3600	≈40*	≈43*	2884	48
H-MPC S_1	T_l=60	T_p=3T_l	3600	≈40*	≈43*	1425	10.5[NH]
	T_h=2T_l	T_k=5T_p					18.2[NB]
MPC S_3	T_s=60	T_p=3T_s	3240	≈46*	≈48*	6218	115.1
H-MPC S_3	T_l=60	T_p= 5T_l	3240	≈46*	≈48*	2949	9.6[NH]
	T_h=3T_l	T_k= 2T_p					51.2[NB]

* *relativement à la procédure standard*
[NH] *Temps de calcul moyen au Niveau Haut*
[NB] *Temps de calcul moyen au Niveau Bas*
^{S_1} *Scénario 1 : MPC avec le profil de la TAC décrit par une fonction linéaire*
^{S_3} *Scénario 3 : MPC avec le profil de la TAC décrit par les polynômes de Lagrange*

Table 2: Résultats comparatifs commande MPC



(a) Profils de la charge TAC

(b) Évolution des stresses

Figure 4: Résultats comparatifs de simulation. Pour les contrôleurs MPC, le paramétrage correspond au Scénario 1 du Tableau 2

0.7 Conclusions et perspectives

Cette thèse a été consacrée à l'élaboration de nouvelles stratégies pour l'amélioration de la performance du démarrage des centrales à cycles combinés. Les stratégies proposées sont basées sur l'utilisation d'un modèle physique du processus et de techniques d'optimisation dynamique pour la réalisation des objectifs prévus. Dans la suite, les principales contributions de la thèse sont listées :

- Un premier apport consiste en une méthodologie qui permet à partir d'une bibliothèque Modelica, conçue pour la simulation, de construire une bibliothèque adaptée à l'optimisation. En utilisant des composants provenant de cette bibliothèque, un modèle de centrale, approprié pour l'optimisation, a été construit et validé.

- La suite des travaux porte sur l'utilisation du modèle pour optimiser le temps de démarrage d'une centrale à cycles combinés. Le modèle a été utilisé pour déduire le profil optimal de la charge de la turbine à combustible, en supposant que ce profil peut être décrit par une fonction paramétrée, dont les paramètres sont calculés en résolvant un problème de commande optimale sous contraintes. La solution proposée optimise en temps continu le profil en recherchant l'optimum dans des ensembles de fonctions particulières (Hill, splines, ...). Les travaux ont précisément porté sur l'étude de diverses familles de fonctions et sur le compromis entre la performance du démarrage et la complexité de l'optimisation. En étendant cette idée, le processus de démarrage a été posé comme un problème d'optimisation "juste-à-temps".

- La dernière partie des travaux considère l'intégration de la solution à temps continu dans une commande à horizon glissant, de type prédictive à modèle non linéaire (NMPC). Cette approche en boucle fermée permet d'une part, de corriger les dérives liées aux erreurs de modèle et aux perturbations, et d'autre part, d'améliorer le compromis entre le temps de calcul et l'optimalité de la solution. Une structure hiérarchisée à 2 niveaux travaillant à des pas de temps et sur des horizons de prédiction différents est finalement proposée pour réduire le temps de calcul.

L'ensemble des travaux a conduit à des résultats intéressants tant sur le plan applicatif, avec une réduction importante du temps théorique de démarrage (jusqu'à 47% par rapport à une séquence à prise de charge constante), que sur le plan méthodologique.

En prolongement de ces travaux de thèse, plusieurs directions peuvent être envisagées pour les développements futurs. Les principales pistes sont les suivantes :

→ l'amélioration de la procédure d'optimisation par le calcul et la prise en compte des informations dérivées (par exemple, les gradients).

→ la conception d'un observateur en mesure de reconstruire les variables d'état du système en utilisant un modèle physique Modelica/Dymola ; la considération des solutions robustes pour initialiser les modèles Modelica ;

→ l'étude de la robustesse des structures de commande proposées ;

→ l'extension de la bibliothèque ThermoOpt avec de nouveaux modèles de composants, et la modélisation des configurations réelles de centrales électriques.

Acronyms

API Application Programming Interface

APROS Advanced Process Simulation Software

ARX AutoRegressive with eXogenous variable

ATT Atemperator (Desuperheater)

BLT Block Lower Triangular

CCPP Combined Cycle Power Plant

CCFF Combined Cycle Fully Fired

DAE Differential Algebraic Equation

DASSL Differential Algebraic System Solver

DCS Distributed Control System

DSH Desuperheater

ECO Economizer

ECO-HP1 High Pressure first stage economizer

ECO-HP2 High Pressure second stage economizer

ECO-HP3 High Pressure third stage economizer

ECO-HP4 High Pressure fourth stage economizer

ECO-IP Intermediate Pressure stage economizer

ECO-LP Low Pressure stage economizer

EDF Electricité de France

ETS Emissions Trading System

EU European Union

EV Evaporator

EV-IP Intermediate Pressure Evaporator

EV-HP High Pressure Evaporator

EV-LP Low Pressure Evaporator

FEM Finite Element Method

G Generator

GT Gas Turbine

GTS Gas Turbine System

H-MPC Hierarchical Model Predictive Control

HAT Humid Air Turbine

HILS Hardware In the Loop Simulation

HL High Layer

HP High Pressure

HRSG Heat Recovery Steam Generator

IEA International Energy Agency

IAPWS International Association for the Properties of Water and Steam

IAPWS-IF97 IAPWS Industrial Formulation 1997

IGCC Integrated Gasification Combined Cycle

IGV Inlet Guide Vane

IP Intermediate Pressure

IPOPT Interior Point OPTimizer

ISE Integral Square Error

LdT Load Transducer (Sensor)

LL Low Layer

LP Low Pressure

LT Level Transducer (Sensor)

MPC Model Predictive Control

NMPC Nonlinear Model Predictive Control

Mtoe Million tones of oil equivalent

NLP Non-Linear Programming

OE Output Error

ODE Ordinary Differential Equation

PT Pressure Transducer (Sensor)

QP Quadratic Programming

RH-IP1 Intermediate Pressure first stage reheater

RH-IP2 Intermediate Pressure second stage reheater

RMSE Root Mean Square Error

SC Supercritical

SI International System of Units

SL Steam Line

SH Superheater

SH-HP1 High Pressure first stage superheater

SH-HP2 High Pressure second stage superheater

SH-IP Intermediate Pressure superheater

SH-LP Low Pressure superheater

SNOPT Sequential Non-linear OPTimizer

SpT Speed Transducer (Sensor)

SQP Sequential Quadratic Programming

ST Steam Turbine

STAG Steam and Gas

STIG Steam Injection Gas Turbine

TT Temperature Transducer (Sensor)

USC Ultra Supercritical

CHAPTER 1

Introduction

1.1 Background

During the last century, the continuous growth of the world's population, the industrial development and the accelerated urbanization have resulted in a major energy demand, which is in full expansion. The energy affects the everyday life of each of us, and the increased need for more energy will require in the next years massive investments and substantial improvement in energy efficiency, all while managing the risks associated with greenhouse gas emission and climate change. The recent energy market liberalization, which aims the prices reduction, the security of supply, the improvement of efficiency and the development of renewable energy production, through competition, is expected to have a positive impact on economy and environment. The international agreements for climate protection and energy security will involve major changes in the energy supply and use, representing a fundamental challenge. The companies that produce energy and also the whole society, in general, will undergo significant changes over the next decades, whether due to government intervention, increasing competition, limited resources or simply public opinion.

Nowadays, at a global level, there is a strong relationship between economy and energy. As a result of the economic crisis and the global spread of its impacts, the future that will unfold in its wake faces unprecedented uncertainty. The strength of the economic recovery holds the key to energy outlook for the following years.

Although global energy demand has recently been reduced by economic recession, the recent annual projection [11] of the International Energy Agency (IEA), suggests that the world primary energy demand might increase with about 36% between 2008 and 2035, from around 12300 million tones of oil equivalent (Mtoe) to over 16700 Mtoe (with 1.2% per year on average). This scenario takes into account the policy commitments and plans that have been announced by the governments around the world, including the national pledges to reduce greenhouse gas emissions and future plans to phase out fossil-energy subsidies. The assumed climate-protection policies and actions make a notable difference to energy trends, presented until now, thus the predicted rate of growth in this case is by 0.8% lower than the rate of the last 27 years and by 0.2% lower than the estimated rate in the previous scenario of the IEA [12]. In this scenario, due to the fact that the fossil fuels still dominate the world's energy mix in 2035, the concentration of greenhouse gases in atmosphere are projected to stabilize at the equivalent of 650 parts per million (ppm) of CO_2, causing most likely an increase in temperature of 3.5°C or more, for a long term. Also the scenario suggests that, in order to meet the Copenhagen Accord's overall goal of limiting the global average temperature increase to below two degrees Celsius (2°C), a much stronger action is needed.

According to the previous scenario, known as reference scenario, the world primary energy demand is expected to grow with about 40% in the next 20 years (see Figure 1.1). In this scenario, the prediction of the energy trends is based on the existing policies of the governments before 2009, e.g. the recent commitments of the Copenhagen Accord are not considered. Due to the lack of new actions to tackle climate change and the increased global demand for fossil energy, an important growth in CO_2 emissions, from 29 gigatones (Gt) to 40 Gt, is expected. The predicted rate of growth in energy demand under this scenario has serious effects on the environment, leading to a global average temperature of 6°C. The projections in this scenario are rather unrealistic, because it is not expected

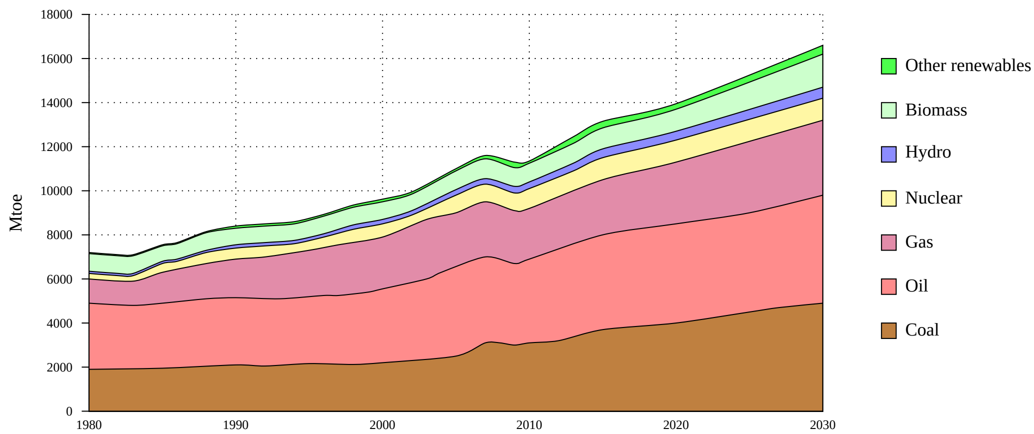


Figure 1.1: World primary energy demand by fuel according to the Reference scenario of IEA

for the governments around the world to take any action, in terms of the main energy challenges: energy demand, security of supply, and carbon dioxide (CO_2) emissions (the main cause of climate change). But, on the contrary, it is more likely that the governments will proceed in this matter and put the global energy system on a sustainable path. So this scenario has been used to analyze the possibilities that the energy holds for the future and what actions should be taken in order to change its evolution.

The major consequences of climate change require strong policies regarding the CO_2 emissions, so according to the climate experts, in order to have a realistic chance to achieve the 2°C goal, the concentration of greenhouse gases in the atmosphere would need to be stabilized around 450 ppm of CO_2 equivalent. To meet this stabilization target, the '450' scenario of the IEA [13] estimates that the primary energy demand would grow with 20%, between 2007 and 2030, less than half than in the case of the reference scenario without climate policy.

Despite the increased interest, manifested for renewable sources (hydro, wind, geothermal, solar, etc.), the traditional energy sources, fossil fuels (coal, oil and natural gas), remain largely dominant in all three scenarios of the IEA. Their share of the total primary energy mix varies depending on the scenario.

Nowadays, more than 65% of world's electrical energy needs are covered by fossil fuel power plants (see Figure 1.2). In line with IEA, between 2008 and 2035, together with the population growth and with the universal coverage of electricity services, the world electricity production is expected to grow more strongly than any other form of energy, the growth rate of demand might exceed 2.2% per year. Globally, the nuclear and renewable

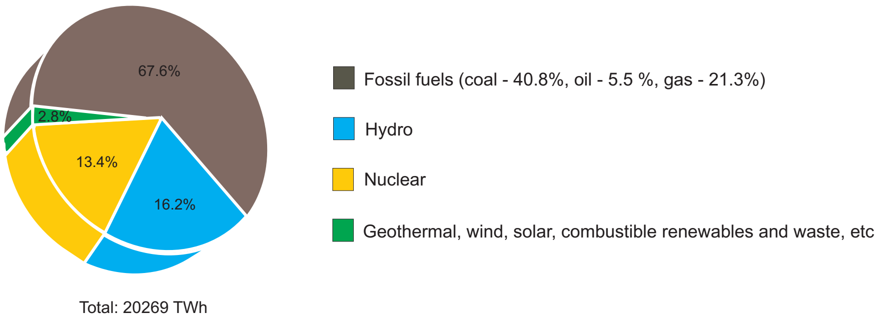


Figure 1.2: World electricity production 2008: partition by primary energy sources (Source IEA 2010)

sources expand, while the fossil fuels continue to dominate but with a decrease from 67.6% to 55% in the share of the overall generation. The coal still remains the principal source of electrical generation, although its share decreases from 40.8% to 32%.

In France, compared to other countries, the present situation is atypical as a consequence of the government action, that, immediately after the oil shock of 1973, decided to expand the country's nuclear power capacity. Currently, the nuclear is the dominant primary energy source, with over 42%. The French electricity production is provided in the ratio of 90% by nuclear and hydroelectric power (see Figure 1.3), this involves advantages in climate protection, making France one of the countries with the lowest greenhouse gas emissions in the world. The nuclear energy offers also a low cost of generation and a greater energy security, allowing France to export significant amounts of electricity. In recent years, the coal began to be gradually replaced by natural gas, while the energy production from renewable sources (besides hydro-power) has been moderately increased. It should be noted that despite the fact that the French model based on a nuclear dominated electricity production, maintains the fossil fuel share of total production around 10%, which is a figure very low relative to other industrialized countries. An energy evaluation in the last years as well as some essential aspects of nuclear energy in France can be found in [14], [15].

Concerning the energy policies, France has ambitious goals aimed not only to the

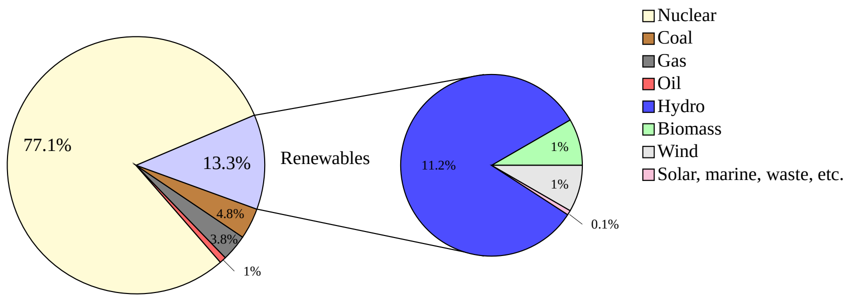


Figure 1.3: France electricity production 2008: partition by primary energy sources (Source IEA 2010)

energy security and competitive supply, but also the protection of the environment and the climate, in order to reduce further the CO_2 emissions (e.g. Grenelle Environment [16]). In the next years, the main targets with regard to sustainable development under European Union (EU) 20-20-20 commitment by 2020 (20% improvement in energy efficiency, 20% reduction in greenhouses gas emissions and 20% renewable energy supply) [17] are:

- a reduction in CO_2 emissions by 14% in the sectors outside Emissions Trading System (ETS);
- an increase in share of renewable energies to 23% of total energy consumption;
- a 20% reduction of the energy consumption, by improving energy efficiency.

All the predictions presented are based on scenarios covering demographic, geopolitical and economic issues which involve risks and uncertainties, because they depend on certain circumstances and are related to events, that will or may occur in the future. But beyond the estimated figures, the information presented above offer us a large view about the global energy evolution.

1.1.1 Electricity market

The electricity is a form of energy which has become an essential part of today's society. Every day the range and quantity of activities which need electrical energy are increasing. So the current energy production methods must cope with the demand growth and society needs. Briefly, we can expect important changes. The electricity industries all over the world are undergoing a period of turmoil and transition. The way in which the electricity is generated, who are the producers, who financed its production, how it is distributed, how prices for it are calculated, they all are in a continuous change. Together with the electricity technology evolution, the role of those involved (e.g. governments, electricity producers, consumers, etc.), are changing.

A key aspect in the electric power markets is to ensure the stability and the reliability of the supply: due to its "non storable" nature, the electricity must be produced, transmitted and consumed in real time. It means that suppliers have no inventory of electricity and have to be able to generate power at levels high enough to meet periods of peak consumption. Another problem is the market volatility, the large fluctuating in demand during the day requires a quick reaction from power generation plants to maintain the balance between supply and demand. The achievement of a reliable electricity supply and an efficient distribution was and remains a major priority.

In recent years a new priority has been set by the global efforts to deregulate the electricity markets. Deregulation should improve the energy market, by opening the competition, offering more options and lower prices to the customers. The private companies have begun to install their own power plants and supply electricity to the grid. Currently, the electricity manufacturers have to compete to sell their own product. If in a regulated market, in the past, the companies obligations were to serve their customers with reliable power, the cost minimization and profitability were considered as secondary endpoints, in this competitive context, overall production costs have become a key driver for the electricity producers. The companies must offer electricity at the lowest cost, while still fulfilling the requirements of grid stability and reliability.

The availability and the reliability of supply have an important role in economical terms. Regarding the availability, if a unit is down, the producer has to either generate power in another station or buy from another manufacturer. Anyway, both cases involve additional costs. A high availability allows the operator to run a plant during longer periods per year, resulting in higher returns. In terms of reliability, this is crucial in deregulated markets. During peak times, an important part of the income is generated, so the plant must be reliable. Also in order to limit losses, the outages can be scheduled outside of peak hours, when costs are lower.

The market risks that can affect the investments (e.g. the price for electricity that changes every hour depending on supply and demand, unexpected costs of complying with greenhouse gas emission reduction regulations, uncertain future fuel costs, etc.) are accepted by the companies which expect that their plants will be competitive and that they get an important profit during their exploitation. The deregulation creates a more dynamic environment where the risks and its minimization have a significant weight. As the risks mitigation is paramount, the investments in new and expanded power stations are performed in close consultation with public policymakers, regulators, and other stakeholders. In this context, the electricity generators are focused to buy or build plants technologies with an industrial maturity and competitive costs. It should be noted that in many cases, the building of new plants, with long construction times and high capital costs, is considered more hazardous from the private investor's point of view, so the use of the existing technologies is more interesting.

An attractive solution is the combined cycle technology [1]. The global electricity supply industry admits that the first choice as production technology for the private investors is represented by the combined cycle power plants (CCPPs), because of their outstanding advantages such as high efficiency and low environmental impact. Also this technology provides several economic advantages, including low investment costs and short construction times compared to the conventional power plants, and even more than the nuclear plants offer.

1.1.2 Combined cycle power plant start-up optimization. Motivation

In recent years, several new combined cycle plants have been installed and many existing units have been re-powered [2]. The continuously growing number of CCPPs installed in the world, together with the increasing energy demand and the competition for energy dispatch to deal with deregulated power markets, require the development and the implementation of innovative strategies, either for control at partial or full load, or for the start-up/shut-down phases.

This second topic is of great importance since in the current context, as a result of competition among energy providers and daily fluctuations in power demand, CCPPs are undergoing frequent start-up/shutdown operations. Indeed, depending on their profitability, the combined-cycle units are shut down for short periods of time or for longer periods. Thus, during operation, these power plants are subject to a large number of transients. Furthermore, considering the fact that the start-up process is only a cost factor (no electrical power is delivered), and that the lifetime consumption of the plant components is severely affected by the thermal and mechanical stresses reached during this phase, the control and optimization of the start-up process represent a major interest.

In general, the traditional start-up techniques are quite conservative, being focused mainly on safety and availability rather than on efficiency of the operation. These procedures are designed in order to naturally preserve the lifetime consumption of the plant components. However, in order to meet the requirements of the present competitive context without compromising the safe operation, efficient start-up strategies are necessary. The new strategies have to ensure reliability and flexibility in operation, and to deliver the power assigned by the economic dispatch in minimal time, while satisfying the operating constraints and minimizing the costs.

Summarizing, the proposed research is motivated by the need to improve the efficiency of combined cycle power plants start-up so as to meet tighter economical and environmental specifications, and to cope with stringent competition induced by the deregulation of energy market.

1.2 Dissertation objectives

Driven by the need to deal with a number of factors such as increased production demand, tight competition and new environmental policies, the main objective of this dissertation is to investigate and propose new methods to improve the CCPP start-up performances, while fulfilling constraints imposed on the physical process variables. The proposed solution is to perform the optimization of the start-up of the combined cycle plants by means of a model-based approach. Moreover, the optimal design of the start-up procedure is considered by means of a model-based predictive approach, the goal being to create a framework for control of the CCPP start-up.

In order to achieve safe and efficient operation, and at the same time to be able to exploit the full potential of the process, the suggested methods are based on an accurate physical model of the plant. For this purpose, a dedicated modeling language together with a specialized library are used.

The goal of this work is not only to investigate and propose new start-up control strategies, but also to emphasize the fact that the availability of a reliable, nonlinear model of a plant can represent an important tool for the design of efficient management strategies and advanced control techniques.

The modeling approach presented in this dissertation is aimed at allowing practitioners to develop and use operational physical simulation models built on the expert's knowledge, for model-based control applications. The proposed methodology also serves as a means of transforming existing knowledge-based models into optimization-oriented models.

1.3 Organization and highlights of the dissertation

The dissertation content is organized as follows:

Chapter 2

The second chapter presents a general overview of the combined cycle power plants. It covers important aspects of power plant design, operation, and optimization. The content of this chapter is focused on the operation of a particular CCPP during the start-up phase, and defines the control specifications for this phase. Thus, plant is partitioned in subsystems and for each subsystem the operational constraints and the existing control loops are indicated. The start-up procedure is described, and the objectives which can be envisaged by the start-up optimization are discussed. A state of the art of the design technologies and approaches used for the optimization of the start-up process concludes the chapter.

Chapter 3

In the third chapter the framework for achieving a physical model of a combined cycle plant is given, with a brief description of the dedicated modeling language and the library for thermo-power systems used to develop process models. The Chapter 3 describes a combined cycle plant model built by using the Modelica language and the *ThermoPower* library. The model is based on the first principle laws and adapted to the start-up phase. The plant data are derived from a detailed simulator used to study the start-up of a real plant and the global behavior of the model has been validated by experts in the domain. Due to their modeling features and complexity the direct use of physical Modelica model for optimization purposes is often difficult. The study presented in this chapter will represent a first step towards achieving suitable models for optimization.

Chapter 4

In order to obtain a model that can be handled by efficient nonlinear algorithms for optimization and control purposes, a methodology is given in Chapter 4. The method aims at transforming the physical model developed in Chapter 3, considered as a reference plant, into an optimization-oriented model. The principle of the method is based on the reformulation of the plant model initially presented with discontinuity sources into a smooth model by means of continuous approximations of the Heaviside function. Based on the derived model components, a new optimization-oriented library for the modeling of combined cycle power plants is developed. Results of validation are presented to demonstrate the consistency of the model components and to emphasize the accuracy of the smooth CCPP model developed for optimization, compared to the original representation.

Chapter 5

A model-based approach to carry out the optimization of the combined cycle plants start-up phases, with focus on the last phase, is proposed in Chapter 5. The approach seeks to find an optimal profile of the control variables, in particular of the gas turbine load, by assuming that these profiles can be described by a parameterized function. The parameters used for the profile-function description are computed by solving an optimal control problem, subject to the plant dynamics represented by the Modelica/Dymola optimization-oriented model and to a number of constraints on the main plant variables, such as pressures, temperatures and stresses. Constrained by the lack of tool support for dynamic optimization relied on Modelica models, the suggested procedure is designed to achieve a good compromise between reliability, optimality of profile and computational efficiency.

Chapter 6

The sixth chapter extends the approach already suggested by implementing a model predictive control strategy. Formulating the start-up problem by means of a MPC controller makes possible to exploit the features of this advanced control strategy (such as, its prediction capability and ability to explicitly handle constraints, its built-in robustness properties) in order to derive fast GT loading rates, which lead to the reduction of the transient time. However, due to the computational effort the application of MPC under real-time conditions is limited. In the second part of the chapter, the MPC framework is redesigned to make it suitable for control of CCPP start-up process under real-time requirements. Specifically, a hierarchical predictive structure with two layers is proposed in which a controller at each level deals with a specific objective.

Conclusions and perspectives

The last chapter reviews the research works presented in this manuscript, and offers several perspectives for future developments.

Appendixes

This manuscript contains several Appendixes in order to complete the content of the work, and to enable the reader to become familiar with the addressed topics.

CHAPTER 2

Combined cycle power plants, start-up problem

2.1 Conventional Power Plants

The thermal power plants represent one of the major sources of power generation. In a thermal power plant, the electricity is produced by converting successively the chemical energy stored in primary sources (fossil fuels such as coal, oil, natural gas, etc) or the nuclear energy, in thermal energy, mechanical energy and finally in electrical energy. The thermal-mechanical energy conversion is performed by means of gas turbine (GT) power plants or steam turbine (ST) power plants, while for the mechanical-electrical conversion, the generators are used. The transformation of the chemical energy in thermal energy is carried out in the furnace of the steam generator (boiler), for steam power plants and in the combustion chamber, in the case of the gas turbine power plants [18], [19], [20], [21]. Regarding the steam turbines, these use a separate heat source and do not directly convert fuel to electric energy. The energy is transferred from the boiler to the turbine through high pressure steam. In other words, the steam turbine plants generate electricity as a by-product of heat (steam) generation, unlike the gas turbine plants, where the heat is a by-product of power generation. In general, ST plants can achieve an efficiency of over 40% and their capacity can vary from 50 kW to several hundred MWs for large utility power plants.

Over time, the GT installation, used mainly in aviation, suffered a considerable progress in order to reach powers unit comparable with other forms of energy production. Thus, the plants based on GT can cover a wide domain of electrical power production (from 3 to over 270 MW) and can reach up to 40% efficiency, even more for the modern units. Nowadays, the GT plants are used rather to cover the peak electricity demand and due to their short start-up time, these have a significant role in emergency services. Both GT and ST power plants offer a wide variety of designs and complexity [19], [18], in order to improve the efficiency and/or to assure the performance specifications.

The GT and ST technologies are based on two thermodynamic cycles: the Joule cycle or Brayton cycle and its derivatives which describe the operation of a gas turbine engine; the Rankine cycle and its derivatives (such as Rankine-Hirn cycle with overheating) which describe the thermodynamic transformations of the water-steam cycle of a steam power plant. In the following only the basic ideas of the thermodynamic cycles and plant operation are presented, a complete description can be found in [4], [22], [19], [20], [18].

As a comparison between the GT and ST plants, besides the fact that each technology presents advantages and disadvantages (e.g. GT plants offer an investment cost lower than ST plants, ST plants can operate with a large variety of fuels, GT has a simple design since no boilers and their auxiliaries are required, GT can be started quickly from initial conditions, etc.), an essential difference is that for two turbines with the same size, the ST will provide an electric power significantly larger than the GT.

For many years, in the past, the conventional steam power plants represented the first option to use fossil fuels to produce electricity. At that time, the thermal plants already used steam turbines with reheating and high steam parameters (170-180 bar, 540-560°C). Currently, these plants represent about 30% of installed capacity and still remain predominant over the world, because of their efficiency and their power unit significantly larger than that the power stations based on GT. A large part of nuclear plants use also the thermal-mechanical energy conversion based on a steam cycle with lower temperatures and pressures.

Initially, the construction of such power plants aimed mainly the electricity production with a minimal cost investments, and the issues as the reduction of the environmental impact and maximization of the efficiency, were not priorities. Nowadays, thermal power plants offer a high availability and a low rate of unplanned or forced outage. In addition, the environment protection has become a priority.

Normally, the average lifetime of a conventional plant is of 30 to 40 years, but in some cases even after this period, the plant components can be in excellent operating conditions. Consequently, a significant number of thermal power plants (which started commercial operation in years 1970 -1980) are currently in a technical state that allows their to operate in acceptable conditions for another 10 to 15 years. However, these plants can not meet the current requirements in terms of efficiency and impact on the environment, so a rehabilitation is needed.

Concerning the rehabilitation of existing plants, there are different techniques, depending on several factors such as the topological conditions, the constructive features, the technical conditions, the life of major equipments, etc. One of the most attractive solutions is to use the GTs [5], [2]. Repowering of the conventional steam plants by completing with a GT is synthesized to the development of particular combined cycle plant.

Over the last years, the improvement of the systems efficiency and the concept of sustainable development related to environmental protection, had to seek more efficient solutions in energy consumption and power plants with significantly improved performance. The improvement of the material quality and the technological evolution have led to a considerable increase in efficiency and productivity of the thermal power plants. Moreover, high efficient cycles are available: the units designed for combined cycles (gas-steam) or conventional supercritical steam parameters, which provide performance considerably higher than the conventional units. For example, the implementation of Supercritical (SC) and Ultra Supercritical (USC) power generation technologies [23], [24], [25], the plants which operate at temperatures and pressures above the critical point of water, provides high efficiency (over 45%), with minimum fuel consumption and gas emission, as well as a reliable supply of electric energy at low cost. Also, the combined cycle plants based on the new technologies of turbines, give an efficiency up to 60%, which in terms of the environment, contributes to the minimization of greenhouse gas emission [26]. Furthermore, the new combustion techniques implemented in the GT technologies allow to achieve lower rates of emissions of NO_x and other pollutants [27], [28], [29]. Nevertheless, the most of combined cycle installations use fuels whose estimated reserves are low and their cost is uncertain on a long term (e.g. use natural gas as the main combustible for the GT).

An interesting solution, developed in recent years envisage the coal exploitation, after the gasification process, in so called Integrated Gasification Combined Cycle (IGCC). IGCC technology converts the coal into a cleaner burning synthesis gas (syngas) and as its name suggest, combines the coal gasification with combined-cycle technology. Actually IGCC represent the most efficient and clean way to use the coal for the electricity production [30], [31], [32], [33].

Currently, thermal power plants have to be flexible to meet rapidly fluctuations of demand levels, as well as to remain reliable, by minimizing the environmental impacts and maximizing the efficiency. So, in the industry, the new challenges are to ensure the flexibility and reliability of the operation with minimum forced outages and the lowest possible costs, implementing innovative strategies that reduce CO_2 emissions and deal with deregulated power markets.

Despite the fact that the current trends target in particular renewable energy sources [34], the new energy technologies will not replace the large conventional units over the short- to medium-term. Energy infrastructure takes time to develop and even longer to achieve the necessary industrial competitiveness and reliability. Therefore the energy production will rely for a long time period on the use of fossil fuels. In this context the combined cycle plants and all technologies that emerge from it (e.g. IGCC), will most likely meet a strong growth in the coming years.

2.2 Combined Cycle Power Plants

2.2.1 General description

Combined Cycle Power Plant (CCPP) is an advanced power generation technology that uses two or more thermodynamic cycles, in order to improve the efficiency [3], [4], [35], [5], [18], [36]. They represent the connection between the energy production technologies based on fossil fuels (which seem to be destined to play a dominant role in the energy scenario at least until 2035, according to IEA) and the evolution towards renewable and innovative sources of energy.

In general, this technology combines the operation of a gas turbine Brayton cycle with that of a steam turbine system Rankine cycle, using air and steam as working fluids. The steam/air combined cycles (the most used working fluids) provide an increased thermal efficiency because the two cycles are complementary in terms of thermodynamic: the heat generated by the GT (Brayton cycle) has a temperature that is conveniently used as energy source for the steam turbine system (Rankine cycle). In addition, the working fluids (air and steam/water) are inexpensive, widely available and non-toxic.

The idea to exploit the second cycle has come from the fact that more than 60% of the thermal energy generated by Brayton cycle is rejected through the exhaust. So, by using the ST a part of the exhaust gas energy generated by the gas cycle is recovered, enabling to obtain higher efficiencies than a GT alone (simple cycle configuration). The connection between the GT and the ST is performed by means of a Heat Recovery Steam Generator (HRSG), that uses the hot gases exhaust from the GT to produce steam that can be expanded in the ST, to generate additional power. The challenge in these systems is to obtain an integration level that maximizes the efficiency at an economic cost. Furthermore, the water-steam cycle, which recovers energy can have different purposes:

- power generation only;
- production of steam for urban heating network or for industrial purposes;
- simultaneous production of steam and electricity: cogeneration [6].

2.2.2 CCPP features

CCPPs represent the most efficient, reliable and economic method of power generation from fossil fuels, currently available. For these reasons, the combined cycle technology is one of the first options for most of the new power plants installed in the world.

The CCPPs commercially available today achieve a net thermal efficiency up to 60%. Further, the recent developments in GT materials and technology to sustain higher firing temperatures and higher pressure ratio [37], have led to efficiency records of more than 60% [38]. The constant technological development of the GT as well as the performance improvement of the steam cycle, is expected to lead to a higher efficiency. Efficiency ratings are very important in economic terms because over a 20-year project life, the cost of fuel represents about 70% of the operating cost for a CCPP. Thus, the efficiency improvements can provide considerable savings in fuel cost for the plant operators and lower electricity rates for the customers.

A high efficiency with regard to fuel resource consumption also means that more electricity can be generated with less environmental damage. So, another major advantage corroborated with the improved performance is the significant reduction in pollutant emissions [39]. In comparison with the coal-fired power plants, the new combined cycle technologies use less fuel per kWh and is much cleaner. Combined cycle plants emit by up to 90% less nitrogen oxide (NO_x) and virtually no sulfur dioxide (SO_2), mercury or particulate matter (PM); they can reduce the CO_2 emissions by up to 75%. These advantages become more and more important in the current context where the operating standards and other factors concerning the environment protection are increasingly stringent.

The combined cycle systems do not represent a new technological innovation, this technology dates back more than 60 years [40], [1]. In spite of this, only after 1990, together with the preoccupation to find solutions to produce electricity at minimal costs, with high efficiency and with a reduced environment impact, CCPPs have experienced a real expansion, proving to be a reliable solution for power generation. Before this period, due to the nuclear energy growth and the low price of the fossil fuels, the development of an alternative high efficiency technology, was not a requirement.

Over the time, the combined CCPPs technological evolution has undergone different generations, progressing in parallel with the GT and steam cycle development. So, at the beginning of 1960s (for the first and second generation), the plants efficiency was around 25 to 30%, then near to 1975 (for third generation), the 40% threshold has been exceeded, to go today to reach 60% (fourth generation). These figures are given just to get an idea of how the combined cycle efficiency has progressed.

Additionally to the advantages listed above, the combined cycle systems provide flexibility that includes several aspects. Combined cycle plants are fuel flexible since they are able to operate by burning a wide variety of fuels, including natural gas, coal, oil fuels, etc. Another aspect of flexibility is the site. CCPPs require less space than equivalent coal or nuclear stations and less constraints on site due to lower environmental impact. CCPPs can be installed close to the demand points, reducing therefore the need for long-distance electricity transmission and decreasing the overall cost of electricity for consumers.

Combined cycles plants are characterized by low investment costs. An example, the capital construction cost of a combined cycle unit is approximately two times lower than a comparable coal plant with the same output. The difference is even higher compared to nuclear plants. The payback period for CCPPs is considerably shorter than conventional power plants. Usually, the payback period for a combined cycle plant is around 5 to 12 years, compared with around 25 years for conventional one. The current trend of the costs is to continue to grow, in the case of the conventional plants, mainly due to increasingly stringent environmental policies, and to decrease for the combined cycle units.

Also CCPPs offer low operation and maintenance costs, by quality equipment designed to allow easy access for components inspection. Furthermore, the modular arrangement of the units also facilitates generation dispatch because the GT can operate independently, if components of the plant are down for maintenance or if only a part of unit total capacity is necessary.

Despite the fact that some private investors consider that in terms of efficiency the plant optimal size is of 50 MW or less, the CCPPs can be profitable operating in a wide range of sizes depending on local needs (i.e. industrial areas, factories, university campuses, commercial buildings, etc.).

CCPPs are able to adapt to fluctuations in the electricity demand; thereby they are suitable for peak and periodic load operations, i.e. frequently start-up/shutdown transients, because their dynamic response characteristics are superior to other power plants, such as fossil-fuel and nuclear power plants. The plant cost and flexibility are major factors in choosing the appropriate plant for a particular application. In general, the plants with low investment cost but with expensive fuels are more suitable for supplying peak electricity demands than those with high investment cost and fuels with low price, which are better suited for base load operation.

CCPPs provide a high availability and reliability due to a robust design that improves parts and components. The design is generally governed by the following aspects: the stresses in the crucial components, the material integrity and the control systems.

Combined cycle plants can be easily shipped, transported and connected in new or existing transmission system of electricity. These plants are commercially available in a worldwide, and in order to minimize installation time and costs their equipment is pre-engineered and factory-packaged.

Due to these outstanding technological advantages, the CCPPs have encountered a large diffusion over the last decades. Nowadays, the CCPPs have a total installed capacity of more than 800 GW, with a share which exceeds 20% compared to less than 5 % fifteen years ago. In the coming years, along with the introduction of new standards relating to environmental protection, the spread is expected to increase even more dramatically.

2.2.3 Configuration types of a combined cycle

CCPPs present a wide range of design options and configurations relative to the number of units, the type of the steam cycle, the connection type between the thermodynamic cycles, the electrical circuits, etc.

Generally, the types of plants are categorized according to the number of its component units. For example, a combined cycle of 1-1-1 type has a GT, a HRSG and a ST.

Moreover, combined cycle power generation includes a range of steam cycle options, where each cycle can have a certain configuration satisfying a number of technical-economic considerations. The choice of the steam cycle topology depends directly on the GT used, in particular of its gas exhaust characteristics (flow, temperature). The amount of potential recoverable energy in the flue gas determines the feasibility of the different steam cycle configurations. Thus there are installations with:

- 1 level of pressure (1P);
- 2 levels of pressure (2P);
- 2 levels of pressure with reheat (2P RH);
- 3 levels of pressure (3P);
- 3 levels of pressure with reheat (3P RH).

The large power stations are often configured 2-2-1, with a water-steam cycle with three levels of pressure and reheat (3P RH) [41].

Also depending on the introducing mode of the fuel and the type of coupling between the thermodynamic cycles, there are several categories of CCPP [5]:

- ◇ Cycles in series: the primary energy is introduced only in gas cycle, the steam cycle is used as a recovery unit. The thermal energy from the fuel is used in both stages of the thermodynamic cascade. This type of combined cycle gives the best thermal efficiency.
- ◇ Cycles in parallel: the primary energy is introduced simultaneously in both cycles (gas and steam). In thermodynamically terms, there is no real coupling between the two cycles, the connection is strictly technological.

◇ Cycles series-parallel: a combination of previous cycles, a part of the primary energy goes through the thermodynamic cascade, the other side is introduced directly in the steam cycle.

Another criterion for classification considers the possible mix of the working fluids (air/gas and water/steam) [5]. Hereby, it can be divided into two: with mix (e.g. STAG, HAT) and without mix (e.g. Steam Injection Gas Turbine (STIG), Combined Cycle Fully Fired (CCFF)).

These elements are technical selection criteria, but also the economic evaluation such as the investment costs, the use in base-load, part-load operations, the fuel cost and quality, the operating and maintenance cost, etc., can influence the topology of such a plant.

2.2.4 CCPP: Components, operation and control

In this section, an overview of the combined cycle components, with a brief description of their operation and their local control strategies, is presented, starting with a simple structure (1 level of pressure) and continuing with one more complex with high performances (3 levels of pressure).

The basic configuration of a combined cycle power plant is illustrated in Figure 2.1. The structure is of type 1-1-1 and the main components are:

- the Gas Turbine (GT);
- the Heat Recovery Steam Generator (HRSG);
- the Steam Turbine (ST);
- the Steam Line (SL): the pipes system between HRSG and ST;
- the Condenser;
- the Generator (G);

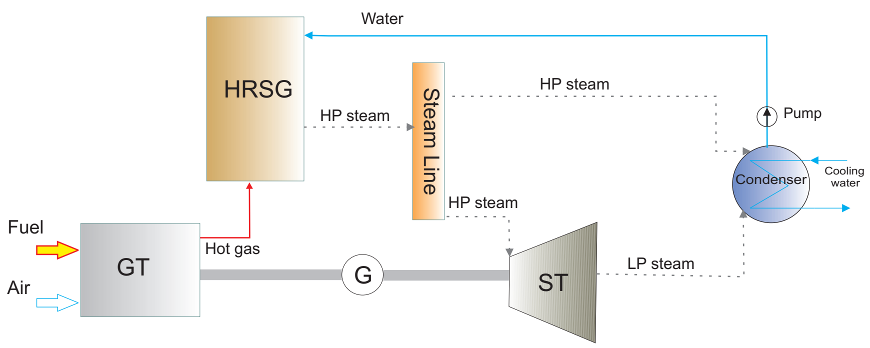


Figure 2.1: Schematic of a combined cycle type 1-1-1

2.2.4.1 CCPP: Single pressure level

The simplest type of combined cycle is the so called single pressure cycle (see Figure 2.1). It represents actually the basis for all possible configurations of combined cycle. In this cycle the HRSG generates steam for ST at only one pressure level (i.e. high pressure - HP). This configuration involves a low installed cost and has been extensively used. Despite the fact it does not provide the highest efficiency, it can be an economical solution when the fuel is of low quality or inexpensive, or when used during peak demand periods.

The normal operation of such a circuit is presented as follows: Ambient air is compressed, then mixed with fuel and burned at constant pressure. The high-temperature, high-pressure gas resulted by combustion is expanded in the GT to generate mechanical power. The GT exhaust hot gases are used to produce steam in the HRSG. The steam produced is expanded in the ST to provide additional power. The exhaust steam from the ST is collected by a condenser. The steam cycle is completed by feeding the HRSG with water from the condenser.

2.2.4.2 Gas Turbine

2.2.4.2.1 Description

The thermodynamic cycle which governs the behavior of GT systems, is a Brayton cycle. This cycle is one of the most efficient for the conversion of gas fuels to mechanical power [20], [18], [42].

The GT (combustion turbine) can be considered as the most important equipment of the combined cycle since it provides a large part of the electrical power (GT produce about 60%, while ST about 40% at base load) and supplies the thermal energy needed by the steam cycle. The three main elements which constitutes the GT are a compressor coupled to a turbine and a combustion chamber in between (Figure 2.2). Briefly their operation

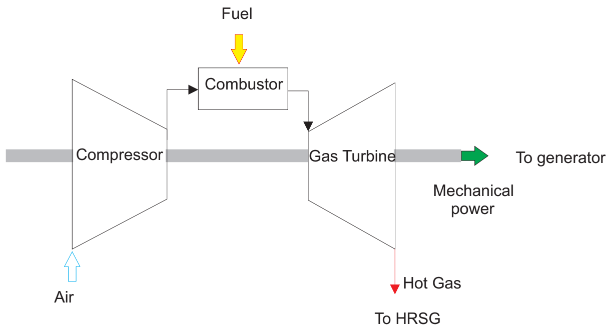


Figure 2.2: Gas turbine structure

is the following. Air at atmospheric conditions enters the compressor and is compressed to a required pressure for combustion. As a result of the adiabatic compression, the air undergoes an increase in temperature and pressure. In the combustion chamber (combustor), the air is mixed with fuel and burned under constant pressure conditions to convert the fuel's chemical energy into thermal energy. The resulting high temperature and high pressure gases are expanded in the turbine, generating mechanical power to drive the compressor and the electrical generator.

2.2.4.2.2 Local control strategy

The functional view of the GT is given in Figure 2.3. The control input variables are the fuel and air flows that are admitted inside the turbine. The main system actuators are: the IGV (Inlet Guide Vane) that regulates the air flow and the fuel admission system that regulates the fuel flow. The interaction variables are the hot gas that is exhausted (temperature and flow) and the generated mechanical power.

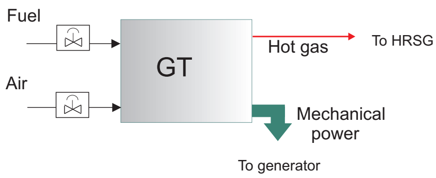


Figure 2.3: Gas turbine

GT is delivered with its control that manages its various operating modes. The control system is designed to start the unit, accelerate it to operating speed, synchronize it with the grid, and increase the load to the predefined value. Concerning the start-up, the GT has three operating modes depending on the load of the machine, each involving a different type of control.

The first mode is the *launch mode*, that is an automatic sequence specified in Figure 2.4. During the first step *purge*, an auxiliary engine is used to drive the compressor in order to purge the chimney. When this is completed the GT ignition is switched on (beginning of step *speed-up*) and the rotation speed is increased, by the auxiliary engine, to reach a given value (considered here as 75% of its nominal value). At the beginning of the next step (*GT speed*) the auxiliary engine is stopped and the speed control is activated in order to lead the GT speed to its nominal value. The speed is controlled by actuating on the fuel flow. The associated generator is then synchronized with the electrical network. Following synchronization, the speed reference becomes a load reference, which is set to a predetermined value (typically 8%). The transition from the *purge* state to the *speed-up* state may also be constrained by external conditions as it corresponds to the ignition switched on and the time when the exhaust gas temperature begins to increase.

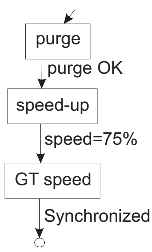


Figure 2.4: Launch mode of the Gas turbine

The second mode is the *temperature matching mode*. In this mode, the exhausted gas temperature is controlled according to a temperature reference signal whereas the load is maintained to a minimal value (typically 8%). Usually, the goal of the regulation is to maintain the GT outlet exhausts temperature constant (equal to the nominal value, in order to optimize the efficiency of the HRSG and of the whole plant).

The last mode is the *loading mode*. In this mode, the load is controlled according to input reference signals. It is used to lead the GT to its nominal load.

The control of the air flow rate is performed in an open-loop way. A typical behavior of the IGV is that when the power is less than a minimal value imposed by the manufacturer, it is partially closed. The control system opens them gradually when the power is between the minimal value and 50-60% of nominal power. After the power exceeds this value, it fully opens. In conjunction with IGV's action, the fuel admission valve adjusts the power in a closed-loop regulation.

In addition to the air and fuel control, a thermo-regulation with the aim to limit the fuel flow rate when the (estimated) combustion chamber temperature exceeds the safety limits, is provided.

2.2.4.3 Heat Recovery Steam Generator

2.2.4.3.1 Description

The HRSG is the part of the combined cycle which makes the connection between the cycle gas and water/steam cycle [43]. A steam generator consists of a flue gas in which are placed different heat exchanger sections. The heat exchangers use the hot gases from the GT to produce steam and to increase the steam temperature beyond the saturation point so that it can be expanded in the ST. HRSG may include one or more water/steam cycles at different pressure levels, typically containing: an economizer, an evaporator associated with a drum boiler and a superheater (see Figure 2.5).

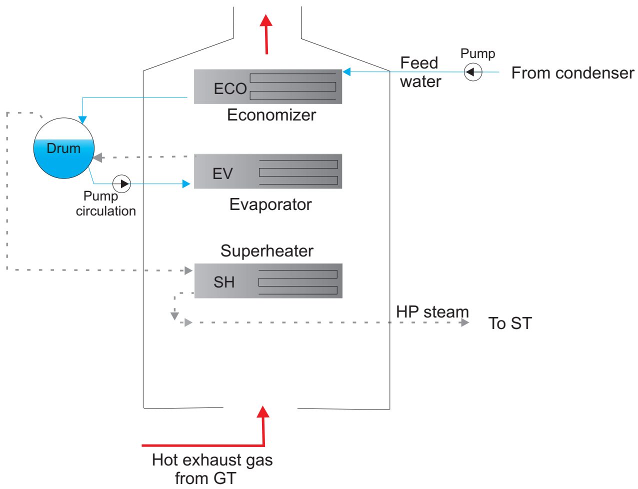


Figure 2.5: Heat Recovery Steam Generator with one pressure level

Economizer. The economizer (ECO) is used to preheat the feed water introduced in the drum in order to replace the steam produced and delivered. It is normally located in the colder gas, downstream the evaporator.

Evaporator. The evaporator (EV) is designed to transform the heated liquid (water) from the drum to a saturated steam. The flow in the evaporator enters as a liquid and exits as saturated steam. The EV modules are connected to the drum via downcorner pipes, riser pipes and distribution manifolds.

Superheater. The superheater (SH) in the HRSG is used to transfer the saturated steam from the steam drum to a superheated steam for injection in the ST. The ”dry” process in the SH is performed at a constant pressure. In some units this steam may only be heated slightly above the saturation point while in other units it may be superheated to a significant temperature for additional energy storage. The SH is normally located in the hotter gas stream close to the inlet of the exhaust gases from the GT.

Drum boiler. The steam drum is the connection point among economizer, evaporator and superheater. The drum is designed to act as a storage tank as well as separator for the saturated steam and liquid phase of water mixture. It is fed with the preheated water from the economizer and the saturated steam from the evaporator.

Certainly, the improvement of the combined cycle performance is linked to the technological progress of the GT and ST, but also the development in steam generators design (the increase of the recovered energy from the GT exhaust, more complex heat exchangers positioning, the strength of materials, etc.) plays a fundamental role [3]. Therefore, in order to maximize the energy recovery from the GT exhaust, multiple-pressure steam cycles are used.

A schematic representation of a system with three circuits - low pressure (LP), intermediate pressure (IP) and high pressure (HP) - is given in Figure 2.6. Each circuit produces steam at a given pressure from water. The LP section is fed with water from the condenser while the IP, HP sections are fed with water that has been heated by the LP circuit. In the case of the systems with reheating, the IP circuit is also fed with the steam return from the HP exhaust ST. An informative range of the thermodynamic values for common configurations of HRSG is given in Table 2.1. Increasing the number

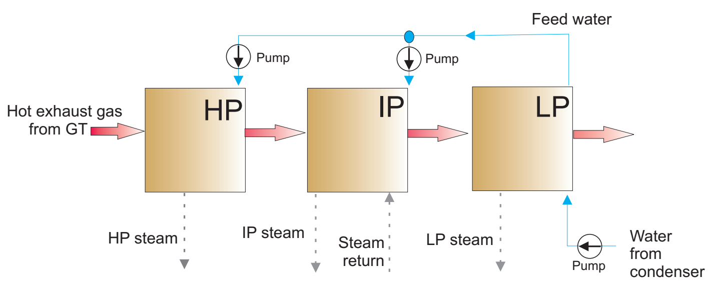


Figure 2.6: HSRG decomposition

of steam pressure levels (two or three-pressure steam cycles) involves the reduction of the difference between exhaust gas and steam/water energy. Mostly, the improvement of the plant performance results from additional heat transfer surface installed in the HRSG. Furthermore, in certain circumstances, the HRSG can be designed to incorporate supplementary firing sources to improve the boiler output [19], [43].

Independent of all these configurations, an HRSG design can be of natural circulation type (by thermosyphon principle) or forced circulation type. The type generally depends on the layout of the boiler, vertical or horizontal [44]. In a forced circulation HRSG, the steam/water mixture is circulated through evaporator tubes by using a pump. The most common type of HRSG in combined cycle plants uses forced circulation. A complete representation of a HRSG with three pressure levels with reheat and forced circulation is shown in Figure 2.7. It should be also noted that the number and the position of the

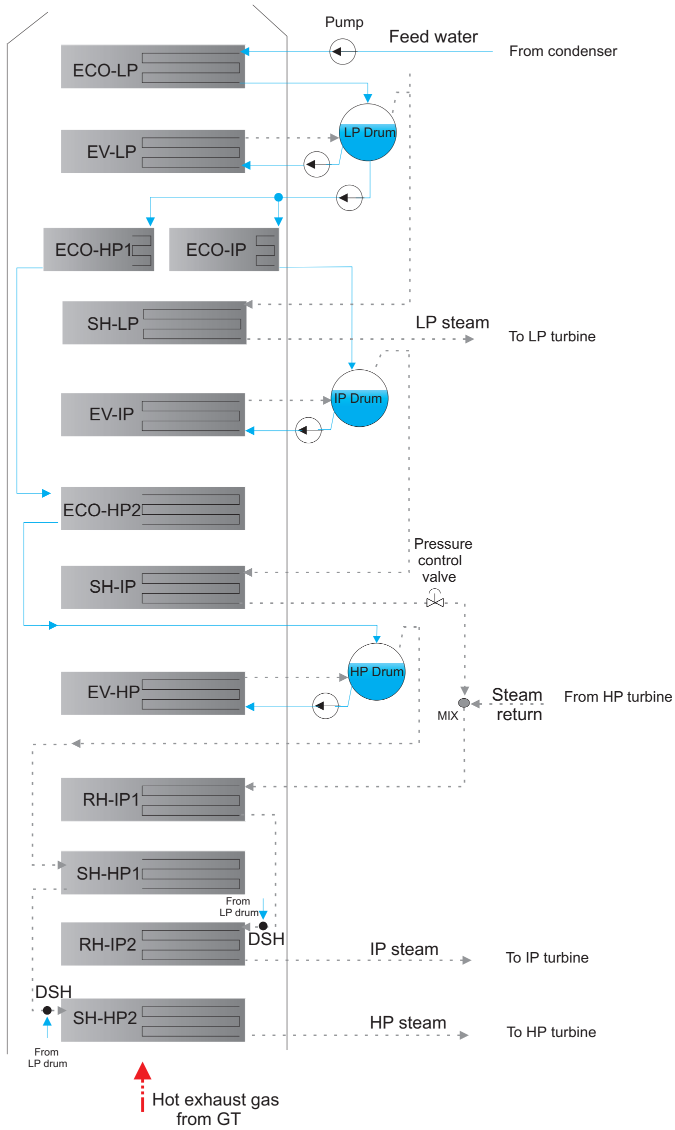


Figure 2.7: HRSG with 3 pressure levels and reheat

heat exchangers in the chimney are in general defined in order to optimize the HRSG efficiency.

HRSG Circuits

In the HP circuit, feed water flows through two economizer stages (ECO-HP1, ECO-HP2 in Figure 2.7), which exchange heat with the hot exhaust gases. The hot gases are flowing in the opposite direction round the circuit, gradually raising the water temperature (approx. 300°C). Then the water goes to the lower part of the HP drum, from where the extraction pumps control its forwarding to the evaporation circuit (EV-HP). Here the water begins to transform in steam and then sinks in the top part of the drum. The steam at this point is at saturation temperature. The HP steam leaves the drum via pipelines, which take the steam to the HP superheater stages (SH-HP1, SH-HP2). The steam temperature in the SH is increased from saturation temperature to superheated temperature (approx. 200°C of superheat). The HP steam exits the superheater and passes through an attemperator or desuperheater (DSH). Here HP feed water is injected into the steam path to control the steam temperature.

The IP and LP circuits operate similarly, with a few differences in terms of circuits configuration, for example for the LP circuit, as the temperature is low, it does not require a desuperheater, and for IP circuit, two additional stages (IP-RH1, IP-RH2) are used, in order to reheat the mix between the IP steam and steam return from the HP stage of the steam turbine.

Steam features		
HRSG	*Pressure*	*Temperature*
	bar	°C
1P	55	500
2P (HP/LP)	75/6	500/200
3P (HP/IP/LP)	110/30/5	550/550/200
3P + RH (HP/IP/LP)	110/30/5	560/550/200

Table 2.1: Thermodynamic values of common configurations of HRSG

2.2.4.3.2 Local control strategy

In the conventional control scheme of the HRSG with three levels of pressure, there are generally three main sets of local regulators: drum levels control, steam temperature control at the outlet of the IP, HP circuits and the water temperature control at the LP circuit inlet. These control strategies are briefly presented in the following. A more detailed reference about it can be found in [45], [46].

2.2.4.3.2.1 High Pressure Circuit

The functional view of the HP circuit is given in Figure 2.8. The manipulated variables are the water flow that feeds the circuit and the desuperheating flow. The controlled variables are the HP drum level and the output steam temperature (HP steam) that is directed through pipes to the ST. In this circuit, the circulation pump is used to assure a constant flow in the evaporator. In certain cases, the circulation in the evaporator is done without pump by natural thermosyphon phenomenon.

The drum level is controlled according to an input reference signal by actuating a feedwater valve/pump. The position of this valve can vary as a function of the various designs of the HRSG manufacturers (e.g. upstream or downstream of the economizer). The regulation of the HP steam temperature is performed according to a temperature reference signal by desuperheating the corresponding steam fluxes, i.e. by injecting water upstream the inlet section of the superheater (SH).

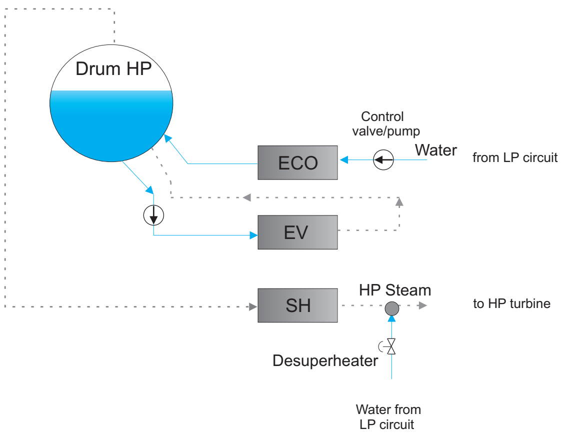


Figure 2.8: HP circuit

2.2.4.3.2.2 Intermediate Pressure Circuit

The control strategy is the same, the IP drum level is controlled according to an input level

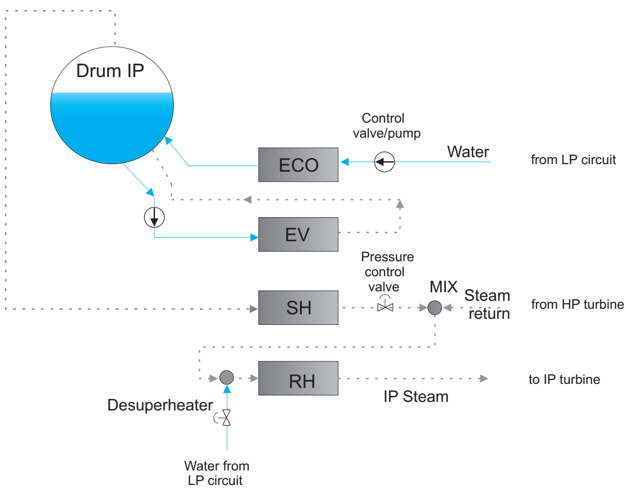


Figure 2.9: IP circuit

reference signal while the IP steam temperature is controlled to a temperature reference signal, by desuperheating, by injecting water upstream the inlet section of the reheater (RH). In addition, IP circuit contains a pressure control valve, in order to handle the pressure difference between the exhaust steam from HP turbine and the IP steam (Figure 2.9).

2.2.4.3.2.3 Low Pressure Circuit

Concerning the LP circuits, with the functional view illustrated in Figure 2.10, the manipulated and control variables are the feed water flow and the drum level, respectively. The LP drum level is controlled according to a reference signal, by actuating the feed-water valve. Additionally, in the LP circuit, a control system is implemented with the purpose to regulate the water temperature at the economizer inlet. The manipulated variable is the recirculation water flow and this control loop has an important role in the prevention of the external condensation on the economizer tubes. It should be noted that

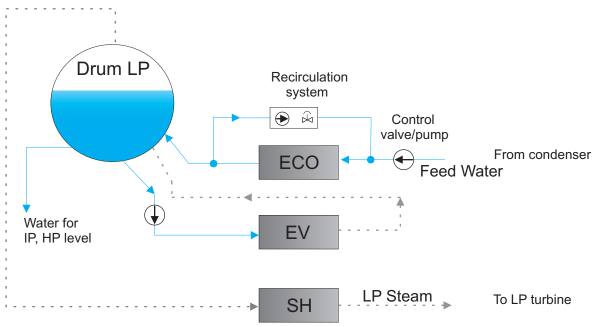


Figure 2.10: LP circuit

in addition to what has been presented these circuits can also contain other elements that can intervene in the system control (e.g. a blowdown system to control the security and maintenance requirements).

2.2.4.4 Steam Line

2.2.4.4.1 Description

The steam line (SL) is the pipes system that makes the connection between the HRSG, the ST and the condenser. Thus the steam from the HRSG passes through pipes and is directed to the ST (when this is started) or to the condenser by means of the bypasses [47]. As long as the ST is not started, the steam from the HRSG is deviated to the condenser or to the reheater in the case of HP steam. The SL is equipped with vent valves and bypass systems, depending on the manufacturer, these can have different dimensions or configurations (e.g. the pipes can reach a length of up to 200 m, the bypass system can be configured in parallel or in cascade) [48]. A schematic representation of the SL is shown in Figure 2.11.

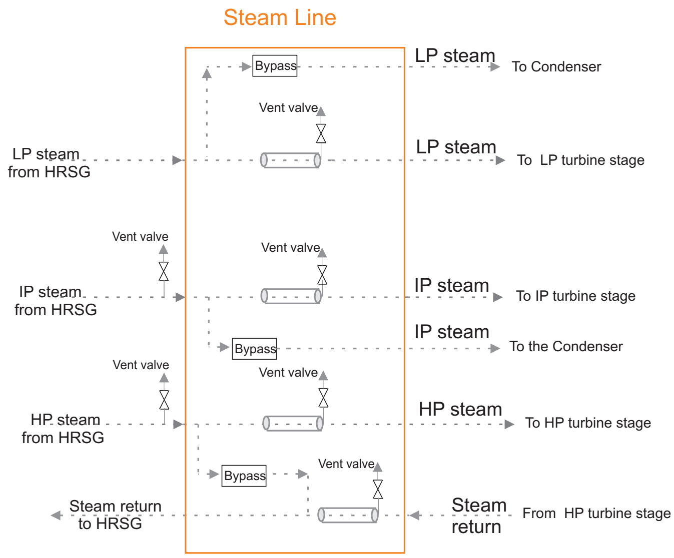


Figure 2.11: SL schematic representation

2.2.4.4.2 Local control strategy

In general, the SL's control role in a combined cycle plant is mainly related to the start-up phase. Its objective is to avoid the sudden warming of the pipes (to limit the thermal shocks and to avoid the condensation). Also the main steam entering the ST has to fulfill certain conditions such as temperature, pressure and quality, necessary for the safe operation of the ST. These conditions are reached by controlling the steam that is admitted inside the pipe with the evacuation and the bypass systems.

The bypass system is used also during the start-up phase to control the steam pressure. Thereby, a control system throttles the bypass valves, in order to maintain the pressure in the circuits at the set points and to ensure that it does not reach the safety limits.

2.2.4.5 Steam Turbine

2.2.4.5.1 Description

The ST operates on the Rankine cycle using the high-enthalpy (high pressure and temperature) steam provided by HRSG to generate mechanical power. In this thermodynamic cycle the working fluid follows a closed loop and is constantly reused, unlike the Brayton cycle that usually works as an open system. The steam turbine converts the thermal energy in the steam into mechanical energy. A turbine contains several sets of blades (buckets): a set of stationary blades (nozzles) which is connected to the casing and a set of rotating blades which is connected to the shaft. The stationary blades convert stored energy of high temperature and high pressure steam into kinetic energy (velocity) and direct the flow onto the rotating blades. The rotating blades transform the kinetic energy in impulse and reaction forces caused by pressure drop, which results in the rotation of the turbine rotor (rotor mounted on turbine shaft) [18].

Depending on the number of pressure levels chosen to optimize the water-steam cycle, the ST units may consist of multiple separate stages. Such a type of ST with three stages, according to the pressure ranges: High Pressure (HP), Intermediate Pressure (IP) and Low Pressure (LP) turbine, is shown in Figure 2.12.

The efficiency of the Rankine cycle can be improved by increasing the pressure and temperature of the steam that enters in the turbine. In a combined cycle plant, high pressure do not necessarily mean a high thermal efficiency. Expanding the steam at higher pressure causes an increase in the moisture content at the exit of the ST, which intensifies the internal losses and causes erosion problems in the later stages of the turbine. These issues can be removed by reheating of the steam. So in the case of the ST with three stages the operation is the following. The HP steam from the HSRG is passed through the HP stage and the HP exhaust steam is returned to the boiler where it is reheated by the flue gases until the superheated temperature, but at a pressure highly reduced. The reheated steam is then passed through the IP stage. The IP steam exhaust is mixed with LP steam and finally expanded in the LP stage of the turbine.

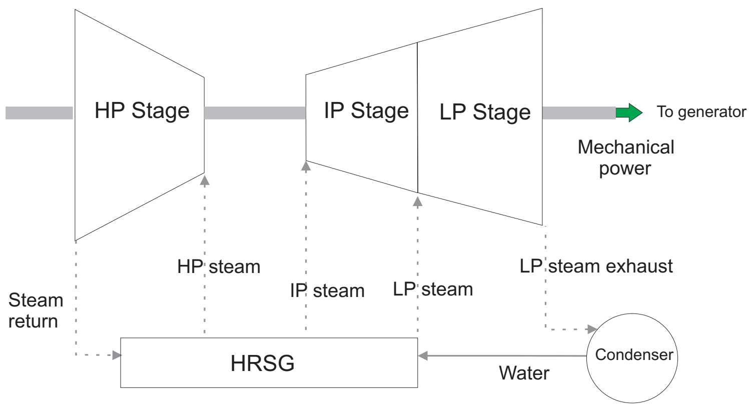


Figure 2.12: Schematic diagram of multi-stage ST

2.2.4.5.2 Local control strategy

The ST or rather the coupling ST/HRSG can be operated in two different modes: fixed steam inlet pressure control and sliding pressure. The combination of these operation modes normally depends on the level of load. During sliding pressure control, the admission control valves (throttle valve) are fully open. The steam pressure is a function of the steam mass flow entering the ST. The load (power output) of the ST is not directly controlled and depends on the steam mass flow from the HRSG and indirectly on the exhaust gas flow from the GT. Thereby, the ST load can only be increased by generating more steam in the HRSG which generally involves an increase in the GT exhaust gas flow and/or temperature. Generally ST operates in sliding pressure mode between approx. 60% and 100% ST load. Below 60%, the ST operates in fixed pressure mode. In this mode the ST control valves are throttled to maintain the HRSG pressure constant at a preset minimum value required by the HRSG (approx. 60% of the nominal steam pressure). In Figure 2.13 is presented the functional view of the ST with three stages.

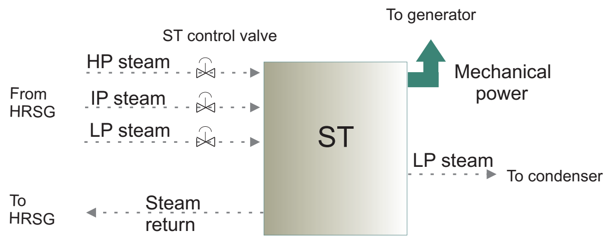


Figure 2.13: Steam Turbine

In the fixed inlet pressure control mode, the valves allow to control the steam admission (HP, IP, LP steam) inside the ST stages. During the start-up sequence, controlling the steam flow, by actuating on the throttle valves, the ST rises gradually in speed and in temperature. The ST accepts the total HRSG steam flow (sliding pressure mode), when the bypass is closed.

2.2.4.6 Condenser

2.2.4.6.1 Description

The condenser is the device that converts the steam exiting from the ST in water. Once the steam has passed through the ST, it is taken by the condenser, which removes the heat until it condenses back into liquid water. This is done by passing the wet steam around several cold water tubes. The condensed steam is collected at the bottom of the condenser and returned to the HRSG using extraction pumps, to continue the water-to-steam cycle. The condenser operates usually at very low vacuum pressures of about 0.1 bar.

2.2.4.6.2 Local control strategy

In the case of the condenser the local regulators aim: the level, the inlet steam temperature and the vacuum pressure. The functional view of the condenser is given in Figure 2.14. The manipulated variables are the water flow, the desuperheating flow and the vacuum flow. The condenser level is controlled according to a reference signal, controlling the water

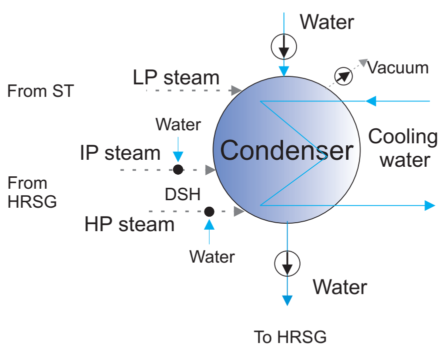


Figure 2.14: Condenser

inflow, i.e. by actuating on the water pump. The regulation of the steam temperature is performed according to a set point signal by desuperheating the corresponding steam fluxes, i.e. by injecting water upstream the inlet section of the condenser. Normally, this regulation is active at start-up when the bypasses are used. The vacuum pressure is controlled by actuating the vacuum pump, according to a reference signal.

2.2.4.7 GT and ST electrical generators

The generator is used to convert the rotating mechanical energy of the GT or ST into electrical energy. The turbine and generator rotors are connected through a coupling. The rotation of the turbine blades causes the rotation of the generator rotor shaft. So, the generator converts the rotating mechanical energy transmitted to it into electrical energy.

In a large part of the CCPPs, the GT and the ST have separate generators, using a multi-shaft configuration. In Europe, a popular option is to use a single shaft arrangement. In this design the GT and ST are on the same shaft using only one generator (see for example Figure 2.1). This configuration represents the economic option, offering a lower overall cost.

2.2.4.8 CCPP: Multi-pressure levels

Summarizing what has been presented so far, the multiple-pressure steam cycles are for the most part similar to the single pressure cycle, with the addition of the LP and IP sections, in the HRSG. In this case, the difference in operation is the following. The GT exhaust gas is used to produce steam at different levels of pressure. This steam is expanded sequentially in the ST stages to obtain additional power. In addition concerning the systems which provide a sufficient high thermal energy to reheat the steam, the HP exhaust steam from the ST is mixed with IP steam and reheated in the HRSG to capture additional energy from the exhaust gases, before the expansion in the IP turbine stage. The steam from the IP turbine exhaust is mixed with LP steam from the HRSG LP circuit and then expanded in the LP steam turbine up to the condenser pressure. The steam cycle is completed by feeding the three circuits in the HRSG with the water from the condenser to the LP drum and then from the LP drum to the IP and HP drums.

Multi-pressure steam generation achieves a better efficiency than a single pressure system but with a higher investment costs. In general, the plants with multi-pressure circuits are the economic option if the fuel price is high or if the duty cycle requires a high plant load factor. This type of system represents the most efficient power plant currently available. A simplified representation of such a CCPP, without any auxiliary system and with the main sensors and actuators, is presented in Figure 2.15.

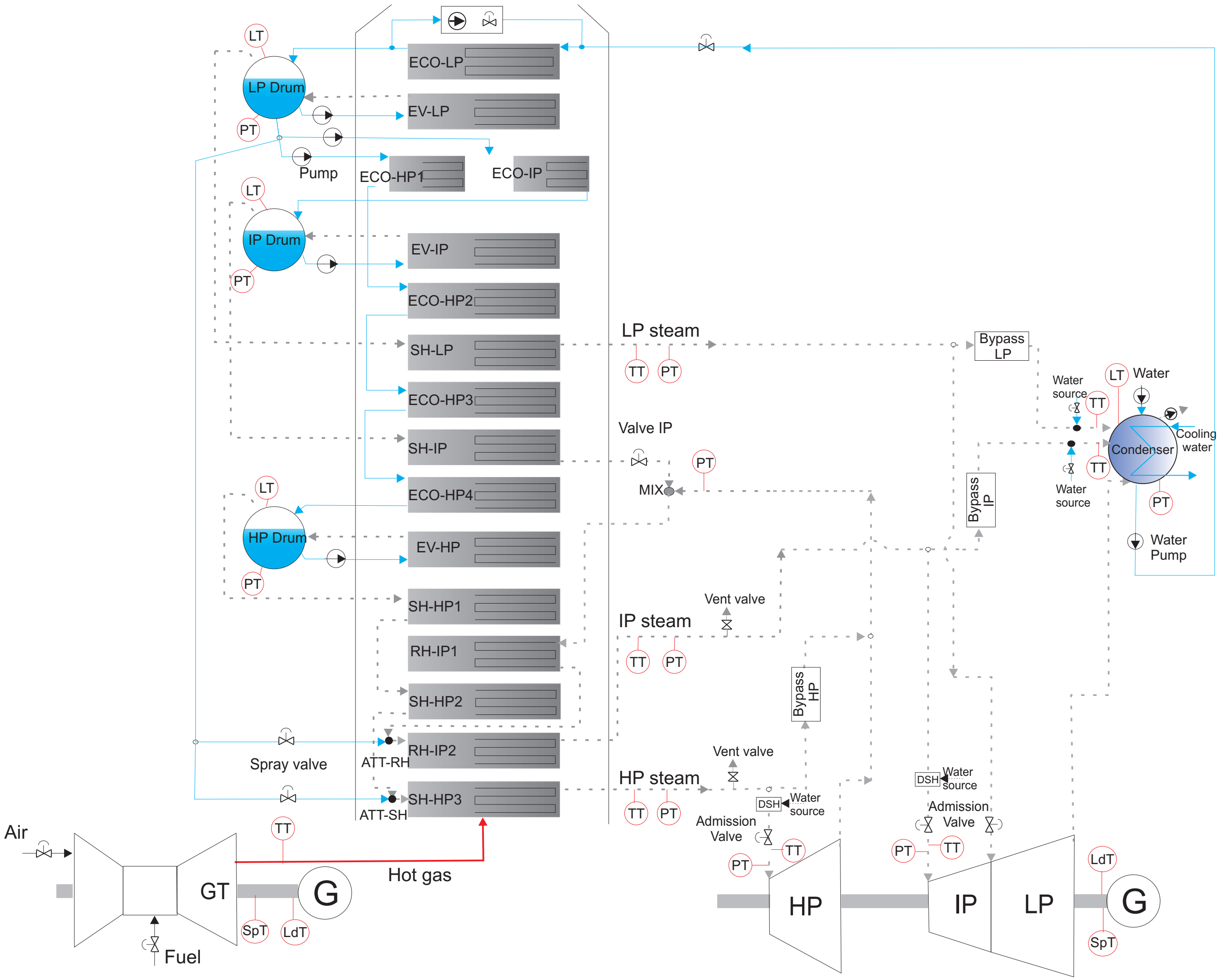


Figure 2.15: CCPP with 3P RH

2.2.5 CCPP start-up process

In the current context, CCPPs have become an important technology for power generation. The combined cycle units have gradually replaced the existing conventional technologies, as a result of their advantages. Initially they were used to meet the base load demands, but subsequently, together with the saturation of the electricity market and the growing in the gas prices, these plants have been used in the intermediate-load range, i.e. plants are started and stopped frequently to meet the daytime peaks. Typically, the average number of power plant start-ups are over 150 per year. The challenge now is to improve the plant flexibility without compromising the component's life or efficiency.

In France, these type of plants are intended to supplement the base load nuclear plants, in particular during the peaks consumption. In figure 2.16 can be seen the evolution of the electricity, price/volume, during a day.

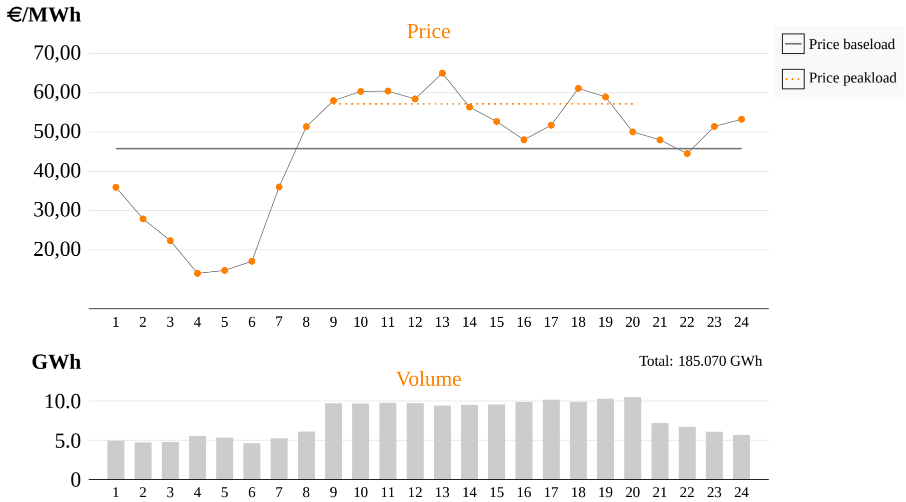


Figure 2.16: Electricity evolution during a day in France (source www.epexspot.com)

Nowadays, as a result of the competition among the energy providers and the daily fluctuations in power demand, the CCPPs undergo frequent shutdowns and start-ups. Indeed, depending on their profitability, the combined-cycle units are shutdown for short periods of time or more extended periods, thus they are subject to a large number of transients during operation.

The frequent start-ups/shutdowns reduce the lifetime of the crucial components and increase the scheduled maintenance stops and forced outage rates. During these transients, the parameter differences that occur can produce thermal and mechanical stresses. The repeated application of these stresses results in fatigue failures. The fatigue damage may cause cracks or fracture after a certain number of cycles. Fatigue fractures are caused by the simultaneous action of cyclic stress, peak values of stress (in tension or in compression), and plastic deformation [49]. The crack is initiated by the cyclic stress and strain, while the tensile stress produces crack propagation. Normally, the compressive stress does not cause fatigue cracks propagation, but the compression load can do it [50].

In addition to these factors, the fatigue behavior is influenced by several other variables, such as corrosion, material structure, component size and shape, residual stresses, etc., which can affect the fatigue of the machine.

The lifetime of the units is usually assessed by the number of cycles N until the final rupture [51]. Currently, there are several standards/codes used to analyze and evaluate the fatigue life of components [49], [52], [53], [54]. In these standards, the maximum temperature ramp rate and the overall range of temperature change supported by a component during the transients are key indicators of creep-fatigue damage. For example, for two start-up types, according to European standard EN 12952-3, the allowable number of cycles to crack initiation for different temperature ramp rates is shown in Figure 2.17.

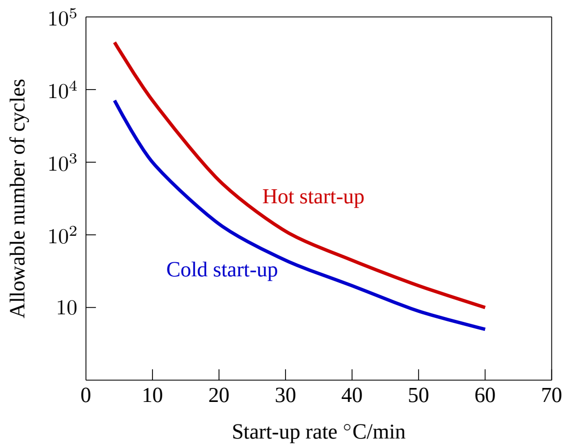


Figure 2.17: Allowable numbers of load cycles for alloy steel materials to crack initiation depending on the temperature ramp rates for two start-up types, according to EN 12952-3 methodology

The start-up type are typically identified by the plant operators as cold, warm and hot, depending on the time period that the unit has not been operational (shutdown or standstill period). Also, another way to define the start-ups is by referring to the metal temperature and/or internal pressure of the components. Thus a hot start is typically defined to have the standstill period less than 8 to 12 hours and high boiler/turbine temperatures (e.g. temperature of the turbine shaft over 370°C, HP drum pressure at least 30 bar and metal temperature over 250°C). A warm start corresponds to a standstill period of 12 to 48-60 hours and has the boiler/turbine temperatures of 140-370°C, and the HP drum pressure of 3-5 bar. In the case of a cold start, the metal temperature and the HRSG internal pressure are at atmospheric conditions. This type of start-up has shutdown periods over 60 hours. These periods and parameters values can vary from a model of unit to another, depending on the unit size, manufacturer, dispatcher etc.

The standstill period is significant for the HRSG and ST operation, influencing the initial value of the component parameters and the start-up times. As the ST and HRSG contain parts that have to be gradually warmed in order to limit the thermal stresses, a longer standstill time involves subsequently a longer start-up time. Unlike the ST, the GT unit is independent of this period, it can reach in less than 30 minutes up to two thirds of its nominal power.

The cycling operations, in particular the start-up, can be quite damaging for the power generation equipment. The main concern in CCPP start-ups is to avoid the phenomena (e.g. thermal stress, mechanical stress) that may shorten the component's life and produce unsafe situations. All this consideration results in the start-up time increasing, while economical requirements are to reduce this time and to minimize the operation costs. Currently, in order to prevent any possible unsafe conditions and to preserve the components life, a number of physical constraints has been introduced. In the following section, the main constraints related to the start-up phase as well as the physical limitations of the systems, are mainly outlined.

2.2.5.1 GT: Physical limitations and constraints

Generally, the GT is not a limiting factor for the CCPP start-up time, since the GT start-up is much faster than the HRSG/ST system start-up time. Also during transients, the GT has a dynamic response (seconds) faster than the HRSG/ST assemblage (minutes). The GT's control system performs the start-up and ensures a progressive increase of the turbine rotor temperature. The system disposes also of protection circuits [18], to ensure safety operation of all the GT's subsystems and prevent abnormal situations, which could lead to the plant shutdown:

- overspeed: when the speed exceeds the security limit (typically 110%);
- overtemperature: when the average exhaust temperature exceeds the set limit point;
- overloading: when the load of the turbine is too high with respect to its nominal value;
- loss of flame: when fuel/air ratio exceeds the flame extinction limit.

In addition, during start-up phase, a number of constraints are imposed on the GT's parameters. These constraints are rather a consequence of the fact that the HRSG/ST system is not able to support a sudden change in steam characteristics (temperature, pressure) and quantity. For a large GT load increase (e.g. more than 5%), in order to guarantee the safety and the availability of the operations, it is necessary to limit the GT load change. In general the load increase is limited to a rate of about 2% -5%/min. Thus, depending on the operating mode, the constraints are specified on the gradients of the gas temperature or the load. The constraint on temperature gradient is mainly considered during the *temperature matching mode*, whereas the one on the load gradient is considered during the *loading mode*.

2.2.5.2 HRSG: Physical limitations and constraints

The physical limitations and constraints related to the HRSG circuits are mainly linked to the component life consumption, in particular to the stress of the material. As the pressure and temperature are higher, these constraints are more critical for the HP and IP circuits.

2.2.5.2.1 High Pressure and Intermediate Pressure Circuits

A first type of constraints is related to stress on certain components. The components in the HRSG which have been identified as critical, in terms of fatigue damage, are the outlet headers of the reheater and HP superheater, the parts of the exchanger located in the hotter gas stream, and to a lesser extent, the thick boiler drums walls. The two main types of stress on the components surface, which can occur, are: the mechanical stress resulting from internal pressure and the thermal stress resulting from thermal expansion. Thus the constraints are mainly specified on the pressure and temperature gradients.

Considering the fact that condensation is an important phenomenon that occurs during the start-up in SL downstream and that can be damaging for the components with thick metallic parts, such as the superheater headers, constraints on the quantity of condensation are imposed. This phenomenon occurs when the steam of the drum can condense in the superheater section, therefore it is necessary to limit the condensate formation in the colder part of the SH and to prevent slugs of condensate to be pushed into the steam pipes by an efficient control of the evacuation system (vent and drain valves).

Also, another type of constraints is specified on the minimum flow rate in pumps, in the case of the plants with forced circulation. This constraint is imposed to avoid the cavitation in pumps [42]. This phenomenon degrades the performance of the pump, resulting in fluctuating flow rate, discharge pressure and in some cases even to the pump components destruction. In general this constraint is imposed on all circuits equipped with a pump.

2.2.5.2.2 Low Pressure Circuit

The LP circuit works at low temperature, so in terms of thermal stress, it is not critical. In this case, in order to keep the mechanical stress in the allowable limits, the constraints are specified on the gradient of pressure in the LP drum.

In addition, at this level a constraint, that aims the temperature of the flue gas, is specified. It is important to maintain the gas temperature above the acid gas dew point, because the gas condensation can occur, causing serious corrosion problems for the metallic parts in contact with these gases.

The values of the gradients used to prevent the fatigue failures, are typically determined in accordance with international standards and codes (see [49], [53], [54]). As mentioned before, the most critical components in the HRSG circuit are the drums and high temperature superheater/reheater headers. For each of them a number of precise limits are established, thus for large steam drums with a wall thickness greater than 100 mm, operating at 100 bar and over, the acceptable temperature gradient is generally of 36°C/min, while for smaller drums operating at 60 bar, the temperature gradient can be up to 8°C/min. For the superheaters depending on the wall thickness, the acceptable temperature gradient can reach up to 17°C/min (e.g. for a SH with a wall thickness greater than 25 mm, the acceptable temperature gradient is of 17°C/min) [46].

The main protection limits for the HRSG (for all circuits HP, IP, LP) are:

- overtemperature: when the steam temperature exceeds the safety limit;

- high drum level: when the level is too high, the drum can give wet steam, leading to turbine blade erosion;
- low drum level (loss of water): when the boiler has a loss of feed water, this can boil dry, leading to a dangerous situation. For example, if in the empty boiler feed water is injected, the water instantaneously boils on contact with the superheated metal and leads to explosion;
- low water pressure in desuperheater: when the water pressure injected is too low;
- feed water flow deviation: when the flow exceeds the maximum capability of the feed water system;
- overpressure: when the pressure in tubes increases over the safety limit.

2.2.5.3 SL: Physical limitations and constraints

Concerning the SL, the main concern is also to limit the components lifetime consumption. Thus, a first type of constraints refers to the gradients of increasing pressure and temperature in pipes.

The second type of constraints aims the condensation. The condensate formation and migration through pipes must be avoided, otherwise, as mentioned above, harmful effects on the plant components can occur.

In some cases, the purge flow rate is too low and a part of condensate can not be blown off. Hence, to ensure that the condensation effect is eliminated, constraints are specified on the purge flow rate.

For safe operation, the system is equipped with protection circuits to avoid the following situations:

- overpressure: when the pressure in pipes exceeds the security limit;
- high condensate level: when the condensate level is too high (flooding).

2.2.5.4 ST: Physical limitations and constraints

It is generally known that during the start-up operations the ST's life is the most affected. As previously pointed out, the main factors for the lifetime consumption and thus the main limitations for the reduction of the start-up time are derived from the temperature gradients which occur in the components with thickest metallic parts. For the ST, these parts are the HP/IP rotors and casings [55]. The most critical constraint is localized in the ST rotor [56], more exactly the part of turbine shaft located close to the admission, where the temperature variations are faster and more intense (temperature increases quickly from about 90°C to 570°C during start-ups). The rotor represents the component with the highest cost and requires a long outage to be replaced. The large temperature variations to which the rotor is subject produce two potential side effects: the development of thermal stresses and the differential expansion between stationary and rotating parts. Both phenomena can cause fatigue damage and reduce the life-time of the turbine if they are not properly handled. Namely, the high thermal stresses reduce the life-time of the rotor, by favoring the cracks apparition, cracks propagation, and even fractures. Also, a large differential expansion can give rise to rubbing, which in turn may lead to permanent deformation (creep) and turbine performance degradation. In general, the creep phenomenon depends much on the stress temporal peak while the fatigue crack is caused by the repeated application of stress.

In addition to the thermal stress, the start-up rotor stress, which is directly related to the ST's life, contains the mechanical stress produced by centrifugal force caused by the rotation speed, and in certain cases the electrically induced stress which occurs due to a faulty synchronization or a short circuit.

Acceleration of the ST start up leads in general to stress increase (mechanical, thermal) in their components [55]. In most cases the limitation of the stress is performed by imposing several constraints on: the gradient of increasing steam temperature at admission, the gradient of speed/load, the difference between metal and steam temperatures as well as on the temperature difference between the static and rotating parts.

Many units experienced corrosion and erosion problems. The imposed constraints to avoid these problems are on the moisture content of steam and the level of water purification. The ST must be started with dry steam, if the moisture content of the steam is over a predefined limit (typically 10% moisture content), major erosion and corrosion problems can occur. In general the steam is superheated and no moisture is present at the inlet. However, moisture can appear at the outlet, particularly for the LP stage. Concerning the water purification, any admission into ST is not possible while the steam does not have the required purity (below 0.03%).

In some cases the constraints are specified on the steam exhaust from LP stage. The pressure of the steam which goes out of the turbine has to be limited, in order to reduce the mechanical stress.

Concerning the abnormal situations and safety operation, which could lead to a procedural execution failure (turbine trip), the system provides protection circuits during several events:

- overspeed: when the rotating speed of the turbine is too high; that can occur when the torque generated by the steam flow exceeds the countertorque generated by the load;
- high HP exhaust steam temperature: when the HP turbine exhaust steam temperature (steam return) is too high;
- water ingress to blade: when the moisture content of steam is over the safety limit (a limit is set at about 90% steam quality);
- low HP turbine pressure: when the pressure is too low;
- high bearing temperature: when the temperature inside the turbine is too high;
- high turbine exhaust pressure (low condenser vacuum): when the turbine exhaust pressure is high, the last stage of the blades in the low-pressure (LP) turbine will become overheated and damaged;

- etc.

In the last decades the effects of the thermal transients on the component life have been continuously studied and a number of precise limits on the stresses have been imposed in order to ensure a safety plant operation [57], [58], [55]. These limits specify the allowable stress levels in the critical components and represent the main operating constraints which are considered in the start-up optimization procedure presented in the following sections.

2.2.5.5 Condenser: Physical limitations and constraints

The condenser usually does not have particular constraints, the safe operation is ensured by protection circuits which avoid situations such as:

- high condenser pressure: when the pressure in condenser exceeds the security limits;
- high condenser temperature: when the temperature exceeds the safety limits;
- water level: when level of liquid water in condenser exceeds the allowable limits (level too high/low).

2.2.5.6 Start-up sequence

CCPP start-up is a complex task including several phases and conditions that must be respected simultaneously. The start-up procedure depends on each technology employed, but in general it can be decomposed into four main stages[1] (Figure 2.18) that correspond to different objectives and active constraints [42]:

1. *Preparation phase*: this first stage prepares the ignition of the GT;
2. *HRSG Start-up phase*: this phase corresponds to the thermal conditioning, and its aim is to start the HRSG in order to obtain the characteristics of the steam (temperature, pressure, quality) suitable to the admission in the ST;
3. *ST Start-up phase*: during this stage the ST is started by admitting steam;
4. *Increasing Load phase*: during the last stage the load of the plant is increased to its expected level.

The CCPP start-up sequence that has been considered in this work begins when the GT is started and finishes when the plant achieves a preset level of power output[2], close to the nominal value. In fact, several conditions are required before the GT start, such as the cooling system, extraction and feedwater pumps circuits started, etc. These actions are supposed to be done. The overall sequence description can be found in [46].

[1] In some cases, the process of start-up can be modified to cope with the different environmental requirements dictated by local regulations

[2] This value can vary from one manufacturer to another

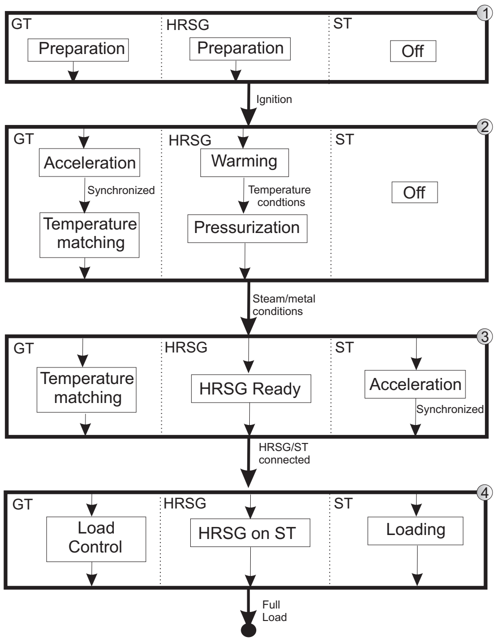


Figure 2.18: Start-up process decomposition

2.2.5.6.1 Preparation phase

During this stage, the GT is started by an external device, and the plant is purged to ensure a non-flammable atmosphere in the turbine and HRSG, prior to ignition. Purging is usually carried out by speeding-up and holding the gas turbine at ignition speed (approx. 25-30% of nominal speed) with an engine or a similar starting equipment, until a specified number of air changes occurs in the GT/HRSG system.

During the purge, the exhaust from the GT is close to ambient temperature and by entering into a hot HRSG will produce a cool off in the entire boiler. In some cases (i.e. warm/hot start-up), the condensate can occur in superheater section. Once it formed, the condensate must be removed to prevent slugs of condensate to be pushed into the system pipes (condensate migration). During this phase, the drums level control is activated to mitigate the swell effect that can appear when the GT is fired and kept active until the end of the start-up sequence. A first constraint that the control system has to respect is on the feed water flow (minimum flow rate). When the purge is completed and the level in drums reaches the reference value, this stage is finished by firing the GT.

2.2.5.6.2 HRSG Start-up phase

The second stage begins with the GT ignition. The GT is then accelerated and synchronized to the grid. After synchronization the GT switches to *temperature matching mode* and the exhaust gas temperature is controlled to gradually warm up the HRSG metal (*Warming mode*). When the metal temperature of the HRSG (mainly of the superheater) is higher than the saturated steam temperature (temperature conditions), the HRSG goes to the *pressurization* phase.

In order to avoid condensate formation and obtain characteristics of the steam (temperature, pressure, quality) necessary to ST start-up while warming the SL, the pressure control (via bypasses and vent system) and steam temperature regulation (output HP, IP circuits, via desuperheating) are started. With the *HRSG start-up phase*, the constraints related to this system become active (e.g. the gradients of increasing temperature in SHs, the gradient of increasing pressure in drums, the condensate level in the SL, etc.).

When the steam conditions (pressure, temperature, quality) for ST start-up are fulfilled, the steam is directed to ST and system goes to the third phase. For the entire time period until the fulfillment of the steam conditions, the generated steam is discharged through the bypass into either the condenser and reheat steam line or only into the condenser, depending on the configuration. In general the steam conditions depend on the initial temperature inside the IP stage of the ST.

2.2.5.6.3 ST Start-up phase

During this stage, the GT is kept on *temperature matching mode* and the HRSG is ready to start the ST. The control of the parameters already activated (i.e. gas temperature, steam pressure, steam temperature and drum level) is maintained. By actuating on the throttle valve opening, the steam is gradually admitted in the ST, that is started, then accelerated and synchronized to the electrical power grid. The ST starts with steam admission in the IP stage and when the HP steam reaches the required temperature limit, it is admitted in the HP stage. After synchronization the ST begins to load (*fixed steam inlet pressure control mode*), tracking a pressure set point imposed by the HRSG.

In conjunction with *ST Start-up phase*, the system control has to respect also the constraints concerning the ST operation (e.g. the gradient of speed, the gradient of increasing temperature, the gradient of increasing load, the steam quality, etc.).

The system goes to the next phase when the connection between the HRSG and the ST is complete (the bypasses are closed and the ST synchronized to the grid).

2.2.5.6.4 Increasing Load phase

In this phase, the bypass systems are closed and the entire amount of steam produced in HRSG is taken by ST (*HRSG on ST mode*). The GT load is controlled in order to achieve the full load (*Loading mode*). The ST follows the GT, increasing steam production, and switches from the fixed pressure mode to the sliding pressure mode. The start-up procedure is finished when the power plant work at full power. In the case when the CCPP has two GTs, the second GT starts up when the ST loads.

During start-up, in addition to the active constraints for each phase, the system control has to guarantee the safety of the operations.

2.2.5.7 Objectives of start-up optimization

The optimization of the start-up can aim several purposes. The ideal solution should allow the plant operator to reach the following objectives, while respecting the constraints.

2.2.5.7.1 Minimization of start-up time

A shorter start-up time represents the most important goal because it may be followed by other benefits presented below as targets. It can improve the operational efficiency, reducing the fuel consumption and consequently the CO_2 emissions. Also, minimizing the start-up time is a crucial competitive advantage in a liberalized market since it is desirable to propose the smallest starting time among all energy providers

2.2.5.7.2 Minimization of the operating costs

Start-up costs include costs for fuel or other consumables such as water and chemicals. A reduction of these consumables means a direct increase of the customer value.

2.2.5.7.3 Minimization of material wear

This minimization has generally two main benefits:

- to reduce the plant life-time consumption;
- to reduce the number of maintenance stops.

2.2.5.7.4 Minimization of environmental impacts

The reduction of pollutant emissions has a major importance in view of the constantly increasing restrictions on the environmental impact by law. The high temperature in the combustion zone (exceeding 1300°C) results in the nitrogen in the air combining with oxygen, also additional nitrogen could be present in the fuel; consequently the combustion temperature and the NO_x concentrations have to be limited.

2.2.5.7.5 Minimization of start-up failure

This represents also an important point since a missed start means time lost and the system impossibility to respond to the dispatching request. This goal can not be easily associated with a criterion as previously case. To minimize the risk of start-up failure, an analysis must be done to determine the main causes (model error, sensor error, etc.) and to choose a criterion that increases the robustness of the control with respect to these causes.

Generally these objectives should be considered all together and not separately; it is obvious that by considering the plant life as the first priority (e.g. minimization of the material wear) will result in a relatively long start-up time. On the other hand, for economic reasons, the start-up procedure should be achieved in the shortest time with a minimum operating cost. For that, a compromise between a fast start-up and the other objectives has to be made, in such a way that the imposed constraints are fulfilled.

2.2.5.8 Approach to improve CCPP start-up

As previously presented, in order to guarantee the safety and the availability of the operations, the start-up process requires to meet a series of constraints and conditions, which involves a longer transient time. Also, the fast start-up capability of the GT is influenced since it must be adapted to the conditions imposed by the HRSG and the ST. Currently there is a number of solutions to address these limitations which makes it possible to optimize the start-up procedure with several benefits. In the sequel, the main works carried out in this area are presented. The information relative to the optimization methods of the start-up are divided in two parts: a first part which summarizes the solutions proposed by the manufacturers and a second one which outlines the so-called "academical" solutions. In general the industrial solutions are focusing rather on the plant design by implementing new technologies to relax the constraints and to eliminate the waiting times during the transients, while the second type aims mainly at improving the start-up profiles, by developing new control strategies and approaches based on the plant model.

2.2.5.8.1 New technological designs

In the recent research, driven by changes in customer requirements, the start-up performance has been improved by implementing new technological designs. An interesting solution is proposed by Siemens, that has developed a fast start-up concept and implemented it in the design of the new power plant units [59], [60]. With this concept, the start-up time is significantly reduced and also the number of starts for which the plant was designed, is increased. To achieve this performance, Siemens has modified the traditional plant design, by introducing new features. One of the most important aspects is the capability of the HRSG, that admits an unconstrained GT start-up. This new steam generator [61] maintains the characteristics of the natural circulation principle of drum type boilers (i.e. flow stability and uniform temperature distribution), but at the same time the HP drum is replaced with thin-walled components, thereby improving the operational flexibility. In the design of the new plant, a key role is played by the dimensioning of the condenser, bypass and desuperheater systems. An optimal sizing of the bypass and condenser allows to take almost the entire quantity of steam produced by the boiler, while a high capacity of the desuperheaters allows to adjust simultaneously the steam temperature to ST requirements, even at GT base load. Thus, by means of the desuperheater, the GT can be "decoupled" from the ST start-up. Also, the proposed design includes other features such as: condensate polishing system to eliminate the waiting time for steam purity, mechanical vacuum pumps to evacuate the condensate during the purge phase and keep it evacuated during the plant standstill, stress monitoring systems, etc.

A further improvement in the start-up time reduction is provided by changing the start-up conventional procedure, in so-called "*hot-start on the fly*". The new concept is based on the parallel start-up of the GT and ST, by monitoring and controlling the temperature gradients for the critical components. The parallel procedure enables plant start-ups without any GT load waiting for the ST start-up. The main innovation is represented by the early starting point of the ST. In the first phase of the procedure, after the synchronization, the GT is loaded continuously with its maximal allowable rate to base load. The exhaust gas is led through the HRSG, and with the first quantity of steam produced, the ST is accelerated and loaded. With this parallel concept, Siemens indicated a start-up time reduction of up to 60% compared to a classical procedure. The announced time for a hot start-up procedure in the case of a single shaft combined cycle plant, is of approximately 40 minutes. Additionally, the times for the other start-up types were considerable reduced (the times obtained for a warm and a cold start-up are of about 120 min and 150 min respectively). Siemens proposes also an optimization module for use with the Distributed Control System (DCS) [62], having two operating modes for the start-up:

- fast start, whose objective is to start the plant in minimal time, which has also indirect benefits by reducing the energy losses;

- low loss start, with the objective to minimize the energy losses during the start-up.

These two methods use a prediction model that estimates the stress margins in the critical components and then provides the optimal trajectories for temperature, pressure and GT load.

Alstom achieves a fatigue-tolerant design [63] to allow an operational flexibility. Their methods do not directly address the optimization of the start-up trajectories. The trajectory conception process follows a series of precise steps based on an accurate modeling of the plant components. The equipment suppliers carry out detailed studies during the design phase of the subcomponents (i.e. valves, pumps, tubes, etc.). Finite Element Method (FEM) techniques are used for this purpose. Each subsystem is modeled with high precision to analyze all the constraints. This analysis allows to determine the operating conditions which guarantee a certain service life. Using the constraints on the subcomponents, a number of start-up rules (tables, diagrams) with the allowable gradients for the main components (i.e. turbines, boilers) are provided. Also detailed parameter histories are obtained from a dynamic simulation of the process. The gradients derived by these approaches are directly used by Alstom to improve the start-up performances. According to Alstom by respecting these operating specifications, the optimization is already performed, the plant starts in minimal time and the life of its units is assured. The confirmed hot start-up time by Alstom with this procedure is about 45 min.

Furthermore, currently there is a wide range of CCPP simulators for the operator training. An example is the simulator from Invensys [64], that although it does not explicitly deal with the start-up optimization, has been used to improve the control procedure during the transients, by considering different test scenarios (i.e. by parallelizing certain operations). The main benefit of the control modification has been the reduction of the start-up time.

The technological efforts result in considerable improvement of the start-up performances, particularly in a start-up times. The new procedures lead to reduced start-up times (e.g. Siemens about 40 min, Alstom about 45 min). These times refer usually to a particular type of combined cycle plant and/or start-up procedure. Also the start-up classification can have different meanings for each manufacturer. For example, concerning the initial conditions, in the Alstom's interpretation, a hot start-up means a ST metal temperature superior to 350°C, or regarding the final point, in the Siemens's conception a "started plant" meaning that the bypasses are closed and the admission valves are completely opened. Therefore, the results generalization and their use as a comparison point between the methods is inappropriate.

2.2.5.8.2 New approaches to improve start-up efficiency

Most of the approaches that deal with the optimization of the CCPP start-up are based on the process model, in particular the optimization principles are applied on the dynamic models. Despite the fact that these models present computational challenges, the dynamic optimization represents one of the most promising solutions. The start-up process is generally treated as a dynamic optimization problem, consisting in finding an optimal profile for each control variable (e.g. gas turbine load, steam turbine throttle valve position) that minimizes an objective function (e.g. time and/or fuel consumption) and fulfills a number of constraints imposed by the plant dynamics and the main process variables. In the following lines, the proposed solutions up to now are analyzed in terms of the three ingredients of an optimization problem: the model, the objective function (cost function, performance index) and the constraints.

CCPP models

In the literature the range of proposed models for the CCPPs is very wide. The models used for the process components are mainly of three types: manufacturer tables/functions-based models, identification-based models and physical-knowledge-based models. The most common are these latter, based on physical equations, which are represented in several ways and with different degrees of complexity.

The works of [65], [66], [67], [68], [69], [70] have shown that the use of simplified models can be a possible option to represent the dynamic behavior of combined cycles for different purposes. Despite the fact that not all can be used to study the CCPP start-up, a few articles are worth mentioning.

Some of the most frequently used gas turbine models in dynamic studies of the combined cycles are those of Rowen [65]. These simplified representations involve the estimation and the identification of certain number of parameters and functions (e.g. the gas temperature). An improved version is presented in [67], where unlike the Rowen's work, the GT model is entirely determined by using the thermodynamic equations of the turbine and therefore no function/parameter identification is required. Further in this article, the complete representation of a combined cycle system is provided, by connecting the GT unit to a steam cycle model (HRSG and ST). In this article (see also [66]), the boiler model conserves the essential non-linear characteristics of the process and its equations are derived from the first principles of mass, energy and volume balance. The modeling of the ST is performed as functions of the steam flow rate. In most of these cases, the immediate applicability of the models in the start-up study is not suitable, mainly because a number of important factors that occur during the transients are not modeled (e.g. no temperature effects, stress, etc.). Therefore, in order to analyze and improve the start-up, these models must be adapted to this sequence.

An appropriate model to the start-up transients, that takes into account the major phenomena during this phase (the evolution of the ST metal temperature, the condensation, the drum level evolution, etc.), is described in [69]. This representation is used by the same authors to derive models adapted to specific stages of the start-up sequence (stage 2 and 4 according to the decomposition from section 2.2.5.6). Based on these models, an interesting solution to minimize the CCPP transient duration phase by phase is proposed in [70]. One of the main drawbacks of this approach is that although the initial model enables to calculate the evolution of the ST metal temperature, the stress evaluation on the ST components is not considered. It must be noted that generally in these works, for the system modeling, the block diagram formalism is used (e.g. by using Matlab/Simulink).

More complete approaches are based on the implementation of the process equations in dedicated languages for physical modeling such as Modelica [71] or other computing language and environments [72], [73], [68], [74]. In these works the dynamic behavior of the CCPP is represented by mathematical models with a level of detail comparable to the models listed above, but their use with more powerful tools/packages for optimization and simulation of dynamic systems enables the study and improvement of the operational parameters profiles during the start-up procedure. To add that in [71] as in [70], the model used to calculate the optimal trajectory is adapted to a stage of the start-up sequence, in particular to the last one (see section 2.2.5.6). The mathematical representation of the plant in these studies includes models of stress, thermal and/or mechanical. Depending on the proposed objective, the models compute the stress only in the ST components, such as [71], [74] or in all critical parts of the steam circuit (ST and HRSG), such as [73], [68].

The models developed in these papers include several simplifications and assumptions such as consideration of constant heat transfer coefficients, neglecting the pressure drops, the friction effects, the heat loss, etc. Moreover, the equations are implemented in a symbolic manner, with no possibility to consider the properties of water and steam. The implementation of these details is required in order to obtain more accurate models. In certain cases the model parameters/functions have to be identified with process data. Although there exist several methods concerning their determination [66], [75], fitting these models with real processes is not always an easy task.

In general, the dynamic simulation is a critical procedure when the systems operate close to the limits of the process. This requires complex models that are as close as possible to real system. Two such models, designed and parametrized with operating data from a real plant are presented in [76] and [10]. These models are adapted to the start-up phase, i.e. they can represent the whole start-up sequence and consider the main phenomena which occur during this phase. In [76] attention is essentially focused on the modeling of an existing HRSG by means of the Advanced Process Simulation software (APROS). The HRSG components are modeled with a high level of detail, while for GT and ST simplified representations are used. As a drawback, the evaluation of the stress is performed only in the HRSG components, therefore it does not take into account the ST stress. The model is validated in two steps: first by performing the verification and evaluation with design data for steady state; then by comparing the dynamic simulation results with measured data from the process.

The article [10] deals with the dynamic modeling of a CCPP by using the Dymola environment [8]. The model is based on the open Modelica library for the dynamic modeling of thermal power plants, *ThermoPower* [9]. In this study, the model validation is performed by replicating a real start-up sequence, as recorded by the plant DCS. A complete representation of the stress in the critical components is provided. In order to keep the model complexity at a reasonable level, some phenomena and components which are not critical for the start-up phase are neglected. Furthermore, it should be noted that in this paper, as in the previous one, the gas turbine model is represented in a simplified way, because as already pointed out, its dynamic is much faster than that of the steam cycle.

In addition to the models proposed in these works, a number of subsystem models can be also mentioned. They are used for the study and optimization of the start-up for each circuit separately, such as for the HRSG [77], [58].

From this general bibliographic study, it appears that the model developed in Dymola/Modelica by Casella and Pretolani [10] is one of the most complete and appropriate representations for the CCPP transients study. For the purposes of the present work, this model and the *ThermoPower* library have represented the starting point.

Operational constraints

The articles, which deal with the CCPP start-up optimization, take into account a series of operational constraints in order to prevent any possible unsafe situation and to preserve the plant components lifetime. These constraints are relevant for the real start-up process, and actually by taking them into consideration the mathematical optimization problem is transformed in a real engineering problem. Thus, in the papers presented above, the following restrictions have been imposed on:

- the maximum allowable stress value of the ST and/or HRSG components, in [73], [68], [74], [71], [76], [10];
- the temperature gradients of the HRSG components (drum, evaporator, superheater), in [72], [74], [70]. These constraints are also related to the stress issues, because according to [72] (and its references), the choice of the maximum values for the temperature gradients is almost equivalent to limiting the stress components. Furthermore, the lack of a model to evaluate the stress can be compensated by imposing these conditions on the temperature rates ([72], [70]). In certain cases, additional restrictions on the steam flow or pressure rate are imposed [70];
- the loading rate of the GT, which must not exceed a maximum rate provided by the manufacturer, in [71];
- the maximum quantity of the NO_x emission, in [74], [68].

It can be seen that the dominant constraints are those related to the material stress, which represents the main limiting factor in the start-up problem. Also it should be pointed out that the list of constraints specified in the previous sections is not entirely covered. Thus, the constraints for example on the steam quality, the gradient of speed/load, etc. are not addressed. Taking into account all these constraints requires on the one hand complex models and on the other hand numerical solutions in terms of accuracy and efficiency.

Cost functions

The optimization problem described in these articles aims in most of the cases at minimizing the start-up time while keeping the operating constraints within their allowable limits. In other words, the objective is to reach the full load or a value close to this point, in minimum time. Nevertheless, two types of problems can be distinguished: the minimal time problems, where the cost function is given only by the final time (the parameter that must be minimized) [73], [70], and optimal control problems, where only the control inputs are optimized (the final time is not an optimization parameter). The objective in this case is formulated by using for example quadratic cost functions [72], further in [71] weighting factors are used to penalize the sum of the quadratic deviations from the target values. With these cost functions, 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 with the maximum allowable loading rate, a minimum start-up time is assumed to be provided. Regarding the optimization problems, they can aim only a stage of the starting procedure [70], [71], or certain operations such as a load variation from 50 to 75%, which although does not correspond to any start-up stage, the issue remains the same as in the case of the transient optimization [72].

As can be seen, the trend is in general to minimize the start-up time in single-objective approaches, managing all other important aspects of the problem (stress, pollutant emission) as constraints. From a certain point of view, this kind of approach does not necessarily ensure the optimization of the overall start-up operations, because an efficient optimization would be in fact multi-objective. A solution is proposed in [74], where the objective function combines the start-up time with fuel consumption, in a way that the two objectives are solved jointly. Also the problem to find the best trade-off between conflicting objectives (time/material stress, production/emissions) can be addressed by means of multi-objective approaches based on the Pareto theory [78], [79]. In these studies, the optimization problem needs to fulfill the following objectives: minimize the start-up time, minimize the fuel consumption, minimize the pollutant emissions, minimize the thermal stress, maximize the energy production.

Optimization

From the analysis of these works, two trends emerge concerning the solving of the start-up optimization problem.

The first trend refers to the conventional methods that use dynamic simulations of the process to generate optimal control profiles. In these approaches, the input/output behavior of the plant is analyzed, for a wide variety of control profiles. These are modified in an iterative algorithm to fulfill the operational constraints and to obtain solutions increasingly better. In this way, [76] and [10] optimize the start-up procedure by using trial-and-error methods. Usually the solutions are sub-optimal and depend on the engineer's intuition and experience. Despite this, the obtained results are promising, thus in [76] the steam turbine reaches the maximal power in 76 min instead 148 min (a start-up time reduction of about 50%), while in [10] the start-up time is reduced from 420 min to 208 min (an improvement of more than 50%, with a trade-off between stress and time).

Also, as a result of the complexity, these models are in general treated as black boxes and methods based on heuristic techniques such as genetic algorithms [78], fuzzy logic [79], [80] and neural network [81], are employed. In general these methods require high computational resources and offer low performance in terms of convergence and execution times. In certain studies, a compromise between the accuracy of the simulator and the computational effort is performed, by using these techniques in different iterative optimization algorithms working on simplified models and validating on more complex models [68]. The solutions found demonstrate a considerable improvement in the start-up performances. In particular in [68] a reduction by 42% and by 29% of the energy loss is indicated, while the solution found in [78], [79] cuts down the start-up time from 351 min to 240 min (32%), consumption (32%) and emissions (11%), keeping the thermal stress relatively low.

In the second trend, the processes of start-up is addressed in the most general way as a dynamic optimization problem. The optimization problem is solved by using a series of dedicated numerical methods that apply non-linear programming (NLP) solvers to the dynamic model. These methods can be separated into two classes: the sequential and simultaneous strategies [82].

In the sequential methods, the control variables are parametrized to form a finite dimensional NLP, and then a standard solver is used. In this category, the solution proposed in [74], [70] where the trajectories control (e.g. GT load, steam pressure) are represented over a finite time horizon as piecewise polynomials, can be included. The coefficients of these polynomials are considered as optimization parameters and then derived using a sequential quadratic programming (SQP) solver. The approach proposed in [74] reduces the start-up time by 6 min (21.7%) compared to a conventional method.

The simultaneous strategies rely on a suitable parameterization of both the state and control variables. By parameterizing the variable profiles, the dynamic optimization problem, as in the case of the sequential methods, is translated into a NLP problem. The resulting NLP is usually very large but with a sparse structure that can efficiently be exploited by several SQP solvers (e.g. IPOPT, SNOPT). Two of the most popular simultaneous methods, direct collocation and direct multiple shooting, see for example an overview [83], are used for the start-up optimization problem. The collocation method is applied in [84], [72], [71], while in [73] the multiple shooting method is employed. These state of the art algorithms are implemented in different environments (e.g. in [73] the optimization package MUSCOD-II, in [71] the platform JModelica.org, in [84] the software OCOMA) and interfaced with the modeling languages in which the combined cycle plants are represented. The solution proposed in [73] is analyzed by means of several simulation experiments, the best compromise between the stress constraints and time reduction providing a start-up with a duration of 162 min (about 52%). In [71] the optimization procedure has been applied to the last part of the start-up sequence, showing that the full load can be reached after 70 min.

It must be noted that in the listed works, although methods for large scale dynamic optimization problems have been developed, the models used with these methods have a relatively low degree of complexity, compared to a real simulator or even to the model presented in [10]. Nevertheless the resulting optimization problems are very large (several thousand variables and constraints), and involve implementation and computational challenges. A solution in this sense applied to the start-up sequence, in so far of our knowledge, does not exist until now, the conventional methods remaining the only ones that treat the start-up optimization problem by means of real complex models.

The emergence of new optimization algorithms as well as the development of the reliable models of the plant dynamics with accurate output prediction, have also led to the use of advanced control strategies. One of the most popular method in the process industry, model predictive control (MPC) [85] has been successfully implemented to improve the start-up efficiency [86], [87], [88]. In the latter, the proposed methodology approximates the non-linear behavior of a combined cycle plant, using a different linear model, derived from a local linearization, at every control step. The resulted large scale quadratic programming (QP) problem has been solved by using an efficient algorithm which exploits the structure data. The MPC solution has been implemented in a real power plant, to the last part of the start-up procedure, with a significant reduction in the start-up time (by more than 40%, about 52 min, the time being counted from the moment when the admission valve is completely open), in the fuel consumption (by 28530 kg) and in the NO_x emissions (by 53 kg), with respect to the previously existing start-up controller.

A general conclusion of these studies is that most proposed solutions are based on simplified models, and not on complex physical-based models validated against experimental data. Only a few articles compare the model behavior with real data to validate it [76], [10]. Moreover, the implementation of new techniques, to optimize the combined cycle start-up, on real power plant is limited [88].

2.3 Conclusions

In this chapter a general overview of the combined cycle power plants has been presented. Attention is mainly focused on the operation of the plant during the start-up phase. Specifically, the control specifications for the start-up of a CCPP have been defined, the plant has been partitioned in subsystems and for each subsystem the operational constraints and the existing control loops have been specified. Also the start-up sequence presentation together with the envisaged optimization objectives of the CCPP transients have been indicated.

In the last part of the chapter, a state of the art with the current design technologies and methods applied for the start-up optimization has been cited.

The content of this part enables to know the operation and the complexity of the system, providing a global view of the start-up problem in order to propose an optimal approach for the start-up sequence. Thus, the information presented in this chapter will be used in the next chapters to build a model of the process and to develop model-based approaches for the optimization of the CCPP start-up.

CHAPTER 3

Physical models for CCPP start-up

3.1 Modeling and Simulation

Driven by increasingly competitive markets, the complexity of products as well as industrial systems is in a continuous growing. Nowadays, the reduction of production time and operating costs, while increasing the products range and their quality, have become essential for companies, not only in terms of profit growth, but also to survive on the market. Innovative methods and strategies to develop new products and/or to improve the existing operation processes are of major interest. In particular, the problems related to the system complexity managing have to be addressed. In this regard, modeling and simulation represent a key enabling technology for dealing with complex systems.

Modeling represents an important part in the development of any engineering field activity. Mathematical models are widely used in many areas, as a standard design methodology. The building of the models and the possibility to perform experiments on them instead of on real systems corresponding to the models, namely to simulate their behavior, involves a series of advantages. Primarily, the modeling and simulation enable to assess the performances, safety and operation of the process before it is built. Accordingly, before proceeding to the implementation phase, the design can be tested, evaluated and/or optimized. Furthermore, in the case of the processes already designed and put into service, this technology is used to analyze and optimize the process operations. In certain cases, the simulation can replace experiments, which from the safety point of view or economical reasons, are not feasible to be realized. Also, in the simulation, all variables can be studied and manipulated, even those that in the real physical system are inaccessible or hard to be changed.

The modeling and simulation technologies are becoming indispensable in industry for engineering needs. Nowadays, these technologies represent key aspects to achieve high performances in manufacturing and to assess the operating risks of the units.

A mathematical model represents a compact description of a real system, where the relationships between the physical variables of the system are expressed in a mathematical form. Thus, by determining a mathematical model of a process, the modeler aims to obtain a quantitative characterization of its operation, as close as possible to reality. An effective modeling requires satisfying the following three requirements:

- universality (the model can be applied to all objects that belong to interest class);
- limited number of parameters;
- identification of parameters.

There are different kinds of models depending on the representation strategy, technique, purposes, etc. In turn this multitude of models can be characterized by different properties which reflect the behavior of the modeled systems (e.g., static or dynamic, continuous-time or discrete-time, linear or non-linear).

In principle, the determination of such models, for a given process, can be obtained either analytically (knowledge-based models) or experimentally (experimental-based models). Knowledge-based models (based on physical laws, chemical, etc.) use the engineers experience from different fields and allow a complete description of the system with a wide range of validity. The involved variables have a direct physical significance (temperature, pressure, flow rate, etc.), and in this sense most laws of nature can be considered as mathematical models. Newton's second law, for example, describes the relationship between the acceleration, mass and force. The models developed in this way are often quite complex and difficult to use for control purposes. Consequently, these models have a relatively low applicability in the automatic control field, being generally employed in practice for process simulation and design. In the second approach (models based on experimental identification), the building of the models is performed by observing and processing of functional variables (inputs, outputs) associated with the system that will be modeled. Most models obtained by processing of experimental data are quite simple and have a limited validity for a specified working point, a certain type of input and a particular process. These models have a reduced physical meaning, in other words, the model parameters do not have a direct physical significance. Such models are relatively easy to build and use, and moreover, in the case of the processes that are based on insufficient knowledge, this representation can be a solution.

Mostly, it is advantageous to combine these two methods, analytical and experimental, so that by analytical modeling a class of models whose parameters are estimated using experimental data is achieved. This type of approach is not always possible, being constrained by the availability of experimental data.

The models that this work is focused on, belong mainly to the first category, so called physical-knowledge-based models, where the mathematical representation is derived from first principle laws, such as conservation of mass, conservation of energy. But nevertheless, for certain components, simplified models based on experimental data have been used. Moreover, in parallel with the physical-based models elaboration, simpler models that represent the main dynamics of the system have been developed by identification and compared with the knowledge-based ones.

3.2 Purpose of Modeling

Modeling represents a creative activity with a broad scope. The attention in this thesis is focused on the modeling of complex industrial processes, in particular a combined cycle power plant.

The determination of a mathematical model requires to get over a number of stages, by starting from the knowledge of the real system. The goal is in general to obtain a model that represents as realistic as possible the system. The real processes are often very complex, and the models do not seek and should not seek to obtain exactly the same complexity. In fact, if a model is too realistic, it can be infeasible from the mathematical perspective. Also a simple model can give good results, but it is likely that the level of details will be unsatisfactory in comparison to the required level. Therefore, a mathematical model must be a compromise between precision and complexity. Furthermore, a model must be appropriate to the proposed purposes. Actually, this is one of the ideas suggested by this thesis, to use knowledge of experts in domain, encoded in models libraries with different levels of detail, to build models which are suitable to the intended purposes. Namely, the models developed and used during this dissertation are adapted to study and to optimize the CCPP start-up procedure. The models reflect closely the real behavior of the system and respect specific requirements concerning the start-up phase; on the other hand these models are suitable for control design and optimization.

3.3 Modelica language

Modeling of complex physical systems requires dedicated high level languages. In particular, such modeling languages must provide a complete mathematical formulation to define the system behavior and, at the same time, structuring concepts for the models description. In this context, a modeling formalism that covers these requirements, combining the object-oriented concept with a declarative mathematical description is Modelica.

The Modelica language implements object-orientation concepts such as the classes, components (class instances), and inheritance, which enable an easy adaptation of the behavior and properties of an existing model in different contexts. These concepts facilitate hierarchical structuring, reuse of modeling knowledge and evolution of large and complex models. In fact, Modelica language uses the structural benefits of object-orientation, emphasizing structured mathematical modeling.

Modelica uses the declarative equation-based modeling formalism (acausal modeling). In acausal modeling the equations do not specify explicitly which variables are inputs or outputs (a conversion), as in the case of assignment based formulations. In the case of equation-based modeling, the causality is unspecified, it is fixed in general when the equation systems are solved, by efficient tools which manipulate the equations to determine the order of execution and which variables are needed as outputs and which are inputs. Thus, this formalism enables the user to state declarative differential and algebraic relations, without the need to transform the model equations into an ODE (Ordinary Differential Equation) representation, a procedure required in the case of block diagram formalism (e.g., Simulink). This concept involves less effort on the part of modeler when the equation are formulated, and moreover, the modeling code becomes more concise and easy to change.

A physical system is usually composed by a number of subcomponents, which are assembled by means of acausal connections. In addition to declarative aspect, Modelica provides support for acausal connections, meaning that the connections are realized as equations. Thereby, acausal physical interfaces to connect the components can be performed, enabling to build models with a structure corresponding to the physical system.

Nowadays, model classes are often employed to create libraries of physical subsystems. The procedure to build a model based on the elements of these libraries - the manner in which the elements can be implemented as well as the possibility that they are acausally connected - is similar to the assembly phase of a real physical system from its components.

Currently, there are several modeling formalism used to represent the behavior of physical systems. In addition to the acausal modeling, Modelica supports other formalisms, including, for example, the block diagram modeling. A complete presentation of the different formalisms can be found in [89].

These formalisms have been applied in several modeling languages, including Omola [90], Dymola [91], ObjectMath [92], that later, by an international effort to design an uniform modeling language for complex heterogeneous physical systems led to the creation of Modelica [93]. In fact, the realization of an unified language, that enables the modeling of various physical domains into one, makes the difference compared to other modeling languages specialized in different fields, such as gPROMS [94] specialized in chemical processes, VHDL-AMS [95] specialized on electrical circuits.

As a general definition, Modelica is an object-oriented equation-based language for modeling of large, complex and heterogeneous physical systems [96]. This language is designed to be an engineering tool for modeling of realistic physical systems. The most important features of this language are [97]:

- Acausal modeling. Modelica supports declarative equation-based modeling instead of assignment statements. This enables the modeler to use directly the equations, without the need to specify a certain data flow direction or execution order. In Modelica, the solution direction of the equations adapts to the data flow context in which it is computed, providing thus a high flexibility and a better reuse of the model components. This is actually the key to the physical modeling capabilities and increased reusability of Modelica models;

- Object-oriented modeling. Modelica like any object-oriented language provides the general concept of class, that represents the fundamental structuring unit and the key to reuse of modeling knowledge. This technique enables the modeler to handle the complexity of large-scale system description, by using the structuring concept such as hierarchies, component-connections and inheritance;

- Multi-domain modeling. In Modelica component models corresponding to physical systems from different domains can be represented and connected, allowing the heterogeneous systems modeling;

- Modelica has a strong support for component-based models, providing constructs to create and connect components. This feature is suited to create modular models as well as interfaces for their connection, becoming thus an architectural description language for complex physical systems.

Modelica is an open language designed to support effective library development and model exchange, which is continuously updated and maintained by Modelica Association [7]. All these properties make Modelica a powerful language for modeling and efficient simulation of large and complex systems. Examples of application areas include robotics, automotive, power plants, involving large-scale models, having more than hundred thousand equations. The main aspects of the Modelica language are briefly reviewed in Appendix A. For a more detailed overview of the subject, see [97], [98], or Modelica tutorial [99].

The Modelica language provides a convenient modeling environment, but its code can not be directly integrated by numerical solvers. Dedicated algorithms to translate the models into an appropriate format for numerical integration are required. Currently, there is a number of symbolic algorithms able to efficiently perform this transformation. The transformation involves in general a series of steps such as structural analysis of the DAE, equation sorting, reduction of size and complexity, and if possible, symbolic solution of equations or code generation for efficient numeric solutions. These algorithms are used by the Modelica-based tools to generate Modelica code that can be interfaced with numerical algorithms for simulation or in certain cases for optimization. A brief presentation of the most popular tools is given in Appendix B.

The primary objective of these tools is most often the modeling and simulation of the complex physical systems. In the present context, together with the attempts to improve systems performance, a new usage of Modelica models emerges, namely the dynamic optimization. In industrial fields, particularly in the process industry, the optimization has become an important task, being used in many applications including design optimization for optimal processes, trajectories optimization, on-line optimal control strategies such as model predictive control (MPC), etc.

In terms of tool for static and dynamic optimization of Modelica models, the support is in general weak, although several efforts have been made to integrate advanced modeling languages with algorithms for dynamic optimization. Dymola (Dynamic Modeling Laboratory) [8], for example, includes add-ons for parameter identification and design optimization. Also, an environment that provides support for dynamic optimization problems is gPROMS [94], but this platform is not based on the Modelica language.

Most of the Modelica models are either not immediately suitable for optimization purposes or limited to a particular optimization algorithm. Also, it must be stated that Modelica was developed for the purpose of an efficient modeling with formalisms well-suited to describe complex heterogeneous physical systems, lacking language constructs which enable to express optimization problems. For example the concept of cost functions, constraints, optimization interval, are not included in language.

A promising open source project, that aims to create a flexible and extensible Modelica environment focused on optimization, is JModelica.org [100]. Currently, the platform is under development, with several constraints on the models formulation, complexity and syntax, limitations that make impossible at the moment the optimization of complex Modelica models, in particular CCPP models. Nevertheless, the potential of this tool has been proved for simplified Modelica models [71].

3.4 CPPP modeling

3.4.1 Model libraries. Generalities

The dedicated languages for physical modeling facilitate the design and the development of process models. Effectively, the modeling languages enable a detailed representation at the component level and a model complexity management at the system level. In general the complex systems require for modeling a quite significant amount of information, which often is difficult to manage. In order to create a more complete modeling framework, at the component level, the expert knowledge from several application domains has been encapsulated in generic and reusable model libraries. The behavior of these models are generally represented by a high level of detail. Thus, by using these collections of reusable model parts and subsystems, corresponding to physical models, on one hand, it facilitates the rapid development of models. The modeler does not have to rebuild already modeled systems, but only to adapt the existing components to meet its specific needs. On the other hand, high accurate models can be developed with a considerable reduced effort. If particular models are required, at the equation level, the modeler can encode physical relations in order to represent the desired behavior, being able to implement their own components library.

Nowadays, the most commonly used system components are available in Modelica libraries (e.g., for electrical, fluid, mechanical components) [7], with the possibility for users to construct model instances corresponding to a particular process or plant. Moreover, in the last years several libraries for power plant systems have been developed. For example, a Modelica library used for modeling of thermo-hydraulic systems is *ThermoFluid* [101], or a library based on the first principle models for modeling of thermal power plants is *ThermoPower* [102]. Also a Modelica library which provides a wide range of model components for modeling of thermodynamic systems of industrial processes is *ThermoSysPro* [103]. Two common particularities of these libraries are worth to be pointed out. The first feature aims at maximizing the readability of the library by using a "flat" structure of the model hierarchy, and the second one is that the models can reach a flexible level of detail advisable for different purposes. In this thesis the library used to developed a combined cycle model is *ThermoPower*.

3.4.2 *ThermoPower* library

The *ThermoPower* is an open-source library, developed at the Politecnico di Milano, for dynamical modeling of thermal power plants at the system level, to support the design and validation of control systems [9]. The library provides the basic components for the modeling of power plants (e.g., pipes, valves, heat exchangers, drum boilers, turbines), having a more restricted purpose than other Modelica libraries for generic thermo-fluid systems [104]. The "specialized" scope of the library enables the adoption of some simplifying assumptions on the nature of the fluids and their phenomena, for example, two fluids are considered (water/steam and ideal gas mixture), to compute pressure drop a turbulent flow is assumed (valid choice only in the case of power plant components involving water and gases). Also, the library is focused on the key components, providing a high level of detail and accuracy. For example, in the pipe models for enthalpies a finer discretization grid than the ones for pressures and flow rates is used, because in general the wave propagation phenomena can be neglected when dealing with power plant dynamics.

The models for water/steam and different gas mixture used in *ThermoPower*, are provided by the *Modelica.Media* library, a package of the Modelica Standard Library [7]. To verify the consistency of the models, each library component has been tested in different configurations. Moreover, the *ThermoPower* library was validated against experimental data coming from a physical model of the evaporation section of a heat recovery steam generator, with a power scaling factor of 1:600. The library is developed according to following main principles:

- The model components are derived from the first principle equations (mass, energy and momentum balance) or from acknowledged empirical correlations;
- The model interface is totally independent of the modeling assumption adopted for each component, in order to achieve a full modularity;
- The level of detail of the models is flexible;
- The inheritance mechanism is used with limitation, to maximize the code readability and modifiability.

A more detail description of the library can be found in [9], [105], also the source code is freely available online [102]. The library features together with the Modelica language make *ThermoPower* to be an attractive option to develop accurate physical models of thermal power systems, and in particular of CCPPs.

Also, the applicability of the library in the modeling, simulation and optimization of the CCPP start-up sequence has been proven in [10], where the behavior of a real plant, with 1-1-1 configuration and three levels of pressure, during the transient phase, is represented by means of Modelica/Dymola and using the components from *ThermoPower*.

3.4.3 Models adapted to start-up phase

In the start-up problem, the process transition involves nonlinear dynamics with multiple steady-state operating points. For the purposes of the present work, the model developed and used to represent the CCPP behavior must have the following features:

- It can represent the whole start-up sequence, from any initial state;
- It includes models of thermal and mechanical stress in some critical components, since the lifetime of CCPP is mainly related to these stresses reached during the start-up;
- It neglects phenomena and components which are not critical for the start-up sequence, in order to obtain a reasonable level of complexity.

The first two points are mandatory for a realistic representation of the start-up phase. Moreover, concerning the second item, the transient regimes which arise particularly during the start-ups give rise to a considerable thermal and mechanical stress in the thick walled components. Therefore, for the integrity and lifetime assessment of the system, the plant model has to include thermo-mechanical stress parts in critical components.

3.4.4 CCPP model

A Modelica model of a typical CCPP with 1-1-1 configuration has been implemented in the Dymola environment, using the elements from *ThermoPower* library. For sake of simplicity, a steam generator with a single level of pressure (high pressure) is considered. The schematic diagram of the plant is illustrated in Figure 3.1. The plant data have been

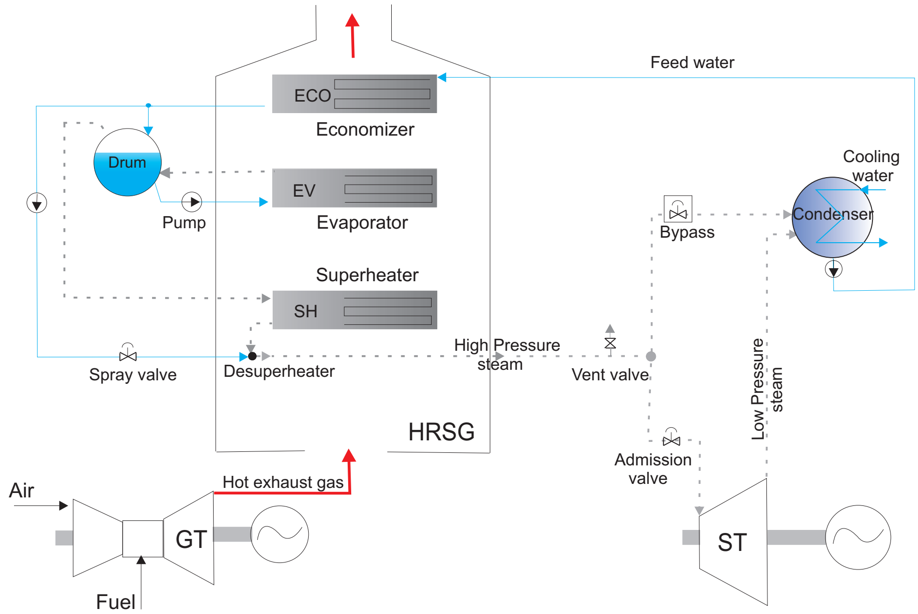


Figure 3.1: Schematic diagram of a CCPP

derived from a detailed simulation model with three levels of pressure used in [10] to study the start-up sequence of a combined cycle plant. The original model was designed and parameterized with data from a typical unit, and validated by replicating a real start-up sequence as recorded by the plant data control system (DCS).

In a first phase, the aim was to build a reliable plant model, accurate enough to represent the main dynamics of a real combined cycle during the start-up phase. Subsequently, the resulting models expected to be coupled with numerical optimization algorithms to compute optimal transients and used for control purposes.

The model built by using the *ThermoPower* library is derived mainly from the first principle laws and adapted to the start-up phase. The components elaboration is based on the object-oriented modeling method: each component is described by means of differential-algebraic equations, and then the components are connected via a-causal connection equations to create the complete model. It should be pointed out that new model components which are not available in the existing libraries, have been developed. For example, in the case of the condensation phenomenon that occurs during the start-up phase, a steam line model has been implemented in order to evaluate the condensation level and to eliminate it, if necessary.

The main features of the plant model with its main components are summarized in Table 3.1. According to the plant structure shown in Figure 3.1 and following the object-

GT data at nominal load	
power	235 MW
flue gas flow rate	585.5 kg/s
exhaust temperature	570 °C
HRSG data at base load	
HP steam pressure	95 bar
steam temperature	520 °C
steam flow rate	64.2 kg/s
ST data at nominal load	
power	80 MW

Table 3.1: Main features of the developed CCPP

oriented modeling principle, the complete CCPP model is obtained by connecting the models of the units (e.g., GT, HRSG, ST), via thermo-fluid connectors [9]. The main model components are briefly described below, a detailed description of the elementary first-principle models of the main CCPP elements is reported in [45].

3.4.4.1 Model components description

3.4.4.1.1 Gas Turbine model

The GT is not generally a limiting factor for start-up time, since the GT has a much faster dynamic response than the HRSG and ST assembly. Accordingly, a simplified description for the GT model has been used. The GT model is described as an ideal source of flue gases, whose flow rate and temperature are specified as a function of the load level input signal (*GTLoad* representing the control variable). The GT model describes a typical behavior of real units with Inlet Guide Vane (IGV) control: from 0% to 50-60% load, the IGVs are closed, thus the exhaust gas flow is approximately constant and the exhaust temperature increases until the base load temperature; for higher load levels, the IGVs begin to open, so the exhaust gas flow increases and the exhaust temperature is maintained constant. The GT behavior description is approximated by piecewise linear functions, which are included in an algebraic model. The schematic representation of its behavior is shown in Figure 3.2.

3.4.4.1.2 Heat Recovery Steam Generator model

The HRSG is the multivariable subsystem of the CCPP with a modular structure (see Figure 3.1). The modular structure enables a separate representation of its component

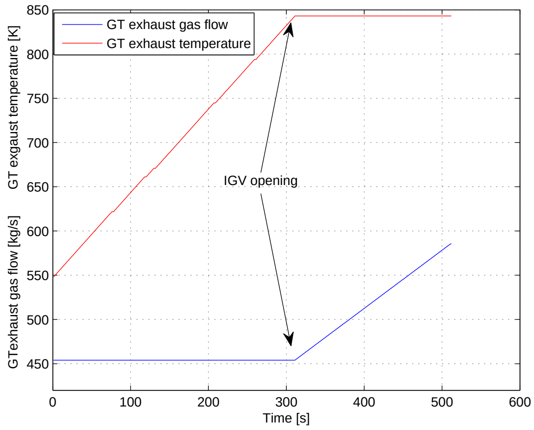


Figure 3.2: Schematic representation of the GT behavior

elements, exploiting thus the advantages provided by the modeling object-oriented language. The complete representation of the steam generator is subsequently obtained by assembling all components according to the plant topology.

The model of the HRSG is composed of several sub-systems, including: heat exchangers, a drum boiler, a desuperheater system and elementary physical components (valves, pumps). Its control inputs are represented by the drum feed-water flow rate and the desuperheater flow rate.

Heat Exchangers

The HRSG model has three heat exchangers: one economizer, one evaporator and one superheater (see Figure 3.1). The heat exchangers together with the drum boiler perform the heat transfer from the GT exhaust gas to the steam/water fluid.

The structure of each heat exchanger includes finite-volume models of the fluid flows (water/steam and/or flue gases), fluid-wall heat transfer and wall thermal inertia. Due to increased efficiency, the adopted heat exchangers configuration is counter-current (i.e. fluxes flowing in opposite directions).

The model of fluid in economizer and superheater is of 1-dimensional type with a single phase, while in evaporator a two-phase model is used to describe the fluid flow. For example, in the case of a 1-D single phase flow in a tube of constant cross-section, assuming that the fluid velocity is uniformly distributed on the cross-section, and that the kinetic term and the thermal diffusion effects are negligible, the basic equations are the distributed-parameter mass, momentum and energy balances:

$$A\frac{\partial \rho}{\partial t} + \frac{\partial w}{\partial x} = 0 \tag{3.1}$$

$$\frac{\partial w}{\partial t} + A\frac{\partial p}{\partial x} + \rho g A\frac{dz}{dx} + \frac{C_f \omega_p}{2\rho A^2} w\,|w| = 0 \tag{3.2}$$

$$\rho A\frac{\partial h}{\partial t}+w\frac{\partial h}{\partial x}-A\frac{dp}{dt}=\omega_p\phi \tag{3.3}$$

where ρ is the fluid density, w is the mass flow rate, p is the pressure, A denotes the pipe cross-section, g is the gravity acceleration, z is the elevation, C_f is the Fanning friction coefficient, ω_p is the pipe perimeter, x is the tube abscissa, h is the fluid enthalpy and ϕ denotes the heat flux entering the tube across the lateral surface. The equation (3.3) is obtained from the standard energy balance by taking (3.1) into account. The equations are then discretized with finite volume method [106].

Drum

The drum model describes a cylindrical cavity containing saturated steam and water. It is equipped with connectors for feedwater inlet (connection with the economizer), steam outlet (connection with the superheater), downcomer outlet and riser inlet (the circuits that connect the drum to the evaporator). Also the drum model disposes of a connector for blowdown that is neglected in our case. The drum representation is based on mass and energy balances, by assuming the thermodynamic equilibrium between the liquid and vapor phases. The basic equations (mass and energy balance) are indicated in the following:

$$\frac{dM}{dt}=w_f+w_r-w_v-w_d-w_b \tag{3.4}$$

$$\frac{dE_t}{dt}=w_fh_f+w_rh_r-w_vh_v-w_dh_d-w_bh_{ls}-Q_{me} \tag{3.5}$$

$$M=M_l+M_v \tag{3.6}$$

$$E=M_vh_v+M_lh_{ls}-pV+E_m \tag{3.7}$$

where M, E_t are the mass and the total energy of the fluid (liquid and vapor), w_f denotes the mass flow rate of feed water, w_r is the mass flow rate from the risers, w_v is the mass flow rate of steam at the outlet, w_d represents the flow rate to the downcomers, w_b denotes the mass flow rate of blowdown, h_f is the specific enthalpy of feed water, h_r is the specific enthalpy of fluid from the risers, h_v is the specific enthalpy of steam at the drum outlet (equal to the enthalpy of vapor at saturation), h_d denotes the specific enthalpy of liquid to the downcomers, h_{ls} represents the enthalpy of saturated liquid and Q_{me} is the heat flow from metal wall to external environment; $M_v=\rho_vV_v$ is the vapor mass (ρ_v, V_v are the density and the volume of the vapor), $M_l=\rho_lV_l$ is the liquid mass (ρ_l, V_l are the density and the volume of the liquid), p is the pressure in the drum, V is the total volume of the drum ($V=V_l+V_v$) and E_m is the internal energy of the metal.

The temperature of the water and steam volume inside the drum corresponds to the saturation state. The metal temperature is assumed to be equal to water/steam temperature and the heat exchange through the metal wall between the drum and the external environment is neglected ($E_m = 0$, $Q_{me} = 0$).

Desuperheater

The desuperheater system plays a fundamental role in the steam temperature regulation at the output of the superheater, by mixing a controlled water flow rate to the superheated steam flux. The system includes a mixer and an ideal flow rate source.

The mixer representation, as in the case of drum model, is based on standard mass and energy balances (3.8, 3.9), assuming uniform pressure and temperature in the control volume.

$$\frac{dM}{dt} = w_l + w_s - w_{ls} \tag{3.8}$$

$$\frac{dE_t}{dt} = w_l h_l + w_s h_s - w_{ls} h_{ls} - \gamma S(T - T_m) + Q \tag{3.9}$$

where M, E_t are the mass and the energy of the fluid, w_l denotes the mass flow rate of the water, w_s is the mass flow rate of the steam, w_{ls} is the mass flow rate of the mixture, h_l is the specific enthalpy of water, h_l denotes the specific enthalpy of steam, h_{ls} is the specific enthalpy of mixture and Q is the heat flow exchanged with the exterior. The heat transfer with the metal is also considered, thus T, T_m are the steam and metal temperatures, γ is the heat transfer coefficient and S is the internal surface.

Thermo-mechanical stress model

The transients that occur during start-up give rise to a considerable stress in the thick walled components. Therefore, for the integrity and the lifetime assessment of the system during the start-up, a thermo-mechanical stress model on the critical components of the heat collector boiler, namely on superheater steam header section, is included.

To compute the thermo-mechanical stress a one-dimensional radial model is considered, by assuming the component geometry similar to that of a hollow, smooth cylinder, of infinite length. Also an axial-symmetric temperature distribution is assumed. This assumption is reasonable as long as in the computation of total stress, acting on the component, appropriate concentration ratios are introduced (dictated by technical standards or requirements of the manufacturers) that take into account the three-dimensional structure of the unit.

Concerning the stress computation, the introduced hypothesis enables the consideration of the cylinder in plane strain, i.e. no component of stress in the axial direction ($\sigma_z = 0$). For headers, if the length is not much larger than the diameter, the axial strain can also be included, this component reducing the total thermo-mechanical stress on the components. Therefore, by neglecting this potential component the stress estimation is performed in order to improve the system security.

To represent the radial distribution of the temperature, a thermal model based on Fourier's equation discretized by a finite difference method, has been used.

Mechanical stress in header

On the header part, the mechanical stress that originates inside the component wall is due to the internal pressure of the fluid. Generally, the most affected part is the internal surface, the mechanical stress on the external surface is often neglected.

The evaluation of the stress depends on the adopted technical standards. In the model two of the most popular standards, the German standard TRD and the American standard ASME, are represented [107], [49]. In the sequel only the ASME evaluation is considered.

According to this standard, the mechanical stress components ($\sigma_{p,r}$- radial, $\sigma_{p,t}$- tangential, $\sigma_{p,z}$- axial) are represented in the following way:

$$\begin{aligned} \sigma_{p,r} &= -p \\ \sigma_{p,t} &= p\frac{r^2+1}{r^2-1} \\ \sigma_{p,z} &= p\frac{1}{r^2-1} \end{aligned} \tag{3.10}$$

where r is the ratio between the external and the inner diameter of the steam header and p is the pressure of the fluid.

Thermal stress in header

The thermal stress is computed on the internal and external surfaces, as a function of the difference between the mean temperature and the surface temperature. Thus on the internal surface the stress components are:

$$\begin{aligned} \sigma_{T,r} &= 0 \\ \sigma_{T,t} &= \frac{\alpha E}{1-\nu}(T_m - T_{int}) \\ \sigma_{T,z} &= \frac{\alpha E}{1-\nu}(T_m - T_{int}) \end{aligned} \tag{3.11}$$

where α is the thermal expansion coefficient, E is the Young's modulus, ν is the Poisson's ratio and T_m, T_{int} are the mean temperature and the internal surface temperature of the material, respectively. The material properties are dependent on its temperature. These characteristics are evaluated during simulation by means of a defined model of material. On the external surface, the thermal stress is calculated in the same manner, by replacing the temperature T_{int} with T_{ext}, where T_{ext} denotes the external surface temperature of the material. Also, it should be noted that usually the axial component is not required to be determined since it is the same as the maximum principal stress (tangential stress).

Equivalent stress

The equivalent stress used to evaluate the lifetime of the unit is computed based on the stress intensity (SI), which represents the largest absolute value of the stress differences:

$$Header\ Stress = max[|\sigma_1 - \sigma_2|, |\sigma_2 - \sigma_3|, |\sigma_1 - \sigma_3|] \tag{3.12}$$

with

$$\sigma_i = K_t \sigma_{T,i} + K_p \sigma_{p,i}, \quad i = \overline{1,3} \tag{3.13}$$

where i corresponds to the stress components (1- axial, 2- tangential, 3- radial), K_t and K_p are stress concentration factors for thermal and mechanical components.

3.4.4.1.3 Steam Turbine model

In order to provide a consistent description of all the start-up phases, the ST unit model is quite detailed. The unit includes a bypass/throttle valve circuit model, a turbine model, a rotational inertia model, an electrical generator model and a simplified model of the connection to the grid.

The modeled turbine has a single stage (HP) and as the most critical constraint in terms of thermal-mechanical stress during the start-up is localized in the ST rotor, in particular in the part of turbine shaft located close to the admission, this turbine part is separately represented. Thus, the steam turbine model is obtained by connecting two sub-models:

- an impulse stage model, which is described by a nozzle, where the steam undergoes a quasi-isentropic transformation, assuming a constant isentropic efficiency and an ideal bucket, which converts the kinetic energy in mechanical energy. The kinetic energy of the steam at the nozzle inlet is neglected. The impulse stage is coupled to a thermo-mechanical stress model of the shaft section.
- a model based on the Stodola's law [108], which computes the performance characteristics of the turbine (i.e. the steam flow rate).

The steam flow through the turbine section described according to Stodola's law is stated as:

$$w_s = k_t \sqrt{p_s \rho} \sqrt{1 - \frac{1}{r^2}} \tag{3.14}$$

where w_s, p_s are the mass flow rate and the inlet pressure of the steam, k_t is the coefficient of the Stodola's law, ρ is the density of the steam and r denotes the pressure ratio (ratio of inlet/outlet stage pressure).
The ST generated power is given by:

$$P_m = \eta_m w_s (h_{in} - h_{out}) \tag{3.15}$$

where P_m is the mechanical power from the steam, η_m is the mechanical efficiency, h_{in}, h_{out} are the inlet and outlet steam enthalpy.

The unit is completed by a rotational inertia and an electrical generator model as well as a simplified model of the connection to the grid. The generator model is represented as an ideal synchronous machine that converts the mechanical power in electrical power:

$$P_m = \frac{P_e}{\eta} + P_{loss} \tag{3.16}$$

where P_e is the electrical power, P_{loss} is inertial power loss and η is the conversion efficiency. The inertial power loss is zero by default, but the generator inertia in the model can be considered:

$$P_{loss} = J\frac{d\omega}{dt} * \omega \tag{3.17}$$

where J is the moment of inertia of the generator and ω is the angular velocity of the shaft.

Also the ST model contains a small torque in order to avoid the turbine stopping, which would lead to model singularities. The ST model has three control inputs: the bypass position, the throttle valve position and the generator grid breaker that connects the turbine to the grid.

Thermo-mechanical stress model

During the start-up sequence, the most limiting factor in terms of time reduction is the lifetime consumption of the ST rotor caused by the thermal and mechanical stresses. In order to evaluate the lifetime of the unit, the ST impulse stage is coupled to a thermo-mechanical stress model in the section of a steam turbine shaft.

Mechanical stress in rotor

The mechanical component of the stress on the rotor is mainly generated by a centrifugal force caused by the rotation speed, the steam pressure effect being neglected. The centrifugal components have been determined by assuming the similarity between the affected area of the rotor and a thick disk (annular wheel), that rotates at the same speed. Considering also a state of plane stress ($\sigma_{m,z} = 0$), the mechanical stress on the internal rotor surface is thus described:

$$\begin{aligned} \sigma_{m,r} &= 0 \\ \sigma_{m,t} &= \frac{3+\nu}{4}\rho\omega^2(r_{ext}^2 - \frac{1-\nu}{3+\nu}r_{int}^2) \end{aligned} \tag{3.18}$$

where ω is the angular velocity of the rotor, ν is the Poisson's ratio, r_{int} is the internal radius and r_{ext} is the external radius of the disk. On the external surface the stress is represented in the same manner, considering that the internal radius corresponds to the external one:

$$\begin{aligned} \sigma_{m,r} &= 0 \\ \sigma_{m,t} &= \frac{3+\nu}{4}\rho\omega^2(r_{int}^2 - \frac{1-\nu}{3+\nu}r_{ext}^2) \end{aligned} \tag{3.19}$$

In general the influence of centrifugal forces is neglected on the outer surface of the rotor.

Thermal stress in rotor

The representation of the thermal stress in rotor is carried out as in the header case, by assuming the same condition. The thermal stress is computed as a function of the difference between the surface temperature and the mean temperature of the rotor.

Equivalent stress

In the case of the rotor, the stress is computed on the internal and external surfaces, but as this latter is the most exposed, the attention is focused mainly on this side. The equivalent stress representation on the external surface of the rotor is represented by using the same principle as in the header case, neglecting the mechanical component:

$$Rotor\ Stress = K_t \sigma_{T,t} \tag{3.20}$$

where $\sigma_{T,t}$ is the tangential component of the thermal stress and K_t denotes a stress concentration factor for thermal component.

Valve and pump models

To complete the system representation, the dynamics of the main actuators (valves and pumps) are included. The valve models are relied on the international standard equations for valve sizing [109]. The pump models are expressed by algebraic characteristic equations derived from the manufacturer's design data. In our model a simplified pump model, which describes the flow rate passing through it, neglecting the specific enthalpy due to the pressure drop between the inlet and the outlet, has been used.

3.4.4.1.4 Simplified component models

The plant parts that work at low temperature (condenser, feed-water system) are represented in a simplified way, because these are not critical concerning the stress and the control, during the start-up. The condenser is described as a sink with the pressure specified as a function of the steam flow rate, while the feed-water system is represented by an ideal flow rate source, controlled by a PI loop, in order to maintain the drum level at a constant value.

Considering the fact that condensation is an important phenomenon that occurs during the start-up phase, a simple steam line model provided with vent valve has been included. Its objective is mainly to evaluate and eliminate the quantity of condensation. A complete description of a steam line model for combined cycle plants can be found in [110].

3.4.5 Model initialization

The initialization of the plant model in a shut-down state is difficult (insufficient knowledge of the initial values, low or zero flow rates). Therefore in a first phase, the Modelica model has been initialized close to the full-load steady state. Then the model has been brought to a state corresponding to the hot start-up of the plant by a shut-down sequence. The obtained state has been used to initialize the model in a feasible region. Thus, the model is designed to study the hot start-up procedure of the plant. Following the start-up sequence specified in the previous chapter, the initial state coincides with second stage of the start-up procedure (see Figure 2.18), when the GT is synchronized and the HRSG is ready (the thermal conditioning is fulfilled, that means the superheater metal temperature is greater than steam temperature).

The first stage, *Preparation phase* (GT start, purge) and also the first part of the second stage, *HRSG Start-up phase* with thermal conditioning (GT acceleration, HRSG warming), are assumed to be completed. Also it should be pointed out that by modifying the shut-down sequence the model can be initialized in a state corresponding to a different type of start-up, e.g., to the warm start-up, or to another stage of the sequence.

The state corresponding to the hot start-up of the plant is then:

- GT:
  - GT Power around 17.6 MW[1];
  - Gas flow rate 454 kg/s;
  - Gas temperature around 312 °C;
- HRSG:
  - Steam pressure in the circuit 65 bar;
  - Water flow rate around 8.9 kg/s;
  - Steam temperature around 305 °C;
  - Water flow rate of the desuperheater 0 kg/s;
  - Drum level[2] around 0 m;
  - Economizer metal wall temperature 281 °C;
  - Evaporator metal wall temperature 282 °C;
  - Superheater metal wall temperature 307 °C;
- ST with no steam flow, stopped (”almost”):
  - Mean temperature of the turbine shaft around 420 °C;
  - Bypass opening around 16%;
  - Admission valve completely closed (0);
  - Generator grid breaker open (0).

[1]Value corresponding to 7.5% of maximal load
[2]Drum level is measured with respect to the middle of the drum

Model specification

The CCPP model has 6 inputs and 11 output signals. In line with the Dymola's statistic information, the model is described by a differential algebraic equation (DAE), with 2721 equations, 867 algebraic variables, 54 state variables, 2664 parameters and constants. Also, it should be noted that the variables considered to be of interest during the start-up phase by engineers have been defined as outputs of the model. Specifically, the output signals are:

- GT power;
- GT fuel flow rate.
- Drum level;
- Steam temperature in SH;
- Steam temperature at ST inlet;
- Steam pressure;
- Steam flow rate;
- SH header stress;
- ST power;
- ST frequency;
- ST rotor stress.

The CCPP model has been designed and parameterized with data from a typical unit. The parameters used in this case study have to be adapted for any real case implementation. The validation of the model has been performed by a series of systematic simulation studies. The overall behavior of the model has been analyzed by experts in the domain, and considered as coherent and representative for the CCPP start-up procedure. Moreover, the model has been developed by using the elements from *ThermoPower* library, components that have been validated against the experimental data. Dymola diagram representation of the CCPP model is given in Appendix F.

3.4.6 Applicability of the CCPP model for optimization

While the availability of dedicated languages for physical system modeling, in particular Modelica language, facilitates the development of complex large-scale models, the resulting code from the modeling process is not immediately suitable for numerical integration. It usually requires a number of transformation steps in order generate an efficient code which in turn can be interfaced with algorithms for simulation or optimization.

As previously pointed out, nowadays there are several tools supporting Modelica, environments having mainly as target the efficient simulation of the models. On the other hand, tool support to formulate and solve optimization problems based on Modelica models is in general weak, although several efforts have been made to integrate the advanced modeling language with algorithms for dynamic optimization.

Most times, in the case of complex heterogeneous physical systems the optimization is usually made by considering the model as a black box; the optimization uses the simulation results without exploring the particular structure of the model, according to some heuristic methods, which do not require derivative information or detailed structural information. Typically, these methods offer low performance in terms of convergence and execution times, essential requirements in real-time optimization problems.

In fact, the key to solve efficiently large-scale optimization problems is related to the structure exploitation of the model to be optimized. An algorithm provides better performance, in terms of convergence properties and execution times, if the model structure can be exploited. In general the applicability of a dedicated optimization algorithm is constrained by the structure of the model (e.g., linear, non-linear ODE, DAE, hybrid systems).

Over the last decades numerous algorithms have been developed to solve efficiently the optimization problems, such a class which offers a fast convergence is represented by the gradient-based methods. These algorithms impose restrictions on the structure when the model is formulated in Modelica. For example this excludes the use of specific constructs that may introduce discontinuities since the convergence relies on a twice continuously differentiable right-hand side. Therefore the model must not contain the possible source of discontinuities, such as hybrid constructs (conditional expressions or conditional equations), lookup-tables, functions like *abs* (absolute value), *min*, *max*.

Indeed this limitation excludes the optimization of many realistic Modelica models, in particular the plant model developed starting from the *ThermoPower* library, which includes several sources of discontinuity. On the other hand, the discontinuities can be reformulated by means of continuous representations in order to enable an efficient optimization.

By analyzing the optimization possibilities and seeing the difficulties to use complex Modelica models, in particular the combined cycle model described in this section, for optimization purposes, two solutions derived from the CCPP model presented above have been suggested. A first solution has been to determine a simple identified model able to accurately reproduce the main dynamics of the concerned power plant. The simplified model has been built by interpolating a number of linear estimated models with local validity. In this sense, an identification and interpolation procedure has been developed and implemented in Matlab/Simulink. A brief description of the procedure is reported in Appendix C, a more detailed presentation can be found in [111], [112].

Although in general the considered procedure allows relatively easy to develop simplified models that can be used with standard tools for different purposes (e.g., simulation, control design and/or optimization), the models obtained through this approach are specific to a particular configuration or system, being thus, limited in their applicability. As the dissertation's objective is mainly focused on physical-based models and their use to improve the efficiency of the start-up phase, another type of approach is proposed. The approach aims at developing more general models with a high accuracy, derived from first principle laws and suitable for control purposes. This approach enables the exploitation of the advantages provided by the dedicated modeling languages.

3.5 Conclusions

This chapter has presented the framework to achieve a physical model of a combined cycle plant able to represent the start-up procedure (modeling language, library specialized for modeling of power plants, requirements for start-up sequence, model initialization), as well as the actual description of the model. The elaboration of the CCPP model has been performed by means of the library for thermal power systems *ThermoPower*, and has been implemented in the Dymola environment. The CCPP model is adapted to the start-up phase, neglecting the irrelevant phenomena and considering in a simplified way components which are not critical for the respective phase. The plant data have been derived from a detailed simulation model used to study the start-up sequence of a real CCPP. The global behavior of the model has been analyzed by experts in the domain, and considered as representative for the start-up procedure.

The physical knowledge-based models developed in Modelica are often difficult to be used for optimization purposes. Therefore, it appears to be interesting to develop knowledge-based models that can be efficiently used in optimal control problems. For this, a different type of approach must be addressed. This aspect will be presented in the following chapter.

CHAPTER 4

CCPP models adapted for optimization

4.1 Introduction

Driven by the increasing use of high-level modeling languages in industrial applications due to their outstanding features and together with the attempts to obtain improved performance to meet increasingly complex and rigorous demands, the investigation and elaboration of Modelica models that can be used for optimization are of interest. The goal is now to employ such models to formulate and solve optimization problems. Moreover, in industrial applications, one of the most important reasons for the limited use of the advanced control techniques (e.g., MPC [85]), is the significant time and effort necessary to develop and validate process models that are applicable over a wide operating range.

Nowadays the availability of the advanced modeling languages and an important number of libraries validated against experimental data facilitates the development of high accuracy models which could be used for control purposes. Nevertheless, most of the models developed with these tools are primarily intended for simulation and process design, and are either not immediately suitable for optimization or limited to a particular optimization algorithm. A solution in this direction has been suggested in the previous chapter (see also Appendix C). Nevertheless, the proposed procedure can be considered quite laborious and in certain cases too application-specific (i.e. in the case of a parameters modification in the plant configuration the whole procedure of model building must be redesigned). In this chapter, a general methodology that enables the building of Modelica models suitable for optimal control problems is proposed.

4.2 Modelica CCPP model suitable for optimization

Modelica together with the *ThermoPower* library forms a powerful tool for a rapid development of accurate physical models of thermo power systems, and in particular of combined cycle systems. On the other hand, the models developed using *ThermoPower* are designed for simulation and use extensively Modelica object-oriented features, such as genericity and inheritance, which make them unsuitable for optimization purposes with tools available at the moment.

Also a drawback when the formulation of optimization problem is based on non-linear models is that a significant computational effort to solve the optimal control problem is needed. To solve efficiently such problems, algorithms that provide high performance in terms of convergence properties and execution times are required. For example, when MPC strategies are applied, algorithms compatible with a real-time applicability are necessary. As previously mentioned in Chapter 3, the key for an efficient resolution of large-scale optimization problems is linked to the exploitation of the model structure. The applicability of dedicated optimization algorithms is usually constrained by the model structure.

An important family of optimization algorithms is that of gradient-based methods. This class of algorithms can typically address large-scale systems and is often used to solve real-time optimal control problems in advanced control structures, offering a fast convergence. In general the convergence of these algorithms is based on the fact that the right-hand side is twice continuously differentiable (e.g., sequential quadratic programming (SQP) [113], interior point methods [114]). The application of gradient-based methods to non-smooth or discontinuous problems can lead to a failure in convergence, in optimality conditions, or in gradient information. In other words, the application of such algorithms to solve optimal control problems based on Modelica models requires a smooth problem[1], more specifically a second-order smoothness[2]. This characteristic restricts thus the use of specific constructs that may introduce discontinuities, when the model is formulated in Modelica. Obviously, this restriction excludes the possibility to optimize the developed CCPP model, since it includes several sources of discontinuity. Therefore some modifications and compromises are required in order to build a ready-to-use Modelica model for optimization purposes. The CCPP *ThermoPower* model designed in the previous chapter will be considered as *reference* plant and will be used to derive and validate models suitable for optimization.

In the sequel, a methodology to transform a physical simulation model to one adapted for model-based control is proposed. The components, used to systematically derive a combined cycle smooth model, have been included into a new library, so-called *ThermoOpt*. This library in turn can be employed to create other plant models that enable an efficient optimization. The library elements as well as the derived CCPP model have been validated by performing a number of simulation tests and comparing the obtained responses with the ones generated by the original model.

4.3 Methodology to create smooth models

In the elaboration phase of the smooth model two priority issues have been considered: primarily, the model must be appropriate for an efficient optimization, and secondly the model has to reflect as closely as possible the behavior of the real system. The idea

[1]A function is *smooth* if it is differentiable and the derivatives are continuous

[2]The second derivatives exist and are continuous

is to transform the existing validated models to be suitable for optimization and not re-create new models components. Thus the new components keep the structure of the original models and can be easily deduced from them. In this sense for each component of the *ThermoPower* library, corresponding to the initial model, a version in which the discontinuities are reformulated by using smooth approximations has been developed. This enables to keep the same structure of the model and the most important aspect, to build a model suitable for optimization in terms of issues such as complexity, accuracy and smoothness. In this context, the challenge is to simplify/approximate the model elements as much as possible while keeping a good precision in terms of behavior with respect to the reference model.

The principle of the approach considered to build the CCPP optimization model has been the following:

- analysis of the simulation model components and identification of the possible discontinuity sources;
- elaboration of the model components without discontinuities, by introducing the continuous representations [115];
- model components validation.

4.3.1 Elimination of discontinuity sources

4.3.1.1 Approximation of conditional expressions

In the components analysis phase, the expressions which can introduce discontinuities and consequently make the optimization problem to become non-smooth have been identified, namely: conditional equation constructs (*if-expression* and *if-equations* to describe different types of models and their behavior), *abs* and *max* functions (which can also be expanded into conditional expressions), piecewise linear functions and steam/water tables.

Most of these discontinuity sources have been reformulated by using continuous representations. The construction of the representations is based on smooth approximations of the Heaviside function:

$$\forall x \in \mathbb{R}, \quad H(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \tag{4.1}$$

replaced by the smooth approximation:

$$\forall x \in \mathbb{R}, \quad H_k(x) = \frac{1}{1 + e^{-kx}} \tag{4.2}$$

where k is the smoothness coefficient (see Figure 4.1).

In the case of conditional equation constructs, such as the conditional model of enthalpy presented below:

$$h_m = \text{if} \quad h \leq h_l \quad \text{then} \quad h_l \quad \text{else} \quad h_v \tag{4.3}$$

the continuous representation is formulated as follows:

$$f(h) = \frac{1}{1 + e^{-k*(h-h_l)}} \tag{4.4a}$$

$$h_m = h_v * f(h) + (1 - f(h)) * h_l \tag{4.4b}$$

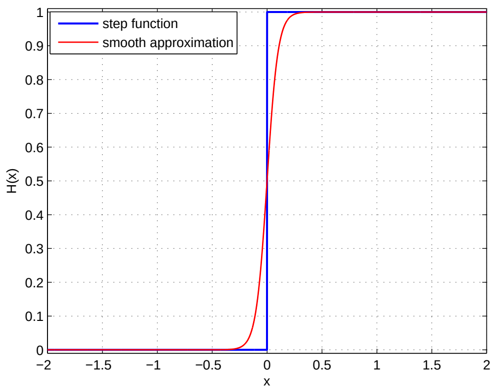


Figure 4.1: Heaviside step function and smooth approximation for k=*20*

On the same principle, the piecewise linear functions used to approximate particular functions (e.g., the behavior of GT unit), are reformulated in order to be smoothed at the border points. Also for this type of functions, in order to reduce the number of equations, a polynomial approximation has been implemented.

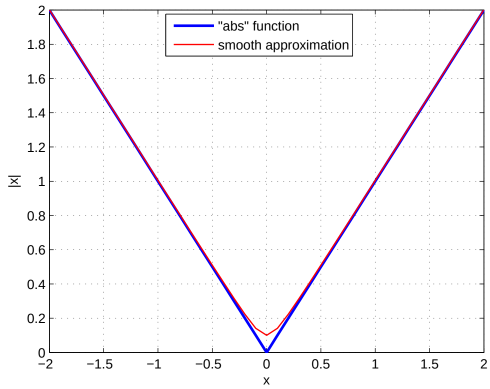


Figure 4.2: *Abs* function and smooth approximation for $\epsilon = 0.2$

In certain cases, e.g., *max*, *abs* functions, for a more rapid implementation a smooth approximation of the following type has been used:

$$\forall x, y \in \mathbb{R}, \quad max(x, y) = \frac{((x-y)^2 + \epsilon^2)^{0.5}}{2} + \frac{(x+y)}{2} \tag{4.5}$$

where ϵ, as in the previous case, represents the coefficient used to set the level of smoothness (see Figure 4.2).

In the elaboration of the smooth model, the most intricate part is linked to the smooth representations of the fluid properties models. The manner in which these steam/water tables have been approximated, aimed at obtaining a consistent level of accuracy with an efficient simulation/optimization. The details of the adopted approach will be presented in the following section.

4.3.1.2 Approximation of steam/water tables

In thermal power plants modeling two types of fluid (water/steam and ideal gas mixture) are mainly considered. For these fluids, specific models are defined. In *ThermoPower*, the fluid properties are computed using the medium models included in the *Modelica.Media*, a package of the Modelica Standard Library [7] that contains property models for single and multiple substance fluids with one and multiple phases. A medium model defines a set of algebraic equations for the thermodynamic variables used in the mass and energy balance of component models. In general the medium equations are independent from the component equations.

The models for the fluid properties calculation are in line with the International Association for the Properties of Water and Steam (IAPWS-IF97) [116], and with accurate ideal gas models based on NASA data, respectively. The implementation of these formulations has been explicitly designed to work well in dynamic simulations, offering high computational performance. On the other hand these functions for the thermodynamic computation include a large number of sub-functions hierarchically implemented; these sub-functions determine the relations between variables using conditional expressions for the affiliation to each thermodynamic region, fact that makes their use with optimization algorithms to generate a series of issues. An alternative to the thermodynamic properties of media for optimization purposes is therefore required.

The industrial standard formulation, IAPWS-IF97, relies on a set of equations which covers a particular range of validity, divided in five regions (see Figure 4.3). According to the variation domain of CCPP model, corresponding to a domain of pressure $\in$ [1-130 bar] and temperature $\in$ [290-865 K], only three regions are covered (the region **1** for liquid state, the region **2** for vapor and ideal gas state and the region **4** for vapor/liquid equilibrium). This facilitates somehow the applicability of the approximation procedure.

Due to the general design of the Modelica medium models, in the *Modelica.Media* library there are several types of functions used to compute the substance properties, covering all regions with large modeling options. The smooth representation of all these functions is beyond the scope of this work. Therefore, attention is focused only on the functions called by the components of the *ThermoPower* library, which are used in the structure of the developed CCPP model. These functions have been approximated and implemented in a new medium model, by using the standard interfaces defined in *Modelica.Media.Interfaces*.

In the Modelica model that derives the medium properties for water in different phases, the independent variable considered are: the *pressure* and the *specific enthalpy*. This is the most popular choice and is recommended for general purpose applications, in particular for power plants.

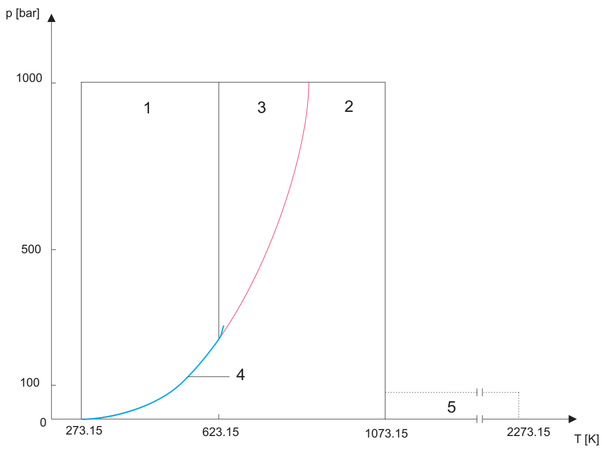


Figure 4.3: Regions according to IAPWS-IF97 standard

The class of functions used in Modelica medium models can be roughly divided into two categories: monovariable and multivariable functions. These functions usually include models with sets of equations corresponding to each thermodynamic regions, models that are coupled by means of conditional expressions. To obtain a smooth approximation, for each type, a methodology is proposed.

The basic principle of the method is the following:

1. the original Modelica function for calculating medium properties is evaluated in N points from the indicated variation interval;

2. a mathematical model that fits the generated data is derived.

The idea is to obtain a mathematical representation for each Modelica function, that provides a good accuracy compared to the original one (a good fit of our data), keeping a reasonable level of complexity and respecting the smoothness restrictions.

Monovariable functions approximation

In the case of the monovariable functions, the determination of a mathematical representation has been performed by using the Matlab Curve Fitting Toolbox [117] or the Polyfitn Toolbox [118]. As a remark, the latter in certain situations, e.g., when the points are not well-scaled, can offer a better fit. Based on regression techniques, different types of functions have been tested (e.g., polynomial, exponential, power series) in order to find the coefficients (parameters) which ensure that function matches as closely as possible with our data points. In the sequel, the method is explained by means of an example.

Consider the function *saturationTemperature()*, defined in *Modelica.Media.Water*, that typically calls in its algorithm section a hierarchy of sub-functions, can be described by a mathematical representation as follows:

$$T_{sat} = f(p) \tag{4.6}$$

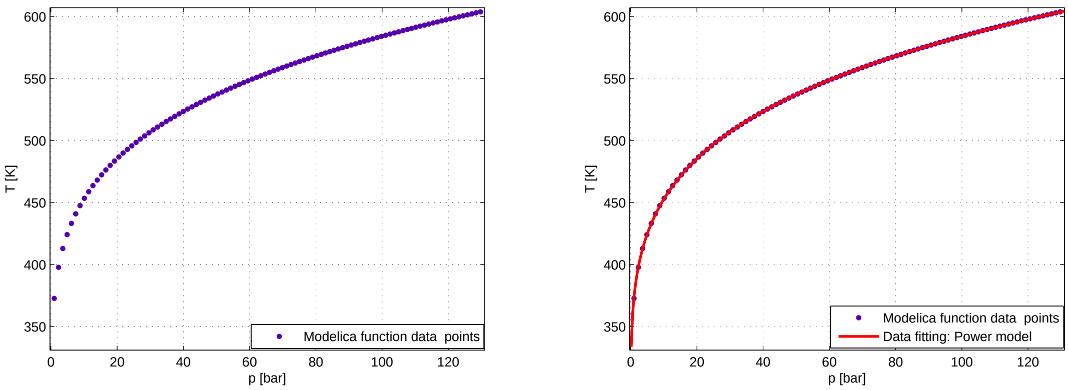
with $f : D_p \rightarrow D_T$, where D_p and D_T are the variation domains for pressure and temperature respectively. The function returns the saturation temperature (T_{sat}) depending on pressure (p).

By using a trivial procedure implemented in Dymola, the function is simulated by varying the pressure in the defined domain of the CCPP model ($p \in$ [1-130 bar]) and evaluated in N points (p^i, T^i_{sat}):

$$T^i_{sat} = f(p^i) \tag{4.7}$$

with $i = \overline{1, N}$. The evaluation of the function is illustrated in Figure 4.4a.

The data points are then imported in the curve fitting tool, which fits this set of points to a mathematical function ($T_{sat} = f_{app}(p)$, see Figure 4.4b). It should be pointed out that the number of evaluation points[3] N, has been chosen so as a trade-off between complexity and precision to be performed. As can be seen from the Figure 4.4b, with a power function, the results obtained in this case indicate a high goodness of fit (*R-square*=1, *Root Mean Square Error*=0.1945).



(a) Data points evaluation

(b) Power model fit

Figure 4.4: Saturation temperature approximation

Multivariable functions approximation

The approximation method previously presented has been tested in the multivariable case. The obtained results have shown a poor performance (high approximation errors). Therefore, in order to improve the approximation results, the approach has been modified and adapted to the multivariable case to cover all considered regions. The idea has been to partition the entire variation domain by regions and to associate an approximation model to each region. Following the same presentation principle as in the previous section, the approach will be explained by means of an example.

[3] N=100 points were selected for most approximations

Consider the function *temperature_ph()*, defined using the IF97 standard in the library of media property models for water, *Modelica.Media.Water*, to compute the temperature (T) as a function of pressure (p) and specific enthalpy (h). The Modelica representation can be interpreted as a mathematical function of the form:

$$T = f(p, h) \tag{4.8}$$

with $f : D_p \times D_h \to D_T$, where D_p, D_h and D_T are the variation domains for pressure, specific enthalpy and temperature.

Over the entire domain of variation, the function is simulated and evaluated in N points (p^i, h^i, T^i):

$$T^i = f(p^i, h^i) \tag{4.9}$$

with $i = \overline{1, N}$. The graphical representation of the function is shown in Figure 4.5.

The derived data points are then partitioned by regions: $(p^j_{R_1}, h^j_{R_1}, T^j_{R_1})$, $(p^k_{R_2}, h^k_{R_2}, T^k_{R_2})$ and $(p^l_{R_4}, h^l_{R_4}, T^l_{R_4})$, with $j = \overline{1, M}$, $k = \overline{1, Q}$, $l = \overline{1, R}$ and $M + Q + R = N$. These points, corresponding to each region, are imported in the surface fitting tool, and an approximation of the original function $T = f(p, h)$ is performed by region (see Figure 4.6):

$$T^j = f_{R_1}(p^j, h^j) \tag{4.10a}$$

$$T^k = f_{R_2}(p^k, h^k) \tag{4.10b}$$

$$T^l = f_{R_4}(p^l, h^l) \tag{4.10c}$$

where f_{R_1}, f_{R_2}, f_{R_4} are the approximation functions associated to the region **1** (R_1), the region **2** (R_2) and the region **4** (R_4).

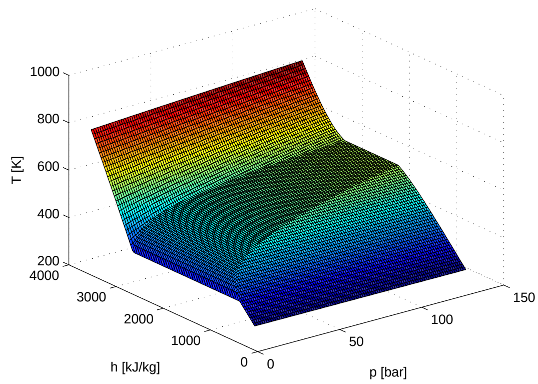
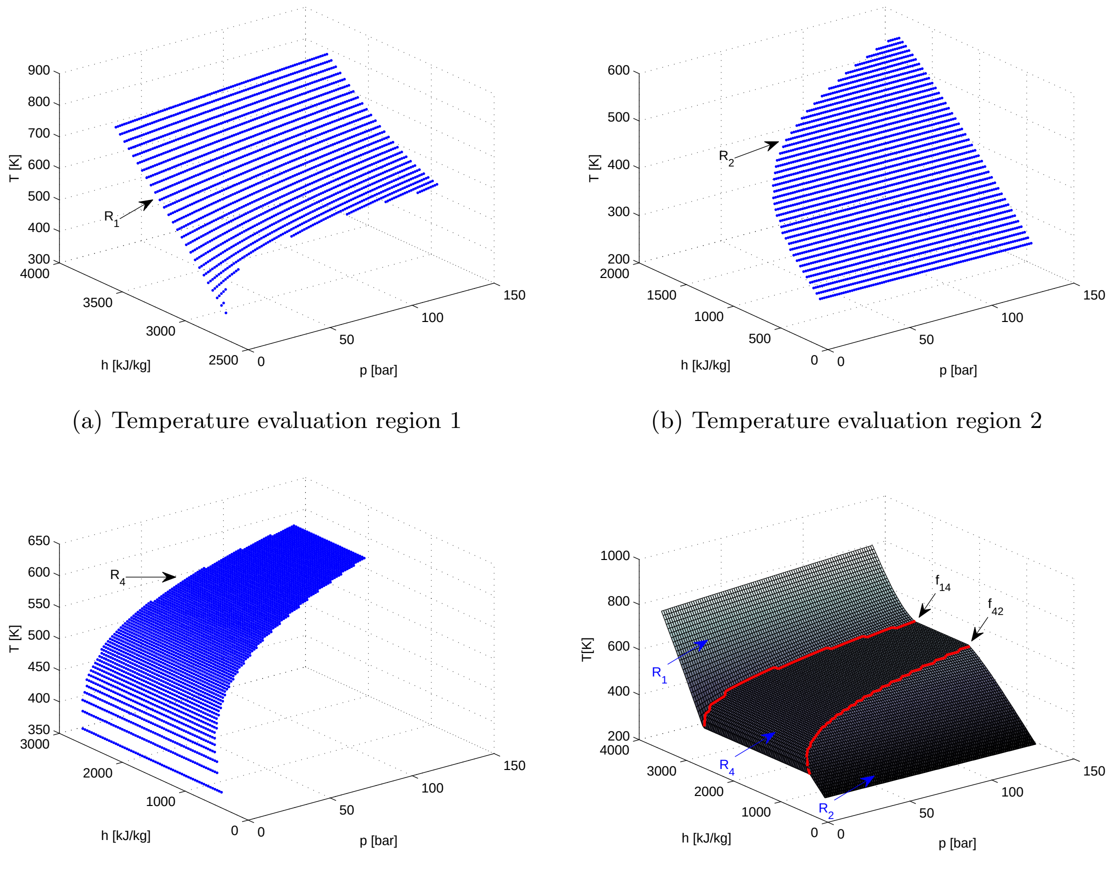


Figure 4.5: Temperature as function of pressure and specific enthalpy (T=*f(p,h)*), according to *Modelica.Media* library



(a) Temperature evaluation region 1

(b) Temperature evaluation region 2

(c) Temperature evaluation region 4

(d) Complete representation with boundaries

Figure 4.6: Temperature function decomposition by regions

Once the fitting problem for each region has been solved, the overall approximation is derived by connecting the functions associated to the regions in a continuous manner, in order to ensure its smoothness. In this case, the principle based on the Heaviside function approximation, previously described, has been adapted by considering the boundary functions among regions. Thus, rewriting equation (4.1) as:

$$\forall x, y \in \mathbb{R}, \quad H(x, y) = \begin{cases} 0, & x - f_R(y) < 0 \\ 1, & x - f_R(y) \geq 0 \end{cases} \tag{4.11}$$

the new smooth approximation used to define the continuous representation of the function becomes:

$$\forall x, y \in \mathbb{R}, \quad H_k(x, y) = \frac{1}{1 + e^{-k(x - f_R(y))}} \tag{4.12}$$

where f_R denotes the boundary function.

By applying this approximation for the temperature function, the continuous representations of the temperature function for adjacent regions (denoted by $H_{k_{14}}$, $H_{k_{42}}$) can be written as:

$$\forall(p,h) \in D_p \times D_h, \quad H_{k_{14}}(h,p) = \frac{1}{1+e^{-k(h-f_{14}(p))}} \tag{4.13a}$$

$$\forall(p,h) \in D_p \times D_h, \quad H_{k_{42}}(h,p) = \frac{1}{1+e^{-k(h-f_{42}(p))}} \tag{4.13b}$$

where f_{14} denotes the boundary function between the region **1** and the region **4** and f_{42} is the boundary function between the region **4** and the region **2**. In turn these boundaries (represented with red in Figure 4.6d) have been approximated by a parametric mathematical model, which describes the specific enthalpy as function of pressure ($h = f_{14}(p)$, $h = f_{42}(p)$).

Finally, the continuous approximation of the temperature function will be a smoothed version of the membership condition. Thus considering that the defined function to represent the temperature over all three regions is formulated as:

$$f(p,h) = \begin{cases} f_{R_1}, & \text{if } f(p,h) \in R_1 \\ f_{R_4}, & \text{if } f(p,h) \in R_4 \\ f_{R_2}, & \text{if } f(p,h) \in R_2 \end{cases} \tag{4.14}$$

By using membership functions related to a region, equation (4.14) can be rewritten as follows:

$$f(p,h) = (1-w_1)f_{R_1} + w_2 f_{R_2} + w_1(1-w_2)f_{R_4} \tag{4.15}$$

where w_1 and w_2 are binary variables (0 or 1), which describe if the thermodynamic variable belongs or not to a particular region (see Figure 4.7). These variables are defined in this case as follows:

$$w_{1,2} = \begin{cases} w_1 = w_2 = 0, & \text{for } f(p,h) \in R_1 \\ w_1 = 1 \text{ and } w_2 = 0, & \text{for } f(p,h) \in R_4 \\ w_1 = w_2 = 1, & \text{for } f(p,h) \in R_2 \end{cases} \tag{4.16}$$

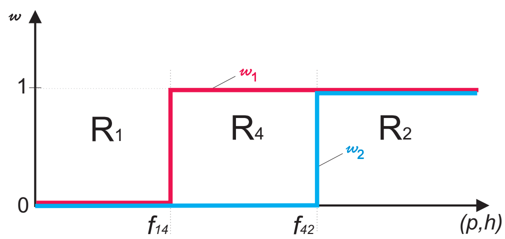
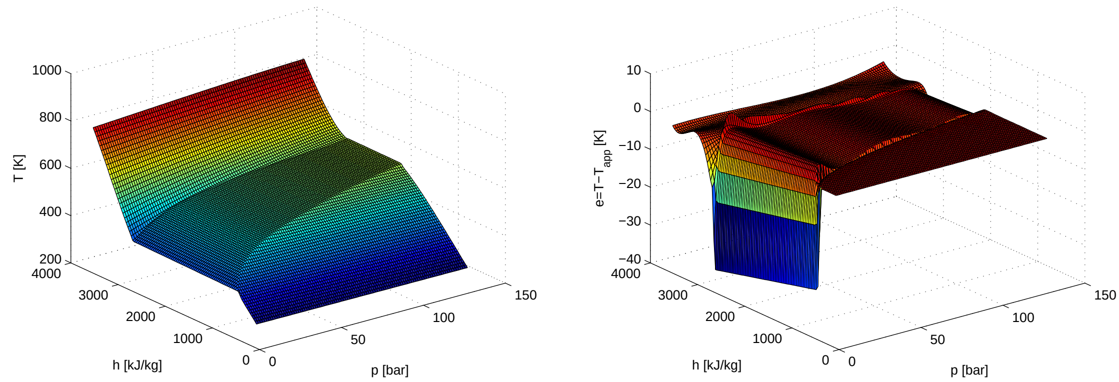


Figure 4.7: Membership functions to regions

Accordingly, the smooth approximation of the temperature function is of the form:

$$f_{app}(p,h) = (1-H_{k_{14}})f_{R_1} + H_{k_{42}}f_{R_2} + H_{k_{14}}(1-H_{k_{42}})f_{R_4} \tag{4.17}$$



(a) Temperature according to smooth approximation (b) Error representation ($e = T - T_{app}$)

Figure 4.8: Smooth approximation of the temperature function and error evaluation

with $H_{k_{14}}$ and $H_{k_{42}}$ the smooth approximations described above. The graphical representation of the function is illustrated in Figure 4.8a

Generally, by adopting this type of approach, the mathematical models obtained with the fitting tools, are quite accurate, providing a good fit (R-square ≈ 1, RMSE close to 0). Also, the introduced errors by the smooth connection between the regions are quite small, resulting in a good approximation (see Figure 4.8b). The exception is the approximation in the first part of the variation interval ([1-5 bar]), where the errors are quite significant (around 35 K). These high errors derive from the approximation of the function for the region **4**, f_{R_4}. A possible solution, in order to reduce these errors and to improve the algorithm fit, is the refinement of the approximation function, by considering additional partitions of the regions into subregions. Such a method has been addressed in the case of the functions that provide poor performance over entire domain of variation (e.g., the density as function of pressure and temperature has been partitioned in 5 sections). For the functions that have high errors only in the interval [1-5 bar], the obtained results have been considered as being in general satisfactory for the intended goals, and therefore an additional refining of the approximation was not necessary. Moreover, the variation domain of the considered Modelica model in the case of a hot start-up sequence is higher than this range. As it will be seen in the validation phase, the approach described in this section enables to obtain a medium model quite consistent compared to the original Modelica one.

4.3.2 ***ThermoOpt*** library

In the modeling field, when a model or a library is designed, one of the first decision to be taken is to specify the modeling purposes. As previously pointed out, the elements of the *ThermoPower* library are primarily intended for simulation and are not designed with the optimization purposes in mind. A new library, so-called *ThermoOpt*, has been developed in this direction.

4.3.2.1 Library purposes

In order to provide model components that can be used for model-based control applications and in the same time that reflect as much as possible the behavior of the real system, the *ThermoPower* library's components have been expanded and transformed. In other words, for each component of the *ThermoPower* library, corresponding to the reference model, a version in which the discontinuities have been reformulated by using smooth approximations has been developed. The idea has been to transform the existing models to be suitable for intended purposes and not re-create new models and library components. This enables, on one side, to keep the same structure of the model and on the other side, the key aspect, to obtain a model suitable for optimization in terms of issues such as complexity, accuracy and smoothness. Then these modified components have been included in *ThermoOpt* library, with a precise purpose, to be able to model, simulate, and optimize the control of a specific power plant. As in the *ThermoPower*'s case, one of the principal objectives of the *ThermoOpt* library is to provide a base framework with common model parts, offering the possibility to the user to extend and adapt the library's elements for a particular application.

Thus, by providing a model library suitable for optimization, even if it does not have a general applicability, being limited to the CCPP start-up case at the time being, it can be considered as an excellent way to package the modeling knowledge in a form that can be further extended by other users with similar purposes.

In this context, the challenge was to find the best trade-off between the simplification/approximation of the model elements and their accuracy with respect to the original representations. What should the library cover and what should be excluded from the library? These decisions affect the elaboration of model components since the design phase. In fact the library *ThermoOpt* aims to provide a basic framework for the modeling of the power plants elements, covering the engineering needs for an intuitive description format and the requirements of the efficient optimization algorithms.

4.3.2.2 Library principles

The *ThermoOpt* library closely follows the modeling principles adopted in *ThermoPower*. In the following sections, the main features of the model library *ThermoOpt* as well as the differences compared to *ThermoPower* are outlined.

Library structure

The *ThermoOpt* library is structured into 9 packages: 7 main packages and 2 auxiliary packages. Structuring of the packages in *ThermoOpt* is generally performed according to two principles: an *object* decomposition and a *subject* decomposition, respectively. The *object* decomposition refers to the system subdivision into its physical parts, e.g., turbines, HRSG. In *ThermoOpt*, this partitioning is used in the package *CCPP* where the main units of the plant are included: GT, HRSG and ST. This subdivision facilitates the user's work that can easily find unit models in the library. The *subject* decomposition deals with the inheritance structure, that in our case is sparingly used to avoid the potential problems linked to the inability of the optimization tools to address the inheritance mechanism. In turn each component, according to the working fluid on which it is based, is grouped in two packages: a package *Water* that provides models of components using water/steam as working fluid, and a package *Gas* that provides models of components using ideal gas mixtures.

The medium models that concern the fluid properties computation are included in the *Media* package. This package contains two sub-packages: *Water* and *FlueGasSingle*. The *Water* package provides a water/steam property model built according to the smoothness requirements, while the *FlueGasSingle* package contains a medium model for a typical gas turbine exhaust, also adapted to the optimization specifications. It should be noticed that in the case of the medium models, the modeling options for the generality of components have been reduced, thus, the general template provided by *Modelica.Media* library has been reduced to a specialized one that includes the basic elements.

The thermal and electrical components are included in two packages: *Thermal* and *Electrical*. In the *Thermal* package the basic modeling blocks to describe heat transfer phenomena are included. The *Electrical* package provides highly idealized models of electric generators and power grid.

The library contains also two auxiliary packages: a package *Test* that provides a few test cases for *ThermoOpt* models and a package *Sequences* with shutdown and start-up sequences used to validate or initialize the developed CCPP model. The different functions used to describe the behavior of subsystems (e.g., for valve characteristics) are assembled in the *Functions* package. The overall structure of the *ThermoOpt* library is given in Appendix D.

Library interface

In the *ThermoOpt* library, the same standard interfaces as in the *ThermoPower*'s case are defined. The connectors structure are designed to be completely independent of the modeling assumptions adopted for the representation of the model components (e.g., the same connectors are used no matter if the fluid is described by a model with one- or two-phase flow, if the momentum balance is static or dynamic).

```
//A-type flange connector
connector FlangeA
       Pressure p;
  flow MassFlowRate w;
  input SpecificEnthalpy hBA;
  output SpecificEnthalpy hAB;
end FlangeA;
```

Listing 4.1: Liquid flow connector type A

```
//B-type flange connector
connector FlangeB
       Pressure p;
  flow MassFlowRate w;
  input SpecificEnthalpy hAB;
  output SpecificEnthalpy hBA;
end FlangeB;
```

Listing 4.2: Liquid flow connector type B

Despite the fact that the same type of connectors is used in the both libraries, the *ThermoOpt* does not handle reversing flows, unlike the *ThermoPower* where a support for this is offered. The reason is quite obvious and relates to the definition way of the connectors, based on the conditional equations. To clarify, a trivial model is given as example.

Consider the definition of the thermodynamic connectors used in the *ThermoPower* library to describe the flanges of the components that carry a water/steam flow, presented in Listings 4.1 and 4.2.

In Listings 4.1 and 4.2, p is the fluid pressure, representing the *potential* variable, w is the mass flow rate entering the component, representing the *flow* variable, h_{AB} and h_{BA} are the specific enthalpies of the fluid corresponding to the flow direction from a flange of A-type to a flange of B-type and vice versa respectively. Although the two types of connector are formally equivalent, the basic rule when two components are connected is that the connection has always to be made between two flanges of complementary type. These types of connectors enable to deal with reversing flows in a quite simple way. This characteristic is illustrated in Listing 4.3 by means of a simple model describing a vessel with a constant quantity of liquid.

The reversing flow support in the models is specified by considering two conditional expressions, thus, the example equations give $h_i = Inl.h_{BA}$ and $h_o = h$, when the mass flow is positive, and $h_i = h$ and $h_o = Out.h_{AB}$, when the mass flow is negative. These expressions can introduce discontinuities, making the optimization problem to be non-smooth. Consequently, in *ThermoOpt* the conditional expressions have been eliminated from the library models, resulting in model components without reversing flow support (see Listing 4.4).

```
model SimpleV "ThermoPower version"
  FlangeA Inl;
  FlangeB Out;
 parameter Mass Ml=10 "Fluid mass";
  Pressure p;
  SpecificEnthalpy h "of Fluid";
  SpecificEnthalpy hi "at Inlet";
  SpecificEnthalpy ho "at Outlet";
equation
 //Mass balance
 0 = Inl.w + Out.w;
 //Energy balance
 Ml*der(h) = Inl.w*hi + Out.w*ho;

 //Boundary conditions
 hi = if Inl.w >= 0 then Inl.hBA
                    else h;
 ho = if Out.w >= 0 then Out.hAB
                    else h;
 Inl.hAB = h;
 Out.hBA = h;
 Inl.p = p;
 Out.p = p;
end SimpleV;
```

Listing 4.3: Reversing flow example

```
model SimpleV "ThermoOpt version"
  FlangeA Inl;
  FlangeB Out;
 parameter Mass Ml=10 "Fluid mass";
  Pressure p;
  SpecificEnthalpy h "of Fluid";
  SpecificEnthalpy hi "at Inlet";
  SpecificEnthalpy ho "at Outlet";
equation
 //Mass balance
 0 = Inl.w + Out.w;
 //Energy balance
 Ml*der(h) = Inl.w*hi + Out.w*ho;

 //Boundary conditions
 hi = Inl.hBA;

 ho = h;

 Inl.hAB = h;
 Out.hBA = h;
 Inl.p = p;
 Out.p = p;
end SimpleV;
```

Listing 4.4: Non-reversing flow example

In general, when models are supposed to not support reversing flows, additional information in connectors (such as specification of the inlet/outlet specific enthalpies) are not required. Although, to keep the compatibility between the library's elements, the structure of the connectors in *ThermoOpt* remains the same as in the *ThemoPower* library. In this case, the flow direction in components is assumed to be always positive. This assumption limits the model components applicability, but it seems to be reasonable for the start-up sequence, being validated by several experimental simulations. However, for extending components which take into account the reversing flow, the same type of approach based on smooth approximation of the Heaviside function could be applied. The considered case does not involve reversing flow, but this feature can be included in a later version.

```
//Distributed Heat Terminal
connector DHT;
 parameter Integer N=2 "Number of nodes";
           Temperature T[N] "Temperature at the nodes";
      flow HeatFlux phi[N] "Heat flux at the nodes";
end DHT;
```

Listing 4.5: Heat flow connector

Regarding the heat flow interfaces, as in *ThemoPower*, the connectors are characterized by a number of nodes uniformly distributed, containing the nodal values of the involved quantities (temperature and heat flux). Their structure remains unchanged in the *ThermoOpt* library (see Listing 4.5).

Library medium property computation

In the *ThermoOpt* library, the medium models for water, steam, and ideal gas mixtures are based on the smooth approximation of the standard substance property representations provided by *Modelica.Media* library. The new medium models are suitable for an efficient optimization and offer a satisfactory level of accuracy to compute the thermodynamic properties of the fluids that are flowing in the combined cycle system.

Library level of details

In *ThermoPower*, the library models provide a flexible level of details. This enables, depending on the situation and/or the modeling purposes, to represent the same component or part of the process with a different level of detail, which is then fully interchangeable. In order to avoid an excessive proliferation of models in library, the level of detail and accuracy of the model components is customizable by users. The user has different modeling options, for example, to include the dynamic momentum term in a pipe model, or to specify the friction coefficient (neglected, constant, variable). For this purpose, conditional equations have been used.

In *ThermoOpt* the modeling options and implicitly the selection conditions for the generality of components have been eliminated and particular models for each modeling option have been developed. This kind of approach is quite rigid and increases considerably the number of models in library, moreover, these conditions do not change over the time span of a simulation and do not represent a source of discontinuities, being usually eliminated during the compilation phase of the model. Nevertheless, the choice has been motivated by the fact that a large part of the actual optimization tools have a series of limitations, particularly in dealing with the conditional expressions.

The modeling assumptions adopted in the optimization library match those from the simulation one, the difference is related to the representation of the models. Thus, in *ThermoOpt* the components of interest from the *ThermoPower* library, in particular those components corresponding to the developed CCPP model, have been implemented in a form that is suitable for optimization, by means of smooth approximation procedure presented above in this chapter. It should be noticed that unlike the *ThermoPower* that provides a wide range of model components for power plants modeling, the *ThermoOpt*'s models range is narrower, including only the elementary components necessary to describe a combined cycle model. A few examples of model components implemented as Modelica code are reported in Appendix E. The library elements have been validated against the data obtained by the *ThermoPower* models. The validation procedure will be presented in the sequel.

4.3.3 Library models validation

The validation of the optimization library elements has been performed by comparing their responses, to the variation of the input signals, with the ones obtained by the *ThermoPower*'s models. The validation procedure presented in this section strictly refers to demonstrate the consistency of *ThermoOpt*'s components with respect to the developed CCPP simulator. Thus, the procedure consists in a systematic replacement of the original elements of the CCPP simulator based on *ThermoPower* with the ones provided by *ThermoOpt* library; then the simulation scenarios are separately applied on the new CCPP "test-model" as well as on the reference simulator, and the so obtained evolution of the main parameters are graphically compared.

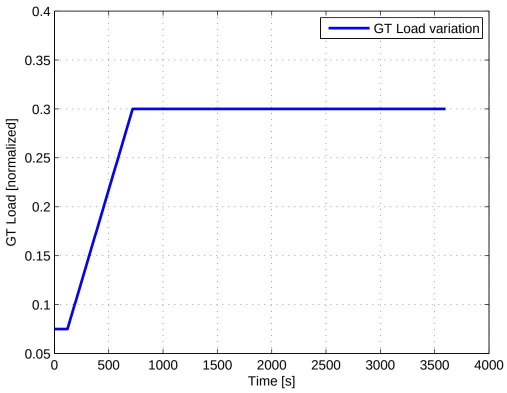
The reliability of the library's components has been investigated through simulation by means of several typical scenarios for the start-up procedure. The main two validation tests and the results of these tests are presented in the following.

Scenario I

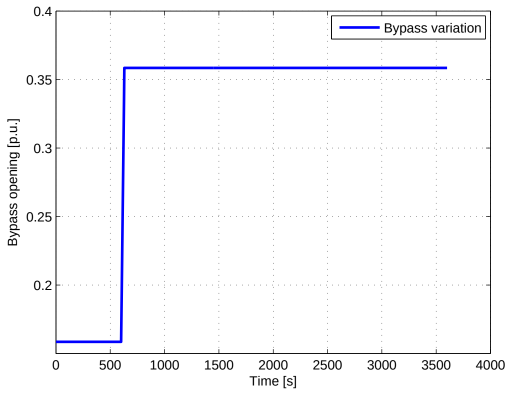
The first scenario refers to a ramp variation of the GT load and of the bypass position. The GT load is gradually increased until 30% of maximal threshold, following a ramp with the slope of 2.25%/min, applied at 120 seconds. According to a ramp variation profile with a slope of 40%/min, the bypass valve is opened up to about 36 %, at 600 seconds. The variations applied on the GT load and on the bypass position are presented in Figures 4.9a and 4.9b. The other signal inputs have been maintained constant and their values as well as the model initialization correspond to the hot start-up phase specified in the previous chapter (see Section 3.4.5).

Scenario II

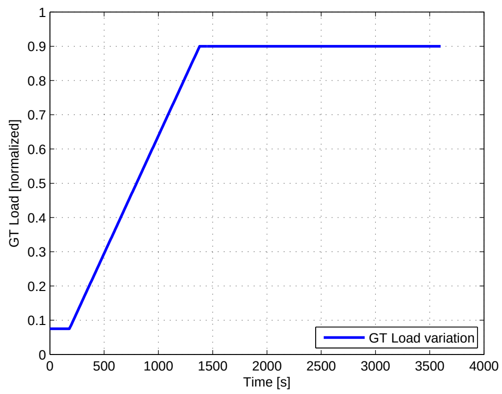
The second scenario deals with a gradual variation of the GT load, which is increased from 7.5% (the initial state of GT, corresponding to a power of 17.6 MW) until 90% of nominal load, following a ramp profile with a slope of approx. 4%/min, applied after 180 seconds. Also, a ramp variation profile with a slope of 60%/min is applied on the ST throttle valve after 600 seconds. The variations applied on the GT load and on the ST valve position are illustrated in Figures 4.9c and 4.9d. During the scenario, the ST is assumed to be connected to the grid and the other inputs are kept constant, corresponding to the hot start-up state.



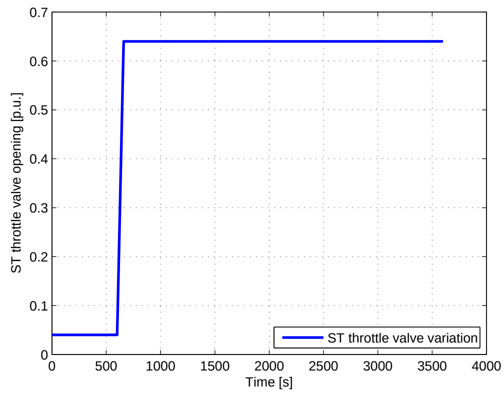
(a) Scenario I: GT Load ramp variation



(b) Scenario I: Bypass ramp variation



(c) Scenario II: GT Load ramp variation



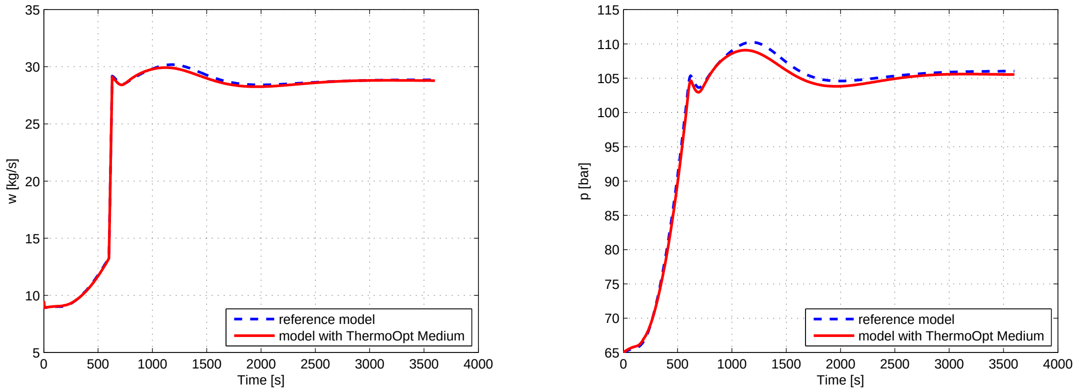
(d) Scenario II: ST throttle ramp variation

Figure 4.9: Scenarios I and II: Applied variations of the input variables

4.3.3.1 Medium models validation

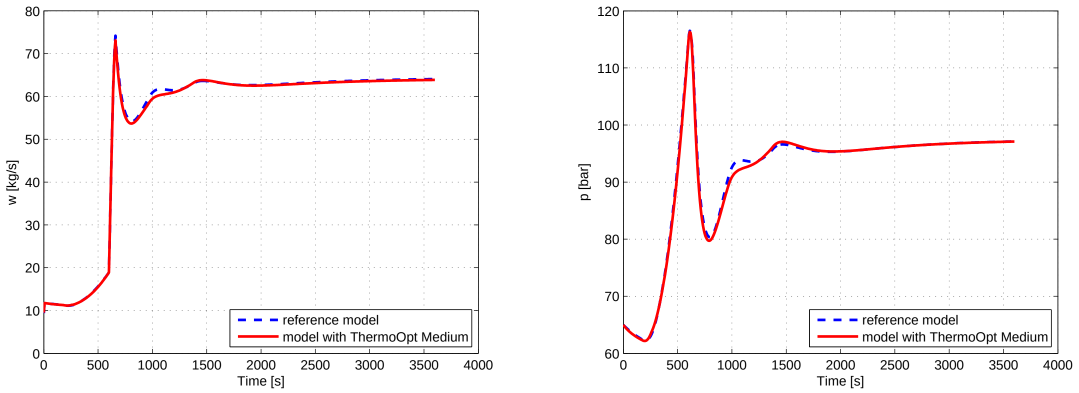
To validate the simplified/approximated models for the fluid properties calculation implemented in the *ThermoOpt* library, a new version of the CCPP reference model has been derived. This new model has been obtained by replacing the standard Medium models with approximated ones, keeping the original components. Then, the validation scenarios presented above have been applied on both models (CCPP reference model and CCPP model based on the *ThermoOpt* Medium models), and their responses have been compared.

In general the parameters that have been investigated are the temperature, pressure and the mass flow rate of the steam. Following the applied scenarios, the steam flow rate through the circuit and the steam pressure are represented in Figures 4.10 and 4.11. As it can bee seen from these figures, the medium models approximation is quite consistent, providing an acceptable level of precision compared to the reference one. The relative errors associated to the considered parameters are in general lower than 2.5%.



(a) Steam flow rate (b) Steam pressure

Figure 4.10: Validation Scenario I: steam mass flow rate and steam pressure in HRSG

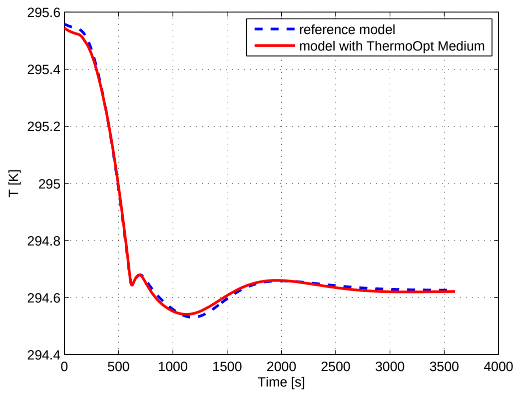


(a) Steam flow rate (b) Steam pressure

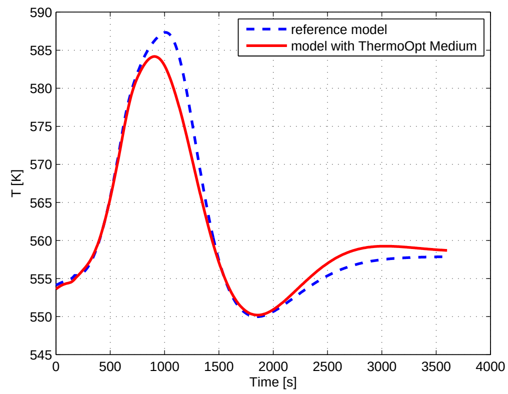
Figure 4.11: Validation Scenario II: steam mass flow rate and steam pressure in HRSG

In the context of the proposed validation scenarios, the steam temperature at the inlet and outlet of all the heat exchangers of the HRSG have been considered. In terms of steam temperature, the model performances are satisfactory, with small discrepancies in their dynamic behavior (see Figures 4.12 and 4.13). The difference is largely acceptable, with a maximum relative error that remains below 1%.

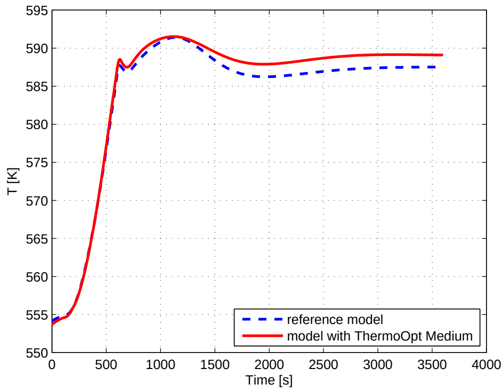
The slight differences which characterize the dynamic behavior of the models are most likely due to the accuracy of the approximated functions. In fact, as already pointed out,



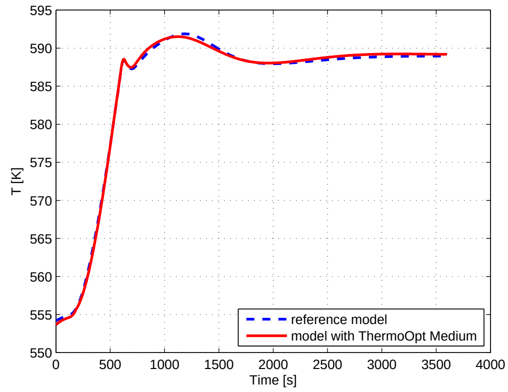
(a) ECO inlet steam temperature



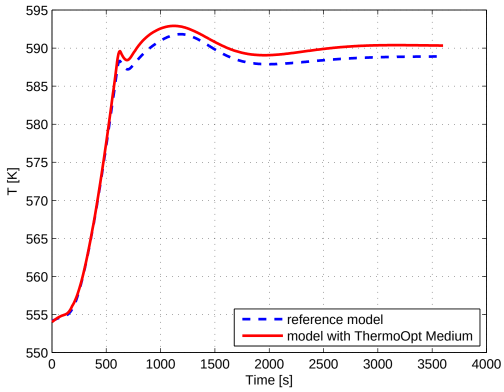
(b) ECO outlet steam temperature



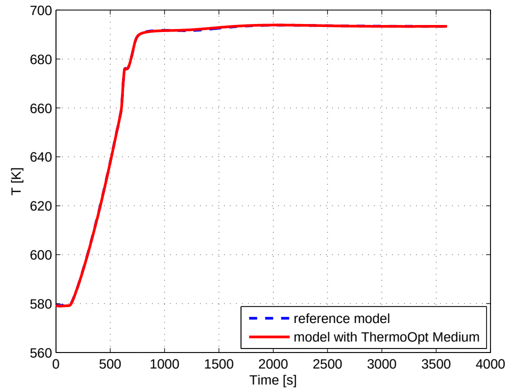
(c) EV inlet steam temperature



(d) EV outlet steam temperature

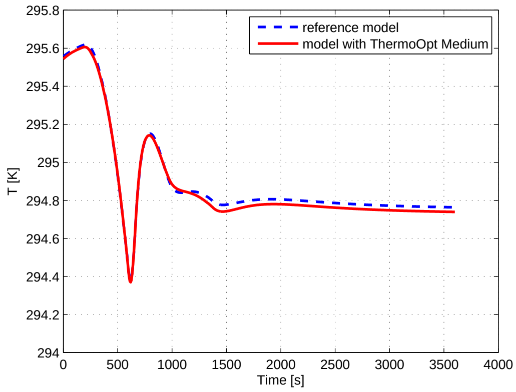


(e) SH inlet steam temperature

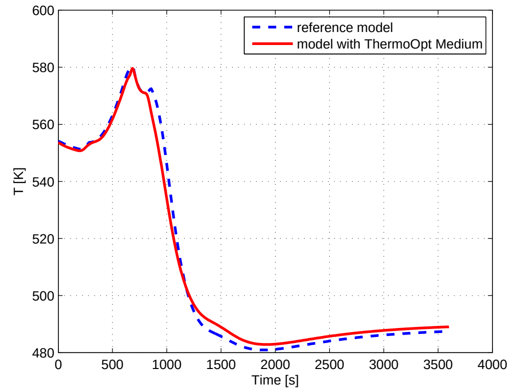


(f) SH outlet steam temperature

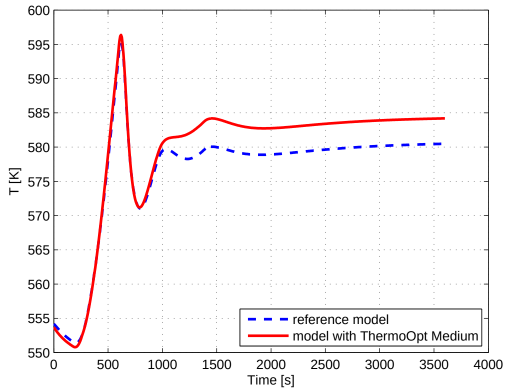
Figure 4.12: Validation Scenario I: steam temperature in HRSG



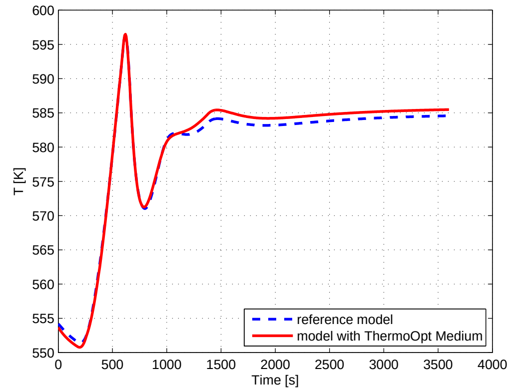
(a) ECO inlet steam temperature



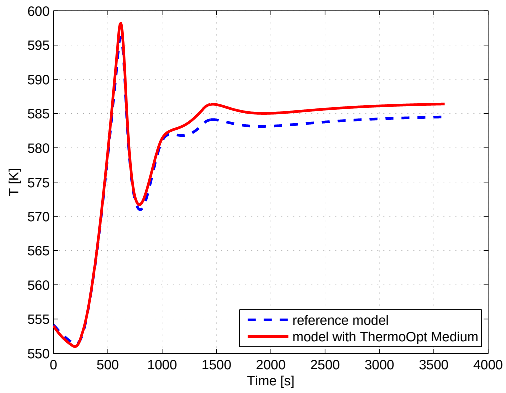
(b) ECO outlet steam temperature



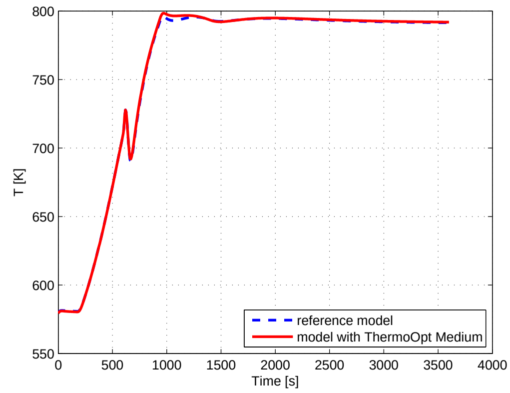
(c) EV inlet steam temperature



(d) EV outlet steam temperature



(e) SH inlet steam temperature



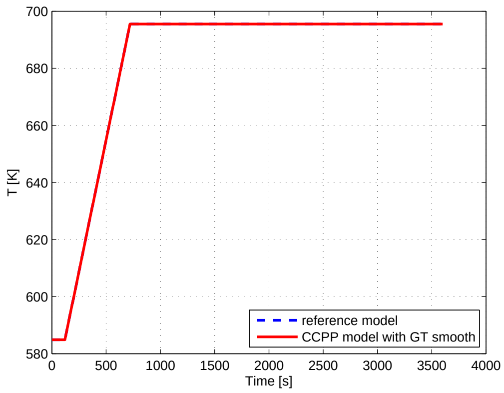
(f) SH outlet steam temperature

Figure 4.13: Validation Scenario II: steam temperature in HRSG

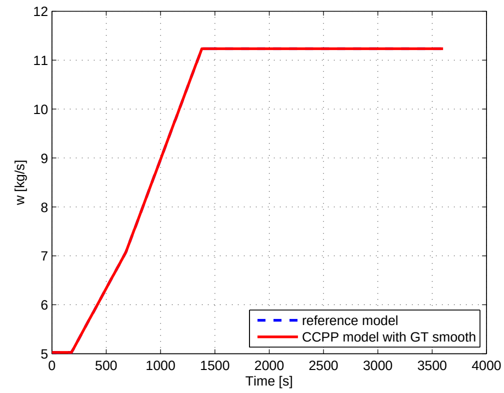
these errors could be reduced by improving the precision of the approximation. Despite these mismatches, the *ThermoOpt* medium is quite reliable, demonstrating a level of accuracy sufficient to be further used to develop optimization-oriented Modelica models.

4.3.4 Component models validation

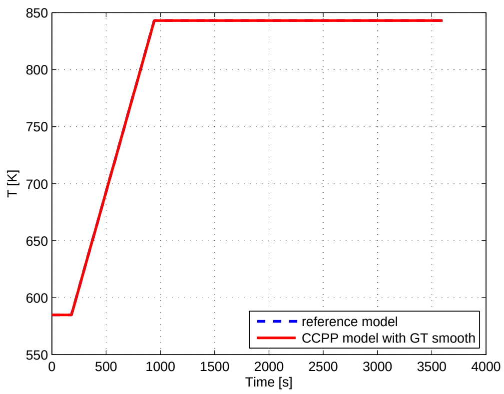
The validation of *ThermoOpt* elements has been carried out following the principle outlined above, i.e. to replace the original *ThermoPower* components of the plant model with their smooth version, and then to compare the dynamic behavior of the "new CCPP model" with the reference one. The procedure involves the testing and validation of each component. In the sequel, the presentation of the validation results is performed on subsystems. The main units of the plant model (i.e. GT, HRSG and ST) have been replaced by their smooth representations, using new medium representations, and tested.



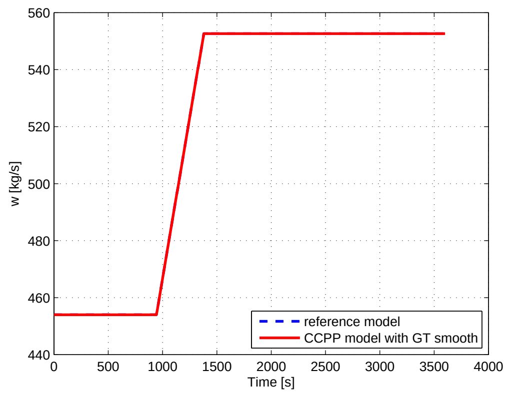
(a) Gas exhaust temperature scenario I



(b) Fuel flow rate scenario I



(c) Gas exhaust temperature scenario II



(d) Gas exhaust flow rate scenario II

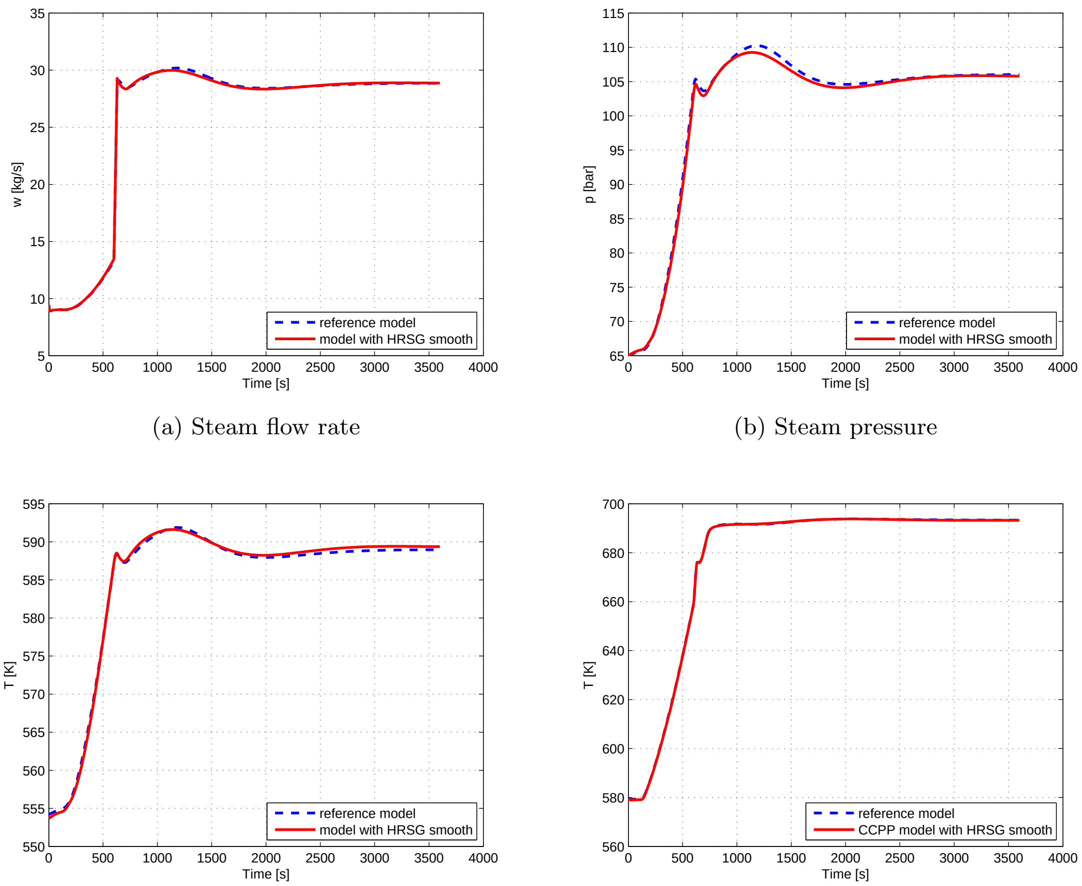
Figure 4.14: Validation scenarios: GT parameters

4.3.4.1 GT unit validation

The responses of the CCPP model with a smooth representation of the GT unit and the ones generated by the reference model, at the application of the test scenarios, are compared in Figure 4.14. As it can be seen the responses of the two models are similar. The adopted smooth approximation does not affect the behavior of the algebraic GT model.

4.3.4.2 HRSG unit validation

For the HRSG unit validation, the comparative responses of the models (the reference plant model and the CCPP representation containing the HRSG subsystem based on *ThermoOpt* components), to the validation scenarios, are presented in Figures 4.15 to 4.17.



(a) Steam flow rate

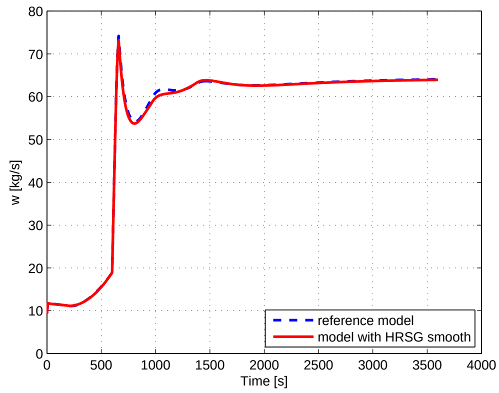
(b) Steam pressure

(c) EV outlet steam temperature

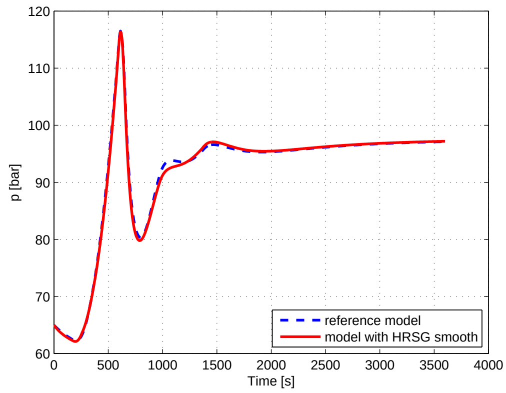
(d) SH outlet steam temperature

Figure 4.15: Validation Scenario I: steam parameters

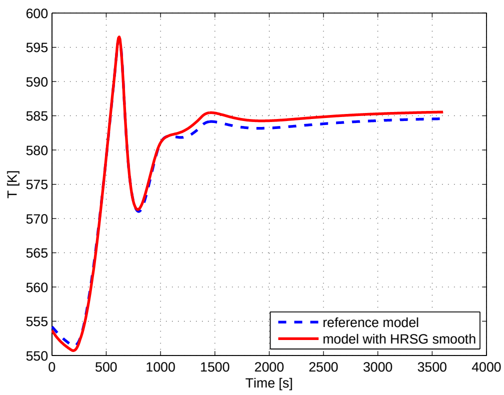
The reformulation of the original representations by using smooth approximation in general does not introduce significant errors, the dynamic behavior of the components remaining close to the reference one (see Figures 4.15 to 4.17). Also an important aspect is the fact that with the smooth model, the evaluation of the stress on the superheater outlet header is quite precise (Figures 4.17a and 4.17b).



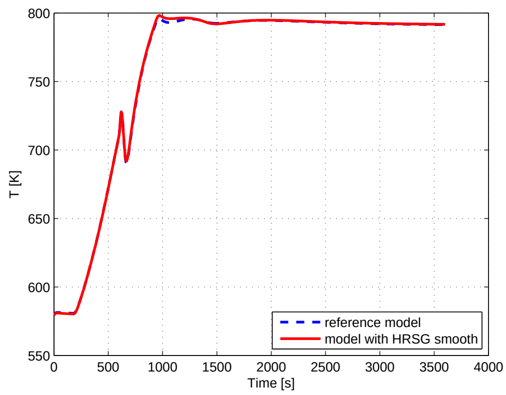
(a) Steam flow rate



(b) Steam pressure



(c) EV outlet steam temperature

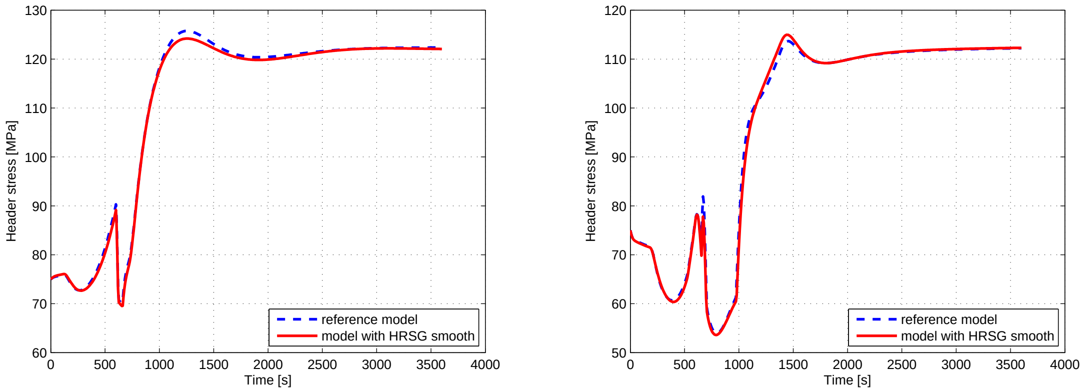


(d) SH outlet steam temperature

Figure 4.16: Validation Scenario II: steam parameters

4.3.4.3 ST unit validation

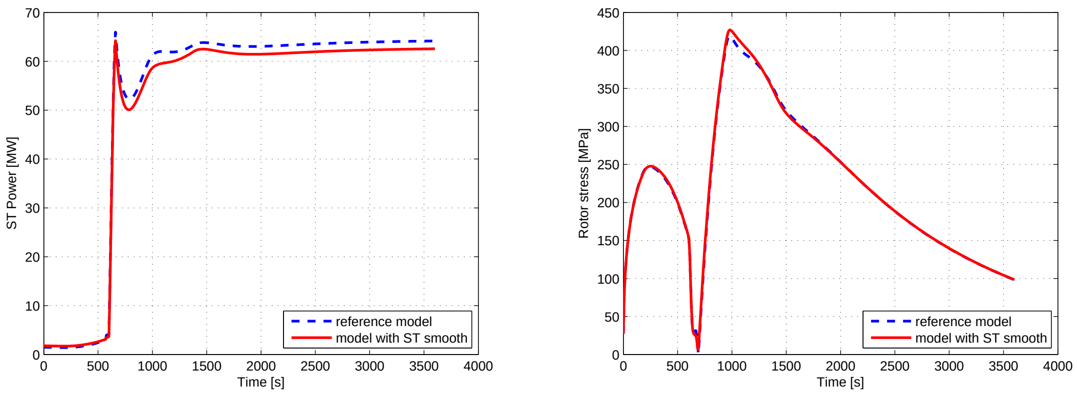
The consistency of the *ThermoOpt* components remains also valid in the case of the ST unit. As it can be seen from Figures 4.18a and 4.18b the dynamic evolution of the smooth model is close to the reference, providing an acceptable level of accuracy (the relative error lower than 2.5%).



(a) Header stress scenario I

(b) Header stress scenario II

Figure 4.17: Validation scenarios: header stress



(a) ST Power

(b) Rotor stress

Figure 4.18: ST power and rotor stress according to scenario II

Generally, the responses of the unit models developed using *ThermoOpt* components are close to the ones of the reference model based on *ThermoPower* library, demonstrating that the smooth approximations made to eliminate the discontinuity sources do not significantly influence their behavior. The existing differences between the models are mainly due to the precision of the approximated medium models, but their relative errors remain below an acceptable threshold of 3%, without compromising the models reliability.

4.3.5 Modelica CCPP smooth model

A smooth version of the typical CCPP model described in the previous chapter has been developed. The smooth model has been completely derived from the *ThermoOpt* library's components and has the same configuration and characteristics as the original combined cycle representation.

The plant model has been elaborated to be handled by numerical optimization methods, and the building procedure follows the object-oriented modeling principle: the system structure is decomposed into a certain number of components, then each component is described by means of differential-algebraic equations, and finally to form the complete model the components are connected via a-causal connection equations.

For the sake of completeness, the main units corresponding to the CCPP optimization model are here recalled:

- Gas Turbine;
- Heat Recovery Steam Generator;
- Steam Turbine;
- Condenser.

The components structure with Dymola diagram representation as well as the plant implementation as Dymola/Modelica model are given in Appendix F.

It is important to note that the plant model in this case does not include the representation of the steam line unit. Concerning the SL model, its main objective during the start-up phase is to avoid condensation (formation and migration) and to get steam quality necessary for the ST's operation. According to simulation results on the reference model, for the considered start-up conditions (initial state corresponding to a hot start-up phase), the level of condensation is quite low, and practically the SL is only a negligible transport delay between HRSG and the ST. Therefore, the SL component has been neglected in the CCPP model adapted for optimization purposes, being used only in the simulation phase to measure the quantity of condensation.

Model specification

The variables defined as input/output signals of the CCPP model for optimization correspond with those of the reference model. According to the statistic information provided by Dymola environment, a brief comparison between the two models is presented in Table 4.1. To keep the coherency of the comparison between the two built models (for simulation and for optimization), the smooth model has been reported to a version of the reference model also without the SL unit.

CCPP	**reference model** (without SL)	**optimization model**
Number of inputs	6	6
Number of outputs	11	11
Number of states	42	42
Number of equations	2405	1956
Simulation time[a]	4.16 [s]	0.5 [s]

[a] *DASSL solver, on a PC with 2GHz CPU*

Table 4.1: Comparative information between the CCPP models

The both models can be rapidly simulated, but nevertheless a significant improvement in the simulation time of the smooth model compared to the reference one (a time reduction of about 90%) can be observed. Therefore, only considering this aspect (simulation time), the effect of the applied transformations is quite important.

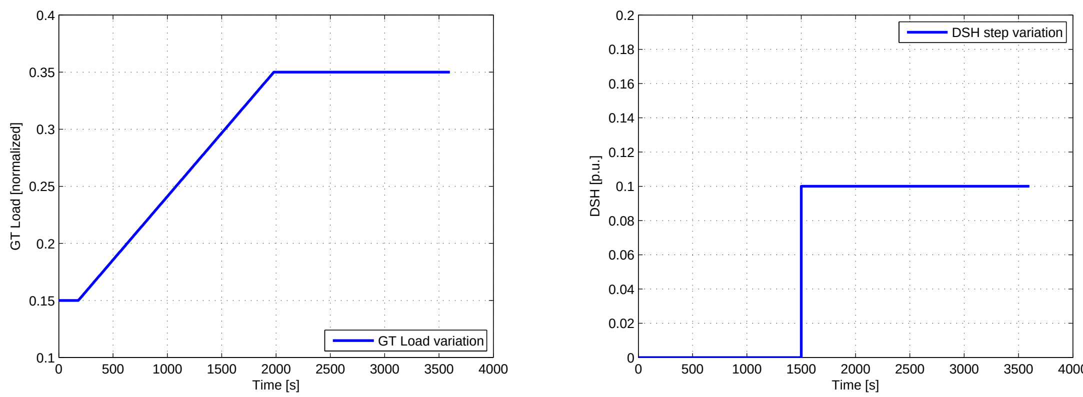
4.3.6 Model validation

As in the validation study of the *ThermoOpt* library, the reliability and consistency of the CCPP smooth model have been investigated by means of several scenarios judged as representative for the start-up phase by engineers. These test scenarios have been derived from a standard start-up procedure and can reflect real transient operating conditions of the plant. Also specific scenarios corresponding to the start-up stages have been considered in order to make a phase by phase validation.

The validation of the smooth CCPP model has been made by strictly comparing its responses to the variation of the input signals, with the ones obtained by the previously described CCPP simulator, considering the same initial conditions for both models. In this section, additionally to the validation scenarios previously presented, two other scenarios have been studied in order to get more confidence in the smooth model consistency: a simple scenario which involves the variation of the input signals, and an experimental test that replicates a classical start-up procedure.

Scenario III

This scenario corresponds to a step variation of the desuperheater flow rate and to a GT load profile that follows a ramp with a slope of approximately 0.65%/min, applied after 180 seconds (see Figures 4.19a). The step variation on the desuperheater flow rate



(a) GT Load ramp variation

(b) Desuperheater step variation

Figure 4.19: Scenario III: Applied variations of the input variables

has an amplitude of 10% and is applied at 1500 seconds (see Figures 4.19b). The initial conditions assumed in this case are:

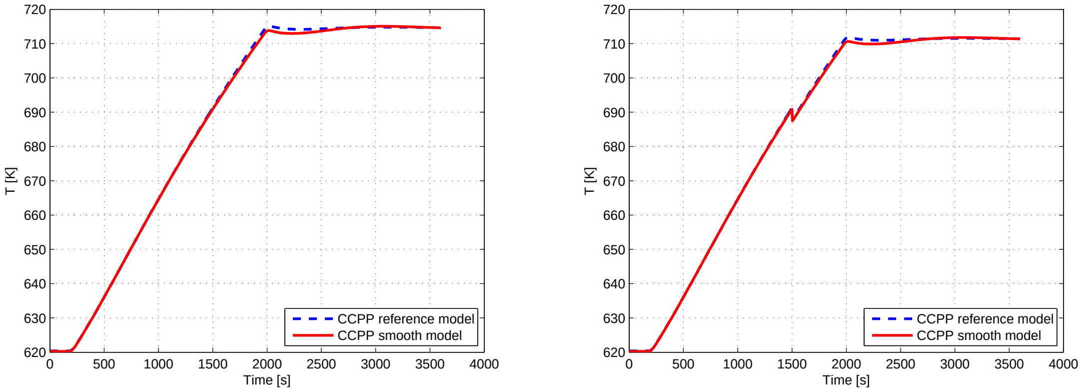
– the GT load is set at 15%;

– the bypass opening is at approx. 40%;

– the desuperheater water flow rate is close to its nominal value (0);

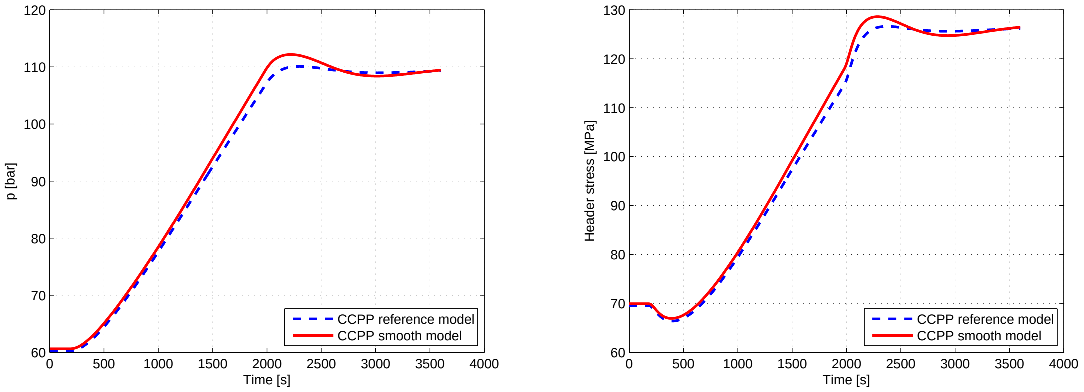
– the ST admission valve is closed;

These conditions can be associated with an intermediate part of the start-up sequence.



(a) SH outlet steam temperature (b) DSH outlet steam temperature

Figure 4.20: Validation Scenario III: steam temperature before and after DSH



(a) Steam pressure (b) Header stress

Figure 4.21: Validation Scenario III: steam pressure and header stress

The comparative results obtained from applying the scenario are shown in Figures 4.20 and 4.21. It can be observed that the smooth model responses are quite close to the reference ones. Relative to the original model, the steam temperature behavior in the smooth model is characterized by a slight difference (see Figure 4.20). In terms of steam pressure, the discrepancy between the two models is considered as acceptable, the relative error being below 2.5% (see Figure 4.21a). In general, the result of these differences, mainly between pressures, can also be seen in the evaluation of stress on the superheater outlet header (Figure 4.21a).

Scenario IV

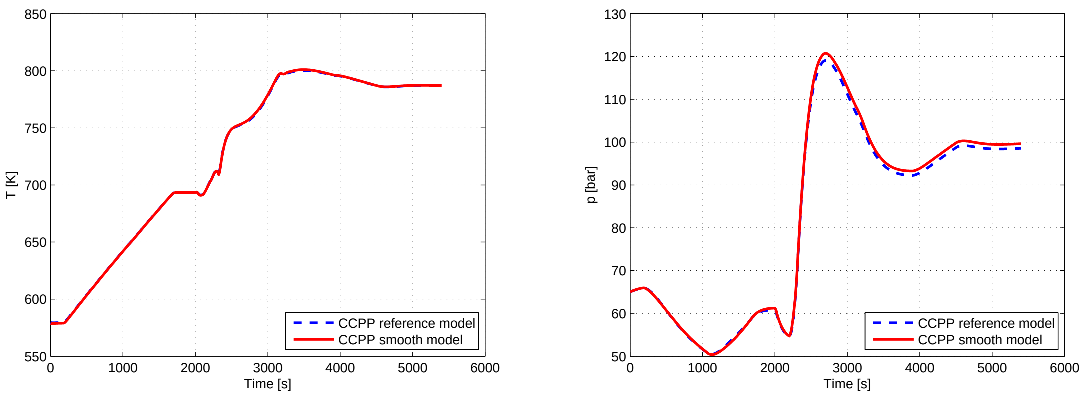
This scenario replicates a simplified start-up sequence. Considering the initial condition corresponding to the hot start-up phase, with GT synchronized and connected to the grid, the scenario follows the steps of the start-up sequence described in Section 2.2.5.6, namely:

1. *Temperature matching*: the HRSG metal is gradually warmed (during 1500 seconds) in order to limit the thermal stress. Using the bypass, the pressure in circuit is maintained close to the value required by the ST (60 bar);

2. *ST Start-up*: the ST admission starts after 2000 seconds when the steam conditions are fulfilled (i.e. steam temperature higher than ST metal temperature, steam pressure above 60 bar). At 2100 seconds, the ST is synchronized to the grid.

3. *Increasing load*: the bypass is closed (2300 seconds) and the ST valve is gradually opened (completely open at 3900 seconds). The GT and ST loads are increased until they reach their nominal values.

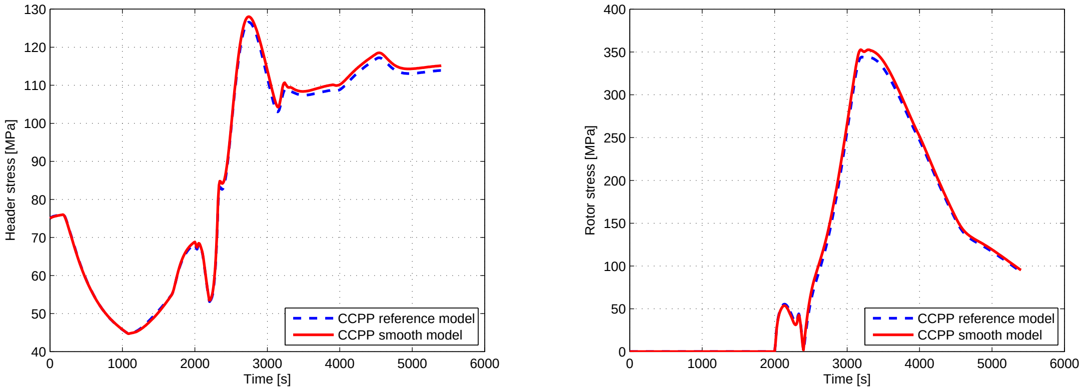
As in previous cases, the difference between the models is below the relative tolerance of 2.5% with respect to the main plant variables (see Figures 4.22 to 4.24).



(a) Steam temperature (b) Steam pressure

Figure 4.22: Validation Scenario IV: temperature and pressure of the steam

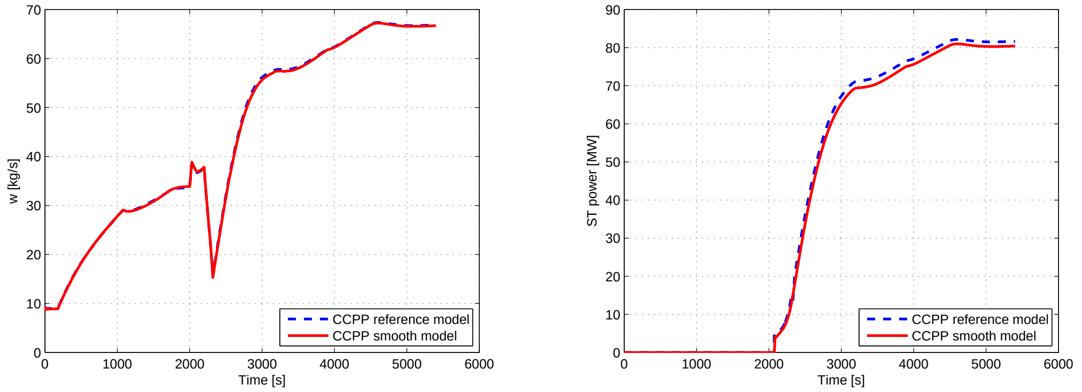
The results of the validation tests show the smooth model consistency and denote that it is accurate enough to be further used for control purposes. Although there are some small



(a) Header stress

(b) Rotor stress

Figure 4.23: Validation Scenario IV: header and rotor stress



(a) Steam flow rate

(b) ST power

Figure 4.24: Validation Scenario IV: Steam flow rate and ST power

differences in the parameters evolution, these model mismatches do not compromise the reliability of the model for intended purposes.

It is worth noting that validity of the model components with efficient optimization algorithms has been studied. As dynamic optimization platforms, JModelica.org [100] and DyOS (Dynamic Optimization Software) [119] have been used. These environments deal with Modelica-based models and use state of the art optimization algorithms.

Despite the fact that the developed model is suitable for optimization in terms of issues such as accuracy and smoothness, it appears that the available optimization tools are not mature enough[4] to cope with the complexity of the CCPP model. Nevertheless, in order to validate the smooth *ThermoOpt* components with nonlinear efficient algorithms,

[4]at the moment of this writing

the optimization problems based on the subsystems of the CCPP model have been solved with JModelica.org tool. The optimization problem formulation as well as the obtained results will be presented in the next chapter.

4.4 Conclusions

The approach adopted in this chapter had as objective the development of Modelica models which can be handled by efficient nonlinear optimization algorithms. In general such algorithms impose a series of restrictions on the model structure which limit their applicability. Therefore certain transformations in order to obtain Modelica models ready to be used for an effective optimization are required.

In this direction, a methodology that aims at transforming a physical simulation model suitable for description of a CCPP start-up behavior, to one adapted for model-based control applications has been proposed. The method enables to keep the same model structure as the reference one and to create models suitable for optimization with respect to complexity, accuracy and smoothness. The reformulation of the models structure is based on continuous approximations.

The modified components have been included in a new library, *ThermoOpt*. Even if the library has a limited operating range, its model components are suitable for optimization purposes, covering engineering needs for an intuitive description format of the power plants and efficient optimization algorithms requirements.

By using the new library components, a smooth model with the same configuration and characteristics as the original CCPP model has been derived. The library elements as well as the smooth CCPP model have been validated by simulation against the data provided by the original *ThemoPower* model. The validation results prove the smooth model consistency and denote that its level of accuracy is sufficient to be used for control purposes.

The proposed methodology is general and can be applied to other complex Modelica models in order to derive optimization-oriented models that comply with the requirements of the gradient-based methods. Moreover, the developed model components included in the *ThermoOpt* library can be used to design different types of plants, for example, a combined cycle power plant with two levels of pressure.

In this context, it is interesting to propose a solution to use such Modelica knowledge-based models in order to improve the CCPP start-up performances. The next chapter deals with an approach that derives the optimal trajectory profiles of the input signals based on the smooth CCPP model.

CHAPTER 5

Start-up optimization procedure

5.1 Introduction

In the current context, the increased production demand as well as the performance improvement to meet the tightening competition and new environmental policies involve the development of new and efficient management strategies for start-up phase. A very promising solution to deal with all these economic and environmental requirements is represented by dynamic optimization. In the case of CCPP start-up several strategies have been proposed in literature for optimization [74], [70], [73], [71] (see Section 2.2.5.8 for an overview).

To systematically address the challenge of applying dynamic optimizations, a model able to operate closely to the process limits is needed. In fact the control problem can be solved as a constrained minimization problem in order to reduce the start-up time while keeping constraints on the plant variables within their allowable limits. The optimization can also aim other objectives (e.g., minimization of material wear, minimization of operating costs, etc). In this study the optimization targets to minimize the start-up time. In the previous chapter, with aid of the advanced modeling language Modelica, an accurate combined cycle model appropriate for the start-up phase and that can be used for control purposes has been developed. By using this model, in this chapter a novel model-based approach is presented in order to optimize the profile of the control variables during the start-up phase. The developed CCPP model is used to derive the optimal profile, assuming that it can 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 main plant variables. Seeing from another point of view, the start-up problem can be posed as a "just-in-time" optimization problem, subject to the same constraints as in the minimum-time case.

The adopted solutions in this chapter are also motivated by the current difficulties to apply efficient numerical methods for the solution of optimization problem based on large-scale Modelica models.

5.1.1 Dynamic optimization

Two main methods for solving optimization problems with constraints in the form of dynamic systems date in literature since the late 1950s, namely, the dynamic programming and the maximum principle. The dynamic programming based on Bellman's principle of optimality [120], is an efficient method to derive optimal control policy as a function of the state variables. The method has been successfully used to solve a large number of problems from different fields. The Pontryagin's Maximum Principle [121] defines more general necessary conditions of optimality. These conditions form a two-point boundary value problem that has to be solved.

Nevertheless, both methods have practical drawbacks that make them difficult to apply to large-scale systems. For example, the difficulty to find the initial guess for adjoint variables in the case of maximum principle or so-called "curse of dimensionality", which is the limited ability of the dynamic programming to solve large-scale problems.

During the last decades, in order to overcome these difficulties, a new class of methods applicable to large-scale problems, referred as direct methods, has emerged. The basic idea behind these methods is to approximate/transcribe the original infinite dimensional problem into a finite dimensional nonlinear programming problem (NLP) by using a finite parameterization. The resulting NLP is then solved by an efficient optimization algorithm, e.g., Sequential Quadratic Programming (SQP).

All direct methods are based on a finite dimensional parameterization of the control trajectory, but differ in the way by which the state trajectory is handled. In general these methods can be separated in two main groups: sequential and simultaneous strategies [82], [83]. In the sequential approaches, only the control variables are parametrized. The parameterization is usually achieved by using piecewise polynomials and the optimization is performed with respect to the polynomial coefficients. The sequential methods rely on numerical integrators and NLP solvers. At each iteration, the dynamic model is integrated to evaluate the cost function and the constraints. The simulation and optimization iterations are sequentially proceed. Typically, the integrators are also able to provide the derivative information (e.g., gradients of the objective function with respect to the control parameters) to the NLP solver. In the simultaneous approach, both the state and the control variables are discretized by means of piecewise polynomials. The strategy generally requires a fine discretization of the states to obtain accurate solutions. Consequently, the resulting NLP is large, but also sparse. To solve the large-scale NLP problem efficient optimization strategies that explore the structure of the problem are required [122]. The most popular variants of the simultaneous method are: collocation on finite elements [123] and direct multiple shooting [124].

Nowadays, to solve constrained optimal control problems in real world applications, the direct methods are by far the most widespread and successfully applied techniques. These approaches represent effective solutions for large-scale optimization problems, but for the time being, their applicability with Modelica models, in particular with the developed combined cycle model, is rather limited. The main reason is the weak tool support for formulating and encoding optimization problems with dynamic constraints represented by Modelica models. Although several efforts have been performed in this sense [100], the available Modelica-based optimization tools are not yet able to deal with the complexity of the CCPP model.

5.1.2 Proposed solution

In view of the difficulty to use Modelica-based optimization tools, the type of approach adopted in this dissertation is directed towards CCPP start-up optimization and follows the principle of the sequential methods, specifically:

1. the control variables are parameterized ($u(t) = u(t, q)$)

2. for a given set of control parameters q_0 the system of equations describing the process is solved by an ODE/DAE solver

3. the solver evaluates the value of the cost function (J), which is used by NLP solver to find the optimal parameters (q^*)

4. the parameters are then updated by the optimization algorithm and the procedure is repeated until the convergence condition is reached

A schematic representation of the approach is illustrated in Figure 5.1. By addressing such a method, the infinite dimensional problem to find the optimal profile is transformed into a finite dimensional problem to find the optimal values of the parameters that describe the control profile. In the sequel, the control parameterization is achieved in different ways, in order to analyze and improve the start-up performances. The optimization

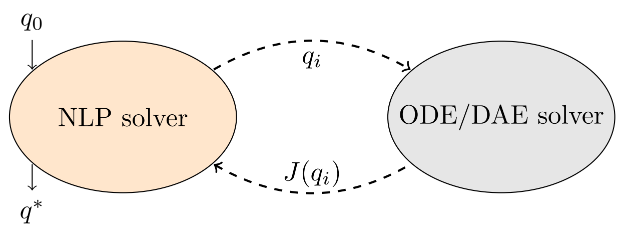


Figure 5.1: Schematic representation of the approach

procedure 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 problem has been solved by means of the Matlab nonlinear constrained optimization solver *fmincon* with active set method that uses as dynamic constraint the Modelica model. It should be pointed out that the gradients of the objective function with respect to the control parameters are not provided to the *fmincon* function, the solver estimates these gradients via finite differences.

All optimization results presented in this chapter were obtained using this procedure. Also, the application of available Modelica-based optimization tools has been tested. Due to the complexity of the process representation, the existing tools have proved unable to handle the entire structure of the derived CCPP model. However, for a comparative idea, optimization problems based on subsystems of the CCPP model have been formulated and solved using the proposed procedure and the platform JModelica.org for optimization of Modelica models [100]. JModelica.org provides efficient algorithms for solving dynamic optimization problems (e.g., algorithms based on the direct collocation method with the interior point optimization solver, IPOPT [125]).

Remark 5.1.1. *The optimization methods often require feasible initial values in order to ensure robust convergence. The initial values of the parameters in the optimization procedure have been derived from the profile corresponding to a classical start-up sequence applied on the CCPP model.*

5.2 Open-loop trajectory optimization

In general the traditional CCPP start-up sequence is quite conservative since in order to guarantee the safety and the availability of the operations, it imposes a series of limitations on the main control variables (e.g., on GT load rate). This type of sequence limits naturally the stress in thick components. In fact, the classical strategy is focused mainly on safety and availability rather than on efficiency of the operation. A new approach for the CCPP start-up process is proposed in this section, the idea is to implement a more flexible start-up strategy that eliminates as much as possible these limitations, without compromising the safe operation and keeping the lifetime consumption of the most stressed components under control.

The approach aims at deriving an optimal profile of the plant control variables (e.g., GT load, ST throttle valve position, DSH flow rate), by assuming that this profile can be described by a parameterized function. The parameters of this function are computed by solving either a minimum-time optimal control problem or an optimal quadratic tracking problem, subject to the plant dynamics and to constraints on the main plant variables, such as pressures, temperatures and stresses.

CCPP start-up is a complex task which includes different phases, involving several interconnected subsystems. In general, the optimization of the entire sequence requires consideration of a hybrid system in which discrete and continuous variables interact. To reduce the complexity of optimization problem of the whole start-up process, the entire problem can be decomposed into simpler sub-problems corresponding to each start-up phase. The idea is to divide the start-up sequence in successive steps which are separately optimized, by performing a phase by phase optimization.

Since the most critical part of the start-up sequence in terms of lifetime consumption (the peak values of stress are reached) involves the ST operation, the optimization problem formulation is focused on the last stage of the sequence, namely, the part after the starting of the steam admission in the ST. The applicability of the approach can be easily extended to other start-up phases.

The optimization procedure can be used to find an optimal trajectory for different system control variables (GT load, bypass, ST throttle, DSH water flow rate, etc.), but as the variable that induces most of the dynamic characteristics during the start-up procedure is the GT load, in particular in the last part of the sequence, it will represent the variable of interest for optimization. Besides the profile optimization of the GT load for a reduction of the start-up time, another study is also performed on the ST throttle for a better control of the thermo-mechanical rotor stress.

5.2.1 Gas Turbine load profile optimization

The goal of the optimization problem is mainly to find the GT load profile that minimizes the time required to reach the final point of the start-up (full load conditions) while fulfilling constraints on the process variables; typically, it is required that temperatures, pressures and stresses remain into specified limits for safety and availability of the operation. The posed problem corresponds to a minimum-time optimal control problem that will be presented below.

Another aspect derived from practice, where in order to provide electricity in time a planning of the plant operation is carried out, the start-up problem can be stated as an optimal quadratic tracking problem subject to constraints. In this type of problem the optimization horizon is fixed a priori, the objective being to reach the final point of the start-up (availability of the nominal operation) at the indicated time ("just-in-time" optimization problem).

5.2.1.1 Problem formulation

Start-up minimum-time problem

Consider that the CCPP nonlinear model developed in the previous chapter can be given into a dynamic explicit form:

$$\begin{aligned} \dot{x}(t) &= f(x(t), z(t), u(t), L(t), t, p) \\ 0 &= g(x(t), z(t), u(t), L(t), t, p) \end{aligned} \tag{5.1}$$

where $x(t)$ is the differential state profile vectors (temperatures, pressures, enthalpies, etc.), $z(t)$ denotes algebraic state profile vectors, $u(t)$ is the vector of input profiles which are not concerned by the optimization procedure (e.g., feed flow to the desuperheater, bypass valve, etc.), $L(t)$ represents the GT load profile (the control profile to optimize) and p is a time-independent parameter vector.

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

Assumption 5.2.1. *The control variable profile can be described by a parameterized function, $L(t, q)$, satisfying the boundary conditions stated above, with q a vector of unknown parameters that have to be derived through an optimization procedure.*

The optimal GT load profile determination consists in finding the value of the parameters vector, q, together with the final start-up time, t_f, as solutions of the following optimization problem:

$$\min_{q,t_f}(J = \int_{t_0}^{t_f} dt) \tag{5.2}$$

subject to the constraints

$$\begin{aligned} \dot{x}(t) &= f(x(t), z(t), u(t), L(t,q), t, p) \\ 0 &= g(x(t), z(t), u(t), L(t,q), t, p) \end{aligned} \tag{5.3}$$

$$L(t_f, q) \geq L_M - \epsilon_1 \tag{5.4}$$

$$|f(x(t_f), u(t_f), z(t_f), L(t_f, q)| \leq \epsilon_2 \tag{5.5}$$

$$c(x(t)) \leq 0, \quad t_0 \leq t \leq t_f \tag{5.6}$$

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 the final time t_f the system has reached the full load with an error specified by ϵ_1 (5.4) and is in stationary conditions with an error specified by ϵ_2 (5.5). The last constraint (5.6) includes, in a general form, all the constraints to be imposed on the plant variables during the start-up procedure. In the following, the main operational constraints included in the optimization procedure are on the level of stress in the critical components, namely, the superheater outlet header (5.6a) and the ST rotor (5.6b). It should be stated that additional constraints on the plant variables can be included (e.g., GT load rate of change, required power production, variation rate of the stress level, drum level), thus according to the intended objectives the procedure can be easily modified.

$$\sigma_{SH}(t) \leq \sigma_{SH}^{max} \tag{5.6a}$$

$$\sigma_{ST}(t) \leq \sigma_{ST}^{max} \tag{5.6b}$$

where σ_{SH} and σ_{ST} are the equivalent stresses on the header and on the external surface of the rotor; σ_{SH}^{max} and σ_{ST}^{max} are the peak values corresponding to admissible stresses on the header and on the rotor surface, respectively.

Start-up "just-in-time" optimization problem

Considering the dynamic system representation (5.1) and the same start-up conditions as in the previous minimum time problem, the optimal GT load profile is determined by finding the value of the parameters vector, q, as solution of the following optimization problem:

$$\min_{q}(J = \int_{t_0}^{t_f} (L(t,q) - L_M)^2 dt) \tag{5.7}$$

subject to the constraints (5.3) to (5.6). In this case an integral square error criterion (ISE) is used in order to drive the system towards the desired set point ($L(t,q) = L_M$) in the interval $[t_0, t_f]$, where the final start-up time, t_f is not an optimization parameters, being fixed a priori.

5.2.1.2 Parameterized functions

The variety of functions that can be selected to describe the GT load profile is quite wide. Based on the experience of the operators and analyzing the GT load transient during a standard start-up procedure, the GT load profile can be associated with a sigmoid function. In fact, the selection of the functions is important, because, as it will be seen in the following sections, the performed choice can lead to a more or less suboptimal solution of the original problem.

In the sequel, several types of functions have been considered, e.g., Hill functions, or a more general situation by using special functions defined piecewise by polynomials, in order to derive the optimal profile with the best compromise between execution time/convergence and start-up performance.

5.2.1.2.1 Parameterization based on Hill function

Hill function

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

$$L(t,q) = L_m + (L_M - L_m)\frac{t^h}{t^h + k^h} \tag{5.8}$$

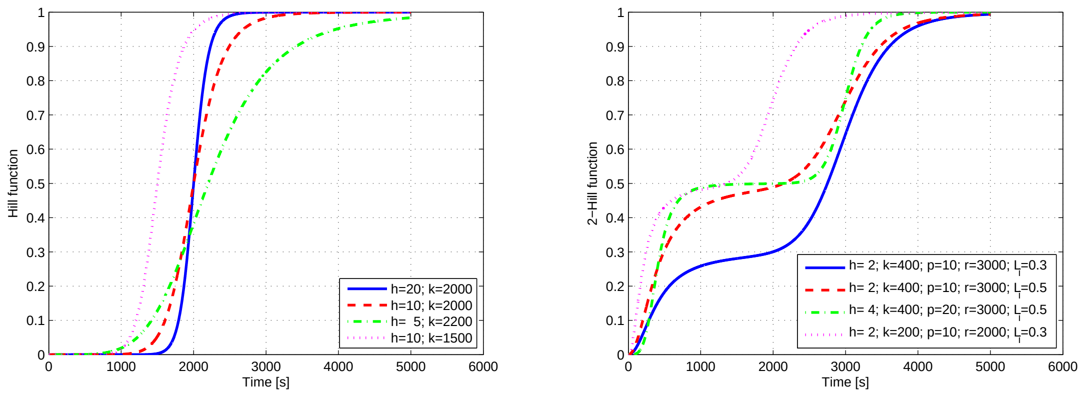
where $q = [h, k]$ is the parameter vector to be determined through optimization. Considering an example with the following start-up conditions, $L_m = 0$, $L_M = 1$, $t_0 = 0$ and $t_f = 5000$ [s], the graphical representation of the *Hill function* for different values of the parameters is shown in Figure 5.2a.

2-Hill function

The GT load profile can be also 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 formulated as follows:

$$L(t,q) = L_m + (L_i - L_m)\frac{t^h}{t^h + k^h} + (L_M - L_i)\frac{t^p}{t^p + r^p} \tag{5.9}$$

where $q = [h, k, p, r, L_i]$ is the parameter vector to be determined through the optimization procedure. The graphical representation of the function for different values of the parameters, considering the same start-up conditions as in the previous case, is shown in Figure 5.2b.



(a) *Hill function* representation (b) *2-Hill function* representation

Figure 5.2: Parameterized functions for GT load profile

5.2.1.2.2 Basis functions

In contrast to the previous section where functions with a predefined form have been used to represent the control profile, in this case the parameterization is made by means of basis functions. This choice involves a higher flexibility, increasing thus the probability to find the global optimum.

The range of basis functions used for control representation is quite rich, the most popular choice being as piecewise constant or piecewise linear functions. In the following the presentation is focused mainly on the use of common spline functions, where the polynomial pieces are joined together with explicit continuity conditions (Figure 5.3). This type of function enables to maintain the flexibility of piecewise polynomials while achieving a specific degree of global smoothness. In principle any polynomial interpolation method can be used to approximate the control profiles.

In CCPP start-up optimization problems studied in this work, in a first phase for control parametrization, linear splines have been used, and then the approach has been extended to quadratic functions in order to obtain a better approximation of the optimal solution.

5.2.1.2.2.1 Spline functions

Definition 5.2.1. *Let $[a, b] \subset \mathbb{R}$ a real interval, and let t_i, $i = \overline{1, N}$, be a set of knots with $t_1 = a$ and $t_N = b$. Denote by I_i the subintervals $[t_i, t_{i+1})$. A function $S : [a, b] \to \mathbb{R}$ is called polynomial function of order m if:*

1. *the restrictions of S to the subintervals I_i are polynomials of order m, $S|_{I_i} = p_{m,i}$;*
2. *S is differentiable to order $m-1$ on interval $[a, b]$, $S \in C^{(m-1)}[a, b]$.*

The second condition contains also the connection mechanism between the knots:

$$p^k_{m,i}(t_{i+1}) = p^k_{m,i+1}(t_{i+1}), \quad k = \overline{0, m-1} \tag{5.10}$$

meaning that at the border t_{i+1} between two subintervals, the polynomial on the left $p_{m,i}$ and its $m-1$ derivatives (denoted by k-order derivative) must have the same values as those of the polynomial on the right, $p_{m,i+1}$ (the function values of adjacent polynomials must be equal at knots).

Linear spline

In the case of linear splines, the piecewise functions in each interval $[t_i, t_{i+1}]$ are formally stated as:

$$p^k_{1,i}(t) = a_i(t - t_i) + b_i \tag{5.11}$$

where the parameters a_i, b_i are determined from C^0 and joining conditions.

Quadratic spline

The spline function consists of quadratic pieces, connected together with the C^0 and C^1 conditions satisfied. The corresponding representation of quadratic function for each data interval $[t_i, t_{i+1}]$ is:

$$p^k_{2,i}(t) = a_i(t - t_i)^2 + b_i(t - t_i) + c_i \tag{5.12}$$

where the parameters a_i, b_i, c_i are determined from C^0, C^1 conditions, adding an arbitrary condition to completely define the problem.

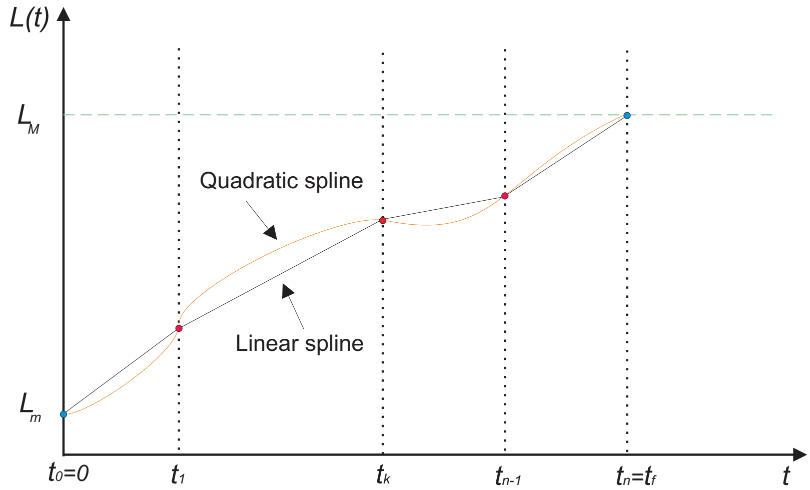


Figure 5.3: Approach based on spline functions

5.2.1.2.2.2 GT load spline parameterization

To parameterize the GT load control profile, a fixed time horizon is considered, namely, $[t_0, t_f^{max}]$, where t_f^{max} denotes the maximal start-up time. The determination of this value is derived by simulation, taking into account the fact that in practice the engineers dispose of a priori knowledge about the start-up time of the system to be optimized. More specifically, t_f^{max} is obtained by applying the control profiles corresponding to a classical start-up sequence with a constant loading on the CCPP model. Thus, the obtained value defines the length of the optimization time horizon, representing in the same time the reference time (maximal time) for our approach.

The approach is based on the following assumption:

Assumption 5.2.2. *The optimal time, t_f, will be always smaller than a reference time, t_f^{max}, given by a standard start-up procedure (i.e. $t_f \leq t_f^{max}$).*

Control parameterization with fixed partition length

Typically, the technique used for the control profile parameterization relies on the partitioning of the time horizon $[t_0, t_f^{max}]$ into N stages, $t_0 < t_1 < t_2 < ... < t_N = t_f^{max}$, each having a fixed length equal to t_f^{max}/N. In each interval, the control profile is represented by using piecewise polynomial functions.

The determination of the optimal GT load profile is performed by considering that the GT load profile is defined by a set of data points $(t_i,\ L_i)$, with $i = \overline{0, N}$, where L_i is the GT load value at the time instant t_i. These points are then interpolated by means of spline functions. Given 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 parameters vector to be optimized, in this case, is $q = [L_j]$, with $j = \overline{1, N-1}$.

In general, the grid density (number of subintervals N) plays an important role in the approximation of the optimal solution, since a too high number penalizes the computation time, and a too small number can lead to suboptimal solutions. This issue will be addressed more in detail in Section 5.3, where an analysis of the simulation results is given (see Figures 5.7a, 5.7b). Also, in the case of a small number of points, the selection of an appropriate piecewise function can influence the optimality of the solution.

Control parameterization with variable partition length

To improve the efficiency of the method, in particular when a low number of sub-partitions is chosen, the previous control parameterization technique has been expanded. The suggested idea is to include in the optimization problem the time instants when the partitioning is made. Thus, the time horizon $[t_0, t_f^{max}]$ is divided in N subintervals of variable lengths, which in turn are derived by the optimization procedure. In this case the parameters vector to be determined through the optimization procedure, q, additionally contains the time instants t_j, $j = \overline{1, N-1}$ (i.e. $q = [t_j, L_j]$).

Including these time instants in the optimization vector, together with an adequate choice of the parameterized function (e.g., quadratic spline), the profile flexibility can be increased, largely offsetting for the lack of an accurate discretization grid.

Remark 5.2.1. *In order to maintain the feasibility of the solution, an additional number* *of linear constraints on the parameters bounds have been imposed:*

$$L_m \leq L(t, q) \leq L_M \tag{5.13}$$

so that the GT load remains between the minimal and maximal load. In addition for the second case:

$$t_0 \leq t_j \leq t_f, \ j = \overline{1, N-1} \tag{5.14}$$

so that the parameters remain in the feasible domain.

5.2.2 GT load profile and ST throttle opening profile optimization

In the procedure outlined above, only the GT load has been used as optimization variable, the other control inputs of the system being considered as constants (e.g., DSH water flow rate), or represented as a predefined function of the GT load (e.g., ST throttle valve). In order to improve the start-up performances even further, the procedure can be easily extended to aim also other inputs, for example, in this section the ST throttle opening is included in the optimization procedure.

Problem formulation

Let *Assumption 5.2.1* hold for ST throttle valve. The ST valve opening profile is represented by using a parameterized function ($O_t(t, v)$), where v denotes the parameters vector. The start-up conditions are specified by $O(t_0) = O_m$ and $O(t_f) = O_M$, where O_m, O_M are the initial and final opening values. The optimization procedure follows the same principle as in previous case; the optimal profiles of the GT load and ST throttle are determined by solving a minimum time problem:

$$\min_{q,v,t_f} (J = \int_{t_0}^{t_f} dt) \tag{5.15}$$

subject to the constraints (5.4) to (5.6)

$$\begin{aligned} \dot{x}(t) &= f(x(t), z(t), u(t), L(t, q), O_t(t, v), t, p) \\ 0 &= g(x(t), z(t), u(t), L(t, q), O_t(t, v), t, p) \end{aligned} \tag{5.16}$$

$$O(t_f, v) \geq O_M - \epsilon_v \tag{5.17}$$

where ϵ_v denotes tightening term of the constraint (ideally equal to zero) and the corresponding constraint (5.17), is included to ensure that at the final start-up time t_f the ST valve is (almost) completely open.

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

5.3 Results

The approach presented in the previous section has been applied to a hot part of the start-up sequence. The part corresponds to the last stage of the sequence, after the steam admission begins and ST is synchronized to the grid (see also Section 2.2.5.6). Namely, the initial conditions of the model assumed in this case correspond to:

- the bypass valve is closed;
- the ST is synchronized and connected to grid;
- the feed-water flow rate is of approximatively 19 kg/s;
- the DSH feed-water flow rate is set at nominal value (0);
- the throttle valve is set to a value that ensures a pressure in circuit of 60 bar;
- the load of the GT is set to 15% ($L_m = 0.15$).

It should be noted that, regarding the ST valve, a loop that controls its position to maintain the pressure in circuit at a specified value (i.e. 60 bar) is used. The control loop is active only when the GT load is below 50%, above this value the opening directly depends on the GT load (sliding pressure mode).

The concerned stage of the start-up is considered as one of the most critical regarding the lifetime consumption of the system components. The optimization procedure has been implemented by imposing a number of constraints on the main plant variables, in order to prevent any possible unsafe conditions and to preserve the life of the units. In particular, the constraints are imposed on the maximum values of thermal-mechanical stresses in the most affected components, namely, the rotor and the superheater outlet header. These peaks determine in general the lifetime consumption during the entire start-up (i.e. the creep phenomenon depends mainly on the temporal peak stress), without considering additional effects caused by the variation rate of the stress level (i.e. the fatigue crack due to fluctuating stress). The peak values for the header stress and the rotor stress have been fixed to not exceed 116 MPa and 460 MPa, respectively (5.18c). These limits are consistent with typical values estimated on a real plant during the start-up phase and correspond to a normal start-up procedure.

As already stated, the objective of the optimization problem is to reach the final point of the start-up as fast as possible, while keeping the lifetime consumption of the most stressed components under control. The start-up sequence is considered as being completed when the GT load exceeds 99.3% ($\epsilon_1 = 0.7\%$) and the stationary conditions are reached with $\epsilon_2 = 1\%$, meaning that the gradient of the steam temperature is lower than 0.01 K/s and the pressure gradient is below 0.01 bar/s (5.18a-5.18b).

$$L(t_f, q) \geq 0.993 \tag{5.18a}$$

$$\begin{aligned} \dot{T}_s(t_f) &\leq 0.01\ K/s \\ \dot{p}_s(t_f) &\leq 1000\ Pa/s \end{aligned} \tag{5.18b}$$

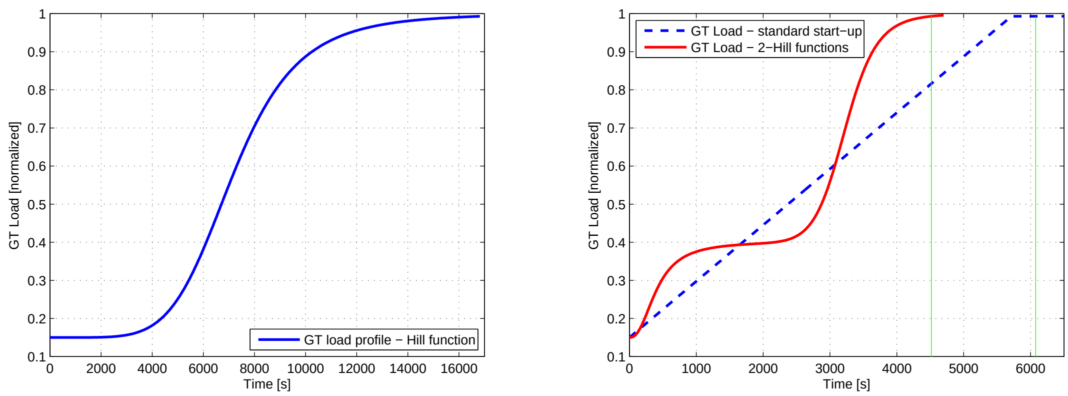
$$\begin{aligned} &\sigma_{SH}(t) \leq 116\ MPa \\ &\sigma_{ST}(t) \leq 460\ MPa \end{aligned} \tag{5.18c}$$

5.3.1 Comparison standard sequence - optimal procedure for minimal start-up time

In order to show the advantages of the proposed approach, based on the above initial conditions, a comparison between the performances in terms of start-up time and consumption of the optimal procedure with respect to a standard start-up sequence has been performed. The classical start-up procedure uses a conservative load rate limit to take into consideration the worst case plant conditions. Thus in our case, during the last part of the sequence the considered GT load ramp has a slope limited to 2 MW/min. This value corresponds to the constant GT load rate that ensures the fulfillment of the constraints imposed on the stress.

5.3.1.1 Hill functions vs. classical profile

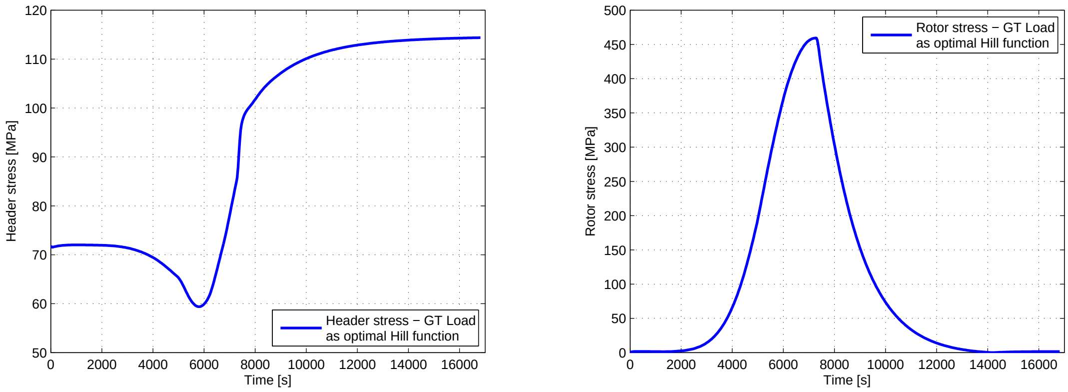
Assuming that GT load profile is described as a *Hill function*, the optimization procedure has been implemented, resulting in the GT load optimal profile shown in Figure 5.4a. As



(a) GT Load profile described as a *Hill function*　　(b) Standard and optimal GT Load profile

Figure 5.4: Start-up with an optimal profile based on Hill functions vs. standard procedure

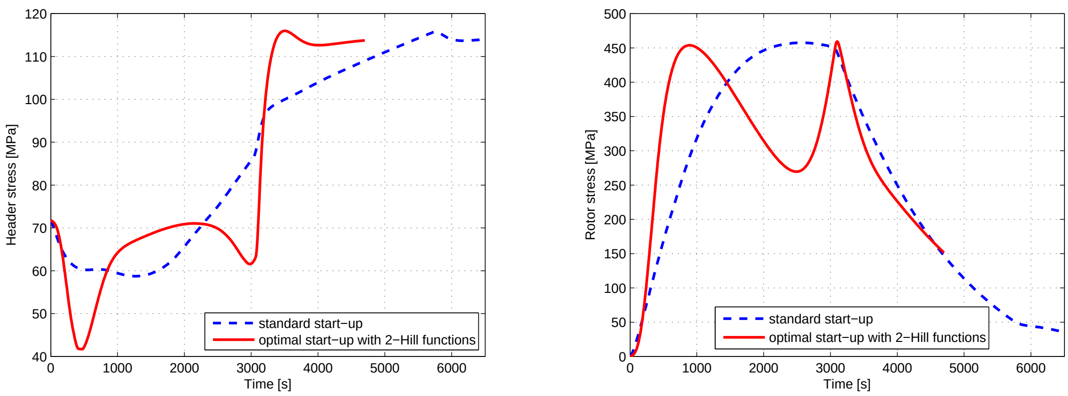
it can be seen, by using this type of functions, the optimization procedure can guarantee a safe operation (see Figure 5.5) but with a very suboptimal solution in terms of start-up time. Nevertheless, this simple choice to describe the input signal could be useful to optimize other start-up phases (e.g., HRSG Start-up phase), or for the concerned phase, by connecting two such functions in cascade. Thus, considering that the load profile is described by a *2-Hill function*, the start-up time can be significantly reduced with respect to the standard procedure (see Figure 5.4b).



(a) Header stress evaluation (b) Rotor stress evaluation

Figure 5.5: Stress evolution according to GT load profile based on *Hill function*

The optimization results show that the new procedure based on *2-Hill function* is able to reduce the start-up time compared to the classical one by approximately 26% (26 minutes), while limiting the peak stress value on the header (Figure 5.6a) and on the rotor surface (see Figure 5.6b). An optimization procedure relied on these functions can guarantee a fast and safe start-up operation.



(a) Header stress evaluation (b) Rotor stress evaluation

Figure 5.6: Stress evolution corresponding to the standard GT load profile and optimal profile based on *2-Hill function*

It should be noted that the simple selection of *2-Hill function* demonstrates adequate results for the last stage of start-up. However different input functions can be selected to improve further the start-up performances, taking into account the fact that a compromise between the start-up time performance and the complexity of the optimization, mainly linked to the number of parameters that define the function, has to be made.

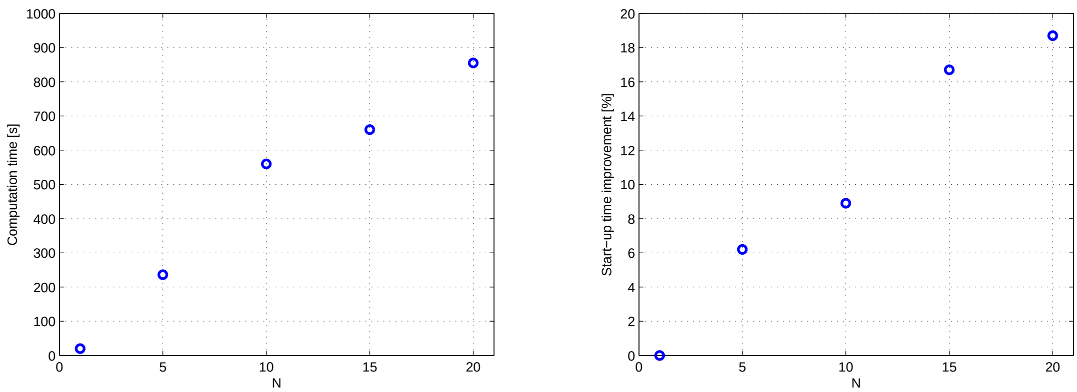
5.3.1.2 Spline functions application results

The application of more flexible parameterizations can lead to a better approximation of the control profiles, increasing the possibility to find the global optimum. The results of the optimization obtained by using spline functions to replicate the GT load profile are presented in Figure 5.8.

5.3.1.2.1 Linear spline

The typical choice of the direct methods to consider piecewise linear functions for the parameterization of the control trajectory has been tested. This technique allows to obtain GT load profiles that improve the start-up performances compared to the standard procedure, but the quality of the results depends on the number of data points used to describe the profile over the optimization horizon. Figure 5.8a illustrates the optimization results for different values of the number of points, N, over the time horizon of the optimal control problem, $[t_0, t_f]$. The efficiency of the derived profiles with this type of parameterization, related to the *2-Hill function*-based approach, is lower even for a significant number of points, i.e. $N = 20$.

Typically, a better approximation of the profile requires a finer discretization; on the other hand a large number of data points for the discretization grid increases the complexity of the optimization problem. In order to show how the grid density can influence the optimization results, a simulation analysis is presented in Figure 5.7. Given



(a) Influence of N on the computation time

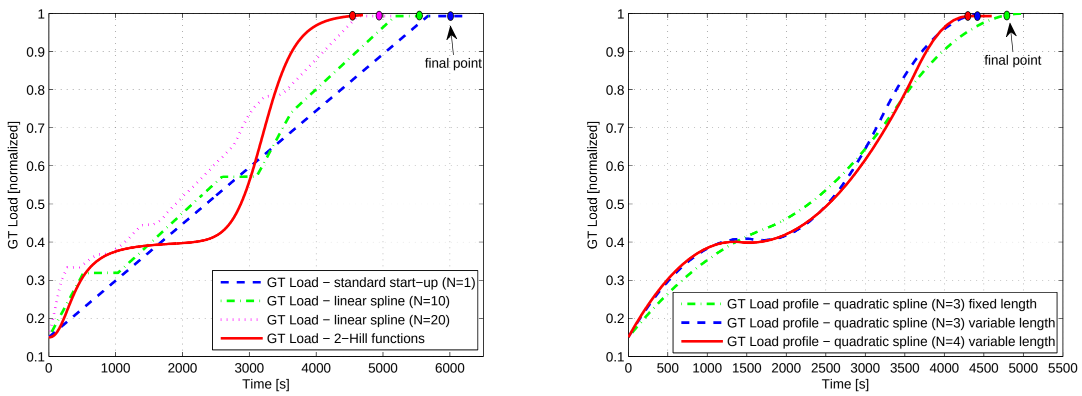
(b) Influence of N on the profile optimality

Figure 5.7: Influence of the parameter N on the optimization performances. Case study for a parameterization with piecewise linear functions of the GT load profile

the fact that the optimization algorithm operates without accurate derivative information, using numerical approximation of the gradients, the current approach provides low performances in terms of convergence properties and execution time when an increased number of optimization parameters is considered. Therefore, in order to make a trade-off between efficiency and convergence/computation time, the approach is generally limited to a reduced number of optimization parameters. Typically, according to the simulation experiments, a lower limit than 15 leads to a good performance. Beyond this limit the performance begins to degrade noticeably.

5.3.1.2.2 Quadratic spline

The use of the quadratic functions enables to obtain a smooth profile of the control variable, feature that can lead to the increase of the simulation efficiency. Moreover, their flexibility can significantly improve the start-up performances. The results show that the quadratic splines are more appropriate to represent the GT load profile when a reduced number of parameters is used for its description. For example, Figure 5.8b illustrates the GT load optimal trajectories corresponding to the set of points, $N = 3$ and $N = 4$. Moreover, the lack of a rich parameterization grid is substituted by the use of quadratic



(a) Optimal profiles using linear splines (b) Optimal profiles using quadratic splines

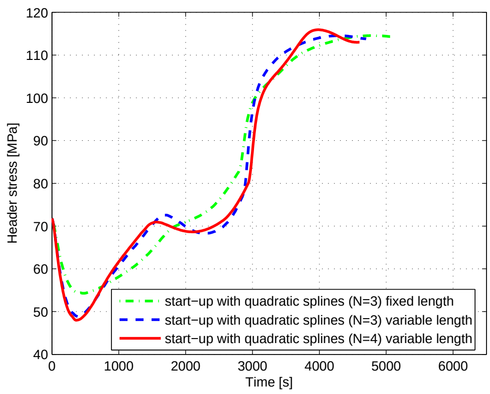
Figure 5.8: GT load profiles derived from optimization procedure

splines with a partitioning of the optimization horizon in subintervals of variable length. The optimization results show that the new procedure is able to reduce the start-up time compared to the classical one by approximately 29% (29 minutes) when the GT load representation is performed with only 4-spline functions (see Figure 5.8b), an approach which is also better than the procedure based on *2-Hill function*.

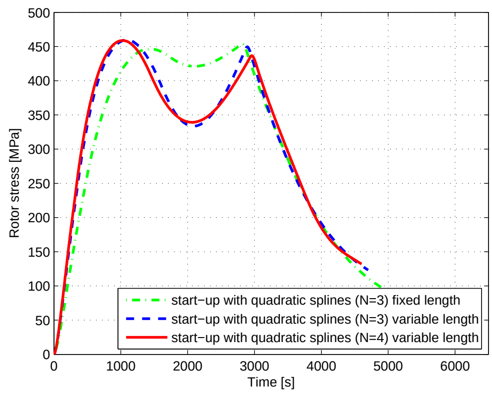
The comparative analysis of the results obtained with the selected functions shows that during the first 1000 seconds, the GT load optimal solutions increase more rapidly compared to the classical one, inducing a rapid growth of the rotor stress (Figure 5.9b). After this first phase, the GT load is very 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 growth of the steam temperature. Therefore, the rotor stress decreases regardless of GT load profile. In the last part, only the Header stress (Figure 5.9a) limits the load increase.

For the piecewise polynomial functions, in particular the quadratic spline functions with a variable subinterval length, the portion of slowing down observed in the *2-Hill* *functions* case is reduced 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. The flexibility can be even higher if the number of the data points used to describe the GT load profile is increased. For example in Figure 5.8b, the GT load is described by using $N = 3$ and $N = 4$ points respectively. The results show that in the second case, the start-up time is reduced by about 2 minutes compared to the first case, but with a growth in the computational effort. This type of approach represents the best trade-off between optimization efficiency and start-up performances. With an increased number of points the gain in start-up time becomes insignificant with respect to the involved computational burden. To give an overview on the computational times, some comparative results with respect to the type of parameterization and the number of optimization parameters are presented in Table 5.1 (see CPU time column).

The evolutions of the header stress and rotor stress corresponding to the optimal profile of the GT load are illustrated in Figures 5.9a and 5.9b. From these figures, it is apparent that the computed optimal profile enables to reach the required final start-up conditions while respecting the imposed state constraints. In particular, the solution guarantees that the level of stress obtained during the transient does not exceed the design value.



(a) Header stress evaluation

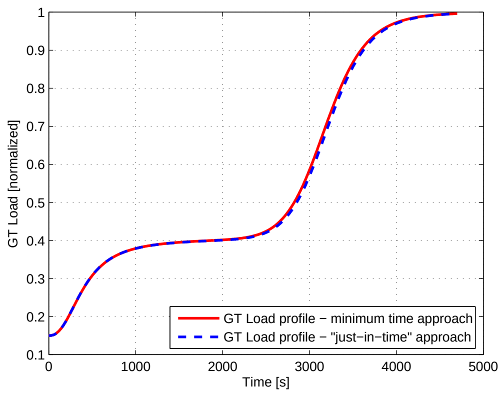


(b) Rotor stress evaluation

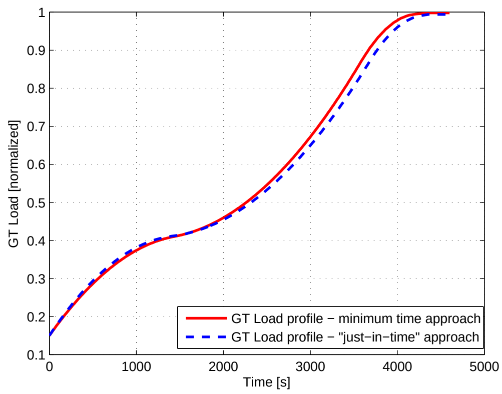
Figure 5.9: Stress evolution corresponding to the optimal profiles based on quadratic spline functions

5.3.2 Comparison minimum time - ”just-in-time” start-up problem. GT load profile optimization

Seeing the start-up process as a ”just-in-time” optimization problem over the time horizon $[t_0, t_f]$, the results of the optimization procedure are compared in the case of *2-Hill* profile (Figure 5.10a) and 4-quadratic spline representation (Figure 5.10b), with the previous minimum-time solutions. The final time has been fixed to 101% of the minimal start-up time provided by the previous approach.



(a) Optimal profiles with *2-Hill function*



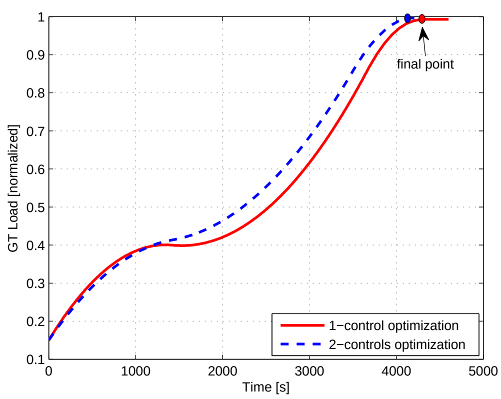
(b) Optimal profiles with quadratic splines

Figure 5.10: GT load profiles derived from the both optimization procedures

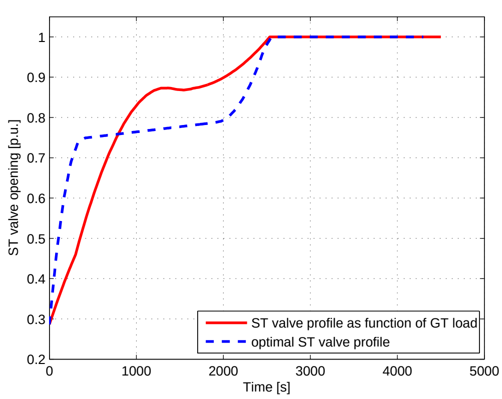
As it can be observed, the approach leads to start-up profiles comparable to the minimum-time solution. Therefore, in the case when a coherent final time is provided, the start-up minimum time problem can be seen as an optimal tracking problem subject to constraints.

5.3.3 GT load profile and ST throttle opening profile optimization

To improve the start-up performances even further, the optimization procedure has been extended to aim also the ST throttle opening profile. If in the previous case the ST throttle valve opening has been described as a function of GT load, in this procedure the profile is represented with quadratic spline functions. The new optimal GT load



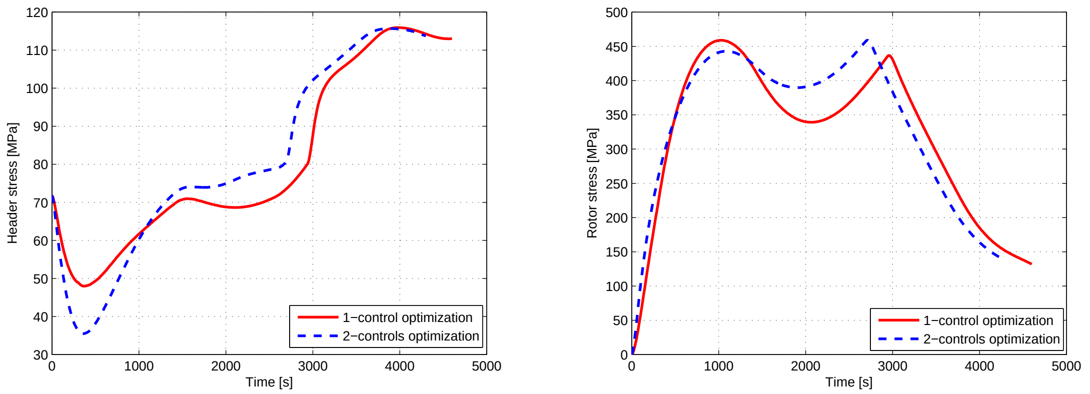
(a) Optimal GT load profiles



(b) ST throttle profile representation

Figure 5.11: Comparison profiles 1-control and 2-controls optimization

profile derived with this procedure as well as the solution found in the case of 1-control optimization problem are reported in Figure 5.11a. In the both cases the control profiles have been parameterized by using 4-quadratic spline functions. Also, the optimized ST throttle profile represented with 3-quadratic spline functions and the ST valve opening as a predefined function of the GT load level are represented in Figure 5.11b. The results



(a) Header stress (b) Rotor stress

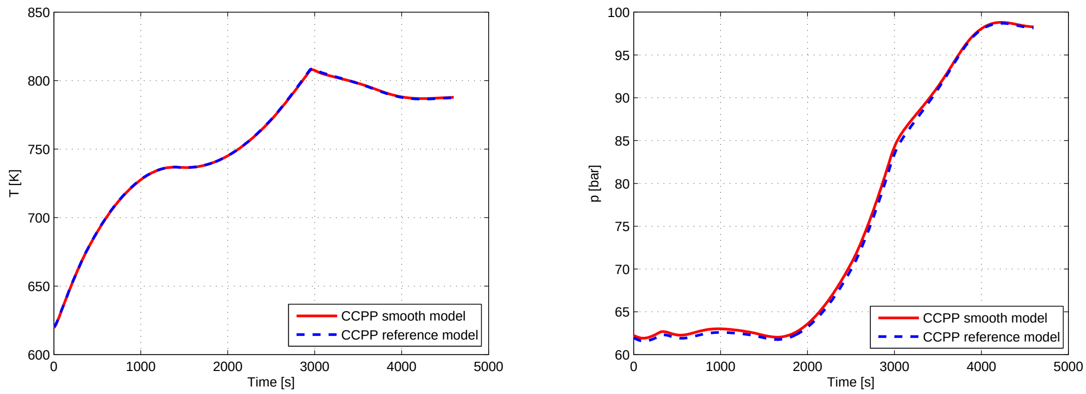
Figure 5.12: Comparison between 1-control and 2-controls optimization stress results

show that by optimizing the ST valve profile (2-controls optimization) a supplementary reduction of the start-up time (approximatively 3 minutes) is obtained compared to the single-control optimization (see Table 5.1 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 has been described as a predefined function of the GT load level, and by optimizing its profile a better control on the ST rotor stress can be performed (see Figure 5.12b). Nevertheless, the gain in start-up time is not significant compared to the involved computation time.

5.3.4 Validation on the reference CCPP model

The different optimal profiles computed with the smooth model have been applied to simulate the last part of the start-up procedure on the original combined cycle model. In this section, as case study only the GT load profile representation with quadratic splines is considered. The transients of some relevant plant variables corresponding to this profile are illustrated in Figures 5.13 to 5.14.

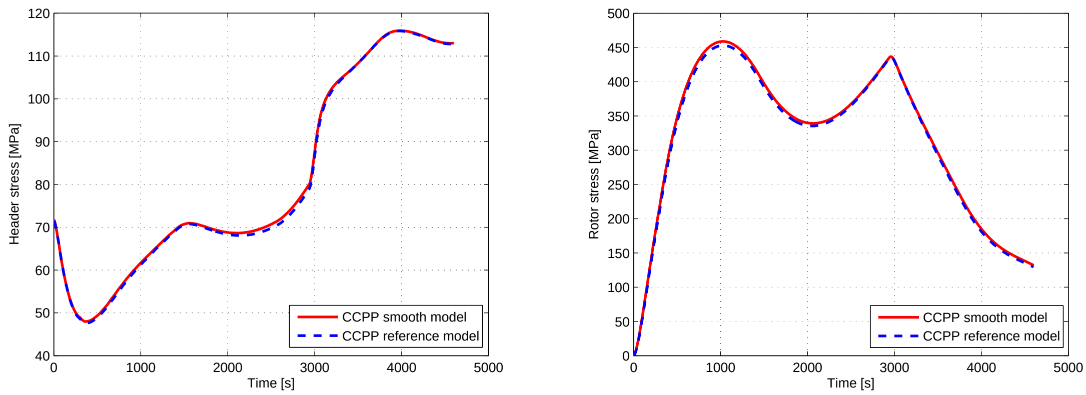
From the derived results it can be observed that the responses of the two models are quite close to each other, validating once again the consistency of the CCPP smooth model . Also, the optimal solution found with the smooth model remains valid in the case of the CCPP reference one, ensuring that the constraints imposed on the main plants variables are fulfilled.



(a) Steam temperature

(b) Steam pressure

Figure 5.13: Evolution of steam parameters corresponding to the smooth and reference model



(a) Header stress

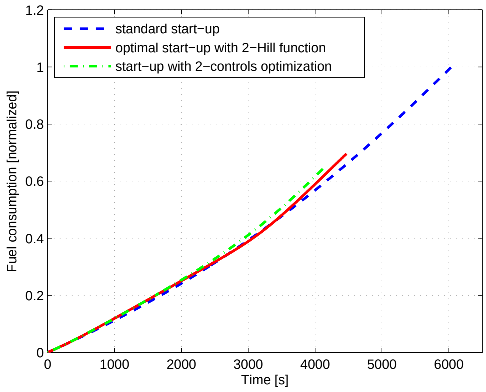
(b) Rotor stress

Figure 5.14: Evolution of the stress corresponding to the smooth and reference model

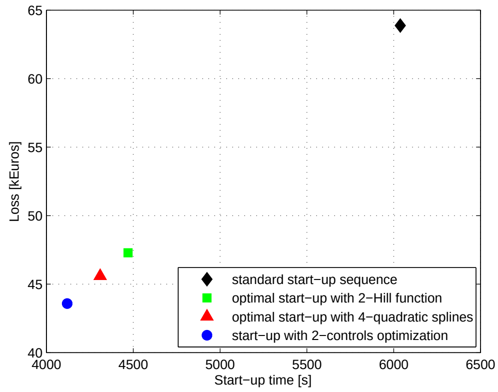
5.3.5 Start-up improvement benefits

The main benefit of shortening the start-up time is the reduction of the operating costs. A first gain is due to lower fuel consumption. A comparison between the fuel consumption for two optimal profiles, related to the classical start-up, is given in Figure 5.15a. The results show that the new optimal start-up procedure consumes up to 36% less than the traditional start-up.

Also, taking into account the losses during the start-up related to an operation of the plant in nominal mode (Figure 5.15b), the reduction of the start-up time involves considerable saved earnings (up to 32%). In Figure 5.15b, the *Loss* indicates the earnings if the plant would have fictitiously operated at nominal power during a period equivalent



(a) Fuel consumption comparison



(b) Losses reduction comparison

Figure 5.15: Start-up reduction benefits

to the start-up time, defined as:

$$Loss = E_n * p_e \tag{5.19}$$

where E_n denotes the delivered electrical energy (KWh), and p_e denotes the tariff (in €/KWh) of the electricity. A complete comparison in terms of benefits, computation time and number of optimization parameters, between all the profile-functions considered in this section can be found in Table 5.1. Also in Table 5.1, a ratio between the start-up time gain and complexity, defined as in 5.20, is presented. This ratio quantifies the trade-off between complexity and performance, showing in some measure the possibility for a real-time implementation.

$$r = \frac{g_t}{c} \tag{5.20}$$

with

$$g_t = \frac{t_{ref} - t_{st}}{t_{ref}} \tag{5.20a}$$

$$c = \frac{t_c - t_{rc}}{t_{rc}} \tag{5.20b}$$

where t_{st} is the start-up time derived by optimization procedure, t_{ref} is the start-up time corresponding to the standard procedure considered as reference time, t_c is the computation time required for the resolution of the optimization problem and t_{rc} denotes the computation time corresponding the standard start-up considered as reference computation time.

It should be pointed out that the computation time and the convergence properties strongly depend on the initial values of the optimization parameters selected in the procedure. In all these cases a feasible set of initial values derived from the standard start-up profile has been used.

Function		Start-up time [s]	Time gain[1][%]	Fuel saving[1][%]	Number parameters	CPU time [s]	Ratio [%]
GT load profile optimization							
linear piecewise	N=1[*]	*6038*	-	-	1	20	-
	N=10	5540	≈8	≈8	10	561	0.3
	N=20	4980	≈17	≈15	20	855	0.4
2-Hill		4466	≈26	≈30	6	169	3.5
quadratic spline	N=3	4777	≈21	≈23	3	58	11
	N=3[v]	4397	≈27	≈30	5	154	4
	N=4[v]	4278	≈30	≈34	7	253	2.5
GT load and ST throttle profiles optimization							
quadratic spline	$N_{GT} = 4$[v] $N_{ST} = 3$[v]	4120	≈32	≈36	12	663	1
Constrained tracking problem "just-in-time"							
2-Hill		4510	≈25	≈29	5	321	1.7
quadratic spline	N=4[v]	4321	≈28	≈31	6	388	1.6

[1] *related to the standard procedure*
[*] *corresponding to the standard start-up with a constant GT Load ramp 2 MW/min*
[v] *parameterization with variable subintervals*

Table 5.1: Comparative results

5.3.6 Start-up subproblem: dynamic optimization with Modelica-based platform

Using effective direct methods for solving the CCPP start-up optimization problem can represent an important solution. This solution has been envisaged, but, its implementation is constrained by the availability of powerful Modelica tools dedicated to optimization. Despite the fact that current Modelica-based tools offer effective methods for dynamic optimization, up to now, their application is often limited to relatively small systems. Thus, due to the CCPP model complexity (i.e. high-index DAE, number of equations), the existing tools are not sufficiently robust to solve the optimal control problem of the start-up process based on the entire system structure.

Nevertheless, in order to show the potential advantages provided by the efficient methods for dynamic optimization implemented in platforms which deal with Modelica models, an optimization problem based on an subsystem of the CCPP model has been formulated and solved. For this purpose, the platform focused on optimization JModelica.org [100] has been used. The off-line dynamic optimization of the start-up profiles for a subsystem has been addressed by means of a state of the art optimization algorithm. The start-up optimization problem was solved using the direct collocation method together with the interior point optimization algorithm IPOPT [125]. Applying an algorithm, such as IPOPT, enables the computation of the solution in a fast and robust way, exploiting the sparsity structure of the resulting NLP.

5.3.6.1 Subsystem model

The dynamic optimization considered in this section is based on a nonlinear model composed of the GT unit and a heat exchanger (economizer). The control input of the subsystem, that will be used as manipulated variable in the start-up optimization problem, is represented by the load level of the GT. According to the Dymola's statistic information, the model is described by a DAE with 481 equations and 12 state variables.

5.3.6.2 Optimization problem

As already stated, the definition of a dynamic optimization problem is an iterative procedure. The result of such a procedure in the case of a start-up problem based on a CCPP subsystem, regarding the adaptation of the model, objectives and constraints definition, is given in a final formulation in this section.

Specifications

As in the case of the start-up optimization procedure already stated, the goal of the optimization problem formulated in this section is to reach the GT full load level as quickly as possible, while limiting the stress value on the metal surface of the exchanger. The objective is formulated by means of a Lagrange-type cost function, penalizing the squared deviation from the target value represented by the full load (5.21). The full load state is considered when the load level of the GT ($L(t)$) has reached the final target of 100% ($L_M = 1$).

In this case a limiting factor in the start-up problem is the constraint acting on the thermal stress in the heat exchanger (5.22). The thermal stress is kept under control by limiting the allowable temperature gradient in the metal $\dot{T}_m$.

In addition, inequality constraints are imposed on the rate of change of the GT load profile. A constraint is imposed so that the GT load must not exceed the maximum rate ΔL_{max} specified by the manufacturer (5.23), and a constraint is considered to impose a positive gradient in order to avoid as far as possible the multiple cycling of the stress level during the transient (5.24).

Optimal control problem

Considering the specifications presented above, the optimization problem has been formulated as:

$$\min_{L(t)} \int_{t_0}^{t_f} (L(t) - L_M)^2 dt) \tag{5.21}$$

subject to the constraints

$$\dot{T}_m(t) \leq \Delta T_{max} \tag{5.22}$$

$$\dot{L}(t) \leq \Delta L_{max} \tag{5.23}$$

$$\dot{L}(t) \geq 0 \tag{5.24}$$

and to the DAE representing the plant dynamics. The initial state for the DAE represents the plant in hot start-up conditions (GT Load set at 0.15, metal temperature set at 569 K). The numerical values of the limits for the considered case study are: the maximum loading rate $\Delta L_{max} = 9MW/min$, the maximum gradient of temperature $\Delta T_{max} = 6K/min$. The final time t_f has been considered fixed and equal to 1500 seconds.

Scaling and initial guess

The physical models, in particular the CCPP subsystem, contains variables with different orders of magnitude (e.g., pressures, temperatures, mass flows). These large difference of values can have negative effects on the efficiency of numerical algorithms, leading to a decrease in their performance and in certain cases even to the algorithm failure. The variables scaling plays an important role in the convergence of the numerical algorithm, therefore all variables (states, controls) have been scaled to the same order of magnitude using the *nominal* attribute offered by Modelica.

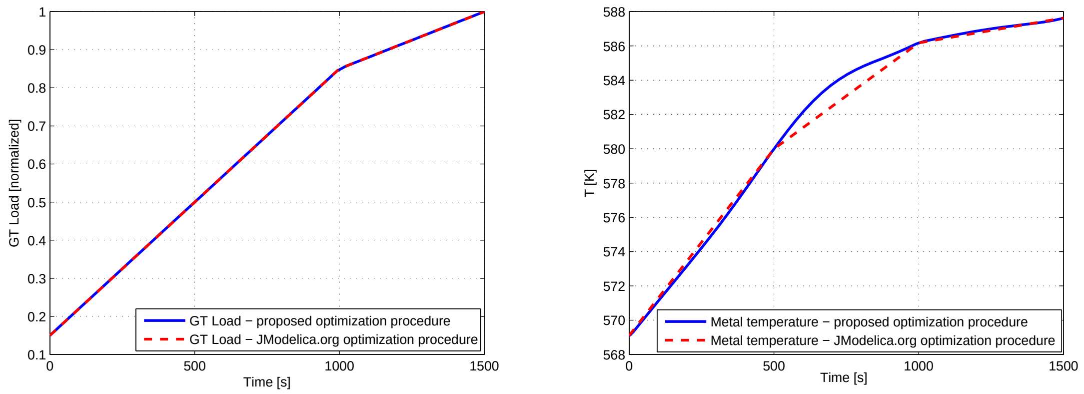
Moreover, the convergence and the execution time of the optimization algorithm strongly depend on the initial guess (start attributes) provided to the NLP solver. JModelica.org provides an algorithm for DAE initialization. Nevertheless, the algorithm is based on starting values fixed a priori and requires coherent values for all variables. Compared to other common procedures (e.g., procedures based on Block Lower Triangular transformation), this approach is generally more sensitive to lacking initial guesses for start values, increasing the burden on the user, that has to provide initial values for all variables. In this case, in order to provide coherent initial values, the Dymola environment has been used to generate a simulation result that has been exploited to initialize the optimization algorithm (to generate initial guesses for all variables).

Also, it should be pointed out that the Dymola tool has been used to obtain a model compatible with the platform requirements. More specifically, the optimization platform provides weak support for high-index DAEs, unlike Dymola that offers a powerful index reduction algorithm, being able to handle high-index systems and to provide to the user information about which equations are differentiated in the index reduction algorithm and which variables are selected as states. These informations have been used to manually perform an index reduction of the Modelica model in order to get a mathematical representation that can be supported by the optimization platform.

Optimization results

The optimization problem has been solved using the direct collocation method in JModelica.org and the proposed Matlab/Dymola procedure, with IPOPT and *fmincon* respectively. For IPOPT, the accurate derivative information and sparsity structures are generated via adjoint automatic differentiation.

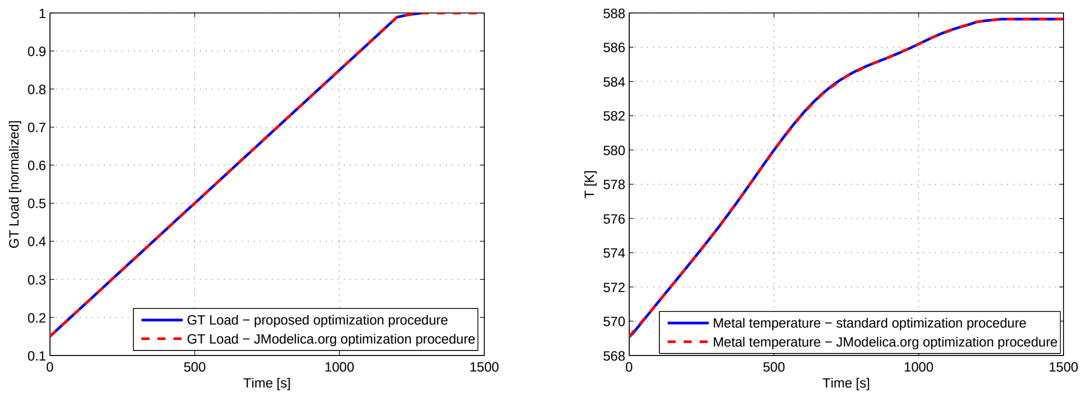
The results of the optimization are shown in Figures 5.16 and 5.17. The GT load control variable has been approximated in both cases by a piecewise linear function, and for the collocation method cubic interpolation polynomials have been used to approximate state profiles.



(a) GT load profile

(b) Economizer metal tube temperature

Figure 5.16: Comparative graphical results for an optimization problem with reduced number of elements of the grid ($N = 3$)



(a) GT load profile

(b) Economizer metal tube temperature

Figure 5.17: Comparative results for an optimization problem with increased number of elements of the grid ($N = 20$)

Table 5.2 shows the performance of the two compared procedures in terms of computation time. As it can be observed, the compared procedures yield very similar results, but with a significant reduction of computation time in the case of the solution computed with the efficient optimization algorithm implemented in JModelica.org. The main drawback of the JModelica.org procedure is that it is no applicable to address the start-up optimization problem based on the complete CCPP model, being limited to plant subsystems. The solution proposed in the first part of this chapter, although, provides performances in terms of computational efficiency and accuracy which may be improved, has the advantage that it can support a complete representation of the plant, enabling handling of large-scale systems.

Optimization Tool	**Number of elements**	**CPU time** [s]
Matlab/Dymola	N= 3	49
fmincon	N=20	8306
JModelica.org	N= 3	8
IPOPT	N=20	38

Table 5.2: Comparative results execution time

5.4 Conclusion

In this chapter the optimal design of the control profiles during the hot start-up of combined cycle plants is considered by means of an approach based on Modelica models. To solve the dynamic start-up optimization problem there is a series of efficient methods, but their application is constrained by the availability of powerful Modelica tools dedicated to optimization. The lack of optimization support able to handle large-scale Modelica models and to provide required information in order to apply efficient algorithms makes difficult the use of such models for optimization and control purposes. However, a version of the sequential method has been implemented in our case. The method exploits the sequential approach principle, solving the optimization problem by using a parameterized control sequence and integrating the system dynamics with this sequence of commands. Thus, the proposed method uses the CCPP model to derive an optimal profile of the input signals, assuming that it can be described by a parameterized function, whose parameters are computed by solving an optimal control problem, subject to the plant dynamics and to a number of constraints on the main plant variables.

The results have showed that the optimization procedure adopted in this work can significantly improve the start-up performances with a limited stress on the components and with numerous economic benefits. The adopted formulation of the optimization problem is quite simple in the definition of the cost function and the constraints. In particular, the variables used to define the constraints, such as allowable stress levels, maximal GT load variation, temperatures, pressures are usual quantities for the plant technicians, so that they can easily modify the optimization problem according to their needs and requirements. Also, although the optimization procedure aims mainly the GT load profile applied on the last start-up phase, the procedure can be easily expanded to the other input signals and other start-up phases.

Nevertheless, the efficiency of the approach is limited by the fact that the nonlinear solver uses numerical derivative information computed by finite-difference approximation. This procedure systematically perturbs each variable in order to compute cost function and constraint partial derivatives. Typically, the problem can be solved more accurately and efficiently if the derivative information is analytically computed and provided to the solver. Thus, the proposed approach makes a trade-off between the start-up performance and the computation time/convergence properties, showing that efficient control trajectories can be derived with a reduced number of parameters and with an adequate choice of the parameterized functions.

Also, in this chapter the applicability of an efficient direct method, implemented in the optimization platform for Modelica models JModelica.org, has been tested for an optimization problem based on a subsystem of the plant, and compared with the proposed procedure. The optimization results have showed that for the same type of approximation of the control profiles, i.e. piecewise linear function, described by the same number of parameters, the obtained solutions by both procedures are quite similar, but with a considerable reduction of computation time when the JModelica.org optimization procedure is used (i.e. direct collocation method with IPOPT). The complete formulation of the start-up optimization problem based on the developed CCPP model has not been possible due to the current limitation of the optimization platform concerning the complexity of the plant model. However, the optimization of physical systems is an area of active research and the level of support is expected to increase even more in the near future, taking also into account the fact that from a dynamic optimization perspective the complexity of Modelica models is a major challenge.

An important aspect that should be noted is the fact that the robustness properties of the optimal solution have not been considered. The robustness of the optimal solutions could be considered in terms of parametric sensitivity. Thus, the robustness properties of a solution can be analyzed by computing the state sensitivity with respect to parametric uncertainty. Nevertheless, as already pointed out, the sensitivities computation requires tool support that is quite difficult to be applied at the moment in the case of power plant Modelica model described in this work.

The proposed open-loop approach can be used in advanced control strategies. Thus, in order to improve the CCPP start-up performances even further, by exploiting the advantages of advanced control techniques, the approach will be implemented in a model predictive control framework. Furthermore, this control technique has built-in robustness properties, representing a robust solution against perturbation and model errors.

CHAPTER 6

Model predictive approach for start-up procedure

6.1 Introduction

Driven by the fact that nowadays the processes need to operate under tighter economical and environmental specifications, in the previous chapter a model-based approach has been proposed. In general these requirements can be met when the systems work over a wide range of operating conditions and often close to its admissible operating region. In these conditions, linear models are often unsuitable to adequately describe the process dynamics and therefore complex nonlinear models, able to operate close to the limits of the process, are required. Thus, the proposed approach is based on the first principle CCPP model, developed in the Modelica language and adapted for optimization purposes in Chapter 4. The model can accurately represent the start-up process, during which the plant operates over wide temperature and pressure ranges, and in which the thermodynamics lead to highly nonlinear behavior.

In this chapter, the approach will be expanded by using the model predictive control strategy (MPC) [85] [126], [127]. During the last decades, the MPC technique (also known as receding horizon control or moving horizon control) has been extensively used in many applications in the process industry to derive control policies for dynamic systems [128]. The predictive control strategy has been successfully applied in the case of combined cycle power plants for control at partial or full load [129], [130], but it has been seldom used for the start-up phase [88].

The main advantages of MPC is that in principle it enables to anticipate the future known situations at en early stage, taking into account the potential perturbations and model-plant mismatch. Due to its prediction capability and ability to explicitly handle constraints, the MPC technique represents an attractive solution for the CCPP start-up problem. Thus, by using the mathematical model of the system, the MPC strategy can exploit in an optimal manner the future effects of the control actions on the system evolution, in particular, the modification of the GT load profile ramp. Indeed, the GT load profile issued from the MPC algorithm is adapted to the current state of the system and can be generated with an improved prediction capability compared to the open-loop approach proposed in Chapter 5.

However, due to the computational burden of the associated optimization problem, the indicated centralized MPC solution can be considered inappropriate for a real-time implementation, being generally limited to an off-line application. In order to obtain real-time implementable results, a hierarchical predictive control structure (H-MPC) in which two controllers tackle different objectives is further suggested.

6.2 Model predictive control

6.2.1 Concept

In general, the principle of the model predictive control strategy [85] [131] consists in the repeated solving of an optimal control problem, over a finite time horizon knowing explicitly the trajectory to be followed over this horizon, subject to system dynamics and constraints involving state and control variables. MPC is an advanced control strategy that explicitly uses a mathematical model to predict the future behavior of the process and an optimizer to derive the control sequence by minimizing/maximizing an objective function. According to the receding horizon paradigm, at each instant, the sequence of optimal control moves is computed by solving the optimization problem over the prediction horizon, and then only the first value of the sequence is applied to the system. At the next sample, the procedure is repeated, and a new optimization is performed, based on the current process measurements or estimations. The basic structure of an MPC control loop is illustrated in Figure 6.1.

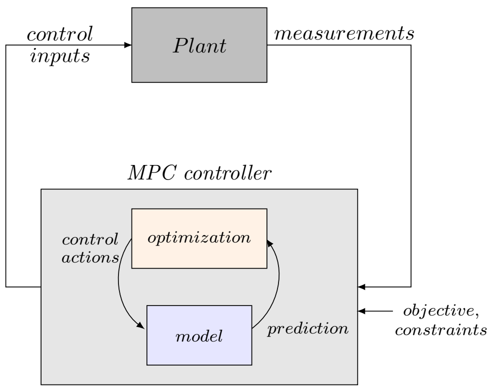


Figure 6.1: Basic MPC scheme

In the following, the basic principle of model predictive control is described by means of Figure 6.2. At time t, by using the current measurements, the controller predicts the future behavior of the system over a prediction horizon T_p and determines over a control horizon T_c (with $T_c \leq T_p$) the input signal trajectory, by optimizing a predefined objective function.

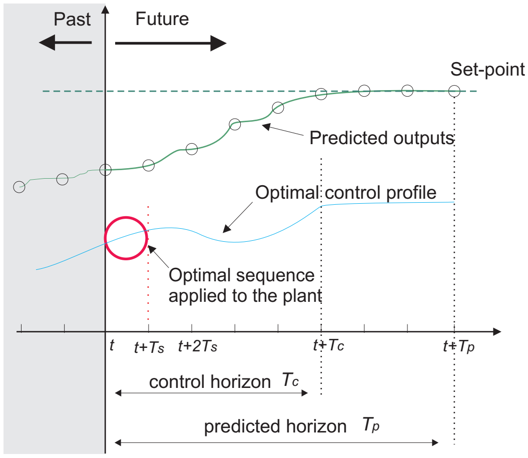


Figure 6.2: Principle of model predictive control

In the ideal case (i.e. no disturbances, no model-plant mismatch) and if the optimization problem can be solved for an infinite horizon, the derived optimal profile at time $t = 0$ can be applied to the system for all $t \geq 0$. However, in a real application this is generally not possible because, due to perturbations and model mismatch, the process behavior is different from the predicted one. Therefore, the obtained control profile will be implemented only until the next measurement becomes available (or in a discrete-time formulation, only the first value of the trajectory is applied to the plant). The difference of time between two measurements can be variable, but in the most common situations it is fixed a priori (at each sampling time, T_s). Based on the new measurements at the time $t + T_s$, the overall procedure - prediction and optimization - is repeated to find a new optimal profile moving the control and prediction horizon forward (receding horizon strategy).

It should be stated that in Figure 6.2 the input profile is described as an arbitrary function of time. As already performed in the open-loop optimization procedure previously presented, and as it will be shown in this chapter, for a numerical solution of the optimal control problem the input variable profiles are parameterized in an appropriate way.

To implement this strategy, the basic structure illustrated in Figure 6.1 is considered. A model is used to predict the future behavior of the plant, based on the past and current information and on the derived optimal control actions. The control actions are derived by the optimizer, taking into account the objective function and the constraints. Summarizing, the operating principle of a standard MPC scheme is:

1. measurement/estimation of the system states;

2. computation of optimal input profile by minimizing/maximizing a specified cost function over a prediction horizon using a prediction model;

3. implementation only of the first part of the optimal signal until new measurements are available;

4. go to step (2).

Consequently, the process model plays a key role in the MPC controller. The control performance depends on the accuracy of the plant model. In fact, in the application of MPC strategy, two key issues have to be addressed. The first concern is that a reliable model able to represent the process dynamics with an accurate output prediction is required. On the other hand, MPC requires the repeated on-line solution of an optimal control problem based on the dynamic model of the plant to be controlled and a complex model involves powerful algorithms to solve efficiently the resulting problem. Thus, the second, which represents one of the main limiting factors of the MPC strategies application, is that at each sampling time the computational effort needed to solve the optimization problem is high and efficient algorithms compatible with real-time implementation are required. The implementation becomes more challenging when nonlinear models are used (NMPC).

6.2.2 Solution of the NMPC problem

In the literature different approaches to solve on-line NMPC problems exist, the most popular being so-called direct methods briefly presented in the previous chapter [132], [133]. In these approaches, the control, constraints and/or state variables are finitely parameterized, and thus an approximation of the original open-loop optimal control problem is sought. The resulting finite dimensional dynamic optimization problem is then solved with standard optimization techniques. Nevertheless, as previously stated, the applicability of these methods is currently limited for Modelica physical models. Therefore, to solve the optimal control problem at each sampling time, the open-loop optimization procedure described in the previous chapter has been adapted and used.

6.3 GT load profile optimization. Predictive approach

The predictive method proposed in this section focuses on the determination of the GT load profile. In general, the severity of thermo-mechanical stresses depend on the initial metal temperatures of the components before start-up, and on the GT loading rates applied throughout the start-up process. Consequently, to achieve a minimum start-up time, the controller must consistently manipulate the fastest instantaneous loading rates, while keeping the stresses under admissible limits. This is specifically the goal of the control strategy described in the following sections. However, the procedure can target also the other control variables of the plant.

For the application in a predictive approach, the mathematical formulation of the open-loop optimal control problem outlined in Chapter 5 has been slightly changed. In the sequel, the adopted continuous time formulation and resolution of NMPC for the CCPP start-up problem are presented.

6.3.1 NMPC problem formulation

Following the notation in Chapter 5, the combined cycle system dynamics can be described as:

$$\begin{aligned} \dot{x}(t) &= f(x(t), z(t), u(t), L(t, q), t, p) \\ 0 &= g(x(t), z(t), u(t), L(t, q), t, p) \end{aligned} \tag{6.1}$$

with differential state variables $x(t)$, algebraic state variables $z(t)$, control variables that are not concerned by the optimization procedure $u(t)$, control variable to optimize represented by using a finite parameterization $L(t, q)$ (GT load profile) and time-independent parameters p.

Objective function definition

The objective in this approach is defined by means of an integral square error criterion, penalizing the deviation of the control input from its target value (L_M). In fact, with this cost function, as in the case of "just-in-time" start-up problem, 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.

In the NMPC problem, at every sampling instant, the input applied to the system is given by the solution of the following finite horizon open-loop optimal control problem:

Problem 1.

Find

$$\min_{q} J(x(t), L(t, q), T_p, T_c)$$

with

$$J(x(t), L(t, q), T_p, T_c) = \int_{iT_s}^{iT_s+T_p} (L(t, q) - L_M)^2 dt \tag{6.2}$$

subject to the plant dynamics (6.1) and

$$c(x(t)) \leq 0, \quad \forall t \in [iT_s, iT_s + T_p] \tag{6.3}$$

where $i \in \mathbb{N}$ is an index used to define the sampling time T_s, T_p and T_c are the output and control prediction horizons. For the approach considered in this section, it is assumed that the output prediction horizon is set equal to the control horizon $T_p = T_c$. The constraint (6.3) represents, in a general form, all the constraints to be imposed on the plant variables during the start-up, in particular on the stress level.

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

6.3.2 NMPC problem solution

The solution of the NMPC problem 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 control values parameterization is to use piecewise constant functions over sampling times. For example, a representation of the control variable $L(t)$ as a piecewise constant function over each predicted sampling interval is:

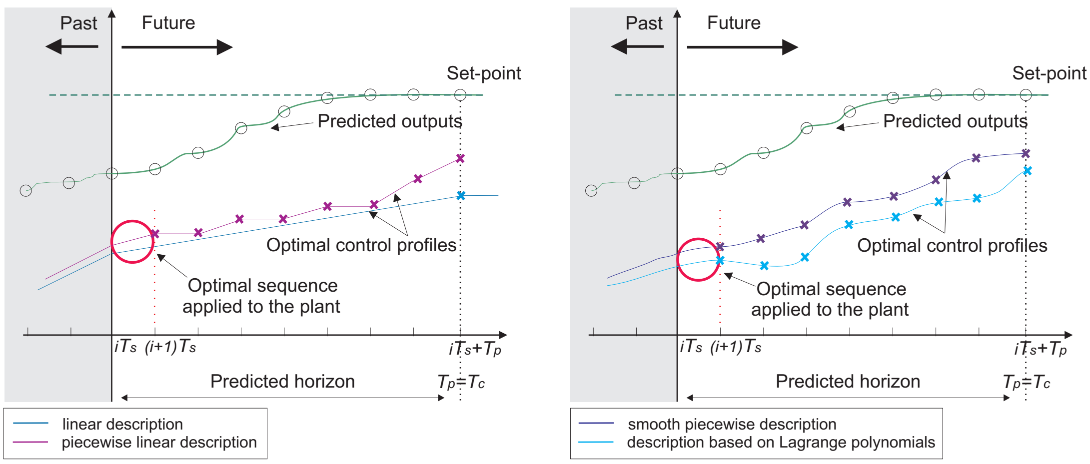
$$L(t) = q_i, \quad \forall t \in [iT_s, (i+1)T_s] \tag{6.4}$$

with $i = \overline{0, N-1}$ and $N = \frac{T_p}{T_s}$. Thus the optimal control problem is reduced to the finding of the parameters vector $q = [q_0, q_1, ..., q_{N-1}]$.

In some cases, in particular for CCPP start-up problem, a continuous control policy is preferred, rather than a policy that requires sudden switching transitions from one level to another. In the predictive methodology described in this chapter, for computing trajectories of the control variable, polynomial parameterizations over a predefined number of intervals have been used.

NMPC: polynomial-based approximations of control profile

In principle, the way in which the control parameterization is performed corresponds to the one for the open-loop optimization procedure presented in the previous chapter. In a first phase, the optimization horizon $[iT_s, iT_s + T_p]$ is subdivided in N control partitions, and then in each subinterval, the GT load control profile $L(t)$ is approximated by a specific function.



(a) Linear representation (b) Smooth representation

Figure 6.3: Predictive approach: control parameterization

As already mentioned, the range of functions that can be used to represent the GT load profile is quite large. The types of polynomial-based functions which have been tested in the start-up problem are illustrated in Figure 6.3. For example, in Figure 6.3a the control profile is described by linear or piecewise linear functions, and in Figure 6.3b by functions with a specific degree of smoothness. It should be pointed out that in the predictive approach for the GT load description, additionally to the case studies presented in the open-loop procedure, a parameterization based on the Lagrange polynomials has been used in order to increase the control performance. This parameterization is described below.

Lagrange polynomial based parameterization

Considering $n+1$ points $t_0, ..., t_n \in [iT_s, iT_s + T_p]$, with $n \geq 1$, and the corresponding GT load values at these points $L_0, ..., L_n$, the Lagrange interpolation polynomial of degree n used to describe the GT load profile is formally stated as follows:

$$L^n(t) = \sum_{j=0}^{n} L_j l_j^n(t) \tag{6.5}$$

where l_j^n represent Lagrange basis polynomials defined as:

$$\begin{cases} l_j^n(t) = 1, & n = 1 \\ l_j^n(t) = \prod\limits_{\substack{k=0 \\ k \neq j}}^{n} \dfrac{t - t_k}{t_j - t_k}, & n \geq 2 \end{cases} \tag{6.6}$$

with $0 \leq j \leq n$.

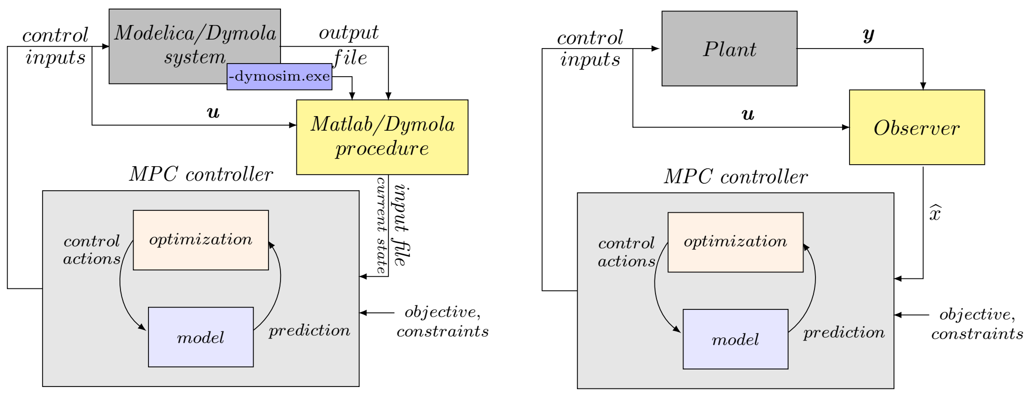
This type of representation has the advantage that it can be easily implemented compared with that based on quadratic spline functions, providing similar performances (see Section 6.3.4). It represents an alternative for the use of quadratic splines, but as for these functions the number of elements in each optimization interval is limited (in this case is typically lower than 10). A description of this parameterization with application results can be found in [134].

6.3.3 Implementation of the predictive approach

The MPC controller has been implemented in Matlab, and uses for the resolution of the optimization problem the nonlinear solver *fmincon*. In order to enable and facilitate the use of the smooth CCPP model for periodic simulations in the MPC algorithm, the model description has been compiled into an executable code (*dymosim.exe*) [8]. The *dymosim* executable is a stand-alone program which can be directly called from Matlab. This program can use specific files to define its inputs and initial conditions, perform a simulation run, store the results into an output file, etc. For example, when a Dymola model is simulated, an input file (*dsin.txt*) is read that contains all the initial values of the variables including the states of the system. Therefore, the Dymola files together with the CCPP description (*dymosim.exe*), have been used to periodically restart the simulation from a given point, making it possible to update the current state of the model and to apply the receding horizon paradigm. More details about *dymosim* executable, Dymola files specification and the way in which a Modelica model can be executed are given in Appendix G.

In general, the derivation of predictive control law is based on the knowledge of the complete state of the system. The feature that the model representation can use files to define conditions of experiment (variables, parameters) and to provide complete information about the state variables of the system has been exploited in the implementation of the MPC controller. Thus, at each sampling time the determination of the current state and the initialization of the model in the specified state have been carried out using the results file generated by simulation.

The entire procedure has been coded in Matlab and included in the control algorithm. The adopted NMPC control loop with the physical plant represented by the Modelica/Dymola system is outlined in Figure 6.4a.



(a) Adopted control loop (b) Classical control loop

Figure 6.4: NMPC control loop

Therefore, the elaboration of the control law has been conducted assuming that the complete states of the system are known, which is not generally true in the case of a real plant. In a real control environment, the measurement of the state variables of the system is often not available, either very expensive or even impossible. The most popular solution to this problem is the use of an observer that reconstructs the state variables using a dynamic model of the physical system, the inputs and the available measurable physical outputs of the system (see Figure 6.4b). However, the design of such an observer is beyond the scope of this work.

6.3.4 Simulation results

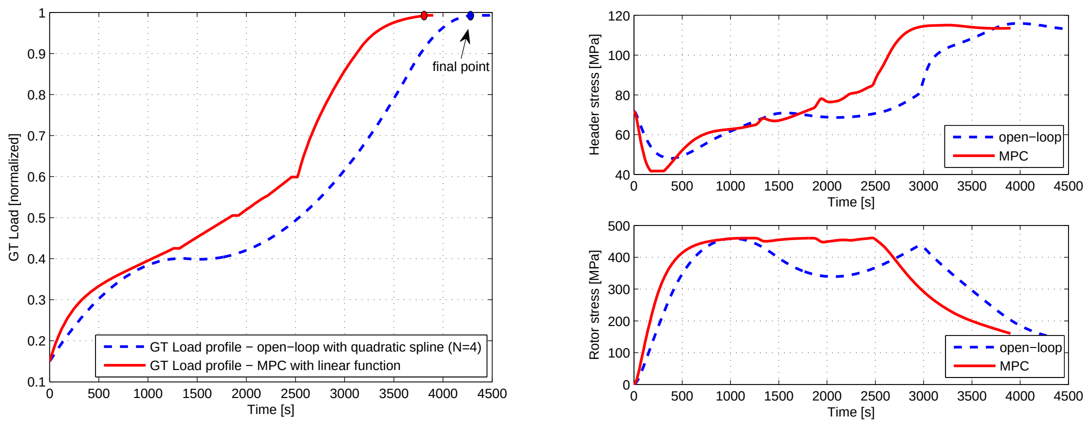
In the preceding paragraphs, the predictive approach of the GT load profile for the CCPP start-up problem has been presented.

The purpose of this section is to compare the predictive algorithm results with the ones obtained by applying the open-loop optimization procedure implemented in Chapter 5. To do this, the control algorithm has been applied to last stage of the start-up sequence, considering the same initial conditions as in Section 5.3. Also, the admissible limits of the constraints that act on the plant variables are maintained the same.

To illustrate the advantages and the limitations of the predictive approach, several scenarios have been studied, varying the MPC control parameters and the type of parameterization used for the GT load profile description. In the sequel, only few of them, considered as being the most representative, will be presented.

Scenario 1

In a first scenario the GT load profile has been considered as being described by a linear function (a ramp profile described by a first-degree polynomial) over the prediction horizon T_p. In this case study the specific tuning parameters of the predictive control strategy are: T_s=60 s and T_p=5*T_s. Figure 6.5a shows that the predictive approach can provide better results than the open-loop optimization. The predictive approach is able to reduce the start-up time compared to the open-loop procedure (by approximatively 10%), while fulfilling the imposed constraints (Figure 6.5b).



(a) GT Load profile (b) Stresses evolution

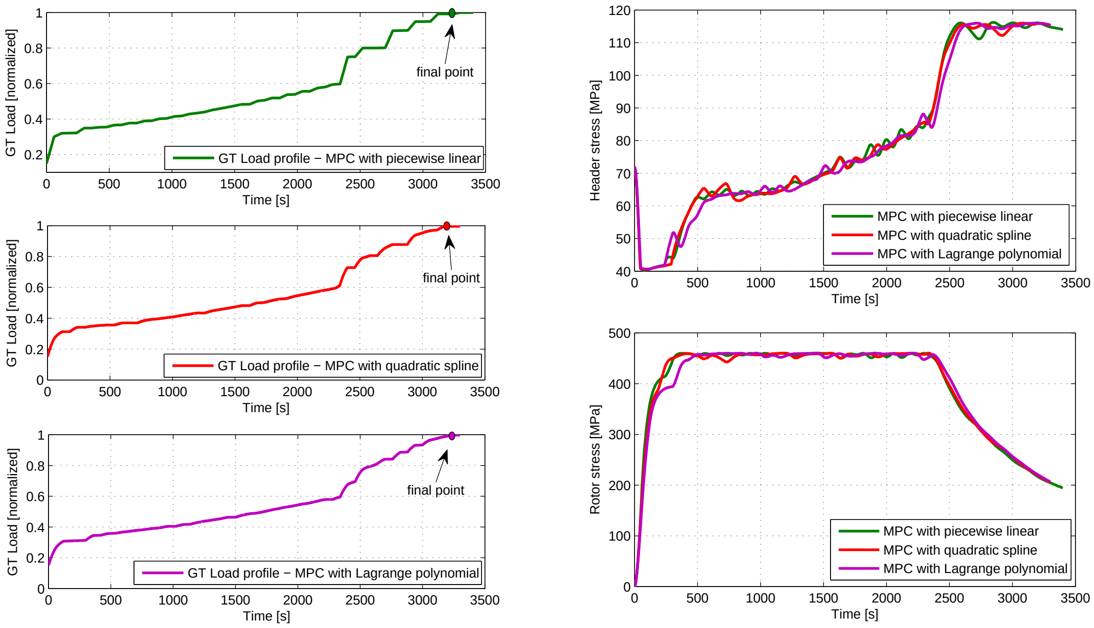
Figure 6.5: Comparative results: open-loop approach vs. MPC GT load profile

Scenario 2

The manner in which the approximation of the control profile is carried out can influence the optimality of the solution. Thus, in order to improve the start-up performance even further, the representation of the GT load profile has been performed with different types of functions. In this scenario, in addition to the linear representation, the GT load profile has been systematically described over the prediction horizon by means of the following functions:

- piecewise linear functions;
- quadratic spline functions;
- Lagrange's polynomials.

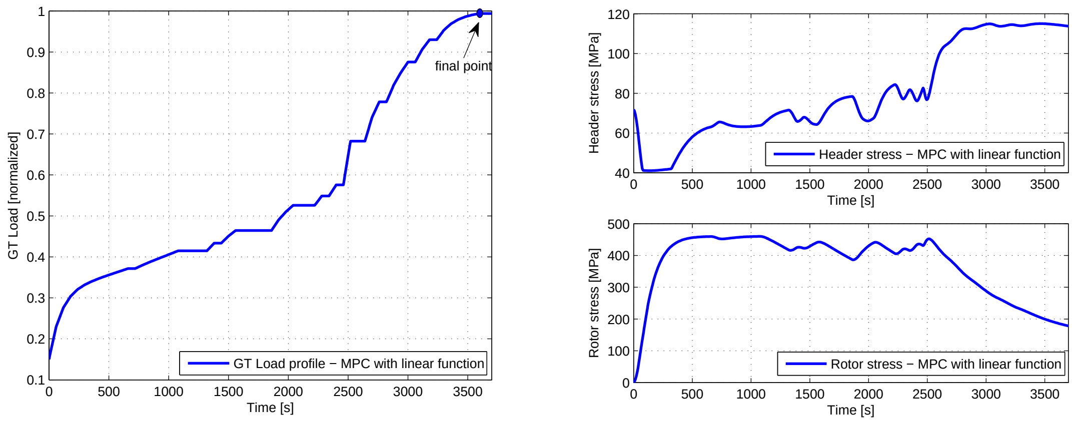
It should be noticed that the study has been conducted by considering the same MPC control parameters as in *Scenario 1* (T_s=60 s and T_p=5*T_s). The resulting GT load profiles are given in Figure 6.6a.



(a) GT Load profile (b) Stresses evolution

Figure 6.6: Comparative results corresponding to different types of parameterization of the MPC GT load profile

By choosing a more flexible parameterization of the control profile, the start-up time is further reduced compared to the MPC of the GT load profile represented with linear functions (by 13-16%), while keeping the stresses under the desired levels (see Figure 6.6b). The difference in terms of start-up time between all three types of parameterization listed



(a) GT Load profile (b) Stresses evolution

Figure 6.7: MPC GT load profile with linear function. Case study for T_s = 60 s and T_p=3*T_s

above is really quite low (equal to the sampling time of 60 s, see also Table 6.1). In this case the gain brought by a more flexible profile (e.g., quadratic spline) is less important than for the open-loop approach. Also, the use of such functions for profile description involves a computation time well above the limits for a real-time implementation.

In the considered predictive approach the solution that ensures the best trade-off between start-up performance and computation time is carried out by approximating the GT load profile by means of a linear function over the prediction horizon, together with an appropriate tuning of the control parameters (Figure 6.7a). It enables to obtain a gain in start-up time comparable to that provided by a more accurate description of the GT load profile (a lower gain up to 12%), without exceeding the stress limits (Figure 6.7b).

In general, simplified representations are desirable from the point of view of the computational effort. This fact can be observed from the computation time needed to solve the overall optimization process for each type of parameterization (see Table 6.1), where for the MPC with a linear representation of the GT load profile the computation time is considerably lower than in other cases.

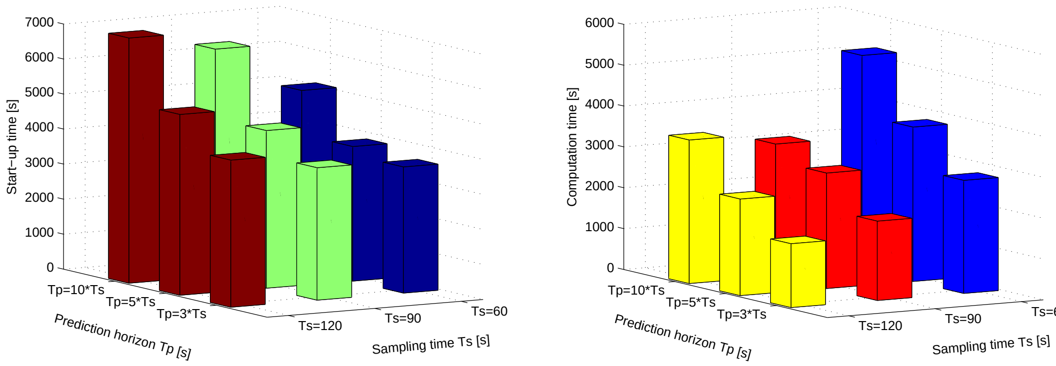
Scenario 3

By adopting the predictive control strategy, the potential delay between changes in the loading rate and their effects on the process variables, in particular on the header and steam turbine stresses, can be anticipated. An important role in this strategy is played by the prediction horizon.

In general in industrial application of NMPC, in order to achieve specific control performance a long prediction horizon is required. Moreover, the nominal stability of the closed-loop is relied on the selection of long horizons. On the other hand, from a computational point of view short prediction horizons are preferable [135].

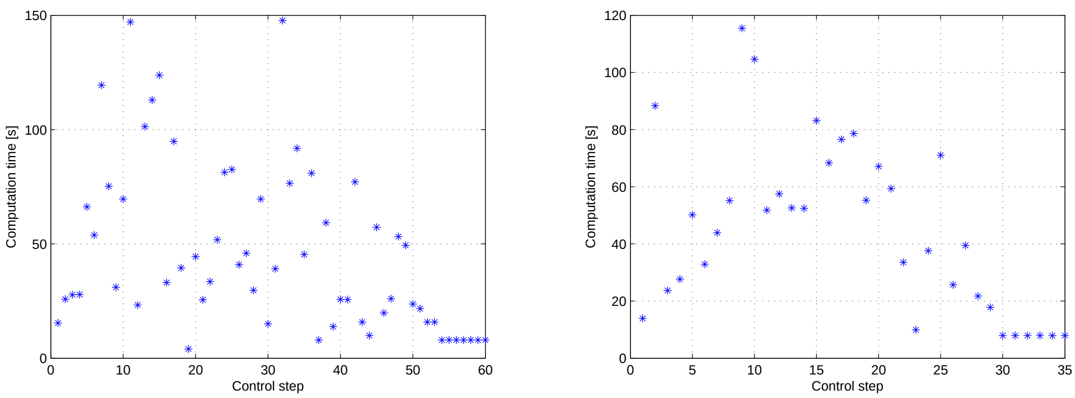
In this scenario the length of the prediction horizon has been varied to show how it can influence the start-up performance. Also, for obtaining high controller performances, different values of the sampling time has been chosen. The obtained results are presented in Figure 6.8.

The selection of sampling time T_s and prediction horizon T_p is important in the implementation of the MPC start-up controller. As shown in Figure 6.8 the trend for obtaining the highest start-up performances is to select a short prediction horizon. In fact, the prediction horizon T_p, which together with the control profile approximation directly influence the computational effort to solve Problem 1, is limited by the efficiency of the optimization algorithm. Also, the manner in which the sampling is chosen can influence the optimality of the solution. The fact that at each control step the approximation of the optimal solution is not performed with accuracy requires a frequent computation of the control profile, involving a short sampling time. In turn, a short sampling time involves an increased number of optimization problems solved during the start-up, resulting in a higher overall computation time (6.8b). It should be pointed out that the minimum value of the look-ahead time has been selected to be longer than the dominant time constant of the system, a necessary condition to avoid issues linked to an insufficient prediction time. A complete comparison in terms of start-up performances and computation time between different types of profile-functions used for the GT load description and control parameters considered in the proposed approach is given in Table 6.1.



(a) Start-up time depending on MPC parameters (b) Total CPU time depending on MPC parameters

Figure 6.8: MPC controller performances depending on the control parameters. Case study for the GT load profile described by a linear function



(a) CPU time for MPC with $T_s = 60$ s (b) CPU time for MPC with $T_s = 120$ s

Figure 6.9: Computation time at each sampling time. Case study for the GT load profile described by a linear function

The analysis of the obtained results shows that effective solutions for the start-up phase can be derived with a simplified representation of the GT load profile and with an appropriate selection of the control parameters. Nevertheless, due to the efficiency of the optimization algorithm the resolution of the nonlinear optimization problem on-line under real-time conditions is quite difficult. Apart from a few cases when a longer sampling time is selected, the solver is not able to provide a solution of the optimization problem within a period defined inferior to the sampling time of the process (see Figure 6.9a). From the computation time point of view, more suitable solutions can be given when a large sampling time is chosen, but in this case the solution is sub-optimal in terms of start-up time (Figure 6.9b).

<table>
<tr><th colspan="7">GT load profile optimization</th></tr>
<tr><th>Function</th><th colspan="2">Control parameters</th><th>Start-up time [s]</th><th>Time gain [%]</th><th>Fuel saving [%]</th><th>CPU time^{1}[s]</th></tr>
<tr><td>standard seq.</td><td>-</td><td>-</td><td>6038</td><td>-</td><td>-</td><td></td></tr>
<tr><td>open-loop quadratic spline</td><td colspan="2">-</td><td>4278</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td rowspan="8">linear</td><td rowspan="2">$T_s = 60$ s</td><td rowspan="2">$T_p = 3 * T_s$</td><td rowspan="2">3600</td><td>$\approx 16^{*}$</td><td>$\approx 14^{*}$</td><td rowspan="2">2884</td></tr>
<tr><td>$\approx 40^{**}$</td><td>$\approx 43^{**}$</td></tr>
<tr><td rowspan="2">$T_s = 60$ s</td><td rowspan="2">$T_p = 5 * T_s$</td><td rowspan="2">3840</td><td>$\approx 10^{*}$</td><td>$\approx 5^{*}$</td><td rowspan="2">3776</td></tr>
<tr><td>$\approx 36^{**}$</td><td>$\approx 38^{**}$</td></tr>
<tr><td rowspan="2">$T_s = 90$ s</td><td rowspan="2">$T_p = 3 * T_s$</td><td rowspan="2">3780</td><td>$\approx 11^{*}$</td><td>$\approx 8^{*}$</td><td rowspan="2">1942</td></tr>
<tr><td>$\approx 37^{**}$</td><td>$\approx 39^{**}$</td></tr>
<tr><td rowspan="2">$T_s = 120$ s</td><td rowspan="2">$T_p = 3 * T_s$</td><td rowspan="2">4200</td><td>$\approx 2^{*}$</td><td>$-^{*}$</td><td rowspan="2">1563</td></tr>
<tr><td>$\approx 30^{**}$</td><td>$\approx 38^{**}$</td></tr>
<tr><td rowspan="2">piecewise linear</td><td rowspan="2">$T_s = 60$ s</td><td rowspan="2">$T_p = 5 * T_s$</td><td rowspan="2">3240</td><td>$\approx 25^{*}$</td><td>$\approx 23^{*}$</td><td rowspan="2">12851</td></tr>
<tr><td>$\approx 46^{**}$</td><td>$\approx 49^{**}$</td></tr>
<tr><td rowspan="2">quadratic spline</td><td rowspan="2">$T_s = 60$ s</td><td rowspan="2">$T_p = 5 * T_s$</td><td rowspan="2">3180</td><td>$\approx 26^{*}$</td><td>$\approx 25^{*}$</td><td rowspan="2">14831</td></tr>
<tr><td>$\approx 47^{**}$</td><td>$\approx 49^{**}$</td></tr>
<tr><td rowspan="2">Lagrange's polynomial</td><td rowspan="2">$T_s = 60$ s</td><td rowspan="2">$T_p = 5 * T_s$</td><td rowspan="2">3240</td><td>$\approx 25^{*}$</td><td>$\approx 23^{*}$</td><td rowspan="2">6218</td></tr>
<tr><td>$\approx 46^{**}$</td><td>$\approx 48^{**}$</td></tr>
</table>

^{1} *overall CPU time computed for optimization process*
* *related to the open-loop procedure*
** *related to the standard procedure with a constant GT Load ramp 2 MW/min*

Table 6.1: Comparative results Model Predictive Control of the GT load profile

Summarizing, the GT load profiles issued from the MPC strategy are adapted for the current plant state and are generated with a specific prediction capability. Therefore, with an accurate estimation of future evolution of the plant behavior, in particular of the stresses (rotor and header), the MPC load can be more aggressive, without exceeding the admissible limits. As a result, the new controller has been able to complete the start-up phase by about 18 minutes faster than the open-loop approach and by 47 minutes faster than a standard start-up procedure. Despite this, due to the computational burden, at this stage, an effective solution in terms of start-up time, which at the same time is implementable under real-time conditions can not be provided. A solution in this sense will be given in the next section.

6.4 Hierarchical model predictive control approach

Hierarchical control has received a significant attention during the last decades [136]. The interest is mainly motivated by the fact that an important number of systems can be better controlled with hierarchical structures than with classic control methodologies. Among them, we can mention the design of controllers for multi-time scale systems characterized by clearly separable slow and fast dynamics (based on e.g., cascade structures), hierarchical structures used to coordinate a number of local controllers (e.g., for the control of a power plant [130]), the design of controllers for plants characterized by an inherent hierarchical structure. A complete discussion can be found in the reports of the HD-MPC project[1] (see for example [137], [138]) and the many references therein.

Also, given the difficulty to control large-scale systems with centralized control structure due to the required computational complexity, the application of MPC approach in hierarchical and distributed control architectures shows a high potential. In many situations, sharing information between local regulator designed with MPC and/or between different control layers about the future control actions and the corresponding state trajectories involves performances comparable to those ideally provided by a centralized control structure.

In the case of the CCPP start-up problem, the application of a hierarchical control structure is motivated by two aspects. The first aspect is that the combined cycle power plants are complex systems composed by many interacting subsystems that are generally controlled by means of a hierarchical structure, where the upper layer coordinates the local controllers. The second aspect is that the computational burden of the optimization problem to be solved at each control step is high, and the proposed MPC solution can be considered inconsistent for a real-time implementation (which was the case in previous section). Therefore, in order to obtain more real implementable results, the MPC framework has been redesigned and extended to a hierarchical predictive control structure (H-MPC) in which two controllers tackle different objectives.

6.4.1 Hierarchical structure

The proposed hierarchical structure includes two layers with different time scales: a high (HL) and a low layer (LL). At the high level a minimum time optimal control problem is solved over a long time period. The solution of this problem is used to update the GT load set-point for the low level, where a constrained tracking optimization problem is solved over a shorter period. It should be noticed that in this case the MPC problems for each layer are formulated based on the same Modelica model and not on models with different dynamic behaviors as in the most common approaches proposed in the literature [139].

The objective of the hierarchical approach is to create a control structure with more suitable optimization problems in terms of computation time. The underlying idea is to have at the high level an optimization problem that derives suboptimal profiles of the GT load, enabling to be quickly solved. Then, at the low level the information provided by

[1] Hierarchical and Distributed Model Predictive Control of Large-Scale Systems, http://www.ict-hd-mpc.eu/

the high level is used for obtaining a more tractable optimization problem, which refines the GT load profiles in order to reach an optimal solution.

6.4.2 H-MPC problem formulation

The problem is formulated for the high and low layers, where the control variable of interest is the load of the GT.

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

- high level:

$$\begin{aligned} \dot{x}(t) &= f(x(t), z(t), u(t), L_h(t), t, p) \\ 0 &= g(x(t), z(t), u(t), L_h(t), t, p) \end{aligned} \tag{6.7}$$

- low level:

$$\begin{aligned} \dot{x}(t) &= f(x(t), z(t), u(t), L_l(t), t, p) \\ 0 &= g(x(t), z(t), u(t), L_l(t), t, p) \end{aligned} \tag{6.8}$$

where $L_h(t)$ is the GT load control variable associated with the high level, while $L_l(t)$ is the refined GT load control variable corresponding to the low level.

Assumption 6.4.1. *The GT load profile for each level is described by a parameterized function, $L_h(t, q_h)$, $L_l(t, q_l)$, with q_h and q_l the parameter vectors that are derived through optimization.*

In the H-MPC approach, for each layer a different sampling time is defined. In the sequel, T_h and T_l represent the sampling times for high level and low level respectively.

Remark 6.4.1. *For simplicity, a relationship between T_h and T_l is fixed, by introducing a time ratio, $k \in \mathbb{N}$ so that $T_h = kT_l$, with $k \geq 1$.*

Remark 6.4.2. *At each level different horizons are associated; a long horizon corresponds to the high level while for the low level a short prediction horizon is considered.*

6.4.2.1 Optimization problem for the high level

The goal of the optimization problem is to find the GT load profile that minimizes the start-up time, while fulfilling constraints on the process variables. The problem corresponds to the open-loop optimization presented in the previous chapter.

At each HL sample time iT_h, the GT load control profile, described by $L_h(t, q_h)$, is computed as a result of the following minimum time problem:

Problem 2.

Find

$$L_h^*(t, q_h) = \arg\min_{q_h, t_f} J_h(x(t), L_h(t, q_h), t_f)$$

with

$$J_h(x(t), L_h(t, q_h), t_f) = \int_{iT_h}^{t_f} dt \tag{6.9}$$

subject to the plant dynamics (6.7) and

$$L(t_f, q_h) \geq L_M - \epsilon_1 \tag{6.10}$$

$$|f(x(t_f), u(t_f), z(t_f), L_h(t_f, q_h)| \leq \epsilon_2 \tag{6.11}$$

$$c(x(t)) \leq 0, \quad iT_h \leq t \leq t_f \tag{6.12}$$

where (6.10-6.12) and the used notations have the same meaning as in Section 5.2.1.1, $i \in \mathbb{N}$ is an index used to define the HL sampling time and q_h is the parameter vectors that are derived through optimization.

6.4.2.2 MPC problem for the low level

The goal at this level is to track the GT load reference given by the HL. Adopting the receding horizon paradigm, the GT load control variable profile is computed at each LL sampling time. The GT load profile, described by $L_l(t, q_l)$, is derived by minimizing an integral square error criterion that penalizes the deviation of the control variable from the target value provided by the HL. The optimization problem is formally stated as:

Problem 3.

Find

$$L_l^*(t, q_l) = \arg\min_{q_l} J_l(x(t), L_l(t, q_l), T_p)$$

with

$$J_l(x(t), L_l(t, q_l), T_p) = \int_{jT_l}^{jT_l + T_p} (L_l(t, q_l) - L_h^*(iT_h + T_k, q_h))^2 dt \tag{6.13}$$

subject to the plant dynamics (6.8) and

$$c(x(t)) \leq 0, \quad \forall t \in [jT_l, jT_l + T_p] \tag{6.14}$$

where T_p is the prediction horizon, $j \in \mathbb{N}$ is a index used to define LL sampling time and T_k is a tunable time constant used to define the future target of the GT load computed by the HL. The constraint (6.14), as in other optimization problems, includes all the constraints imposed on the plant variables during start-up phase.

Then, according to the receding horizon paradigm, only the optimal profile computed for $t \in [jT_l, (j+1)T_l]$ is applied to the system and the overall procedure is repeated at a new sampling period.

The optimization problem at the low level (Problem 3) aims the same objective as the 1-level MPC controller (Problem 2), namely the minimization of the squared deviations from the target point. The difference is made by the target point that in the hierarchical structure is defined at each HL sampling time by the upper optimization problem, while in the 1-level MPC it corresponds to the full load.

Remark 6.4.3. *The reference from HL is considered as a constant along the interval* $[jT_l, jT_l + T_p]$ *to facilitate the communication between layers.*

6.4.3 H-MPC multilevel algorithm

The hierarchical approach presented above has been integrated in an MPC multilevel algorithm. The algorithm with the principle adopted for the communication of information between the layers can be stated as:

Basic algorithm:

HL: - at the time instant iT_h:

- the optimization problem corresponding to the high level (Problem 2) is solved
- the solution determined at the predefined instant T_k, $L_h^*(iT_h + T_k, q_h)$, is sent to the low level

- the procedure is repeated after each time instant T_h ($i := i + 1$) with the new states provided by the plant

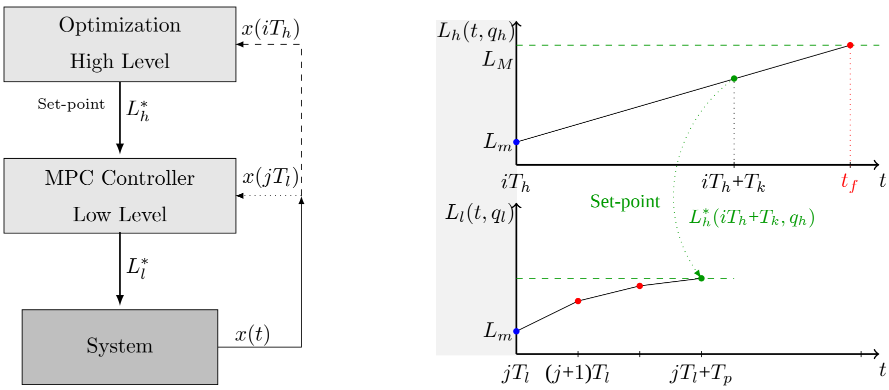
LL: - at every instant jT_l:

- the corresponding MPC problem for the low level (Problem 3) is solved by using the reference signal provided by high level optimization

- according to the receding horizon paradigm:

- only the optimal profile computed for $t \in [jT_l, (j+1)T_l]$ is applied to the system
- the overall procedure is repeated at a new sampling period T_l ($j := j + 1$), with the current plant states

The adopted structure of the hierarchical controller is illustrated in Figure 6.10a. Also, Figure 6.10b shows the temporal diagram of the approach.



(a) Hierarchical structure (b) Hierarchical approach temporal diagram

Figure 6.10: H-MPC approach

Definition of the reference time constant

In the proposed multilevel MPC algorithm, a key role is played by the constant T_k, that defines the time at which the reference signal is set for the low level. The selection of this time constant is important, since a value that is too small can lead to suboptimal solutions and a value that is too high can penalize the computation time. Specifically, a higher value of T_k involves an intermediate target in the LL objective function closer to the full load, which makes the tracking problems to approach the 1-level MPC case. On the other hand, a value of T_k close to T_p involves target values for the LL which can be rapidly reached, but which lead to loading rates less aggressive. In this case the solution is dictated rather by the HL open-loop optimization, limiting potential benefits of a refinement of the profile at the lower level. Therefore, a trade-off between the profile efficiency and the computation time can be done by the choice of T_k. A parametric study is presented in the following paragraph in order to show the manner in which the selection of the tunable constant T_k can influence the algorithm performances (Figure 6.13).

Remark 6.4.4. *For a minimum level of optimality, the tunable time constant T_k must be selected according to the condition $T_k \geq T_p$.*

6.4.4 Results

Hierarchical structure approach aims at achieving real-time implementable results regarding computation time. In order to show the advantages of the proposed approach, in this section the hierarchical structure has been applied to the last phase of the start-up sequence, considering the same initial conditions as given before, and then compared with the 1-level MPC solution. Some comparative results are summarized in Table 6.2.

The study of the effectiveness of the H-MPC algorithm is conducted in a similar manner to that described in Section 6.3.4, by considering several scenarios in which the controller parameters and the GT load representation are varied.

GT load profile optimization							
Approach	Control parameters		Start-up time [s]	Time gain [%]	Fuel saving [%]	CPU time [s] total	CPU time [s] mean
standard	-	-	*6038*	-	-	-	-
open-loop	-		*4278*	-	-	-	-
MPC S_1	$T_s = 60$ s	$T_p = 3 * T_s$	3600	≈40*	≈43*	2884	48
H-MPC S_1	$T_l = 60$ s $T_h = 2 * T_l$	$T_p = 3 * T_l$ $T_k = 5 * T_p$	3600	≈40*	≈43*	1425	10.5 HL 18.2 LL
MPC S_3	$T_s = 60$ s	$T_p = 3 * T_s$	3240	≈46*	≈48*	6218	115.1
H-MPC S_3	$T_l = 60$ s $T_h = 3 * T_l$	$T_p = 5 * T_l$ $T_k = 2 * T_p$	3240	≈46*	≈48*	2949	9.6 HL 51.2 LL

* *related to the standard procedure*
HL *High Level mean computation time*
LL *Low Level mean computation time*
S_1 *Scenario 1: MPC with GT load profile described by linear functions*
S_3 *Scenario 3: MPC with GT load profile described by Lagrange polynomials*

Table 6.2: Results comparison MPC vs. H-MPC

Scenario 1

In this scenario, a linear representation of the GT load profile has been considered for each MPC controller. Also, to compare the performance of the H-MPC algorithm with respect to the 1-level MPC solution, the parameters of the 1-level MPC controller correspond to those of the LL controller of H-MPC, namely:

- MPC: $T_s = 60$ s, $T_p = 3 * T_s$;
- H-MPC:
  - LL: $T_l = T_s$, $T_p = 3 * T_l$;
  - HL: $T_h = 2 * T_l$, $T_k = 5 * T_p$.

The experimental results in terms of computation time at each control step are given in Figure 6.11. Visibly, the main advantage of the application of the hierarchical structure is that it enables the reduction of the computation time. In the case of the H-MPC algorithm application, the time required to solve the optimization problem at each control step is lower than the sampling period (see Figure 6.11). Furthermore, the hierarchical approach based on a consistent estimation of the future behavior of the plant, together with a periodic computation of the GT load profile, derives an optimal profile that leads to a fast start-up phase (see Figure 6.12a), while maintaining the lifetime consumption of the most stressed components under control (see Figure 6.12b). As can be seen from Table 6.2, the hierarchical approach leads to start-up performances comparable to those of the one level MPC solution, but with a significant reduction of the total computation time (over 50%).

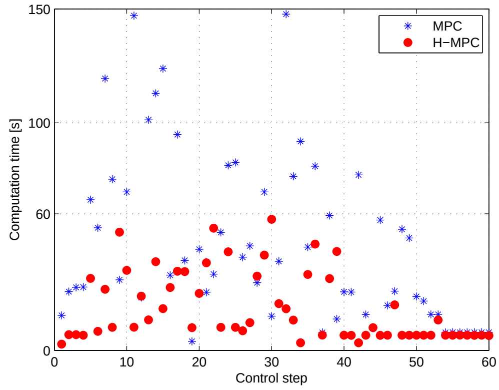
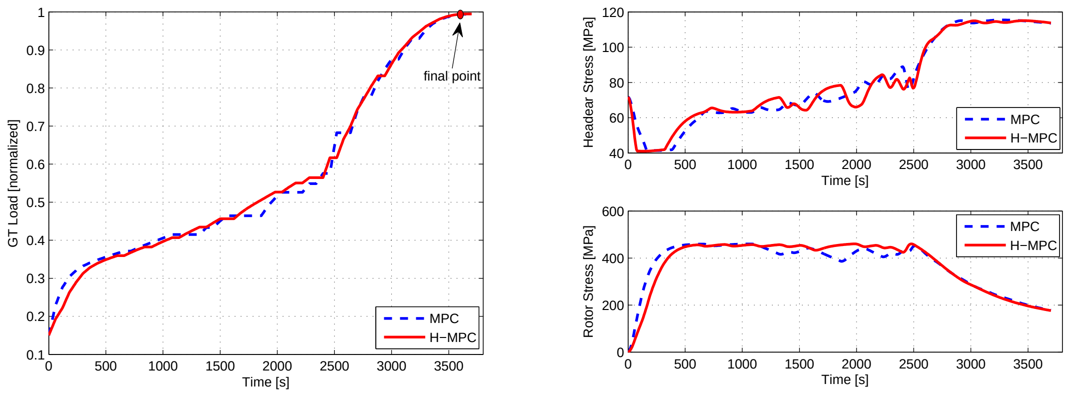


Figure 6.11: Computation time at each sampling time MPC vs. H-MPC. Case study for: $\{T_s = 60\text{ s}, T_p = 3 * T_s\}_{MPC}$, $\{T_l = T_s, T_p = 3 * T_l, T_h = 2 * T_l, T_k = 5 * T_p\}_{H-MPC}$, with GT load profile described by a linear function



(a) GT Load profile

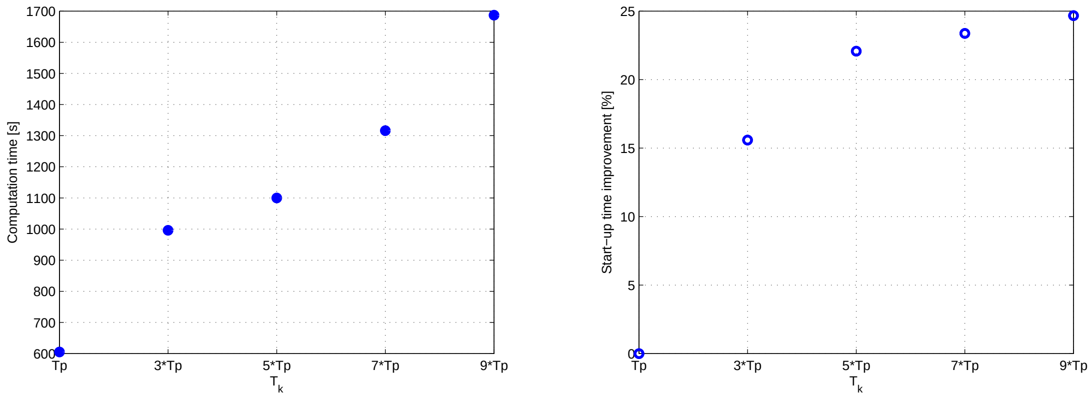
(b) Header stress and Rotor stress results

Figure 6.12: Results comparison MPC vs. H-MPC. Case study for: $\{T_s = 60\text{ s}, T_p = 3*T_s\}_{MPC}$, $\{T_l = T_s, T_p = 3*T_l, T_h = 2*T_l, T_k = 5*T_p\}_{H-MPC}$, with GT load described by a ramp profile

Scenario 2

The key of the hierarchical algorithm is represented by the use of intermediate load-points to define the target in the low level objective function. The full load target in the LL controller, initially considered in the 1-level MPC, has been replaced with intermediate points generated by the HL in order to provide to the solver a more suitable tracking problem.

In the H-MPC algorithm the definition of the LL targets-load is updated at each T_k period. The selection of the constant T_k at which the reference signal is set to the LL influences the quality of the solution. In order to show how the tunable algorithm parameter T_k can influence the solution of the H-MPC problem, the results of a parametric study are presented in Figure 6.13. The case study has been performed for a description of the GT load profile with a linear function and using the following parameters of the H-MPC controller: $T_l = 60$ s, $T_h = 2 * T_l$, $T_p = 3 * T_l$.



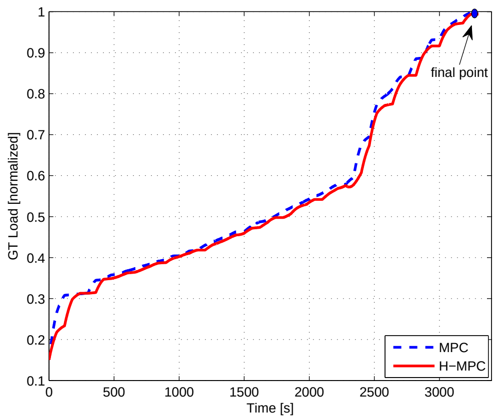
(a) Influence of T_k on the overall computation time (b) Influence of T_k on the profile optimality

Figure 6.13: Influence of the parameter T_k on the optimization performances. Comparison start-up time related to the case T_k=T_p

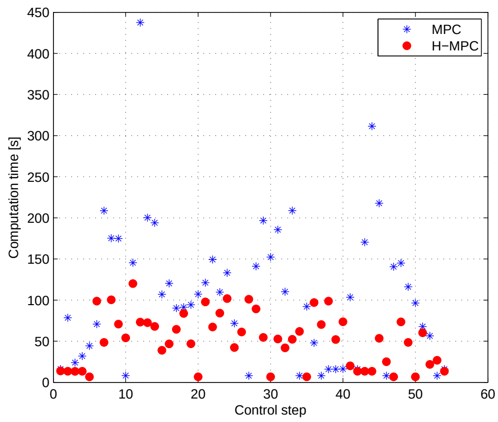
According to the simulation results, it is apparent that a value of T_k close to the LL prediction horizon leads to profiles which can be quickly derived, but less effective in terms of start-up time; on the other hand, if a higher value is chosen, the corresponding targets tend to full load, resulting in optimal profiles determined with inappropriate computation times, but close to those provided by the 1-level MPC solution. In general, a compromise start-up time/computation time must be done in order to obtain real-time implementable results. In fact, the proposed approach offers the possibility to make a trade-off between start-up performance and computation time, by selecting the constant that defines the time at which the set-point is set for the low level.

Scenario 3

The performances of the H-MPC algorithm, as for other proposed solutions, depend on the accuracy with which the GT load profile is approximated. In the third scenario, the efficiency of the hierarchical algorithm has been tested for more flexible representations of the GT load profile at the low level. The results presented in Figure 6.14a have been obtained by considering the GT load profile at the LL described by Lagrange polynomials. At the HL, the same type of representation as in the previous cases, based on a linear function, has been used. Also, the case study has been achieved by using the following parameters of the H-MPC controller: $T_l = 60$ s, $T_p = 5 * T_l$, $T_h = 3 * T_l$, $T_k = 2 * T_p$. It should be pointed out that the hierarchical control strategy has been compared with



(a) GT Load profile



(b) CPU time at each sampling time

Figure 6.14: Comparative results H-MPC vs. MPC with GT load profile based on Lagrange polynomials. Case study for: $\{T_s = 60\ \text{s}, T_p = 5 * T_s\}_{MPC}$, $\{T_l = T_s, T_p = 5 * T_l, T_h = 3 * T_l, T_k = 2 * T_p\}_{H-MPC}$

the 1-level MPC controller of the GT load ramp-profile, whose parameters ($T_s = 60$ s, $T_p = 5 * T_s$) correspond to those of the low level H-MPC.

The H-MPC approach can derive GT load profiles with a considerable reduction of the computation time, while achieving start-up performances comparable to the 1-level MPC controller (see 6.14a). However, with the available computing power, the resolution time in the case of a more specific description of the control profile is incompatible with the time constants of the process for a real-time implementation (see Figure 6.14b, where a sampling time of the process equal to 60 s has been considered).

Summarizing, the results have shown that the use of the H-MPC solution can significantly reduce the start-up time (more than 40%) compared to the standard optimization procedure, keeping the start-up performances close to those provided by 1-level MPC solution, but with a considerably lower computation time (see Table 6.2). The H-MPC algorithm, based on an appropriate selection of the profile-function representation and of the control parameters, provides solutions compatible with a real-time implementation. Nevertheless, the efficient application of the H-MPC algorithm under real-time conditions depends on the accuracy with which the GT load profile is approximated.

6.5 Conclusion

In this final chapter, a Nonlinear Model-based Predictive Controller (NMPC) for CCPP start-up problem has been proposed. The optimization problem has been formulated in order to minimize the start-up time subject to the plant dynamics represented by the smooth Modelica model of the plant and to constraints on the main plant variables. The optimization is focused on the GT load profile. Based on the finite parameterization of the manipulated variable profile, the same technique as in Chapter 5 has been used for the resolution of the optimization problem. Different types of profile functions have been tested to achieve efficient solutions.

The predictive approach is a promising control technology for the CCPP start-up. Using a periodic computation of the GT load profile and based on an accurate prediction of the future behavior of the plant, the proposed approach provides high start-up performances (time, fuel consumption), being able to significantly improve the transient time as compared to the standard start-up procedure, and even than the open-loop procedure previously presented.

The optimal GT load profiles derived from the MPC controller are computed with a specific predictability and adapted to the current process state, resulting in a more aggressive GT loading rate, without exceeding the stress limits.

However, due to the computational effort to solve the optimization problem at each control step, the indicated MPC solution has been considered inconsistent for a real-time implementation, providing an insufficiently low computation time. A solution in order to obtain real-time implementable results has been proposed in the second part of this chapter. The suggested solution has been to use a hierarchical predictive control structure (H-MPC) with two levels, in which each controller tackles a different objective. The hierarchical algorithm has a certain degree of flexibility (start-up efficiency/computation time) regarding the determination of the optimal solution.

The results emphasize the fact that the use of the H-MPC solution reduces significantly the start-up time compared with the standard start-up procedure and even with open-loop optimization procedure, keeping the start-up performances close to those provided by 1-level MPC solution, but, the most important aspect, with a lower computation time.

This chapter completes the proposed solutions for the improvement of the CCPP start-up throughout the dissertation. In the next chapter, a synthesis with general conclusions regarding the presented study is outlined and some perspectives with suggestions for a future work are provided.

CHAPTER 7

Conclusions and perspectives

The content of this thesis is dedicated to the development of new strategies for the improvement of the combined cycle power plant start-up efficiency. The proposed strategies are based on the first-principle model of process using dynamic optimization techniques for achieving the intended purposes. The presented chapters include systematically the main phases to develop and implement a model-based approach, detailing every aspect in part: physical modeling, optimization and control. In general, the results are specific for the CCPP start-up, but many parts can be generalized to other applications of optimization and control design. The posed optimal start-up problem has been analyzed and formalized, before explaining the proposed solutions, while emphasizing the contributions brought. Each adopted solution has been tested and validated by simulation. In this final chapter, the main results are summarized, and some future research directions are discussed.

7.1 Synthesis of the work

In the process industry, the safety and the operational availability represents imperative requirements. Thus, conventional strategies are generally used for the start-up of combined cycle power plants. Nevertheless, the traditional techniques are quite conservative, using a series of limitations on the control variable which leads to suboptimal solutions (i.e. a relatively long start-up time, high operating costs, etc.). These limitations are intended to preserve the lifetime consumption of the plant components under control. Due to increasingly stringent economical and environmental specifications, new and efficient management strategies are required for the start-up phase, which ensure the safe operation of the plant, while keeping the lifetime consumption of the components under specific levels.

The dynamic optimization for optimal control of the system represents an important solution to deal with the improvement of the start-up efficiency. Based on a dynamic mathematical model of the process, powerful strategies can be implemented. The optimal control of the CCPP start-up can be considered to derive control and state profiles, minimizing different performance criteria.

Predictive control strategy, widespread used in industry for optimal control of complex systems, can be successfully applied for the start-up problem. The results presented in this manuscript emphasize the benefits of the model-based approaches, in terms of reduction of the start-up time, and implicitly of the fuel consumption. The application of the model-based approaches in the case of large-scale systems, in particular the MPC, is generally constrained by the necessity to have an accurate model of the plant, which can be further used by optimization algorithms compatible with real-time requirements.

Several research works have been conducted in recent years on modeling of thermal power plants, encapsulating the experts knowledge in model libraries. The use of dedicated languages for physical modeling of complex heterogeneous systems together with the current available model libraries facilitate the design and the development of accurate models. This type of approach has been exploited to design knowledge-based models that can be used in different procedures for obtaining high control performance. However, in general the models developed with these tools are not immediately suitable for optimization and control purposes. The present work provides a solution in this sense, proposing a methodology to transform physical model components into optimization-oriented models. Based on the developed models a series of approaches to improve the performance of the start-up phases, with a special emphasis on the last phase, are suggested. The approaches derive optimal profiles of the control variables, being designed to reduce the start-up time and the operating costs, while maintaining the life consumption of the most stressed components of the plant under specified levels. Due to weak tool support for

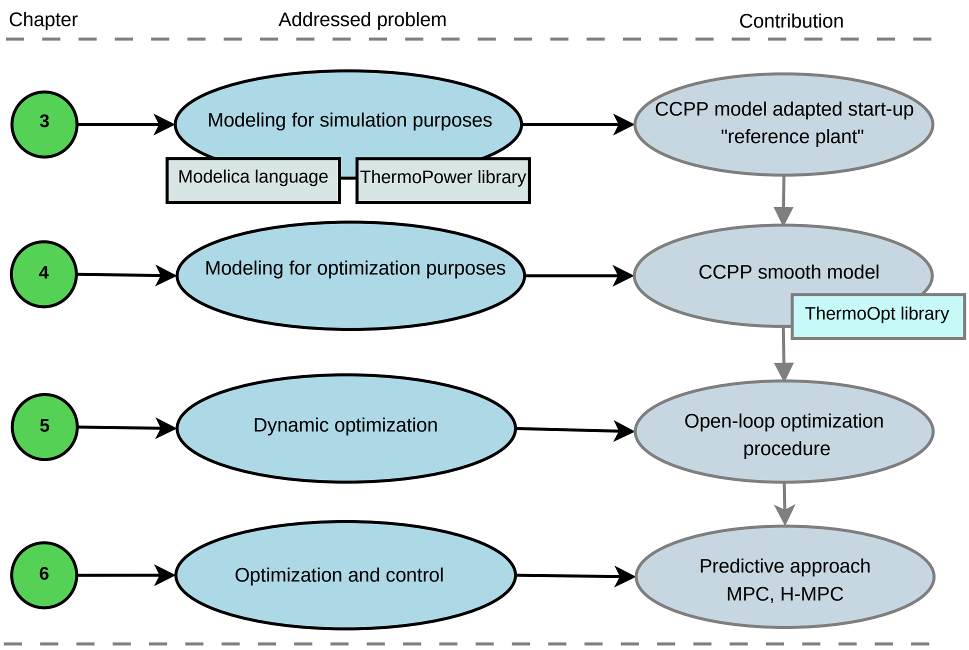


Figure 7.1: Schematic representation of the content addressed in manuscript

dynamic optimization, in particular for the models developed in Modelica language, the provided solutions are based on a trade-off between start-up efficiency (i.e. optimality of profile), on the one hand, and computational performance (i.e. computation time, convergence properties), on the other hand.

The development of the approaches described in this thesis is progressive. The main problems addressed in the manuscript with the distribution by chapter are summarized in Figure 7.1.

Model of the process

In a first part, a nonlinear model of a typical CCPP with one single level of pressure has been derived based on the first principle library *ThermoPower* [9]. The model has been developed in the Modelica language and implemented in the Dymola environment. In order to enable a complete investigation of the transient phases, the CCPP model has been adapted to the start-up. A detailed model, previously validated by studying a real start-up sequence [10], allowed to derive all the plant data (dimensioning, masses, capacities, etc.). The global behavior of the model has been validated by experts in the domain.

It has been shown that the availability of reliable plant models, able to operate close to the process limits, capturing the most significant dynamics and nonlinearities, is a key aspect to carry on different plant management strategies and to implement innovative control techniques. The use of simulation models, in particular the CCPP model developed in Modelica, for optimization and control purposes can represent a solution. However, the efficient application of these models, regarding computation time, optimality, etc., in model-based methods requires some transformations and compromises in the models structure. Therefore, the CCPP *ThermoPower* model initially designed has been considered as *reference* plant and used to derive and validate models suitable for optimization.

Model of the process for optimization and control purposes

In order to obtain a ready-to-use Modelica model for intended purposes, a methodology which transforms physical models designed for simulation to optimization-oriented models is suggested. The goal has been to derive a CCPP physical model that can be further used with efficient optimization algorithms to improve start-up performances. The class of algorithms envisaged for resolution of optimization problems based on such a model has been gradient-based methods, which can typically address large-scale systems and efficiently solve real-time optimal control problems. In general, the convergence of these algorithms relies on the fact that the right-hand side is twice continuously differentiable, imposing a series of restrictions on the model formulation, in particular that of the lack of discontinuity source. Thus, the proposed methodology is based on the reformulation of the plant initially represented with discontinuity sources into a smooth model, by means of continuous approximations of the Heaviside function. It enables to develop models with the same structure as the original one, on the one side, and suitable for optimization in terms of issues such as accuracy and smoothness, on the other side.

By applying this methodology, a CCPP model is built, and then validated by comparing its dynamic responses to specific simulation scenarios with the ones provided by the original *ThemoPower* model. The validation results demonstrate the model consistency and emphasize the fact that it is sufficiently accurate to be used for optimization and control purposes.

ThermoOpt library

The components systematically derived to build a combined cycle smooth model have been encapsulated into a first-principle models library, so-called *ThermoOpt*. The *ThermoOpt* is an optimization-based version of the *ThermoPower* library, covering engineering needs for an intuitive description format of the power plants and efficient optimization algorithms specifications. The validation of the library elements has been performed against simulation data provided by the original models. Also, in order to validate the fact that the library elements are adapted to optimization, optimal control problems based on the subsystems have been formulated and solved with state-of-the-art solvers.

The model components of the *ThermoOpt* can be further used in different applications with similar purposes, for example, to design a model of combined cycle power plant with two levels of pressure for start-up study. In addition, as a general feature of models written in Modelica, the library provides:

- high flexibility for a rapid model development;
- full access to the source code, enabling an easier model verification and validation;
- declarative encoding of modeling knowledge based on equations;

Optimal control of the start-up process

The start-up control problem is challenging since the process involves transitions over a wide range of operating conditions, exhibiting a highly nonlinear behavior. Moreover, the thermal-mechanical stresses which occur in thick walled components need to be kept under specified limits, and multiple manipulated variables must be coordinated.

In this work, the start-up process has been treated as a dynamic optimization problem, specifically: to find optimal profiles of the control variables which minimize a specific objective function, while respecting the constraints imposed by the plant dynamics and process variables. Two types of optimal control problems have been formulated: the first corresponds to a minimum-time problem which aims at minimizing the start-up time, the second corresponds to a constrained quadratic tracking problem which uses an integral square error criterion to drive the system towards the nominal set point. The optimization problem has been implemented based on the optimization-oriented model (CCPP smooth model) and by imposing a series of constraints on the main plant variables, such as pressures, temperatures and stresses, in order to prevent emergency stops and to preserve the lifetime consumption of the units. Due to the fact that the lifetime consumption during the entire start-up is generally dictated by the maximum values of thermal-mechanical stresses in the most affected components (rotor and superheater header), the constraints have been mostly imposed on the peak values of the stresses.

The CCPP start-up is a complex procedure which includes several phases, and the optimization of the entire procedure requires consideration of a hybrid system containing discrete and continuous variables which interact. To reduce the complexity of optimizing the whole start-up process, the problem has been decomposed into simpler sub-problems corresponding to each start-up phase. Attention has been focused on the last part, the increasing load phase when both turbines are synchronized and connected to the grid, phase considered as the most critical in terms of lifetime consumption.

The proposed approach aims at deriving optimal profiles of the plant control variables, by assuming that these profiles can be described by a parameterized function, whose parameters are computed by solving a constrained optimal control problem (either a minimum-time problem or a quadratic tracking problem, subject to variable limitations and plant dynamics). The idea behind the approach is deduced from the direct methods [82] that approximate the original infinite-dimensional problem by a finite-dimensional problem by using a finite parameterization. The variable of interest for optimization has been the gas turbine load, because it induces most of the dynamic characteristics during the start-up procedure, in particular in the considered phase. It has been also shown that the applicability of the approach can be easily extended to other control profiles.

Different types of parameterization have been proposed in order to obtain the best balance between quality of solutions and computation time required. The approach is able to considerably improve the start-up performances compared to a standard start-up procedure. Also to emphasize the consistency of the approach, the derived profiles computed with the smooth model have been validated by simulation on the original combined cycle model.

Although the performances of the approach are limited by the accuracy and efficiency of resolution of optimization problem, the approach offers the advantage that it can be applied to the whole combined cycle system, providing a solution for optimization of large-scale problems based on Modelica/Dymola models.

Predictive approach for the start-up process

The control design based on Model Predictive Control represents an important solution for the improvement of the start-up performance, due to the fact that it can handle multivariable process models, input and state constraints, providing flexibility in the specification of the control objectives.

In this manuscript, MPC strategy for the gas turbine load control has been implemented and investigated for the start-up process. The specific goal of the control technique has been to reduce the start-up time by means of a consistent manipulation of the instantaneous GT loading rates, while keeping the thermo-mechanical stresses under admissible limits.

The predictive approach relies on the optimization-oriented model developed in Modelica/Dymola, using at each sampling time the proposed optimization strategy with a finite parameterization of the control variable profile. The objective has been defined by means of an integral square error criterion, penalizing the deviation of the control input from the target value represented by the full load.

The results demonstrate that a periodic computation of the GT load profile together with an accurate prediction of the future behavior of the plant make it possible to derive fast loading rates, leading to a significant reduction of the transient time compared to standard procedures, and even compared with the presented open-loop procedure, without exceeding the stress limits.

Despite the proven benefits offered, the indicated MPC solution has been considered as inappropriate for a real-time implementation. The main drawback is the computation time, which at each control step is generally superior to the defined sampling time (apart from a few cases which provide implementable solutions, but with a penalization in start-up time). In order to obtain real-time implementable results a hierarchical predictive control structure (H-MPC) has been suggested. The hierarchical structure includes two layers with different time scales: a high and a low layer. The control problem for each layer has been formulated based on the CCPP smooth model, and aims at determining an optimal profile of the GT load. At each level a controller tackles a different objective. Thus, at the high level a minimum time optimal control problem is solved over a long time period. The solution of this problem is used to update the GT load set-point for the low level, where a constrained tracking optimization problem is solved over a shorter period.

The hierarchical algorithm has been designed with a certain degree of flexibility, which makes it possible to derive optimal profiles with a trade-off between a low computation time and a high start-up efficiency. The results emphasize the fact that the H-MPC leads to start-up performances close to those provided by 1-level MPC solution, but with a computation time compatible with real-time implementation.

The collaboration and the assistance provided by EDF have been important in the development of this research. The support of experts in the field, regarding issues such as operation, modeling and control of combined cycle power plants proved to be, undoubtedly, beneficial. It is important to stress this aspect, not only to emphasize the "applicative/industrial" character of the suggested approaches, but also to underline that the objective has been to provide a study with a general validity, in the sense that, the modeling methodology, optimization procedure and control design methods proposed can be further used for other typologies of combined cycle plants.

7.2 Perspectives

Several possible extensions of the work presented in this dissertation can be envisaged for future developments. The availability of a smooth CCPP model together with an optimization-oriented library opens a wide field of research projects. One direction further concerns the modeling of different real power plant configurations. Specifically, the extension of the CCPP configuration with one level of pressure to the model with three levels of pressure that has been initially used to derive all the plant data implemented in the current CCPP representation. Also, another study is to continue the development of model components for the representation of thermal power plants in order to complete the *ThermoOpt* library.

A future work concerns the improvement of the optimization procedure. Two directions are under consideration:

- analytical computation of information related to derivative terms (i.e. gradients) that can be provided to the optimization solver. The optimization problem can be more accurately and efficiently solved if the derivative information is provided;
- application of state-of-the-art optimization algorithms. For control purposes, it could be beneficial to couple the model with state-of-the-art optimization algorithms which exploit the structure of the model in order to obtain a better performance in terms of convergence properties and execution times.

For a successful implementation of the MPC technique a series of studies are envisaged. Among them, the following are recalled:

- extension of the predictive procedure in order to take into account the use of the original physical model as plant. MPC schemes discussed in this work consider that the actual system is identical to the model used for prediction, i.e. that no model/plant mismatch or unknown disturbances are present. Considering the original physical model as plant, a robustness study with respect to the current implementation can be performed. Also, for a practical application, an MPC framework which addresses robustness issues can be intended.
- design of an observer able to reconstruct the state variables using a Modelica/Dymola model of the physical system. In the formulation of the start-up optimization problem, the system states have been considered totally accessible through measurements. This fact is generally not possible and a state observer must be included in the control loop. Moreover, the initialization of large-scale models in the current states is often difficult, and new solutions to robustly initialize Modelica models must be taken into consideration.

Regarding the hierarchical approach, its implementation requires the consideration of a number of factors about the robustness and reliability of the control with respect to the lower level, considerations that can be envisaged for the future.

An important alternative to deal with the complexity of the control task is the implementation of distributed approaches. A distributed control structure, where the global system is decomposed in several subsystems interacting with each other, has been studied beyond the information presented in this manuscript [134]. The encountered issues regarding the feasibility of the profiles which are changed among the interacting subsystems, as well as the lack of tool support have made difficult the application of the distributed approach in the case of the CCPP start-up problem. Nevertheless, the applicability of a distributed control structure for the CCPP start-up remains questionable, and once with the maturity of optimization Modelica-based tools, a more extensive approach should be envisaged.

The extension of the phase by phase optimization into a hybrid procedure that will cover the whole start-up sequence represents an interesting future study.

APPENDIX A

Modelica language

The features of the dedicated modeling language Modelica make it to be widely used in several industrial fields. In the sequel, the main aspects of the language are briefly reviewed. The presentation is separated into two categories, according to the constructs for behavior definition and according to the mechanisms of model structuring and handling complexity.

Constructs to describe the system mathematical behavior

Equations

As previously pointed out, Modelica is primarily an equation-based language. The equations are used to define behavior of physical systems. The relationships between physical variables, such as temperature, enthalpy, pressure etc., are expressed by equations in a declarative way. Thus, by stating equations in a declarative way provides a higher flexibility compared to other standard programming constructs like assignment statements.

The model equations are placed in sections indicated by the keyword *equation* (see Listing A.1 also the examples presented below). It should be mentioned that the equations in Modelica can occur in different syntactic contexts, not only in equation sections. For example to declare variables or to modify attributes. Also, initial equations to solve the initialization problem can be specified.

The expressions of equations in Modelica can contain repetitive structures, often stated as arrays, the only restriction being that all variables must have the same dimension.

The equations of a Modelica model form an equation system. This system to be simulated, must be primarily solvable. Therefore, a necessary requirement for the system to be solved is that the total number of equations must be equal to the total number of unknowns, in other words the equation system must not be neither undetermined nor overdetermined.

In Modelica language different categories of equations can be represented, including ordinary differential equations, algebraic equations, differential algebraic equations, difference equations, hybrid DAEs and so on. Nevertheless there are a few types such as partial differential equations that are not yet supported by the language.

Algorithms

Even if the equations are suitable to express physical phenomena, there are situations when the systems behavior is more conveniently represented by algorithms. A popular example is the case when the model has to include not only the physical system but also the associated control system.

The algorithms, expressed as sequences of instructions (statements), can occur only within the *algorithm* sections. As in the equations case, there is a specified keyword *algorithm*, which indicates these sections (see Listing A.2). Typically the algorithm sections are used when procedural descriptions are required.

In Modelica, there are different kinds of non-declarative algorithmic constructs available, such as assignment statements, conditional statements(e.g. *if-then-else*) and iteration (e.g. *for-loop*).

```
equation
  ...
  <equations>
  ...
<other allowed keyword>
```

Listing A.1: Equation section

```
algorithm
  ...
  <statements>
  ...
<other allowed keyword>
```

Listing A.2: Algorithm section

Functions

The Modelica language has a limited number of predefined mathematical functions (e.g. *abs*, *mod*, *sqrt*, etc.). This list is completed by other functions (e.g. *sin*,*log*, *exp*, etc.) available in the Modelica standard library, developed together with the language by the Modelica Association [7]. Also the user-defined mathematical functions are supported by the Modelica. In general, the body of a Modelica *function* is represented by an algorithm section that is executed when the function is called. Each formal parameter of the functions are prefixed by either *input* or *output* keywords, the later being used to return the function results (see Listing A.3).

The Modelica functions calls are in general declarative[1], despite the fact that they encapsulate non-declarative constructs. The Modelica functions correspond to the mathematical functions, meaning that for the same arguments it always returns the same results.

Another feature is the possibility to call functions that are defined outside of the Modelica language (external functions). Thus, it enables easy reuse of tested functions and interface with the routines of other software. The declaration of these functions is marked with the keyword *external* (see Listing A.4).

[1]Exception: external functions called in when-equations/when statements or during initialization

```
function sum
  input Real x;
  input Real y;
  output Real s;
algorithm
  s := x + y;
end sum;
```

Listing A.3: Function

```
function ext
  input Real x;
  input Real y;

external

end ext;
```

Listing A.4: External function

Mechanism for model structuring and handling complexity

Models/Classes

In Modelica, as in any object-oriented language, the class and object concept is fundamental. The classes, referred to as models in Modelica, provide the structure for objects (instances), and serve as blueprints to create objects. A Modelica class can contain elements (variable declarations, local class definitions), and sections containing equations and algorithms. Variables contain data associated with classes and their instances. The behavior of these classes is specified by equations together with possible algorithms.

The members of a class can have two levels of visibility: public or protected. The latter, meaning that only the elements inside the class as well as the elements in the classes that inherit this class, are accessible. The default declaration is public.

Also in Modelica the variability of the variables during the transient analysis of a model, can be specified, by using the variability prefixes: *parameter*, *constant*, *discrete*. The term variability is considered as time variability, and define the cases when the value of variables can change and when it is constant. A variable of type Real for example, with the *parameter* prefix, means that it is constant during time-dependent simulation, but can change values between simulations. The *constant* prefix used usually to define mathematical constants, signifies that the variable value can not be changed once it has been defined. The variables with the prefix *discrete* are piecewise constant signals that change their values only at event instants during simulation. When no variability prefix is specified, the default declaration of all quantities in Modelica is assumed to be *continuous*, meaning that its value evolves continuously during the simulation.

A trivial example is considered in Listing A.5. The class *Ex* contains five elements: a local class definition, *Mcl*, and four component declarations: d (the parameter), y, z and *var* (the variables). The behavior of the class *Ex* is defined by three equations, relating to its variables. The function *der* represents the time derivative operator $\frac{d}{dt}$, used to express the dynamic behavior.

```
class Ex
   class Mcl
     Real x;
   end Mcl;
   parameter Real d=1;
   Real y;
   Real z;
```

```
   Mcl var;
equation
   z = d;
   der(y) = y;
   var.x = z;
end Ex;
```

Listing A.5: A class example

In Modelica, almost everything is a class. The class concept is used for different purposes. In order to make the programs code more accessible and easy to read and maintain, special *keywords* are introduced for specific use of the class. There are several restricted classes (specialized), specified by keywords: *model*, *connector*, *record*, *block*, *type*.

A *connector*, is a class used to declare the structure of connection points, which may contain variables but no equations. The *model* restricted class is the most widely used type of class for modeling purposes. Its semantics are almost the same with the general class concept, the only restriction is that a model may not be used in connections. A *type* restricted class is usually used to introduce new type names and can be an extension to a record, an array, or one of the Modelica built-in data types: *Real*, *Integer*, *Boolean*, *String*, *Enumeration*. It should be noted that in Modelica all kinds of classes define types, not only the *type* restricted class.

Another specialized class, *record*, is used to declare a record data structure and may not contain equations or algorithms. Also a class with a specified causality, is the *block*. The interface member variables of this class are declared by using the causality prefix, *input* or *output*.

As the restricted classes are specialized versions of the general concept of class, for a valid model, the specific keywords can be replaced by the *class* keyword without changing the model behavior.

From the modeler point of view, the idea of specialized classes is useful, because it is not necessary to learn several different concepts, but only one: the class concept. Most of class properties, such as syntax and semantic of definition, instantiation, inheritance, genericity are the same for all kinds of specialized classes.

There are another two concepts in Modelica: *package* and *function*. These concepts are more than restricted classes, they can be also considered as enhanced classes. The *function* concept corresponds to the mathematical functions and its syntax and semantics are close to the *block* restricted class. A *package* is mainly used to manage namespaces and organize Modelica code. The *packages* can contain definitions of constants and all kinds of restricted classes. The requirements for a *package* is to not contain non-constants, i.e, parameters, variables. By using the *import* statement, it can get access to the definitions of other packages.

Typing

Modelica is a strongly typed language. The strong typing in Modelica has been introduced to provide a partial verification of internal consistency, thus reducing the risk of errors in modeling. By typing, precise data types are defined. For example, in the case of the dominant data type in physical systems modeling, the *Real* variables, for a more precise characterization and differentiation, have a set of attributes such as unit of measure, quantity, initial value, minimum and maximum value, etc. Some of these attributes can be modified when a new type is declared. Listing A.6 shows an example of derived type *Temperature* and a variable declaration to be a temperature in Kelvin instead of a *Real* variable.

```
type Temperature = Real(unit="K");
     Temperature T;
```

Listing A.6: Example derived type

where *Real* is the predefined variable type, and the units of measure are set according to the International System of Units (SI).

Partial Models and Inheritance

One of the key feature of the object-orientation is the ability to extend the behavior and the properties of an existing class. The original class, referred as superclass or base class, is expanded to create a more specialized class, referred as the subclass or derived class. In this process, the subclass *inherits* (reuse) the data and the behavior of the base class, in the form of variable declarations, equations, and other contents. In Modelica the inheritance is specified by using the *extends* clause. Furthermore, multiple inheritance, i.e. several extends statements, is supported.

An example of extending a simple Modelica class is presented in Listing A.7. The base class *PointData* has two variables (x, y) representing the coordinates of a point in space, and the derived class *Point*, that inherits these variables to represent the point and adds an equation as a constraint on the point position. The equivalent of the expanded *Point* class is shown in Listing A.8.

Also, in Modelica there is the notion of *partial class*, usually known as *abstract class* in other object-oriented languages. In general, it is used to define an "interface" model class that captures the common property of components. The keyword *partial* indicates that the model is incomplete, and in order to obtain a well-defined model, constitutive equations must be added.

```
class PointData
  Real x;
  Real y;
end PointData;

class Point
  extends PointData;
equation
  x + y = 0;
end Point;
```

Listing A.7: Example inheritance

```
class PointExpanded
"Equivalent to Point class"
  Real x;
  Real y;

equation

  x + y = 0;

end PointExpanded;
```

Listing A.8: Equivalent expanded class

Hierarchical structuring

Complex physical systems are naturally composed of several connected components, where a part of these components can be hierarchically decomposed into distinct subcomponents through multiple levels. This technique to decompose the complex models in simpler parts which are combined in a hierarchical approach is fully supported by Modelica. Thus, an organization of structured models is performed. In order to formulate structured models, the interaction between components on the same hierarchical level is achieved via a connection mechanism, by using specialized class *connector* and the built-in function *connect*.

The *connector* class acts as an interface between components. The communication interface, defined by the connectors (often called ports in literature), consists of a set of variables. These variables are basically of two types: *non-flow* (by default) and *flow* (prefixed with the keyword *flow*). Non-flow refers to potential variables, e.g. pressure, temperature, voltage, etc., while variables of flow type usually represent some kind of flow, e.g. fluid flow rate, electrical current, force, etc.

Depending on the type of variables, two kinds of coupling, relied on laws of nature, are established by connections: for potential variables, these are set equal (equality coupling), while the sum of the flow variables is set to zero (sum-to-zero coupling), at the connection point.

The Modelica connector classes can be used to design new interfaces and can be formulated for different application domains. Currently a large number of well-designed connectors in Modelica libraries are available. The only restriction, to realize the connection between systems, is the fact that the components must have the same type of connector.

The connection between connectors are performed as equations. During translation of Modelica models, the *connect* statements are transformed in normal equations. For example, considering the Listing A.9, the *Flange* connector class describes an interface for interaction in thermodynamic domain by specifying the flow variable represented by the mass flow rate w and the potential variable represented by pressure p at the interaction point (Figure A.1).

```
connector Flange
  Pressure p;
  flow MassFlowRate w;
end Flange;
```

Listing A.9: Flange connector

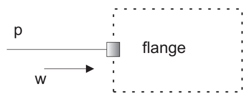


Figure A.1: A component with a flange connector

In Modelica there are three kinds of connection, reflecting three design situations. The first is *acausal connection*, when the interaction is between two physical components involving energy flow. In this case the data flow direction in connection is not explicitly specified. The second is *causal connection*, when signals (information) are exchanged between components. In the connector class, in this case, the data flow is specified by using the input/output prefixes. The last type is *composite connection*, for the complex interaction between components, including several connections of type acausal and causal.

In Figure A.2, an example of two connected thermodynamic flanges *flangeA* and *flangeB* is shown. The connectors *flangeA* and *flangeB* are the instances of the connector class *Flange* (Listing A.10).

```
Flange flangeA;
Flange flangeB;

connect(flangeA, flangeB);
```

Listing A.10: Connect equation

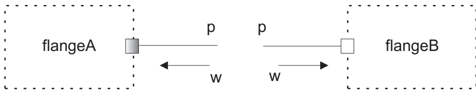


Figure A.2: Connection two flanges

The connection equation: **connect**(flangeA, flangeB) connects the two flanges so that these form a node. This produces two equations, namely:

$$flangeA.p = flangeB.p \qquad * \textit{the connected pressures are equal};$$
$$flangeA.w + flangeB.w = 0 \qquad * \textit{the sum of the flow rate is zero.}$$

```
partial model BaseM
"Basic interface"
  Flange infl;
  Flange outfl;
end BaseM;

model FM "Fluid flow model"
  extends BaseM;
  parameter Real k=100;
  MassFlowRate w;
equation
//Mass and Momentum balance
  infl.w + outfl.w = 0;
  infl.p - outfl.p = k*w;
  infl.w = w;
end FM;

model SourceW "Flow rate source"
  parameter MassFlowRate w0=70;
  MassFlowRate w;
  Flange infl;
equation
  infl.w = -w;
  w = w0;
end SourceW;
```

Listing A.11: Component models

```
model SinkP
"Pressure sink"
  parameter Pressure p0=1.01325e5;
  Pressure p;
  Flange outfl;
equation
  outfl.p = p;
  p = p0;
end SinkP;

model SwPSp
"Model assembly of components"

//creation of instances
  FM Pipe;
  SourceW SW;
  SinkP SP;

equation
//connection SW with Pipe
  connect(SW.infl, Pipe.infl);

//connection Pipe with SP
  connect(Pipe.outfl, SP.outfl);
end SwPSp;
```

Listing A.12: Additional models

In Listings A.11, A.12, a few classes describing elementary thermodynamic components are presented. The presentation of these examples is intended to show some important features of the Modelica language, above listed, such as hierarchical modeling, inheritance, and the connection mechanism. The *Flange* connector (see Listing A.9) is used to formulate a partial model, *BaseM*, for a generic thermodynamic component with two ports. This model, at its turn, is completed with equations in order to define a specific behavior of component. In this case, two equation based on energy and mass balance are given in the specialized model, *FM*. It should be noted that different models corresponding to thermodynamic components relied on the partial model can be defined. To create a simple model that can be simulated, two models corresponding to a flow rate source (*SourceW*) and a pressure sink (*SinkP*), are presented. Thus, a model (*SwPSp*), consisting of flow rate source, a fluid flow (pipe) and a pressure sink, is simply obtained by their connection.

Modifications and Class Parameterization

As mentioned in previous sections, the key concept to reuse the modeling knowledge in Modelica is the class and the inheritance mechanism. Moreover, in order to provide a more flexibility in reuse, Modelica supports inheritance through modification by means of so-called modifiers. Actually, in a single Modelica declaration three operations can be expressed: object instantiation, inheritance and specialization by modification.

By using the modification mechanism, the classes behavior can be changed, e.g. add/change the default value in a declaration or replace the original declaration of the elements by new declaration. This latter type of modification is called a *redeclaration*. In order to reduce the risk of modeling errors the redeclaration requires in the modifier the use of the special keyword *redeclare*.

In most cases the redeclaration requires that the modified element be declared with the prefix *replaceable*, except the case when the change aims only to restrict the array dimension size and/or variability of a declared variable. The Listings A.13, A.14 show these types of redeclaration and the equivalent class respectively. In fact, the main mechanism in Modelica, to create very flexible models, is represented by replaceable classes. By redeclaration, a part of the original declaration is automatically inherited by the new declaration, thus for modeler it is easier to write code, without the need to repeat common parts of the declarations.

```
class EP
  Real[:] x;
end EP;

class B = EP(redeclare parameter Real x[5]);
```

Listing A.13: Element redeclaration

```
class Bexp
"Equivalent to B class"
  Real[5] x;
end Bexp;
```

Listing A.14: Equivalent class

The components redeclaration in general has to fulfill some type compatibility conditions, defined by the declaration of the original component. For example the element used in a redeclaration must be a *subtype* of the corresponding replaceable element in the modified class. These compatibility rules are in detail treated in [96], [97].

A few different cases of modification/redeclaration, supported by Modelica, which respect these conditions are illustrated in Listings A.15, A.16. In Listings A.15, the modifiers are applied to elements that are variables (*T1* of type *Real* default declaration replaced by unit="K"), in an extends clause (the class *Pex* inherits the class *Ppa* and applies the modifier *p0=1e5*, or in the case of *Pbar*, the modifiers are applied to the local class *Pressure* within the class *Ppa*, changing thus the unit), hierarchical modification (the instance *Ph* of the class *Ppa*, with modified parameter, is created).

The Listing A.16 shows an example of redeclaration, *redeclare MP m(x=1)*, that replace the original declaration *MP m*, from the class *EP*, and also a redeclaration of the subtype *SS*, both used in an extends clause.

```
//Modifier to variable
Real T(unit="K");

class Ppa
  type Pressure = Real(unit="Pa");
  Pressure P;
  parameter Real p0;
end Ppa;

//to local class
class Pbar
  extends Ppa(Pressure(unit="bar"));
end Pbar;

//nested extends clause
class Pex
  extends Ppa(p0=1e5);
end Pex;

//hierarchical
Ppa Ph(p0(unit="bar") = 1);
```

Listing A.15: Modification examples

```
//Redeclaration
class MP
  parameter Real x;
end MP;

class SS "subtype of MP"
  parameter Real x=10;
  parameter Real y;
end SS;

class EP
  replaceable MP m;
end EP;

class EPs
  extends EP(redeclare MP m(x=1));
end EPs;

class EM
  extends EP(redeclare SS m(y=1));
end EM;
```

Listing A.16: Redeclaration examples

The redeclaration approach enables the use of the parametrized generic classes, commonly known as generics. Instead to write code for several similar models, a significant amount of coding as well as software maintenance can be reduced by expressing a general structure of the problem ("prototype model"), with specific information provided as formal parameters. The *formal class parameters* makes from Modelica a flexible modeling language, thus depending on situation and purpose, simplified models can be refined into a high precision models, by exchanging sub-models or even whole sub-systems.

In Modelica two cases of class parameterization can be noted: formal class parameters are components (i.e. when objects are used as values) or *types* (i.e. when types are used as values). In Listing A.17, the first case when class parameters are variables (components) is presented.

Consider the description of the thermodynamic circuit (*SwPSp*), previously presented in Listing A.12. In this case, *SwPSp* has a formal class parameter *Pipe* (components of *SwPSp*), marked by the keyword *replaceable*, and two components *SW*, *SP* that are not class parameters because are not declared *replaceable*.

Assuming that there is a more detailed model of the fluid flow, *FMc*, subtype of *FM*, the target is to create a model (called *C*), with the same structure but containing *FMc*.

This is easily performed, by exchanging the *FM* with the new model, *FMc*, in other words, the default declaration of the class parameter *Pipe* is changed in the definition of the class *C*. The equivalent of this class is also illustrated (*CC*).

An example relating to the second case, when the class parameters are types, is shown in Listing A.18. Considering this example, it is possible to change a set of models by defining a type parameter *Mr* in the generic model *SwPSp*, all objects having this type can be then changed (e.g. from the default type *FM* to *FMc* type).

```
model SwPSp

  replaceable FM Pipe;
  SourceW SW;
  SinkP SP;
equation
  ...
end SwPSp;

model C = SwPSp(redeclare FMc Pipe);

//class equivalent to C
model CC
  FMc Pipe;
  SourceW SW;
  SinkP SP ;
equation
  ...
end CC;
```

Listing A.17: Modification examples

```
model SwPSp
  replaceable model Mr=FM;
  Mr Pipe;
  SourceW SW;
  SinkP SP ;
equation
  ...
end SwPSp;

model C = SwPSp(redeclare Mr=FMc);

//class equivalent to C
model CC
  FMc Pipe;
  SourceW SW;
  SinkP SP ;
equation
  ...
end CC;
```

Listing A.18: Redeclaration examples

Annotations

Apart from the features above mentioned, Modelica provides a particular construction, called annotation, to store additional information about the elements of the models. This feature is used by Modelica environments for example to describe a graphical representation of the icons and connections associated with models or to support model documentation. In general, the annotations are widely used by library developers. By graphical representation, the models become easier to understand, and the users can create models based on elements of libraries, in a drag and drop manner, without directly working with the Modelica code. The graphical annotations are also represented textually in the model code, thus making possible a textual editing. This is seldom made because most tools provide a graphical modeling environment. In addition, in Modelica language, special annotations can be defined for functions, in order to obtain a higher performance and/or numerical accuracy.

APPENDIX B

Modelica tools

Currently there are several software tools for physical system modeling based on the open-source standard Modelica. Some such tools are briefly presented in the following, for more information, see [7].

Generally any Modelica environment needs to contain the following modules:

- a Modelica compiler (interpreter), that translates the Modelica model into a form which can be efficiently simulated. Symbolic manipulation algorithms are required;
- an execution and run-time system, to simulate the translated model with standard numerical solvers and plotting of the results;
- a text editor for models editing, that can be a specialized Modelica or any standard text editor;
- a graphical editor for graphical representation of models and browsing.

These features can be found in different variants in all Modelica tools presented in the following lines.

Dymola

Dymola, Dynamic Modeling Laboratory [8], is a modeling and simulation environment based on Modelica language. Dymola provides a powerful Modelica translator that is able to perform symbolic manipulation for large-scale systems (more than 100 000 equations), as well as for real-time applications, currently being considered the leading Modelica software. Apart from the common features above listed, graphical model editing, simulation and plotting, Dymola enables model calibration, parameter optimization and real-time Hardware in the Loop Simulations (HILS). In addition, Dymola can be connected with Matlab by means of Dymola-Symulink interface (DymolaBlock). Dymola offers also a wide range of libraries, covering different engineering domains (e.g. mechanical, electrical, thermodynamical, etc.), with the open possibility of the user to create models according to their specific needs, using the templates and elements from these libraries.

OpenModelica

The OpenModelica [140] is an open source project that supports the full Modelica language but with a lower level of efficiency, in terms of computation and solving the equations systems, than Dymola. OpenModelica uses MetaModelica, an extension of the Modelica specifically designed to expand the tool functionality.

MathModelica

MathModelica [141] is a commercial Modelica environment relied on the integration of Modelica with the Mathematica language. The subset Modelica is used for object-oriented modeling and simulation whereas the Mathematica part is used for interactive scripting, providing extensibility and flexibility in functionality. MathModelica provides a customizable set of Modelica component libraries but with a reduced number compared to Dymola.

MapleSim

MapleSim [142] is a multi-domain modeling and simulation tool, supporting Modelica. MapleSim combines Modelica libraries with advanced symbolic computing technology and high-performance numerical solvers, taking advantage of Maple's features.

SimulationX

SimulationX [143] is a graphically-interactive tool for modeling, simulation and analysis of multi-domain systems. The whole range of the SimulationX computation methods, can be applied to all kinds of Modelica models. SimulationX offers also the possibility to export models to various platforms such as Matlab/Simulink, dSPACE, Simpack etc.

This section presents only a few most popular environments with different levels of support for Modelica, the list goes on. As the adoption of Modelica as standard for dynamic modeling is steadily increasing, in the near future, it is expected that the level of support as well as the number of tools for Modelica to increase.

JModelica.org

The project is described as *an extensible Modelica-based open source platform for optimization, simulation and analysis of complex dynamic systems*. JModelica.org tries to fill the gap between modeling/simulation tools and the numerical optimization algorithms. Effectively, the platform enables a high-level description of the models, simulation, analysis and optimization, in a fully integrated way. The formulation of dynamic optimization problems is performed into an extension of the Modelica language, called Optimica [144]. This extension complements Modelica with language constructs dedicated to optimization (e.g. cost functions, constraints, quantities to optimize), enabling the user to use high-level descriptions to express optimization problems. As a key feature of the Optimica is the fact that the formulation of the optimization problem is performed independently of the numerical algorithm used to solve the problem. The algorithm and associated control parameters can be specified using language constructs, making easy the applicability of different algorithms. The JModelica.org platform implements state of the art numerical algorithms e.g. simultaneous optimization method based on Lagrange polynomials on finite element with Radau points. In particular, Jacobians and sparsity information are exploited in order to increase the efficiency of the algorithm. The non-linear program resulting from collocation is then solved by the solver IPOPT.

Currently the platform is under development, with several constraints on the models formulation, complexity and syntax, limitations which are highlighted in the content of this thesis, limitations that have made our attempt to use this environment to deal with the start-up optimization problem based on the developed CCPP models to fail.

APPENDIX C

CCPP identified model

A solution to derive simple models of combined cycle power plant suitable for control purposes is to use an identification/interpolation procedure. The principle of the method is here presented.

Identification and interpolation procedure [1]

According to the approach proposed in [145], [146], the nonlinear combined cycle model can be considered as composed by a number of local linear identified models combined by means of a scheduling variable, which describes their local validity through suitably defined membership functions. As in the start-up phase one of the most important variables is represented by the load level signal *GTLoad*, this has been selected as scheduling variable, i.e. the variable specifying the current operating condition. The Modelica developed model has been integrated in Matlab by using Dymola-Simulink interface, and several transients to identify models for different operating points have been generated. The identification of each simulation transient has been performed by means of the Matlab System Identification Toolbox [147]. The considered approach is based on the following main steps:

I. Consider that the operating set of the model is partitioned into n (n=10) disjoint operating regimes. To each regime corresponds a significant value of the *GTLoad*. According to the normal functioning of this type of plant, the selected plant operating points are:

   - 7.5% of full load;
   - 15% of full load;
   - 25% of full load;
   - 40% of full load;
   - 50% of full load;
   - 57% of full load;
   - 60% of full load;
   - 65% of full load;
   - 75% of full load;
   - 100% of full load.

[1] Work carried out in collaboration with Dipartimento di Elettronica e Informazione, Politecnico di Milano

It should be noted that this set of values has been selected to cover the overall operating range of the system and is more dense around 60% where the plant model exhibits a strongly nonlinear behaviour.

II. In order to generate appropriate identification data, different simulations of the Dymola/Modelica model, with small step variations around each operating point of the interest variable, have been performed. For the determination of the steady state conditions at any point, the Dymola simulator has been used. It should be noted that in the case of the CCPP system, the model to be identified has been initialized in a state corresponding to a hot start-up phase and has three signals of interest: GT level load, ST throttle valve position and the desuperheater flow rate. The drum feed-water flow rate is controlled by a local regulator to avoid the problems of an excessive increase/decrease of the level inside the drum, while the other inputs: the bypass position and the grid breaker are maintained constant, with the bypass valve completely closed and the ST connected to the grid.

III. The generated data have been subsequently used to identify linear models with local validity. Thus, a local model (M_i, with $i = 1, ..., n$) has been associated to each operating regime. These models have been estimated by means of standard identification procedures (e.g. based on the Least Squares method), using ARX (AutoRegressive with eXogenous variable) and OE (Output Error) structures. In general the use of first and/or second order models are adequate to describe the transients of the outputs with satisfactory results for the proposed purposes. For example, the comparison between the transients corresponding to the simulation data model with those provided by the estimated model, when a step positive variation of the *GTLoad* at the 75% of its nominal value is applied, are shown in Figures C.1a and C.1b. As it can be seen, the obtained responses with the two models, in the case of two interest variables during start-up, steam temperature and header stress, are quite close, showing thus the consistency of the identified model.

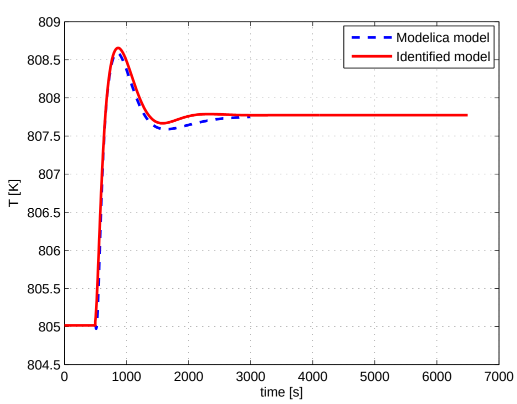
IV. Once the local linear models have been estimated, the overall identified plant model (M) has been obtained by suitably interpolating at any operating point the local linear models through the use of weights:

$$M = \sum_{i=1}^{n} \mu_i(\phi) M_i \tag{C.1a}$$

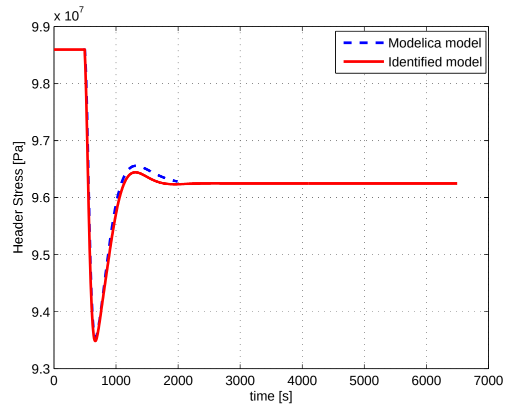
$$\mu_i(\phi) \in [0, 1] \tag{C.1b}$$

$$\sum_{i=1}^{n} \mu_i(\phi) = 1, \quad \forall \phi \in \Pi \tag{C.1c}$$

where ϕ represents the scheduling variable, *GTLoad*, chosen to cover if possible the overall operating set of the system (Π) and μ_i denotes the weighting functions which define the validity of the local estimated models. The weights have been specified by



(a) Step responses of steam temperature

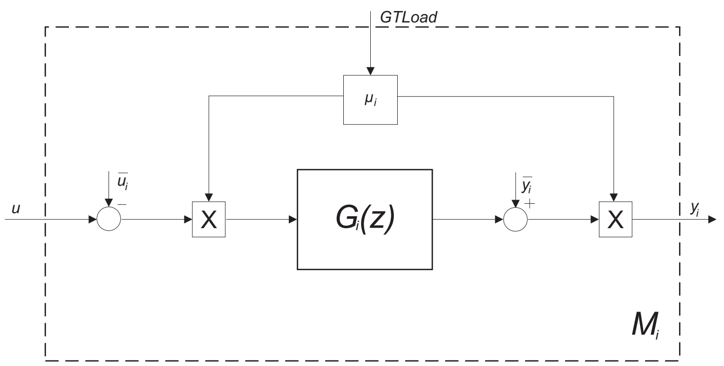


(b) Step responses of header stress

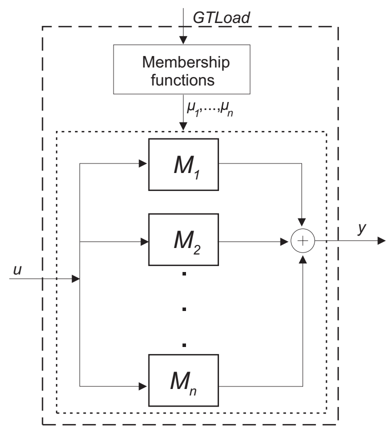
Figure C.1: Comparison between the Modelica and identified model responses at 75%

a series of adequately chosen membership functions. The tuning and the definition of these functions have been performed according to a ”trial-and-error procedure’, each one providing the ”degree of truth” (C.1b) of the *i-th* local model based on the current value of the GT load. In other words, the membership functions have been selected so that for any value of the *GTLoad*, the weight associated to the model estimated at the nearest operating point to provide a higher contribution than the others. It must be noted that the sum of the membership functions associated to the operating points, where the local models have been identified, is always equal to one (C.1c).

The structure of each local linear model as well as the complete interpolated model are reported in Figures C.2a and C.2b. In these figures, u denotes the input vector, y_i is the



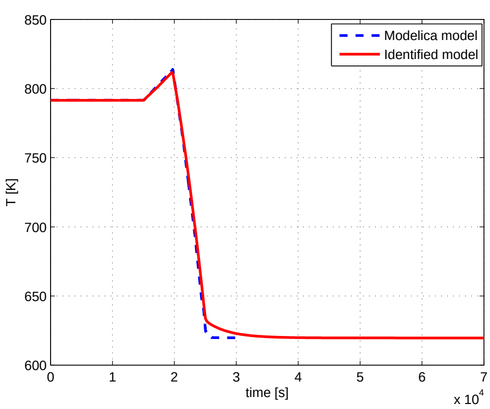
(a) *i-th* local linear model with membership function



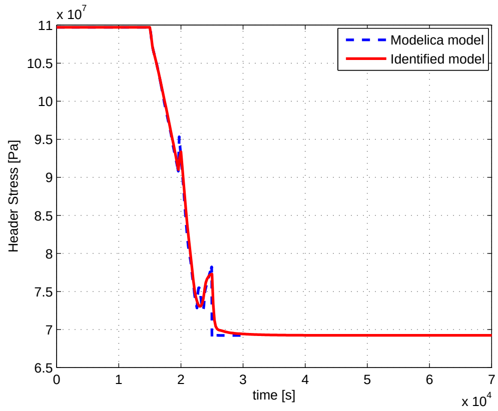
(b) Structure of the overall model

Figure C.2: Local and overall models structure output of the local model M_i, $G_i(z)$ is the identified transfer function of the local linear model, $(\overline{u}_i, \overline{y}_i)$ is the input-output equilibrium pair for the *i-th* working point specified by the corresponding value of the $GTLoad$ and y represents the output of the overall model obtained as the sum of the outputs y_i, $i = 1, ..., n$ of the local models M_i.

The consistency of the model developed by the identification and interpolation procedure has been tested in several simulation scenarios. In addition to the local comparison of responses generated by each identified model with those provided by the Modelica simulator, several large transients have been performed. The results of such a scenario when a wide ramp variation of the $GTLoad$ (from 100% to the 15% of the nominal load) is applied, are reported in Figures C.3a and C.3b.



(a) Steam temperature responses



(b) Header stress responses

Figure C.3: Comparison between the Modelica and identified model responses corresponding to a large transient (from 100% to the 15% of the GT nominal load))

It is apparent that the identified model can reproduce the plant dynamics of interest with satisfactory accuracy, both for local and for large variations of the working point. The only variable that requires a deeper investigation is represented by the rotor stress of the ST, that according to the results given by an optimization procedure exhibits an incompatibility between the real simulator behavior and the one obtained with the identified model. This is probably due to the insufficient accuracy of the identified model for the rotor stress variable, and as a possible solution, a more appropriate identification of the stress model in the rotor section has to be done.

Concerning the adopted procedure to obtain the interpolated identified model, the number of operating points is important but it must be emphasized that the most cumbersome and time consuming phase concerns the definition and tuning of the membership functions. In this regard, automatic procedures based, e.g., on the optimization of the quality of the open-loop simulation results over large transients could be conceived to reduce the development time. Furthermore, a direct analysis of a classical start-up transient to tune the membership functions can be envisaged.

Concluding, the proposed method appears to provide promising results. Typically, the method performance depends strongly on the quality of the identified linear models as well as on the membership functions specification. The procedure can be further improved, in particular with an optimized selection of the membership functions and with the consideration of the other signals (e.g. throttle valve position and desuperheater flow rate). Indeed, the later will require a more complete procedure that deals with the definition of the operating points where to perform local model identification and the subsequent optimization of the corresponding membership functions.

APPENDIX D

Structure of the ThermoOpt library

The optimization-oriented model components developed in this dissertation have been included in the *ThermoOpt* library.

The *ThermoOpt* library is composed of 7 main packages and 2 auxiliary packages. The overall structure of the *ThermoOpt* library is shown in Figure D.1.

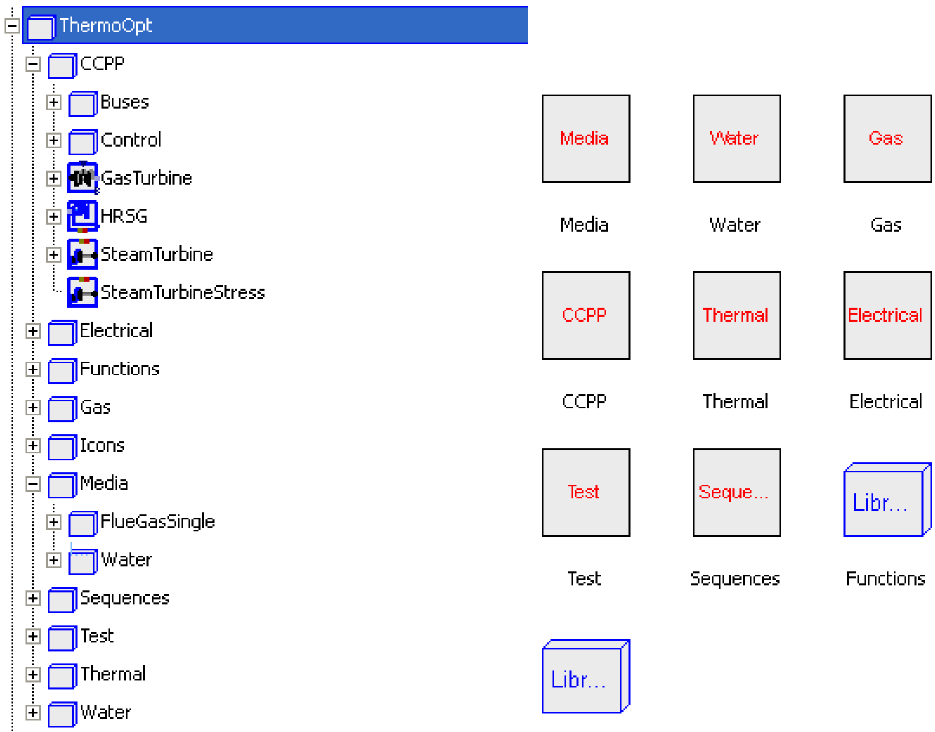


Figure D.1: Structure of the *ThermoOpt* library

Package *CCPP*

The package *CCPP* contains the models of the main units of the plant: GT, HRSG, ST. The package includes also two sub-packages: *Buses* and *Control*. The sub-package *Buses* has bus definitions (empty connectors defined by expansion) used to make the connections simpler for users. The package *Control* has a block model which can provide P, PI, PD, and PID controllers with anti-windup

Package *Media*

The package *Media* contains two sub-packages: *Water* and *FlueGasSingle*. The package *Water* provides a water/steam properties model developed according to the smoothness requirements, while the package *FlueGasSingle* provides a medium model for a typical gas turbine exhaust, also adapted to the optimization requirements.

Package *Water*

The provides models of components that use water/steam as working fluid. The package contains models of elementary components such as valves, pumps, mixers, headers, drum boilers, ideal flow sources, turbines, etc.

Package *Gas*

The package *Gas* provides models of components that use ideal gases as working fluid. Similarly to the *Water*, this package contains models of elementary components.

Package *Thermal*

The *Thermal* package includes basic modeling blocks to describe heat transfer phenomena.

Package *Electrical*

The *Electrical* package contains idealized models of electric power components.

Package *Functions*

The *Functions* package includes several functions used to describe behavior of subsystems or for a smooth approximation (e.g., the smooth approximation of the *abs* and *max* functions).

Package *Sequences*

The package *Sequences* contains several shutdown and start-up sequences used in the phases of validation and initialization of the CCPP model.

Package *Test*

The package *Test* contains a few test cases for the component models of the library.

APPENDIX E

Modelica components representation

The CCPP model components included in the *ThermoOpt* library have been developed for optimization purposes. A few examples of models implemented as Modelica code with Dymola tools are given in Listings E.1-E.3.

```
model GT "Gas Turbine model"
 replaceable package FlueGas = Media.FlueGasSingle "Exhaust gas medium";

  parameter Power maxPower=235e6 "GT nominal power";
  parameter MassFlowRate FGNom=585.50 "Nominal flue gas flow rate";
  parameter MassFlowRate FGMin=454 "Minimum flue gas flow rate";
  parameter MassFlowRate FGOff=FGMin/100 "Flue gas flow rate GT off";
  parameter MassFlowRate fuelNom=12.1 "Nominal fuel flow rate";
  parameter MassFlowRate fuelInt7.08 "Intermediate fuel flow rate";
  parameter MassFlowRate fuelMin=4.58 "Minimum fuel flow rate";
  parameter MassFlowRate fuelOff=0.1 "Flue gas flow rate GT off";
  parameter Temperature FGNomT=843 "Maximum flue gas temperature";
  parameter Temperature FGMinT=548 "Minimum flue gas temperature";
  parameter Temperature FGOffT=363.15 "Flue gas temperature GT off";
  parameter Real constTL=0.60 "Fraction of load temperature constant";
  parameter Real intLoad=0.42 "Intermediate load for fuel consumption";
  parameter Real k=1000 "Smoothness coefficient";

  FlueGas.BaseProperties gas "Gas properties";
  Gas.FlangeB flueGasOut "Flange connector";
  Modelica.Blocks.Interfaces.RealInput GTLoad "GT unit load in p.u.";
  Modelica.Blocks.Interfaces.RealOutput GT_Power;
  Modelica.Blocks.Interfaces.RealOutput GT_Fuel;

  MassFlowRate w "Gas exhaust flow rate";
  Power P_el = GTLoad*maxPower "Electrical power output";
  MassFlowRate fuelFlowRate "Fuel flow rate";

//Heaviside approximation
```

```
  Real Ht ;
  Real Hw ;

equation
// Smooth approximations
 Ht = 1/(1+exp(-k*(GTLoad-constTL)));
 Hw = 1/(1+exp(-k*(GTLoad-intLoad)));
//Gas exhaust temperature
 gas.T = Ht*FGNomT+(1-Ht)*(FGMinT+GTLoad/constTL*(FGNomT-FGMinT));
//Gas Exhaust flow rate
 w = Ht*(FGMin+(GTLoad-constTL)/(1-constTL)*(FGNom -FGMin))+(1-Ht)*(FGMin);

 fuelFlowRate = Hw*(fuelInt+(GTLoad-intLoad)/(1-intLoad)*(fuelNom-fuelInt))
               +(1-Hw)*(fuelMin+GTLoad/intLoad*(fuelInt-fuelMin));

 flueGasOut.w = -w;
 flueGasOut.hBA = gas.h;
 GT_Power = P_el;
 GT_Fuel = fuelFlowRate;


//initial ThermoPower model
 gas.T = noEvent(if GTLoad > constTL then FGNomT
                                   else
                if GTLoad > 0 then FGMinT+GTLoad/constTL*(FGNomT-FGMinT)
                             else
                FGMinT*(1+GTLoad)-FGOffT*GTLoad);
 w = noEvent(if GTLoad > constTL then FGMin+(GTLoad-constTL)/
           (1-constTL)*(FGNom-FGMin)
                               else
            if GTLoad > 0 then FGMin
                         else
            FGMin*(1+GTLoad)-FGOff*GTLoad);
 fuelFlowRate = noEvent(if GTLoad > intLoad then fuelInt+(GTLoad-intLoad)/
                      (1-intLoad)*(fuelNom-fuelInt)
                                          else
                       if GTLoad > 0 then fuelMin+GTLoad/intLoad*
                       (fuelInt-fuelMin)
                                     else
                       fuelMin*(1+GTLoad)-fuelOff*GTLoad);
//
end GT;
```

Listing E.1: GT model

```
model STStage "Impulse stage of a steam turbine"
 replaceable package Medium = Media.Water "Medium model";

  parameter Pressure pstart_in "Inlet pressure start value";
  parameter Pressure pstart_out "Outlet pressure start value";
  parameter MassFlowRate wstart "Mass flow rate start value";
  parameter SpecificEnthalpy hstartin "Inlet enthalpy start value";
  parameter SpecificEnthalpy hstartout "Outlet enthalpy start value";
  parameter MassFlowRate wnom "Inlet nominal flowrate";
  parameter Real eta_mech=0.98 "Mechanical efficiency";
```

```
  parameter Real eta_iso_nom=0.92 "Nominal isentropic efficiency";
  parameter Area An "Nozzle exhaust area";
  parameter CoefficientOfHeatTransfer gamma_n "coef. at nominal flow rate";
  parameter Real k=1000 "Smoothness coefficient";

  Medium.BaseProperties steam_in(p(start=pstart_in),h(start=hstartin));
  Medium.BaseProperties steam_iso(p(start=pstart_out),h(start=hstartout));

  Angle phi "shaft rotation angle";
  Torque tau "net torque acting on the turbine";
  AngularVelocity omega "shaft angular velocity";
  MassFlowRate w(start=wstart) "Mass flow rate";
  SpecificEnthalpy hin(start=hstartin) "Inlet enthalpy";
  SpecificEnthalpy hout(start=hstartout) "Outlet enthalpy";
  SpecificEnthalpy hiso(start=hstartout) "Isentropic outlet enthalpy";
  SpecificEntropy s_in "Inlet entropy";
  Pressure pout(start=pstart_out) "Outlet pressure";
  Real PR "pressure ratio";
  Power Pm "Mechanical power input";
  Real eta_iso "Isentropic efficiency";
  Velocity u(start=0.1) "Nozzle outlet velocity";
  Real S_abs "smooth approximation";

  Modelica.Blocks.Interfaces.RealInput partialArc;
  Modelica.Mechanics.Rotational.Interfaces.Flange_a shaft_a;
  Modelica.Mechanics.Rotational.Interfaces.Flange_b shaft_b;
  Water.FlangeA inlet(p(start=pstart_in));
  Water.FlangeB outlet(p(start=pstart_out));
  Thermal.DHThtc fluidAtRotorSurface(final N=1) "Fluid at rotor surface";

equation
//basic definition

 PR=inlet.p/outlet.p "Pressure ratio";
 partialArc =1 "Default value";

//outlet enthalpy computed by isentropicEnthalpy function
 steam_iso.p = pout;
 s_in=Medium.specificEntropy(steam_in.state);
 s_in=Medium.specificEntropy(steam_iso.state);

 hiso=steam_iso.h;
 hin-hout=eta_iso*(hin-hiso) "Computation of outlet enthalpy";
 Pm=eta_mech*w*(hin-hout) "Mechanical power from the steam";
 Pm = -tau*omega "Mechanical power balance";

//mechanical boundary conditions
 shaft_a.phi = phi;
 shaft_b.phi = phi;
 shaft_a.tau + shaft_b.tau = tau;
 der(phi) = omega;

//steam boundary conditions and inlet steam properties
 steam_in.p=inlet.p;
 steam_in.h=inlet.hBA;
```

```
 hin=steam_in.h;
 hout=outlet.hBA;
 pout=outlet.p;
 w = inlet.w;

 inlet.w + outlet.w = 0 "Mass balance";
 inlet.hAB = outlet.hBA "Close balance equation";

//impulse stage description
 eta_iso = eta_iso_nom "Constant efficiency";
 hin = hiso + u^2/2;
 w = steam_iso.d*An*u;
 fluidAtRotorSurface.T[1] = steam_iso.T "Fluid temp. at rotor surface";

//initial ThermoPower model
///////////////////////////
 fluidAtRotorSurface.gamma[1] = gamma_n*(abs(w)/wnom)^0.8 "heat transfer";
//////////////////////////

//replaced by
 S_abs=(((2*w)^2+k^2)^0.5)/2;
 fluidAtRotorSurface.gamma[1] = gamma_n*(S_abs/wnom)^0.8 "heat transfer";
end STStage;
```

Listing E.2: ST impulse stage model

```
model Condenser "Pressure sink for water/steam flows"
 extends Icons.Water.SourceP;
 replaceable package Medium = Media.Water "Medium model";

  parameter Pressure p0=1.01325e5 "Nominal pressure";
  parameter Pressure pmin=0.1e5 "Minimum pressure";
  parameter Pressure pmax=0.15e5 "Maximum pressure";
  parameter MassFlowRate wmin=18 "Minimum flow";
  parameter MassFlowRate wmax=82.5 "Maximum flow";
  parameter SpecificEnthalpy h=1e5 "Nominal specific enthalpy";

  Pressure p;
  Modelica.Blocks.Interfaces.RealInput in_w;
  Water.FlangeA flange;

equation
 flange.p = p;
 p = ((pmax-pmin)/(wmax-wmin))*in_w;
 flange.hAB =h;
end Condenser;
```

Listing E.3: Condenser model

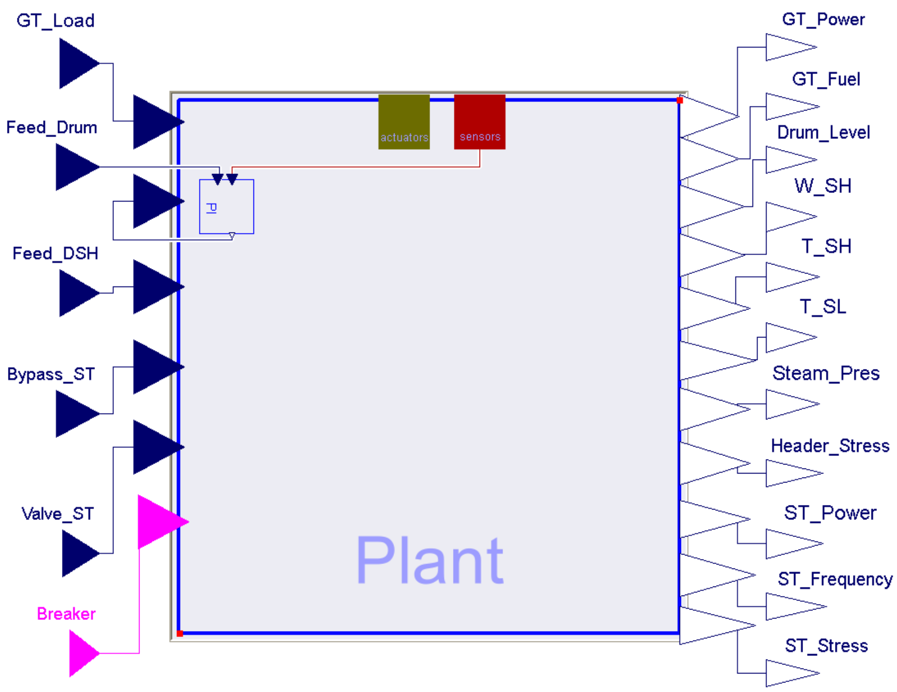
APPENDIX F

CCPP Dymola diagram

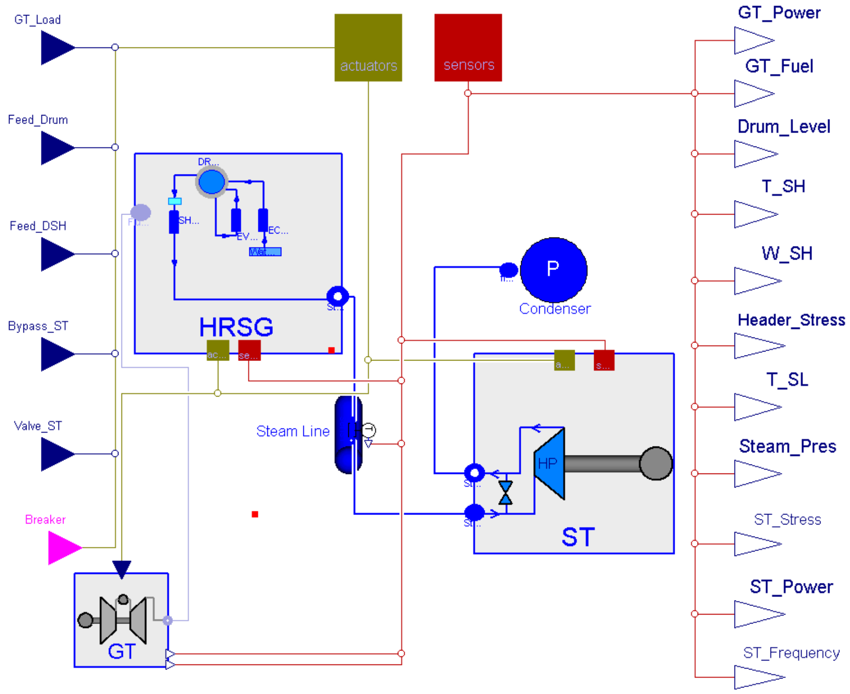
The CCPP model presented throughout this manuscript has been developed in the Dymola environment. The model corresponds to a combined cycle plant with one single level of pressure and includes the following units:

- Gas Turbine;
- Heat Recovery Steam Generator:
  - 1 drum boiler;
  - 3 heat exchangers;
  - 1 desuperheater system (Mixer + water source);
  - 1 steam header;
- Steam line;
- Steam Turbine:
  - turbine;
  - bypass/valve circuit;
  - rotational inertia of the turbine;
  - electrical generator;
  - power grid;
- Condenser.

The Dymola representation of the plant model with the main components is presented in Figures F.1a-F.2b. It should be pointed out that in the case of optimization-oriented model the steam line unit has not been included in the structure of the system.

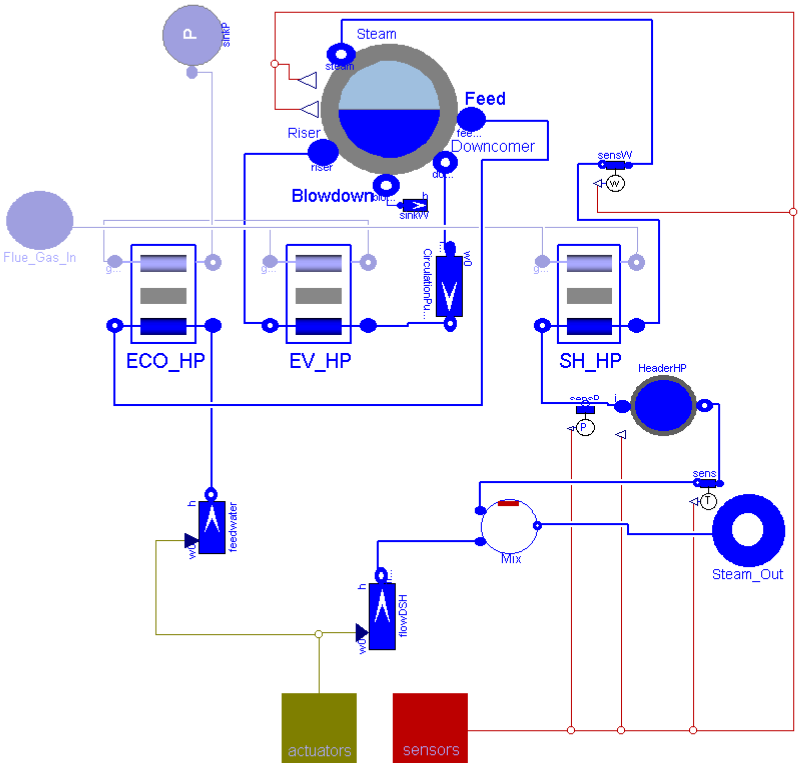


(a) Plant implementation as Dymola/Modelica model

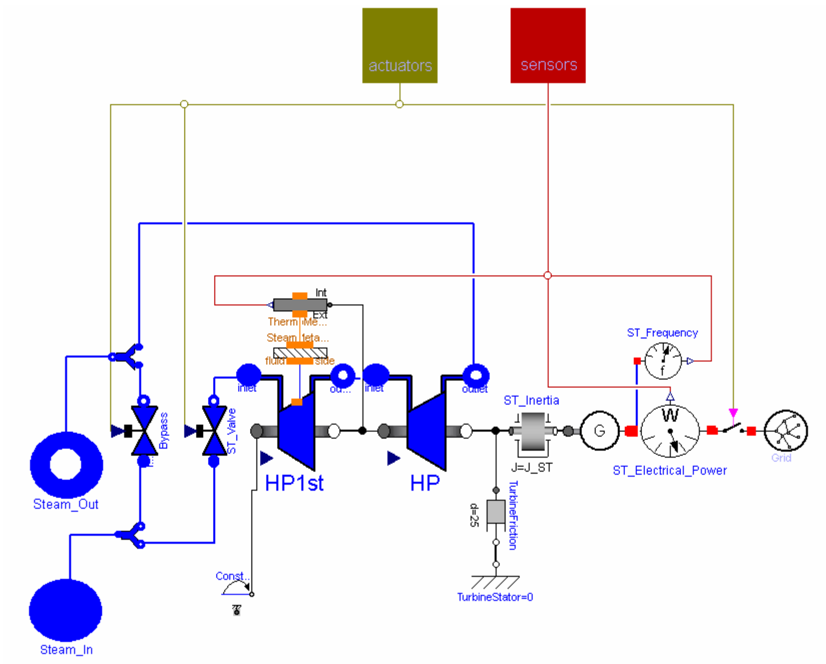


(b) Structure of the plant model

Figure F.1: Structure of the CCPP model represented in Dymola



(a) Structure of the HRSG model



(b) Structure of the ST model

Figure F.2: Structure of the model units represented in Dymola

APPENDIX G

Execution of Modelica/Dymola model in Matlab

For optimization and control designed, the smooth CCPP model developed in Modelica/Dymola has been imported in Matlab by means of Dymola-Matlab interface.

Dymola-Matlab interface

Dymola provides convenient interfaces to Matlab and to the simulation and modeling platform Simulink. It enables to combine the powerful physical modeling capabilities of Modelica language with efficient tools for control design and optimization given by Matlab.

In addition to the fact that a model can be transformed into a S-function file and simulated in Simulink as an input/output block, Dymola compiles the model into an executable *Dymosim* file (*dymosim.exe*). *Dymosim* contains the code necessary for continuous simulation, and event handling. Model descriptions are compiled into machine code such that to reach maximum execution speed during simulation. The *dymosim* is a stand-alone application which can be used in several different environments.

Dymosim has no graphical user interface, and reads the experiment description from an input file, performs a simulation run, stores the result into an output file and terminates. The *dymosim* executable can be directly called by user from Matlab:

>>*dos('dymosim.exe');*

or as a *m-function* :

>> *[s,n] =* ***dymosim****(exp,x0,p);*

The simulation run of the Dymola model is performed by reading an input file '*dsin.txt*' with the initial state, by storing the simulation result into the binary file '*dsres.mat*', and optionally by loading the result from file into '*s*, *n*'. The executable '*dymosim*' contains the model to be simulated and the most important simulation parameters can be given when calling this function (*exp* is the experiment setup vector, $x0$ is the initial state vector and *param* is the parameter vector). Also, *dymosim* can read trajectories of input signals from a file called '*dsu.txt*' if this is provided. The schematic representation of the *dymosim* with all involved input and output files is illustrated in Figure G.1.

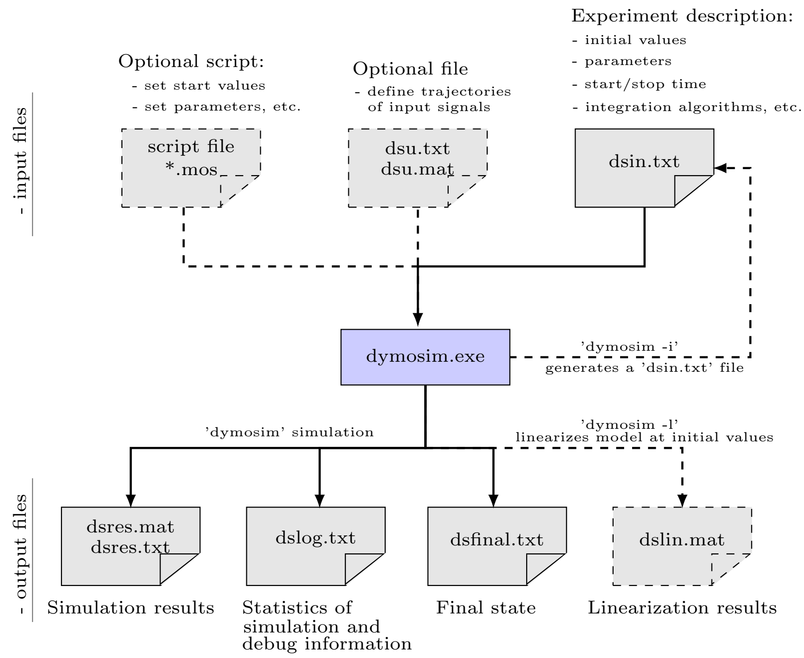


Figure G.1: *Dymosim* executable and its input and output files

Periodic simulation from a given state

The '*dsin.txt*' file contains all the initial variable values including the states of the system, and is read by *dymosim* when a model is simulated. Also, if the trajectories of input signals are specified ('*dsu.txt*' is provided), *dymosim* uses these profiles for simulation. When a simulation is finished another file called '*dsfinal.txt*' is created. This file contains the final values of the system. The running of the *dymosim.exe* together with an adequate modification of the '*dsin.txt*', '*dsfinal.txt*' and optionally '*dsu.txt*' files enable the simulation of the model from a specified point.

The periodic simulation from a given state for the MPC strategy has been performed by using the final values of the previous simulation as the initial values in the new simulation. Specifically, the procedure implemented can be described as follows:

I. a simulation is performed over a certain time interval, specifying '*dsin.txt*' and '*dsu.txt*', and the results are saved into '*dsfinal.txt*;

II. the new resulting '*dsfinal.txt*' is copied into '*dsin.txt*', changing the state of the model, and *dymosim.exe* is run for a new simulation.

List of Figures

List of Tables

Bibliography

[1] G. Ryder. The advantages of combined cycle plants: A 'new generation' technology. *Probe International*, pages 1–12, November 1997.

[2] G. Heyen and B. Kalitventzeff. A comparison of advanced thermal cycles suitable for upgrading existing power plant. *Applied Thermal Engineering*, 19:227–237, March 1999.

[3] R. Kehlhofer. *Combined-Cycle Gas & Steam Turbine Power Plants*. PennWell, first edition, 1997.

[4] J.-M. Monteil. Centrale à cycle combiné: Théorie, performances, modularité. *Techniques de l'Ingénieur*, pages [BE 8 905] 1–15, October 2003.

[5] G. Darie. Réhabilitation des centrales thermiques. *Techniques de l'Ingénieur*, pages [BM 4189] 1–14, 2005.

[6] C. Lévy. Les techniques de cogénération. *Techniques de l'Ingénieur*, pages [B 8910] 1–25, August 1996.

[7] Modelica Association. The Modelica Association, https://www.modelica.org/.

[8] Dynasim AB. *Dymola, Multi-Engineering Modeling and Simulation. User Manual Volume 1-2*. Dassault Systemes, 2009.

[9] F. Casella and A. Leva. Modelica open library for power plant simulation: design and experimental validation. In *Proceedings of the 3rd International Modelica Conference*, November 2003.

[10] F. Casella and F. Pretolani. Fast start-up of a combined-cycle power plant: a simulation study with Modelica. *Modelica Conference*, pages 3–10, September 2006.

[11] International Energy Agency. World energy outlook 2010.

[12] International Energy Agency. World energy outlook 2009.

[13] International Energy Agency. World energy outlook 2008.

[14] M. Dimitrova. French nuclear energy and its environmental impact. *Perspectives on global issues Fall 2009*, 4(1):23–26, 2009.

[15] EDF. Renewable origin electricity production: details by region and country. *Observ'ER*, pages 116–120, 2010.

[16] T. Tuot. Rapport general: le grenelle environnement, october 2007.

[17] European Union Committee. The EU's target for renewable energy: 20% by 2020. Technical report, House of Lords, October 2008.

[18] P. Kiameh. *Power Generation Handbook: Selection, Applications, Operation, and Maintenance*. McGraw-Hill Professional, 2002.

[19] A. Lallemand. Production d'énergie électrique par centrales thermiques. *Techniques de l'Ingénieur*, pages [D 4 002] 1–11, 2005.

[20] M. P. Boyce. *Gas Turbine Handbook*. Gulft Professional, second edition, 2002.

[21] C. Bolard and J.-P. Metz. Production d'énergie électrique à partir des combustibles fossiles. *Techniques de l'Ingénieur*, pages [B 1 250] 1–25, 2005.

[22] R. K. Kapooria, S. Kumar, and K. S. Kasana. An analysis of a thermal power plant working on a rankine cycle: A theoretical investigation. *Journal of Energy in Southern Africa*, 19(1):77–83, February 2008.

[23] M. Boss, T. Gadoury, S. Feeny, and M. Montgomery. Recent advances in ultra super critical steam turbine technology. In *GE Energy*, January 2002.

[24] R. Quinkertz, A. Ulma, E. Gobrecht, and M. Wechsung. USC steam turbine technology for maximum efficiency and operational flexibility. In *Power-GEN*. Siemens AG, October 2008.

[25] F. Cziesla, J. Bewerunge, and A. Senzel. Lunen - state-of-the art ultra supercritical steam power plant under construction. In *Power-GEN*, May 2009.

[26] D. L. Chase and P. T. Kehoe. GE combined-cycle product line and performance. *GER-3574G: GE Power Systems*, pages 1–40, October 2000.

[27] L. B. Davis and S. H. Black. Dry low NO_x combustion systems for GE heavy-duty gas turbines. *GER-3568G: GE Power Systems*, pages 1–22, October 2000.

[28] C. Johnson, B. Pepperman, M. Koenig, K. Abou-Jaoude, A. Gulati, and A. Moradian. Ultra low NO_x combustion technology. In *PowerGen International*. Siemens Power Generation Inc., December 2008.

[29] V. Cenusa, R. Benelmir, M. Feidt, and A. Badea. On gas turbines and combined cycles. *International Journal of Energy, Environment and Economics*, 11(4):15–26, 2003.

[30] P. E. Campbell, J. T. McMullan, and B. C. Williams. Concept for a competitive coal fired integrated gasification combined cycle power plant. *Fuel*, 79(9):1031–1040, July 2000.

[31] M. Klaeylé and F. Nandjee. Technologie de gazéification intégrée à un cycle combiné. *Techniques de l'Ingénieur*, pages [B 8920] 1–25, April 1997.

[32] O. Shinada, A. Yamada, and Y. Koyama. The development of advanced energy technologies in japan: Igcc: A key technology for the 21st century. *Energy Conversion and Management*, 43(9-12):1221–1233, June-August 2002.

[33] G. J. Stiegela and R. C. Maxwell. Gasification technologies: the path to clean, affordable energy in the 21st century. *Fuel Processing Technology*, 71(1-3):79–97, 2001.

[34] B. Multon. Production d'énergie électrique par sources renouvelables. *Techniques de l'Ingénieur*, pages [D 4 005] 1–11, 2003.

[35] D. L. Chase. Combined-cycle developement, evolution and future. *GER-4206: GE Power Systems*, pages 1–14, October 2000.

[36] P.A. Pilavachi. Power generation with gas turbine systems and combined heat and power. *Applied Thermal Engineering*, 20(15-16):1421–1429, 2000.

[37] I. G. Wright and T. B. Gibbons. Recent developments in gas turbine materials and technology and their implications for syngas firing. *International Journal of Hydrogen Energy*, 32(16):3610–3621, Novembre 2007.

[38] Siemens AG. New order for combined cycle power plant featuring world-record siemens gas turbine. *Energy Business Review (EBR)*, January 2011.

[39] J. L. Silveira, J. A. de Carvalho Jr., and I. A. de C. Villela. Combined cycle versus one thousand diesel power plants: pollutant emissions, ecological efficiency and economic analysis. *Renewable and Sustainable Energy Reviews*, 11(3):524–535, April 2007.

[40] W. J. Watson. The 'success' of the combined cycle gas turbine. In *IEEE International Conference on Opportunities and Advances in International Electric Power Generation*, pages 87–92, March 1996.

[41] J.-M. Monteil. Centrale a cycle combine - fonctionnement, exploitation, exemple. *Techniques de l'Ingénieur*, pages [BE 8 907] 1–16, 2009.

[42] F. Sandoz. Principes généraux des centrales à cycles combines et de leur demarrage. In *Les centrales a cycles combines*, pages 1–29, October 2008.

[43] J.-M. Monteil. Centrale à cycle combiné: Composants potentiels. *Techniques de l'Ingénieur*, pages [BE 8 906] 1–15, October 2003.

[44] H. Nessler, R. Preiss, and P. Eisenkolb. Developments in HRSG technology. In *The 7th Annual Industrial & Power Gas Turbine O&M Conference*, November 2001.

[45] C. Aurora. *Power Plants: Modeling, Simulation and Control.* PhD thesis, University of Pavia, 2004.

[46] B. G. Liptak. *Control and Optimization of Unit Operations*, volume II, chapter 8, pages 1473–2208. CRC Press, 2006.

[47] International Society of Automation (ISA). Fossil fuel power plant steam turbine bypass system, 1999.

[48] D. Faille. Instruction de la modélisation du conditionnement des lignes vapeur d'un cycle combiné. Technical report, EDF R&D, 2008.

[49] R. B. Dooley. Evaluation of thermal-, creep- and corrosion-fatigue of heat recovery steam generator pressure parts. Technical report, Electric Power Research Institute (EPRI), 2006.

[50] H. E. Boyer. *Atlas of fatigue curves.* ASM International, 1986.

[51] I. Dumitru and N. Faur. *Elemente de calcul si aplicatii în rezistenta materialelor.* Editura Politehnica Timisoara, 1999.

[52] R. Miroshnik, Y. Shaked, and D. Elmakis. Life assessment evaluation of piping branch connection under creep and fatigue. *International journal of pressure vessels and piping*, pages 147–154, 1997.

[53] P. Fontaine and J.-F. Galopin. HRSG optimization for cycling duty. *Power Engineering*, pages 170–184, November 2007.

[54] American Boiler Manufacturers Association. Comparaison of fatigue assessement techniques for heat recovery steam generators. Technical report, ABMA, 2003.

[55] G. Kosman and A. Rusin. The influence of the start-ups and cyclic loads of steam turbines conducted according to european standards on the components life. *Energy*, (26):1083–1099, 2001.

[56] G. Song, B. Kim, and S. Chang. Fatigue life evaluation for turbine rotor using Green's function. *Procedia Engineering*, pages 2292–2297, 2011.

[57] R.C. Spencer and D.P. Timo. Starting and loading of large steam turbines. In *36th American Power Conference*, May 1974.

[58] T. S. Kim, D. K. Lee, and S.T. Ro. Analysis of thermal stress evolution in the steam drum during start-up of a heat recovery steam generator. *Applied Thermal Engineering*, pages 977–992, 2000.

[59] N. Henkel, E. Schmid, and E. Gobrecht. Operational flexibility enhancement of combined cycle power plants. In *Power-Gen.* Siemens AG, October 2008.

[60] U. Grumann, H. Kurz, G. Meinecke, and A. Pickard. Improvement of operational efficiency based on fast startup plant concepts. In *XXIst World Energy Congress*. Siemens AG, September 2010.

[61] Siemens AG. Benson once-through heat recovery steam generator. Power Plants, www.siemens.com/powergeneration, 2006.

[62] Siemens AG. SPPA-P3000 solutions for process optimization. Instrumentation & Controls, www.siemens.de/ic, 2006.

[63] M. Ladwing and M. Stevens. KA26 combined cycle power plant as ideal solution to balance load fluctuations. In *Power Plant Technology Forum - Hannover Messe 2011*, April 2011.

[64] Invensys Operations Management. The benefits of using dynamic simulation and training systems for expanding operator knowledge and understanding. www.iom.invensys.com, 2010.

[65] W.J. Rowen. Simplified mathematical representations of heavy duty gas turbines. *ASME Journal of Engineering for Power*, 105(4):865–869, 1983.

[66] F.P. Mello. Boiler models for system dynamic performance studies. *IEEE Transactions on Power Systems*, 6(1):66–74, 1991.

[67] Working Group on Prime Mover and Energy Supply Models for System Dynamic Performance Studies. Dynamic models for combined cycle plants in power system studies. *IEEE Transactions on Power Systems*, 9(3):1698–1708, 1994.

[68] T. Akiyama, H. Matsumoto, and K. Asakura. Dynamic simulation and its applications to optimum operation support for advanced combined cycle plants. *Energy Conversion and Management*, 38(15-17):1709–1723, 1997.

[69] D. Faille, F. Davelaar, and A.C. Deroide. Combined cycle start-up hybrid simulation. In *51th Annual ISA Power Industry Division Symposium*, June 2008.

[70] D. Faille and F. Davelaar. Model based start-up optimization of a combined cycle power plant. In *6th IFAC Symposium on Power Plants and Power Systems Control*, July 2009.

[71] F. Casella, F. Donida, and J. Akesson. Object-oriented modeling and optimal control: A case study in power plant start-up. In *18th IFAC World Congress*, August 2011.

[72] J. Bausa and G. Tsatsaronis. Dynamic optimization of startup and load-increasing processes in power plants-part II: Application. *Transactions of the ASME*, 123:251–254, 2001.

[73] C. Albanesi, M. Bossi, L. Magni, J. Paderno, F. Pretolani, P. Kuehl, and M. Diehl. Optimization of the start-up procedure of a combined cycle power plant. In *Proceedings of the 45th IEEE Conference on Decision & Control*, pages 1840–1845. IEEE, December 2006.

[74] M. Shirakawa, M. Nakamoto, and S. Hosaka. Dynamic simulation and optimization of start-up processes in combined cycle power plants. *JSME International Journal*, 48(1):122–128, 2005.

[75] H. E. Shalan, M. A. M. Hassan, and A. B. G. Bahgat. Parameter estimation and dynamic simulation of gas turbine model in combined cycle power plants based on actual operational data. *Journal of American Science*, 7(5):303–310, 2011.

[76] F. Alobaid, R. Postler, J. Strohle, B. Epple, and H.-G. Kim. Modeling and investigation start-up procedures of a combined cycle power plant. *Applied Energy*, 85(12):1173–1189, 2008.

[77] K. Kruger, R. Franke, and M. Rode. Optimization of boiler start-up using a nonlinear boiler model and hard constraints. *Energy*, 29:22392251, 2004.

[78] I. Bertini, M. De Felice, F. Moretti, and S. Pizzuti. Start-up optimisation of a combined cycle power plant with multiobjective evolutionary algorithms. In *Applications of Evolutionary Computation: EvoApplications (2)*, pages 151–160. Springer, April 2010.

[79] I. Bertini, A. Pannicelli, and S. Pizzuti. Fuzzy optimization of start-up operations for combined cycle power plants. In *Soft Computing Models in Industrial and Environmental Applications, 5th International Workshop (SOCO)*, pages 153–160, June 2010.

[80] H. Matsumoto, Y. Ohsawa, S. Takahasi, T. Akiyama, and O. Ishiguro. An expert system for startup optimization of combined cycle power plants under NO_x emission regulation and machine life management. *IEEE Transactions on Energy Conversion*, 11(2):414–422, 1996.

[81] H. Matsumoto, Y. Ohsawa, S. Takahasi, T. Akiyama, H. Hanaoka, and O. Ishiguro. Startup optimization of a combined cycle power plant based on cooperative fuzzy reasoning and a neural network. *IEEE Transactions on Energy Conversion*, 12(1):51–59, 1997.

[82] A. Cervantes and L. T. Biegler. Optimization strategies for dynamic systems. In C.A Floudas and P.M. Pardalos, editors, *Encyclopedia of Optimization*, January 2000.

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

[84] J. Bausa and G. Tsatsaronis. Dynamic optimization of startup and load-increasing processes in power plants-part I: Method. *Transactions of the ASME*, 123:246–250, 2001.

[85] E. F. Camacho and C. Bordons. *Model Predictive Control in the Process Industry. Second Edition*. Springer-Verlag, London, England, 2004.

[86] A. Nakai, M. Nakamoto, A. Kakehi, and S. Hayashi. Turbine start-up algorithm based on prediction of rotor thermal stress. In *SICE'95. Proceedings of the 34th SICE Annual Conference*, pages 1561–1564. IEEE, July 1995.

[87] J. Patanian. Plant automation methods for predictable steam turbine fast starts in a combined cycle. In *53th Annual ISA Power Industry Division Symposium*, June 2010.

[88] F. J. D'Amato. Industrial application of a model predictive control solution for power plant startups. In *Proceedings of the 2006 IEEE International Conference on Control Applications*, pages 243–248, October 2006.

[89] K. J. Astrom, H. Elmqvist, and S. E. Mattsson. Evolution of continuous-time modeling and simulation. In *12th European Simulation Multiconference, ESM'98*, June 1998.

[90] Mats Andersson. *Object-Oriented Modeling and Simulation of Hybrid Systems*. PhD thesis, Lund Institute of Technology, 1994.

[91] Hilding Elmqvist. *A structured model language for large continuous systems*. PhD thesis, Lund Institute of Technology, 1978.

[92] P. Fritzson, L. Viklund, D. Fritzson, and J. Herber. ObjectMath - an environment for high-level mathematical modelling and programming in scientific computing. In *IEEE Software*, July 1995.

[93] H. Elmqvist, S. E. Mattsson, and M. Otter. Modelica - an international effort to design an object-oriented modeling language. In *Proceedings of the 1998 Summer Computer Simulation Conference, SCSC'98*, July 1998.

[94] Process System Enterprise Limited. gPROMS, http://www.psenterprise.com/gproms/index.html.

[95] E. Christen and K. Bakalar. VHDL-AMS - a hardware description language for analog and mixed-signal applications. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 46(10):1263–1272, 1999.

[96] Modelica Association. *Modelica - A Unified Object-Oriented Language for Physical Systems Modeling. Language Specification Version 3.2*, March 2010.

[97] P. Fritzson. *Principles of Object-Oriented Modeling and Simulation With Modelica 2.1*. John Wiley & Sons, Inc., 2004.

[98] M. Tiller. *Introduction to Physical Modeling with Modelica*. Kluwer Academic Publishers, 2001.

[99] Modelica Association. *Modelica - A Unified Object-Oriented Language for Physical Systems Modeling. Tutorial*, December 2000.

[100] J. Akesson, K-E. Arzen, M. Gafvert, T. Bergdahl, and H. Tummescheit. Modeling and optimization with Optimica and JModelica.org—languages and tools for solving large-scale dynamic optimization problem. *Computers and Chemical Engineering*, 34(11):1737–1749, November 2010.

[101] H. Tummescheit, J. Eborn, and F.J. Wagner. Development of a Modelica base library for modeling of thermohydraulic systems. In *The Modelica Workshop 2000*, October 2000.

[102] Politecnico di Milano (F. Casella). The ThermoPower Modelica library v. 2.0, http://home.dei.polimi.it/casella/thermopower/.

[103] B. El Hefni, D. Bouskela, and G. Lebreton. Dynamic modelling of a combined cycle power plant with thermosyspro. In *Proceedings 8th Modelica Conference*, March 2011.

[104] F. Casella and A. Leva. Object-oriented modelling & simulation of power plants with Modelica. In *44th IEEE Conference on Decision and Control*, December 2005.

[105] F. Casella and A. Leva. Modelling of thermo-hydraulic power generation processes using Modelica. *Mathematical and Computer Modelling of Dynamical Systems*, 12(1):19–33, 2006.

[106] F. Casella and F. Schiavo. Modelling and simulation of heat exchangers in Modelica with finite element methods. In *Proceedings of the 3rd International Modelica Conference*, November 2003.

[107] TRD. Technical rules for steam boilers. Technical report, Deutscher Dampfkesselausschuß (DDA), 1975.

[108] D. H. Cooke. Modeling of off-design multistage turbine pressures by Stodola's ellipse. In *Energy Incorporated PEPSE user's group meeting*, November 1983.

[109] Fisher Controls International LLC. Introduction and sizing valves for liquids, November 1999.

[110] M. Outtas, H. Gueguen, and D. Dumur. Modélisation du conditionnement des lignes vapeur d'un cycle combiné. Master's thesis, Supelec, France, 2009.

[111] F. Casella, M. Farina, F. Righetti, R. Scattolini, D. Faille, F. Davelaar, A. Tica, H. Gueguen, and D. Dumur. An optimization procedure of the start-up of combined cycle power plants. In *18th IFAC World Congress*, August 2011.

[112] F. Righetti. Control and optimization of the start-up of a combined cycle power plant. Master's thesis, Politecnico di Milano University, 2010.

[113] J. Nocedal and S.J. Wright. *Numerical Optimization*. Secon Edition. Springer, 2000.

[114] A. Neculai. *Programarea matematica. Metode de punct interior*. Editura Tehnica, 1999.

[115] M. Gapaillard. Représentations continues de systèmes hybrides. *Journal européen des systèmes automatisés*, pages 913–940, 2007.

[116] International Association for the Properties of Water and Steam. Release on the iapws industrial formulation 1997 for the thermodynamic properties of water and steam. Technical report, IAPWS, Erlangen, Germany, 1997.

[117] John D'Errico. Polyfin matlab toolbox, http://www.mathworks.com.

[118] MathWorks. Matlab curve fitting toolbox, http://www.mathworks.com/products/curvefitting/.

[119] M. Brendel, J. Oldenburg, M. Schlegel, and K. Stockmann. *DyOS User's Guide, Release 2.1. Aachen: Lehrstuhl fur Prozesstechnik*. RWTH Aachen University, 2002.

[120] R. Bellman. *Dynamic Programming.* Princeton University Press, 1957.

[121] L.S. Pontryagin, V.G. Boltyanskii, R.V. Gamkrelidze, and E.F. Mishchenko. *The Mathematical Theory of Optimal Processes.* John Wiley & Sons Inc., 1962.

[122] L.T. Biegler, A.M. Cervantes, and A.Wachter. Advances in simultaneous strategies for dynamic process optimization. *Chemical Engineering Science*, 57:575–593, 2002.

[123] L.T. Biegler. Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation. *Computers & Chemical Engineering*, 8:243–248, 1984.

[124] H.G. Bock and K.J. Plitt. A multiple shooting algorithm for direct solution of optimal control problems. In *9th IFAC World Congress Budapest*, July 1984.

[125] A. Wachter and L. T. Biegler. Interior Point OPTimizer (IPOPT). *Mathematical Programming*, 106(1):25–57, 2006, https://projects.coin-or.org/Ipopt.

[126] S.J. Qin and T.A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, 11:733–764, 2003.

[127] P. Boucher and D. Dumur. *La commande prédictive.* Éditions Technip, Paris, 1996.

[128] M. Morari and J.H. Lee. Model predictive control: past, present and future. *Computers and Chemical Engineering*, 23:667–682, 1999.

[129] C. Aurora, M. Diehl, P. Kuhl, L. Magni, and R. Scattolini. Nonlinear model predictive control of combined cycle power plants. In *Proceedings of the 16th IFAC World Congress*, pages 128–132, July 2005.

[130] M. R. Katebi and M. A. Johnson. Predictive control design for large-scale systems. *Automatica*, 33(3):421–425, 1997.

[131] J. Maciejowski. *Predictive Control with Constraints.* Prentice-Hall, 2001.

[132] T. Binder, L. Blank, H. G. Bock, R. Bulirsch, 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*, chapter 3, Online Optimization of Large Scale Systems, pages 295–339. Springer-Verlag, 2001.

[133] M. Diehl, H. J. Ferreau, and N. Haverbeke. *Efficient Numerical Methods for Nonlinear MPC and Moving Horizon Estimation*, chapter 32, Nonlinear Model Predictive Control, pages 391–417. Springer, 2009.

[134] A. Tica, H. Gueguen, D. Dumur, D. Faille, and F. Davelaar. Predictive approaches for the combined cycle plant start-up. Technical report, Hierarchical and Distributed Model Predictive Control of Large-Scale Systems project (HD-MPC), 2011.

[135] H. Chen and F. Allgower. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34(10):1205–1217, 1998.

[136] R. Scattolini. Architectures for distributed and hierarchical model predictive control. *Journal of Process Control*, pages 723–730, 2009.

[137] H. Scheu and W. Marquardt. Report on literature survey on hierarchical and distributed nonlinear mpc, including analysis and comparison, and description of the resulting methodological framework. Technical report, Hierarchical and Distributed Model Predictive Control of Large-Scale Systems project (HD-MPC), 2009.

[138] A. Nunez, B. Picasso, D. De Vito, R. Scattolini, P. Colaneri, J. Espinosa, F. Valencia, B. De Schutter, and K. Stankova. Report on newly developed coordination mechanisms for hierarchical and distributed mpc. Technical report, Hierarchical and Distributed Model Predictive Control of Large-Scale Systems project (HD-MPC), 2010.

[139] B. Picasso, C. Romani, and R. Scattolini. On the design of hierachical control systems with MPC. In *Poceedings of the European Control Conference*, pages 731–736, August 2009.

[140] Open Source Modelica Consortium (OSMC). OpenModelica Open Source Tools, http://www.openmodelica.org/.

[141] MathCore Engineering AB. MathModelica, http://www.mathcore.com/products/mathmodelica/.

[142] Waterloo Maple Inc. MapleSim, http://www.maplesoft.com/.

[143] ITI GmbH. SimulationX, http://www.itisim.com/.

[144] Johan Akesson. Optimica - an extension of Modelica supporting dynamic optimization. In *The 6th International Modelica Conference*. Modelica Association, March 2008.

[145] T. A. Johansen and B. A. Foss. Constructing narmax models using armax models. *International Journal of Control*, 58(5):1125–1153, 1993.

[146] T. A. Johansen and B. A. Foss. Identification of non-linear system structure and parameters using regime decomposition. *Automatica*, 31:321–326, 1995.

[147] MathWorks. System identification toolbox, http://www.mathworks.fr/products/sysid/index.html.

Publications

The work presented in this thesis is based on the following publications:

International journals with reading committee:

- A. Tică, H. Guéguen, D. Dumur, D. Faille, F. Davelaar. "Design of a combined cycle power plant model for optimization". *Applied Energy*, 98(10): 256-265, 2012.

International conferences with reviewing committee:

- A. Tică, H. Guéguen, D. Dumur, D. Faille, F. Davelaar. "Optimization of the Combined Cycle Power Plants Start-up". *54th ISA Power Industry Division Symposium*, Charlotte, North Carolina, 2011.

- F. Casella, M. Farina, F. Righetti, R. Scattolini, D. Faille, F. Davelaar, A. Tică, H. Guéguen, D. Dumur. "An optimization procedure of the start-up of combined cycle power plants". *18th IFAC World Congress*, Milan, Italy, 2011.

- A. Tică, H. Guéguen, D. Dumur, D. Faille, F. Davelaar. "Hierarchical Model Predictive Control Approach for Start-up Optimization of a Combined Cycle Power Plant". *8th Power Plant and Power Systems Control Symposium*, Toulouse, France, 2012.

- A. Tică, H. Guéguen, D. Dumur, D. Faille, F. Davelaar. "Hierarchical Nonlinear Model Predictive Control for Combined Cycle Start-up Optimization". Submitted to *51st IEEE Conference on Decision and Control*, USA, 2012.

Reports in the European FP7 STREP project HD-MPC (Hierarchical and Distributed Model Predictive Control of Large-Scale Systems)

- A. Tică, H. Guéguen, D. Dumur, D. Faille, F. Davelaar. "Control specification for the Combined Cycle start-up". Report D7.1.1, HD-MPC ICT-223854, 2009.

- A. Tică, H. Guéguen, D. Dumur, D. Faille, F. Davelaar, F. Righetti, R. Scattolini. "Model and open-loop simulation results for the combined cycle start-up". Report D7.1.2, HD-MPC ICT-223854, 2010.

- A. Tică, H. Guéguen, D. Dumur, D. Faille, F. Davelaar. "Closed-loop validation results for the combined cycle start-up". Report D7.1.3, HD-MPC ICT-223854, 2011.

International conferences without proceedings:

- A. Tică, H. Guéguen, D. Dumur, D. Faille, F. Davelaar. "HD-MPC poster on the application to the start-up of the combined cycle power plant". CPS Week, Stockholm, Sweden, 2010.

Titre : Conception, optimisation et validation des séquences de démarrage des systèmes de production d'énergie

Cette thèse porte sur l'application des approches de commande prédictive pour optimiser les séquences de démarrage des centrales à cycles combinés. L'élaboration des approches est progressive. Dans une première partie un modèle de centrale est construit et adapté à l'optimisation, en utilisant une méthodologie qui transforme des modèles physiques Modelica conçus pour la simulation en des modèles pour l'optimisation. Cette méthodologie a permis de construire une bibliothèque adaptée à l'optimisation. La suite des travaux porte sur l'utilisation du modèle afin d'optimiser phase par phase les performances du démarrage. La solution proposée optimise, en temps continu, le profil de charge des turbines en se restreignant à des ensembles de fonctions particulières. Le profil optimal est déterminé en considérant que celui-ci peut être décrit par une fonction paramétrée dont les paramètres sont calculés en résolvant un problème de commande optimale sous contraintes. La dernière partie des travaux consiste à intégrer cette démarche d'optimisation à temps continu dans une stratégie de commande à horizon glissant. Cette approche permet d'une part de corriger les dérives liées aux erreurs de modèles et aux perturbations, et d'autre part, d'améliorer le compromis entre le temps de calcul et l'optimalité de la solution. Cette approche de commande conduit cependant à des temps de calcul importants. Afin de réduire le temps de calcul, une structure de commande prédictive hiérarchisée avec deux niveaux, en travaillant à des échelles de temps et sur des horizons différents, a été proposée.

Mots clés : Commande Prédictive, Commande Hiérarchisée, Optimisation, Systèmes de grande taille, Modélisation physique, Centrales à Cycles Combinés, Séquence de démarrage.

Title: Design, optimization and validation of start-up sequences of energy production systems

This thesis focuses on the application of model predictive control approaches to optimize the combined cycle power plants start-ups. The development of the proposed approaches is progressive. In the first part a physical model of plant is developed and adapted to optimization purposes, by using a methodology which transforms Modelica model components into optimization-oriented models. By applying this methodology, a library suitable for optimization purposes has been built. In the second part, based on the developed model, an optimization procedure to improve the performances of the start-up phases is suggested. The proposed solution optimizes, in continuous time, the load profile of the turbines, by seeking in specific sets of functions. The optimal profile is derived by considering that this profile can be described by a parameterized function whose parameters are computed by solving a constrained optimal control problem. In the last part, the open-loop optimization procedure has been integrated into a receding horizon control strategy. This strategy represents a robust solution against perturbation and models errors, and makes it possible to improve the trade-off between computation time and optimality of the solution. Nevertheless, the control approach leads to a significant computation time. In order to obtain real-time implementable results, a hierarchical model predictive control structure with two layers, working at different time scales and over different prediction horizons, has been proposed.

Keywords: Model Predictive Control, Hierarchical Control, Optimization, Large-scale Systems, Physical modeling, Combined Cycle Power Plants, Start-up.