Analytical target cascading: convergence improvement by sub-problem post-optimality sensitivities

S. Tosserams

SE 420 389

M.Sc. thesis

Supervisor: Prof.dr.ir. J.E. Rooda
Coach: Dr.ir. L.F.P. Etman

EINDHOVEN UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF MECHANICAL ENGINEERING
SYSTEMS ENGINEERING GROUP

Eindhoven, June 2004
Preface

What lies opened before you is the result of a year’s work on my final assignment for becoming Master of Science. This thesis finishes a five year curriculum at the Department of Mechanical Engineering of the Eindhoven University of Technology. After three years of basic courses and group assignments, I decided to join Professor Rooda’s Systems Engineering group for the last two years of the curriculum. This group develops methods, techniques, and tools for design, analysis, optimization and control of advanced industrial systems. During these two years I particularly enjoyed an external internship at the toolroom of PDL Industries Ltd. in Christchurch, New Zealand in corporation with the University of Christchurch. During these four months at the other side of the world, I enjoyed both researching the scheduling of the toolroom and staying in a foreign country and culture. During my final assignment, I noticed that I quite enjoyed researching and working with my coach, Pascal Etman. As a result, I am fortunate to tell you that I have been invited to take part in a PhD project on multi-level decomposed optimization of micro-technological systems at the Systems Engineering group.

A word of thanks goes to Pascal Etman, my coach, for his enthusiastic and constructive support, and fruitful discussions; I am looking forward to working with him during the next four years of my PhD project. A word of thanks also goes to Ad Kock for taking the time to review my report, and Professor Rooda, for his support and his invitation to join the PhD project.

Furthermore I would like to thank my parents for providing a firm genetic foundation, and for supporting me during the last five years in Eindhoven. I also want to thank my friends Roel, Dennis and Rob for studying together, and for providing the necessary distractions from our study. I enjoyed the five years we spent in Eindhoven together. Miel and Ward also deserve credits for their corporation in making computer isle 1 the most vivid one in the lab. Finally, I especially want to thank my girlfriend Moon for her interests, support and her ability to endure my occasionally very lively spirit.

Simon Tosserams
Assignment

Student S. Tosserams
Supervisor Prof.dr.ir. J.E. Rooda
Coach Dr.ir. L.F.P. Etman
Start May 2003
Finish May 2004
Title Analytical target cascading: convergence improvement by subproblem post-optimality sensitivities

Subject
Analytical target cascading (ATC) is a methodology for the design of large and complex engineering systems at the early product development stages [Kim01]. ATC starts from a multi-level hierarchical decomposition of the system. It translates the overall system design targets into individual targets for the subsystems and components that make up the complete system. Each subsystem (component) is represented by a design optimization problem to meet cascaded targets as closely as possible while satisfying local design constraints.

Analytical target cascading has been demonstrated to be convergent under standard smoothness and convexity assumptions [Mic03]. It has been applied successfully in several vehicle design case studies. However, numerical experience with analytical optimization test problems revealed slow convergence and difficulties to arrive at a highly accurate final solution compared to the known ‘all-at-once’ solution [Etm02, Hul03]. One reason may be that the ATC coordination passes among the various optimization subproblems target values that are treated as fixed within the subproblems.

Assignment
Through post-optimality sensitivity analysis gradient information on the computed targets may be obtained. The MSc thesis project aims to investigate whether post-optimality sensitivities computed at the ATC subproblems can be used to improve the (speed of) convergence of the ATC coordination.
References


Summary

Analytical target cascading (ATC) is a decomposition methodology for the design of large and advanced systems and follows a natural decomposition of the system. ATC has been originally developed at the University of Michigan by Kim, Michelen, Papalambros, and co-workers. The methodology defines a method for propagating overall design targets throughout the hierarchy of the system. Each element in the hierarchy receives propagated targets and computes responses as close as possible to these targets. The exchange of targets and responses is formalized in a multilevel decomposed optimization problem, consisting of a number of connected sub-problems defined for each element in the hierarchy.

ATC has been applied to a number of practical cases, and a convergence proof exists under convexity assumptions. The numerical behavior of the method has also been investigated, showing that the ATC solution process is a computationally expensive process. The algorithm for solving the decomposed problem is defined in a coordination strategy. Current coordination strategies only exchange target and response values between the sub-problems of the hierarchy. Sensitivities of targets and responses with respect to each other can be obtained relatively cheap from the sub-problems by using standard post-optimality techniques. The objective of this research is to investigate the possibility of using these sensitivities in the coordination algorithm.

The ATC methodology is reviewed and classified in the framework of decomposition-based optimization approaches, both from a mathematical and engineering point of view. In the previous work on ATC, no general method for partitioning a (large) optimization problem into a hierarchy of multiple connected sub-problems has been presented so far. The underlying ATC partitioning structure is however needed to properly classify ATC. Therefore a partitioning method for ATC is developed in this thesis building on the earlier work of Wagner. On this basis, ATC can be classified as a dual-feasible method for decomposition, which was only partly recognized in earlier work. The ATC dual-feasible decomposition method comprises of four steps: (1) problem structure identification and manipulation, (2) variable partitioning, (3) sub-problem formulation, and (4) solution coordination.

Previous work on ATC indicated the need for more test examples. Again, the missing partitioning approach is the reason for this. With the proposed partitioning method
Summary

for ATC, optimization problems can be decomposed and formulated as ATC problems. The partitioning method is used to decompose several optimization problems found in the literature into ATC problems, providing a large number of test problems for ATC.

The feasible and dual master problems of ATC have received little attention. In the founding ATC papers only a feasible master problem is presented to update the values of responses and targets computed in the individual sub-problems. In the feasible master problem, a coordination strategy defines in what sequence the sub-problems in the problem hierarchy are solved. Recently, a dual master problem for ATC has been proposed, reducing the computational efforts required to solve the decomposed optimization problem.

This report investigates five alternative feasible coordination strategies. The first two strategies only use target and response values, and the latter three also use sensitivities of targets and responses with respect to each other. Strategy 1 is a hierarchy-based coordination strategy that iterates between solving sub-problems at all even levels and all odd levels of the hierarchy. Strategy 2 is an all-parallel coordination solution algorithm. This hierarchy independent strategy simply solves all sub-problems at once, and then updates all targets and responses for all sub-problems. Strategies 3, 4, and 5 are sensitivity-based methods that use gradient information of target and responses, in addition to the values of targets and responses. The gradients used in the sensitivity-based strategies can be computed relatively easily from post-optimality sensitivities of sub-problem solutions.

The basis for sensitivity-based coordination is an alternative point of view on the solution to the decomposed problem. Elements in the hierarchy represented by optimization sub-problems are seen as black-box functions that compute outputs from an element’s inputs. Two relations between inputs and outputs exist: input-output functions defining how outputs of an element depend on the inputs that the element receives, i.e. the black-box functions, and secondly, hierarchy relations defining what output is connected to what input. By combining the input-output functions and the hierarchy relations of all sub-problems, a feasible master problem is formulated that searches for a solution with minimal inconsistencies. In this master problem, the black-box functions are approximated by first-order Taylor series. The gradients used in this algorithm, when applied to ATC, are the sensitivities of outputs with respect to inputs.

First-order algorithms often require safeguards to keep linearisations valid. The properties of the linear approximations of the black-box functions for ATC are investigated. These approximations for realizable inputs are unable to predict constraint activities, and therefore suggest unboundedness of outputs. Outputs however are bounded by local constraints, and therefore two safeguarded strategies are presented. Safeguarded strategy 4 uses move limits, a more traditional safeguard, and strategy 5 uses a hybrid method, which is specifically designed for ATC. The hybrid method uses gradients obtained with non-realizable inputs, and only uses output values for realizable inputs.
Strategies 2 through 5 are compared to a benchmark strategy, a state-of-the-art hierarchical scheme commonly used in ATC. The strategies are compared for three different ATC problems. Two test examples clearly show that the sensitivity-based strategy 3 outperforms all other strategies with respect to computational cost and convergence rate. For the third test example however, all coordination strategies have similar performance. What decreases the performance of the sensitivity-based approaches for this last example however remains to be investigated. Based on the three examples it is concluded that sensitivity-based coordination performs better or similar to the benchmark coordination strategies. As yet, no clear explanation has been found why the sensitivity-based ATC coordination does not yield an improved performance for the third test example.
Samenvatting (in Dutch)

Analytical target cascading (ATC) is een methode voor het gedecomponeerd ontwerpen van omvangrijke en geavanceerde systemen en volgt een natuurlijke decompositie van het ontwerpprobleem. Oorspronkelijk is ATC ontwikkeld aan de Universiteit van Michigan door Kim, Michiena, Papalambros en hun medewerkers. De methode beschrijft hoe eisen (targets) die aan het gehele ontwerp gesteld worden, doorvertaald kunnen worden naar targets voor de individuele componenten in het systeem. Een element in de hiërarchie ontvangt eisen van een ander hoger gelegen element, een zogenaamde ouder, waarop vervolgens responsies bepaald worden die beschrijven hoe goed een element aan de targets kan voldoen. Tevens bepaalt een element targets die doorgegeven worden aan zijn kinderen, gelegen op lagere niveau’s in de hiërarchie. Het proces van het doorgeven van targets en responsies wordt vastgelegd in een multilevel gedecomponeerd optimaliseringsprobleem dat bestaat uit een aantal optimaliserings-subproblemen die verbonden zijn door de targets en responsies.

In het verleden is ATC toegepast op een aantal praktische ontwerpproblemen uit de automobiëndustrie. Een convergentiebewijs voor convexe problemen is opgesteld. Onderzoek naar het numerieke gedrag van de methode wees uit dat veel rekentijd vereist is voor het oplossen van een gedecomponeerd ATC optimaliseringsprobleem. Een coördinatiestrategie is het algoritme dat gebruikt wordt om het gedecomponeerde probleem op te lossen. Beschikbare coördinatiestrategieën wisselen alleen targets voor subproblemen en de bijbehorende responsies uit. Gevoeligheden van de berekende responsies ten opzichte van de doorgegeven targets kunnen betrekkelijk eenvoudig verkregen worden door middel van een post-optimale gevoeligheidsanalyse die uitgevoerd kan worden na het oplossen van een subprobleem. Het doel van het onderzoek beschreven in dit verslag is onderzoeken of deze gevoeligheden in een coördinatiestrategie voor ATC gebruikt kunnen worden.

De ATC-methode is geanalyseerd en geclassificeerd als een methode voor gedecomponeerd optimaliseren. ATC wordt vergeleken met decompositiemethodes met zowel een wiskundige als praktische achtergrond. Uit de analyse blijkt dat er tot op heden geen algemene decompositiemethode beschikbaar was voor het partitionen van een (groot) optimaliseringsprobleem in een aantal onderling verbonden kleinere optimaliserings-subproblemen. Voor het classificeren van ATC is deze algemene decompositiemethode echter wel nodig. In dit verslag is een decompositiemethode voor ATC ontwikkeld.
gebaseerd op het werk van Wagner. Aan de hand van deze methode kan ATC geclassi-
ﬁceerd worden als een dual-feasible methode voor decompositie. De dual-feasible eigen-
schappen van de ATC methode zijn slechts gedeeltelijk onderkend in voorgaand onder-
zoek. De dual-feasible decompositiemethode voor ATC bestaat uit de volgende stappen:
(1) het identiﬁceren en manipuleren van de structuur van het originele probleem, (2) het
partitioneren van variabelen, (3) het formuleren van subproblemen en (4) het oplossen
van het gedecomponeerde probleem.

Door het ontbreken van een algemene decompositiemethode kwam in eerder onderzoek
naar ATC de behoefte aan een groter aantal analytische testproblemen naar voren.
Met de voorgestelde decompositiemethode is het mogelijk om optimaliseringsproble-
men te decomponeren en te formuleren als ATC problemen. De decompositiemethode is
aangewend om een aantal optimaliseringsproblemen uit de literatuur te decomponeren.
Hiermee is een aantal analytische test-problemen beschikbaar gemaakt voor ATC.

De duale en feasible master problemen van de dual-feasible decompositiemethode voor
ATC hebben vooralsnog weinig aandacht gekregen. Het feasible master probleem is
voorheen gebruikt om de targets en responsies berekend in een subprobleem te up-
daten in de overige subproblemen. Tevens werd in het feasible master probleem een
coördinatiestrategie gedefinieerd waarin vastgelegd in welke volgorde subproblemen opgelost
worden. Eén duaal master probleem voor ATC is voorgesteld. Met het opnemen van
dit duaal master probleem in het oplossingsalgoritme zijn de rekentijden benodigd voor
het oplossen van het ATC probleem teruggedrongen.

In dit verslag zijn vijf verschillende feasible coördinatiestrategieën voorgesteld. Twee
ervan gebruiken enkel targets en responsies berekend in de subproblemen, wat vergelijk-
baar is met bestaande coördinatie-strategieën. De overige drie voorstellen maken gebruik
van de gevoeligheid van targets en responsies ten opzichte van elkaar. Strategie 1
beschrijft een oplossingsvolgorde gebaseerd op de hiërarchie van het gedecomponeerde
probleem. De methode wisselt tussen het oplossen van subproblemen op alle even-
genummerde levels in de hierarchy en het oplossen van subproblemen of oneven levels.
Strategie 2 beschrijft een methode voor het volledig parallel oplossen van alle sub-
problemen. Nadat alle subproblemen opgelost zijn worden de targets en responsies
ge-update in de subproblemen, en worden al subproblemen weer gelijktijdig opgelost.
Strategieën 3, 4 en 5 zijn methodes die gebruik maken van de gevoeligheid van targets
en responsies die te bepalen zijn met een post-optimale gevoeligheidsanalyse.

Een andere kijk op het oplossen van het gecomponeerde probleem ligt aan de basis
voor methodes 3, 4 en 5. Subproblemen in een hiërarchie kunnen beschouwd worden
als onbekende black-boxfuncties die targets en responsies berekenen (de outputs van
een subprobleem) aan de hand van ontvangen targets en responsies (de inputs van een
subprobleem). Inputs en outputs van subproblemen in het gedecomponeerde probleem
hangen op een tweetal manieren samen. De eerst set relaties zijn de black-boxfuncties
die een relatie beschrijven tussen inputs en outputs die gekoppeld zijn door de subprob-
lemen. Een tweede set vergelijking zijn de hiërarchische relaties die beschrijven welke output met welke input verbonden is binnen de hiërarchie van het gedecomponeerde probleem. Na het samenvoegen van deze twee sets van functies kan een feasible master probleem gedefinieerd worden. In dit master probleem wordt een vector van inputs gezocht die de verschillen tussen targets en responsies, de inconsistenties, minimaliseert. Voor strategieen 3, 4 en 5 worden de black-boxfuncties lineair benaderd door eerste-orde Taylorpolynomen. De gradiënten gebruikt in de benaderingen komen overeen met de gevoeligheden van de outputs ten opzichte van verandering in de inputs bepaald met de post-optimale gevoeligheidsanalyse van subproblemen. Strategie 3 maakt gebruik van dit feasible master probleem.

Eerste-orde algoritmen maken vaak gebruik van voorzorgsmaatregelen, of safeguards, om de lineaire benaderingen betrouwbaar te houden. De lineaire benaderingen van de black-boxfuncties zijn geanalyseerd om te beslissen of en welke safeguards gebruikt kunnen worden voor ATC. Uit deze analyse blijkt dat benaderingen verkregen voor realiseerbare inputs (outputs gelijk aan inputs) niet in staat zijn veranderingen van constraintactiviteit te voorspellen, waardoor wordt gesuggereerd dat outputs onbeperkt zijn. In werkelijkheid worden de outputs begrensd door de constraints, en zijn de benaderingen dus onbetrouwbaar voor realiseerbare inputs. Voor niet-realiseerbare inputs (outputs ongelijk aan inputs) zijn de lineaire benaderingen meer betrouwbaar omdat ze informatie verschaffen over actieve constraints. Over niet-actieve constraints daarentegen is geen informatie beschikbaar. Om de onnauwkeurigheid van de benaderingen te ondervangen zijn twee safeguards voorgesteld. Strategie 4 is gelijk aan strategie 3 maar maakt gebruik van move limits die de bewegingsvrijheid beperken in het master probleem. Strategie 5 is een hybride methode ontworpen voor ATC die gebruik maakt van strategie 3 wanneer lineaire benaderingen betrouwbaar zijn (voor niet-realiseerbare inputs), maar schakelt over naar strategie 2 wanneer de lineaire benaderingen onbeperkte outputs veronderstellen (voor realiseerbare inputs).

Strategieën 2 tot 5 zijn vergeleken met een bestaande referentiestrategie. Deze referentiestrategie is een veelgebruikte hiërarchische coördinatiestrategie voor ATC. De strategieën worden vergeleken aan de hand van resultaten verkregen na het oplossen van drie testproblemen met alle vijf de strategieën. Voor twee testproblemen blijkt dat strategie 3, die gebruik maakt van gevoeligheden, beter presteert dan de overige strategieën op het gebied van convergentiesnelheid en hoeveelheid benodigde rekentijd. Voor het derde testprobleem blijkt echter dat alle vijf de strategieën vergelijkbaar presteren. Waarom alle vijf de strategieën vergelijkbaar presteren is echter niet duidelijk geworden en vereist nog verder onderzoek.

Aan de hand van de drie testproblemen kan geconcludeerd worden dat strategieën die gebruik maken van gevoeligheidsinformatie, met uitzondering van de hybride methode, beter dan de referentiestrategie presteren of ten minste vergelijkbaar.
Contents

Preface  i
Assignment  iii
Summary  v
Samenvatting (in Dutch)  ix
Symbols and notation  xvii

1 Introduction  1

2 Optimization by decomposition  5
  2.1 Optimization problem formulation: primal and dual form  5
  2.2 Decomposition: partitioning and coordination  6
  2.3 Classification of decomposition methods  11
  2.4 Dual hierarchical decomposition methods  13
  2.5 Feasible hierarchical decomposition methods  16
  2.6 Dual-feasible hierarchical decomposition methods  20
  2.7 Non-hierarchical decomposition methods  21
  2.8 Summary  23

3 Analytical target cascading  25
  3.1 The ATC methodology  26
  3.2 Sub-problem formulation  29
  3.3 Original AAO problem  32
  3.4 Relaxation of coupling constraints of AAO problem  39
  3.5 Variable allocation  42
  3.6 Formulation of ATC sub-problems  44
  3.7 Problem formulation example  46
  3.8 Coordination method  50
  3.9 Discussion  57

4 Sensitivity based coordination  61
4.1 Concept of sensitivity-based coordination .......................... 61
4.2 Computation of Jacobian matrix of sensitivities .................. 67
4.3 General sensitivity properties .................................... 73
4.4 Sensitivity based coordination algorithms ......................... 80
4.5 Discussion ....................................................... 86

5 Coordination performance comparison 89
5.1 Description of the numerical experiments ....................... 89
5.2 Geometric optimization problem 1 ............................... 94
5.3 Geometric optimization problem 2, partition 1 .................. 102
5.4 Geometric optimization problem 2, partition 2 .................. 107
5.5 Overall performance comparison ................................. 112
5.6 Discussion ....................................................... 112

6 Conclusions and recommendations 115
6.1 Review and classification of ATC ................................. 115
6.2 Sensitivity-based coordination .................................... 117
6.3 Recommendations ............................................... 120

Bibliography 123

A ATC-formulation for several problems 129
A.1 Identifying problem hierarchy in the FDT ...................... 130
A.2 Example problems ............................................. 134
A.3 Problem 1 ...................................................... 135
A.4 Problem 2 ...................................................... 136
A.5 Problem 3 ...................................................... 137
A.6 Problem 4 ...................................................... 139
A.7 Problem 5 ...................................................... 140
A.8 Problem 6 (Colville No.6) .................................... 142
A.9 Problem 7 ...................................................... 144
A.10 Problem 8 (Geometric optimization problem 1) ............... 146
A.11 Problem 9 (Static Power Scheduling) ......................... 148
A.12 Problem 10 (Heat exchanger design I) ....................... 150
A.13 Problem 11 (Heat exchanger design II) ....................... 152
A.14 Problem 12 (Geometric optimization problem 2) .......... 155
A.15 Problem 13 (Geometric optimization problem 3) .......... 160

B Discussion paper on objective function properties for ATC 165
B.1 Introduction .................................................... 165
B.2 Differences between [Kim01a] and [Etm02] .................... 165
B.3 Proposed top-objective formulation ............................. 166
B.4 Example Problem .............................................. 167
B.5 Geometric Optimization Problem ............................... 168
B.6 Conclusions and Discussion ........................................ 170
B.7 Appendix: Results for the Geometric Optimization Problem .......................... 171

C Notational differences ............................................. 173

D Augmented Lagrangian relaxation for ATC ............................................. 177
  D.1 General augmented Lagrangian penalty function .................................. 177
  D.2 Augmented Lagrangian penalty function for ATC .................................. 179

E Derivation of sensitivity equations ......................................... 181

F Sensitivities for MIMO sub-problems ........................................ 183
  F.1 Sensitivities for MIMIO sub-problems ........................................... 183
  F.2 Sensitivities for MIMDO sub-problems ........................................... 186

G Results ................................................................................. 193
  G.1 Geometric optimization problem 1 ................................................. 193
  G.2 Geometric optimization problem 2, decomposition 1 .............................. 195
  G.3 Geometric optimization problem 2, decomposition 2 .............................. 197

H MatLab files ........................................................................ 199
  H.1 Global functions ................................................................. 199
  H.2 Geometric optimization problem 1 ................................................. 201
  H.3 Geometric optimization problem 2, decomposition 1 .............................. 204
  H.4 Geometric optimization problem 2, decomposition 2 .............................. 209
Symbols and notation

Abbreviations

AAO  All-at-once
ATC  Analytical Target Cascading
FDT  Functional Dependence Table
KKT  Karush-Kuhn-Tucker
MDO  Multi-disciplinary Optimization
MIMIO Multiple Input Multiple Independent Output
MIMDO Multiple Input Multiple Dependent Output
NLP  Non-linear Programming
SISO Single Input Single Output
WUM  Weight Updating Method

Symbols

\(\epsilon\) error
\(\gamma\) convergence rate constant
\(\lambda\) Lagrange multiplier associated with equality constraint
\(\mu\) Lagrange multiplier associated with inequality constraint
\(\pi\) penalty function
\(\rho\) convergence rate
\(\tau\) ATC termination tolerance
\(e\) element
\(f\) objective function
\(g\) inequality constraint function
\(h\) equality constraint function
\(K\) number of required iterations
\(L\) Lagrangian dual function
\(n\) number of optimization variables
\(n^q\) number of inputs
\(n^Q\) number of outputs
\(n^{Q'}\) number of independent outputs
Symbols and notation

\( N \) number of levels in the problem hierarchy
\( N_f \) number of required function evaluations
\( m \) number of constraints
\( p \) number of sub-problems
\( P \) optimization sub-problem
\( T_i \) target for the value of the objective function

\( \mathcal{E} \) set of elements
\( C \) set of children
\( \mathcal{Q} \) realizable domain

\( \lambda \) vector of Lagrange multipliers associated with equality constraints
\( \mu \) vector of Lagrange multipliers associated with inequality constraints
\( \theta \) vector of inconsistencies

\( a \) vector of response analysis functions
\( c \) vector of constraints of the feasible master problem
\( F \) vector of input-output functions
\( g \) vector of inequality constraints
\( g_C \) vector of inequality coupling constraints
\( h \) vector of equality constraints
\( h_C \) vector of equality coupling constraints
\( m \) vector of move limits
\( r \) vector of child level copies of response target variables
\( R \) vector of parent level copies of response target variables
\( s \) vector of shared optimization variables
\( w \) vector of weights
\( w^R \) vector of weights associated with response target variables
\( w^Y \) vector of weights associated with linking target variables
\( x \) vector of local optimization variables
\( \bar{x} \) vector of all optimization variables
\( y \) vector of child level copies of linking target variables
\( Y \) vector of parent level copies of linking target variables
\( z \) vector of optimization variables

\( A \) connection matrix
\( B \) right hand matrix of sensitivity system of equations
\( C \) left hand matrix of coefficients of sensitivity system of equations
\( J \) Jacobian matrix of sensitivities
**Notation**

Let $x$ and $y$ be scalars, and $\mathbf{x}$ and $\mathbf{y}$ be vectors. Scalar $x_k$ denotes element $k$ of vector $\mathbf{x}$. Also, the scalar $a_{(k,l)}$ is the element at row $k$ and column $l$ of matrix $A$. If matrix $A$ has $n$ rows and $m$ columns, then $A = [a_{(k,l)}]_{n \times m}$.

Furthermore:

- $\mathbf{x}^\ast$ optimal solution of $\mathbf{x}$
- $\mathbf{x}|_y$ evaluation, value of $\mathbf{x}$ evaluated at $y$
- $\langle \mathbf{x} \rangle_i$ selection, element $i$ of vector $\mathbf{x}$
- $||\mathbf{x}||_i$ $l_i$ norm of vector $\mathbf{x}$
- $\frac{dx}{dy}$ full derivative of $x$ with respect to $y$
- $\frac{\partial x}{\partial y}$ partial derivative of $x$ with respect to $y$
- $\mathbf{x} \circ \mathbf{y}$ element by element multiplication of vectors $\mathbf{x}$ and $\mathbf{y}$, the result is a vector

In this notation:

\[
\langle \mathbf{x} \rangle_i = x_i,
\]

\[
\mathbf{x} \circ \mathbf{y} = [x_k]_{k \times 1} \circ [y_k]_{k \times 1} = [x_k y_k]_{k \times 1} = [x_1 y_1, x_2 y_2, \ldots, x_k y_k],
\]

\[
\frac{dx}{dy} = \begin{bmatrix} \frac{dx}{dy_1}, & \frac{dx}{dy_2}, & \ldots, & \frac{dx}{dy_l} \end{bmatrix},
\]

\[
\frac{\partial x}{\partial y} = \begin{bmatrix} \frac{\partial x}{\partial y_1}, & \frac{\partial x}{\partial y_2}, & \ldots, & \frac{\partial x}{\partial y_l} \end{bmatrix},
\]

\[
\frac{d\mathbf{x}}{d\mathbf{y}} = \begin{bmatrix} \frac{d x_1}{d y_1}, & \frac{d x_1}{d y_2}, & \ldots, & \frac{d x_1}{d y_l} \\
\frac{d x_2}{d y_1}, & \frac{d x_2}{d y_2}, & \ldots, & \frac{d x_2}{d y_l} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{d x_k}{d y_1}, & \frac{d x_k}{d y_2}, & \ldots, & \frac{d x_k}{d y_l} \end{bmatrix},
\]

\[
\frac{\partial \mathbf{x}}{\partial \mathbf{y}} = \begin{bmatrix} \frac{\partial x_1}{\partial y_1}, & \frac{\partial x_1}{\partial y_2}, & \ldots, & \frac{\partial x_1}{\partial y_l} \\
\frac{\partial x_2}{\partial y_1}, & \frac{\partial x_2}{\partial y_2}, & \ldots, & \frac{\partial x_2}{\partial y_l} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{\partial x_k}{\partial y_1}, & \frac{\partial x_k}{\partial y_2}, & \ldots, & \frac{\partial x_k}{\partial y_l} \end{bmatrix}.
\]
Symbols and notation
Chapter 1

Introduction

The design of technologically advanced systems is becoming an increasingly complicated task. The size of and the required level of expertise for design projects often prohibit the design problem to be solved as a whole; consequently the problem has to be decomposed into smaller, more manageable parts. Each of these parts, or sub-problems, is tackled by different design teams or disciplines, which act rather autonomously. As a result, one design group does not know its decisions affect the other disciplines. To deal with multiple disciplines, a systematical approach to product design is desired; designing a product is thus equivalent to designing a system of coupled disciplines.

Design optimization is a mainstream activity in product development; it comprehends the selection of the ‘best’ design within the available means [Pap00]. Software tools for analysis and numerical optimization are nowadays indispensable for this selection. The constantly increasing abilities of these tools stimulate the design optimization activities. Due to the systems approach to product design, there has been an increased need to study design optimization methods that are able to deal with this systems approach.

An important phase of the design process, involving design optimization, is the conceptual design phase. The selection of a ‘best’ conceptual design is based on meeting overall design specifications (or targets), maximizing revenues, or minimizing total costs. These overall objectives have to be translated into objectives for the individual disciplines. Propagation of the overall objectives throughout the hierarchy is a major issue. Disciplines are never independent of each other. Moreover, discipline design problems should be solved such that the overall objective is met as well as possible.

Decomposition of large problems into a system of coupled elements has been an active field of research during the last four decades. In mathematical programming, decomposition refers to the process of exploiting the problem structure to both define and solve a set of smaller problems. In mathematical programming, the separation into smaller elements is called partitioning. A brief review of classic decomposition methods for mathematical programming is presented in [Wag93], amongst which dual...
decomposition methods [Dan60, Las68], goal-coordination methods [Wis78, Dia91], and separability based methods [Ben62, Wis78, Sob82a]. All methods partition the original problem into a number of connected sub-problems. Sub-problems are placed on a single level and a hierarchical framework is imposed to coordinate the communication between elements. Lootsma discusses the potential reduction in computational effort by solving sub-problems on parallel computers [Loo89].

A second approach to decomposition emerged from multi-disciplinary (MDO) approaches in practice. In contrast to the rigorously formulated mathematical approaches, these engineering-based methods tend to be heuristic. Partitions of MDO methods are either aspect-driven, or component-driven. They deal with “black-box” discipline analysis models that cannot be represented by analytical equations in the system optimization problem. Sobieszczanski-Sobieski and Haftka present a critical overview of MDO approaches [Sob97]. One of the most promising MDO methods was Collaborative Optimization [Bra96, Tap97]. However, inherent to its formulation, the method experienced very poor analytical and computational behavior causing convergence difficulties [Ale02]. More recently, Sobieszczanski-Sobieski and Kodiyalam [Sob01], and Haftka and Watson [Haf04] presented new MDO methods.

Another relatively new approach to decompose engineering systems is analytical target cascading (ATC). Kim [Kim01a] presented analytical target cascading a decomposition method for solving multilevel optimal design problems. The hierarchical partition of problems is based on a natural hierarchy present in the original design problem. Specifications or targets are set for the overall system. Target cascading describes a method for propagating appropriate targets to various disciplines that form the system. For each discipline an optimization problem is formulated, the objective is to find responses that meet these targets as closely as possible while satisfying the disciplinary constraints. Theoretically convergent multilevel optimization methods are presented in Michelena et al. [Mic03].

### Previous work on analytical target cascading

The ATC methodology has been developed at the Optimal Design Laboratory of the University of Michigan, and was first formalized by Kim [Kim01a]. Michelena et al. [Mic03] theoretically investigated convergence properties of ATC. They proved that, under convexity assumptions, the ATC process converges to the optimal solution of the original design target problem. Michalek and Papalambros revised the notation and problem statements for ATC for clarity and rigor [Mic04b].

Analytical target cascading has been successfully applied to a wide range of vehicle design problems [Kim01b, Mic01, Kim02], and simple manufacturing system design [Hul03]. Kokkolaras et al. extended the target cascading formulation to the design of product families [Kok02].
Etman et al. discussed the advantages of concurrent programming languages for the coordination of multidisciplinary optimization [Etm04]. They proposed the use of the $\chi$ language and software tools (Kleijn and Rooda [Kle01]) for specifying the coordination of the ATC process. Preliminary experiments with the ATC coordination implemented using the $\chi$ language showed fast convergence of the ATC process for a three-level example problem but inaccurate results [Etm02]. Hulshof et al. continued on these experiments and implemented truncation of communicated target values to obtain more accurate solutions of the ATC process, at the expense of increased computational cost [Tze03, Hul03]. Michalek et al. presented a weighting update method to reduce computational cost by iteratively increasing the inconsistency penalties [Mic04a]. However, all these numerical experiments on the ATC convergence were carried out for just one analytical optimization problem decomposed into two or three levels [Kim01a, Etm02]. A large set of ATC test problems for numerical experimentation has not been proposed so far.

ATC coordination methods currently only communicate target and response values, which results in a slowly converging process that requires a large computational effort. Many other numerical methods use sensitivity information to speed up the convergence of the iterative solution process. Compare, for instance, algorithms to find fixed points of a function; zero-order methods only use function information to determine the next step, while faster first-order methods (e.g. Newton’s method) also use gradient information to define the next step. With this additional sensitivity information, convergence speed is significantly increased. This suggests that for ATC sensitivity information of responses with respect to targets may be helpful to generate ‘smarter’ targets and increase the convergence speed.

**Objectives and approach**

The objective of this research is to investigate how sensitivities of responses with respect to targets can be used to speed up the ATC coordination. With this additional information, the great computational effort required to solve the ATC problem may be decreased.

To assess the applicability of a sensitivity based coordination method, first the original ATC process is given a closer look. ATC is presented in the general context of decomposition-based optimization. Chapter 2 gives an overview of this area of research, and relevant terminology is introduced as well as general characteristics of decomposition methods. Chapter 3 describes analytical target cascading in greater detail. The classification and formal description of ATC provides new insights in the methodology and allows it to be compared to other decomposition methods. Interesting observations and opportunities are presented in a discussion at the end of this chapter. An illustrative example for ATC is presented as well. On the basis of this thorough analysis of ATC, specifically regarding the decomposition structure, a new series of example problems has been derived from original all-at-once test problems such as found in e.g. [Hoc81].
After the investigation of ATC and several decomposition strategies, the use of sensitivity information in the coordination process is discussed in Chapter 4. After presenting a gradient-based coordination concept for ATC, the computation of sensitivities through post-optimality analyses is discussed. With this post-optimality approach, the properties of sensitivities for ATC are analyzed. After this analysis, three sensitivity-based coordination algorithms are defined. In Chapter 5, the numerical behaviors of the new coordination algorithms and other existing strategies are compared after which suggestions for possible improvement of the method follow.

Major findings of this research are presented in the conclusions (Chapter 6) as well as recommendations for future research.
Chapter 2

Optimization by decomposition

Before starting with the analysis of the analytical target cascading methodology, in this chapter, the use of decomposition in optimization problems is discussed. The formulation and terminology for optimization problems is introduced, as well as terminology and techniques for decomposition-based optimization. For most of the decomposition strategies several example methods are mentioned as well as some of their characteristic properties. The overview of these methods is by no means exhaustive but serves to illustrate the variety of methods for decomposition-based optimization.

2.1 Optimization problem formulation: primal and dual form

An optimal design problem can often be formulated as an optimization problem with a set of decision variables, one objective function depending on these decision variables that reflects the goal of the design problem, and a set of constraints representing technological bounds on the design and (equilibrium) relations on the physics.

Assuming that design variables and mathematical relations for the objective and constraints have been identified and formulated, the optimal design problem can be stated as a non-linear programming (NLP) optimization problem in negative-null form [Pap00]:

\[
\min_{z \in \mathbb{R}^n} f(z) \\
\text{subject to } h(z) = 0 \\
g(z) \leq 0
\]  

(2.1)

with design variables \(z, \mathbb{R}^n\) the \(n\)-dimensional real space, objective function \(f(z)\), equality constraints \(h(z)\) and inequality constraints \(g(z)\). The research presented in this
report also holds when some design variables are integer, in which case appropriate optimization techniques have to be selected. Throughout the report, Eq. (2.1) will be referred to as the \textit{primal problem}, and design variables $z$ will be referred to as the \textit{primal variables}.

Next to the primal problem (2.1), the closely related \textit{dual problem} formulation has been proposed. Among the various duality formulations, consider the Lagrangian duality formulation. The Lagrangian dual function $L$ is defined as:

$$L(z, \lambda, \mu) = f(z) + \lambda^T h(z) + \mu^T g(z) \quad (2.2)$$

with $\lambda$ and $\mu$ the vectors of Lagrange multipliers associated with equality and inequality constraints, respectively. Lagrange multipliers $(\lambda, \mu)$ are also called the \textit{dual variables} of the primal problem. With the Lagrangian dual function (2.2) the \textit{Lagrangian dual problem} of problem (2.1) can be formulated [Baz93, Ber95]:

$$\max_{\lambda, \mu} \quad L(\lambda, \mu, z),$$

subject to $\mu \geq 0$, \hspace{1cm} (2.3)

where $L(\lambda, \mu, z)$ is the solution to problem:

$$\min_z \quad L(\lambda, \mu, z). \quad (2.4)$$

Since the dual problem consists of maximizing the minimal solution of the Lagrange function [Baz93], it is sometimes referred to as the max-min dual problem. For some problems the dual problem is easier to solve than the problem in primal form.

\section*{2.2 Decomposition: partitioning and coordination}

Decomposition methods aim at exploiting the structure of the primal or dual problem by \textit{partitioning} problem variables and functions, and implementing an appropriate \textit{coordination strategy} to solve the partitioned problem. A decomposition method refers specifically to the procedure outlined in Figure 2.1, consisting of four steps.
PARTITIONING

1. Identify the structure of the NLP primal problem statement.

2. Define a master problem and one or more sub-problem(s).
   (a) Partition variables based on the structure in (1).
   (b) Partition functions into a master problem and sub-problem(s) based on the variable partition.

COORDINATION

3. Select a coordination strategy.
   (a) Select a solution algorithm for the master problem.
   (b) Select a solution algorithm for the sub-problem(s).

4. Implement the coordination strategy iteratively to solve the partitioned NLP problem.

Figure 2.1: Basic steps of a decomposition method [Wag93]

Using the problem structure identified during step 1, a decomposition method partitions the primal problem variables and functions. Variables and functions are grouped into a number of weakly linked partitions; these partitions can be linked through either variables or functions. For each partition, an optimization sub-problem is formulated. Variables assigned to a single master sub-problem are called coordinating variables, while variables and constraints assigned to the sub-problems are labelled local variables. The resulting two-level problem hierarchy consists of one master problem placed over a number of sub-problems. A multi-level partition can be obtained by further decomposing the sub-problems.

Several partitioning strategies exist: aspect (or discipline), object, or model-based partitioning methods. Aspect and object partitions are partitions based on a system’s aspects (e.g. aerodynamics, structures, propulsion) or components (e.g. engine, transmission, chassis). These natural partitions often match the human organization of an optimal design problem, i.e. the partition is based on design teams each with its individual field of expertise. Model-based partitioning is a mathematical method for finding a more ‘balanced’ partition of the original problem. These methods aim at finding a balance between the number of partitions and the amount of linking between these partitions. In theory, model-based partitions should be preferred over natural partitions since in principle the ‘best’ partition for the design problem is sought. In practice however natural partitions may be required for matching the human organization of the design
teams. Note that for many optimal system design problems, model-based and natural partitioning are likely to result in similar partitions, because objects or disciplines often have individual technological constraints and physical relations.

To solve the partitioned problem a *coordination strategy* is defined. The strategy usually exploits problem properties identified from the partition. Common manipulation techniques are the introduction of support variables and formulation of the dual problem. Coordination methods often involve techniques such as gradient methods, penalty methods and constraint relaxation strategies.

A method for decomposed optimization is defined by both the partitioning of a problem and the solution of the partitioned problem using a coordination strategy. Most of the methods are especially designed to exploit one specific characteristic of a problem, and are therefore only applicable to problems with specific properties. The following section introduces these properties of problems relevant in decomposition and how the properties can be identified.

### Characteristic problem properties

Five properties of the primal problem are relevant in the context of decomposition methods:

1. Additive separability of functions
2. Linearity/convexity of functions
3. Monotonicity
4. Linking variables
5. Linking functions

A function is said to be *additive separable* when the function is a sum of terms, each term depending on disjoint subsets of primal problem variables (unlinked partitions). A function is *linear* when it depends linearly on the optimization variables, i.e. the function gradient is constant over the domain. For linear problems, more classical methods can be used. *Monotone* functions have a gradient that does not change sign over the domain. When all functions of a problem are monotone, analytical solution of sub-problems is facilitated. Functions are *convex* when the Hessian of the function is either semi-positive or semi-negative definite over the domain. Convexity assumptions
are often required for convergence proofs. Linearity and convexity can easily be identified, and techniques for exploiting monotonicity are well documented [Pap00]. Linking variables are variables common to more than one sub-problem, and linking constraints are constraints depending on more than one subset of local variables.

Linking variables and functions are commonly identified from a matrix representation of a problem’s structure. Two of the most common structure representations are the functional dependence table and the adjacency matrix. The functional dependence table (FDT) is a matrix representation of the structure of the Jacobian of the objective and constraint functions. Rows are labelled with function names, and columns with variables names, element \((i, j)\) of the FDT is non-zero when function \(i\) depends on variable \(j\), otherwise the element is zero. Shading is often used to indicate non-zero elements of the matrix.

The adjacency matrix can be defined for both variables and functions. For variables, rows and columns are both labelled with variable names, and element \((i, j)\) has a non-zero entry when one or more functions depend on both variable \(i\) and variable \(j\), otherwise the element is zero. For functions, the adjacency matrix can be constructed similarly, where element \((i, j)\) is non-zero when both functions \(i\) and \(j\) depend on at least one common variable.

Although the structure of a problem can be identified from both the FDT and the adjacency matrix, the FDT will suffice to illustrate the identification of structure in optimization problems. Model-based partitioning methods for finding variable partitions exist for both the FDT (See e.g. [Wag93, Kri97]) and the adjacency matrix (See e.g. [Mic97]).

To illustrate the use of the FDT, consider problem (2.5) with six design variables \(z_1, \ldots, z_6\), objective \(f\) and seven equality constraints \(h_1, \ldots, h_7\). The FDT of this problem, after rearranging the columns, is depicted in Figure 2.2(a). Note that the objective \(f\) is additive separable: \(f = f_1 + f_2\), with \(f_1 = z_1 z_2^2\), \(f_2 = z_1 z_3^3\). These two terms of the objective are represented in two separate rows of the FDT.

\[
\begin{align*}
\min_{z_1, \ldots, z_6} & \quad f : z_1 z_2^2 + z_1 z_3^3, \\
\text{subject to} & \quad h_1 : z_1 + z_2 - 2 = 0, \\
& \quad h_2 : z_2 + z_3^2 = 0, \\
& \quad h_3 : z_1 z_2 z_3 - 5 = 0, \\
& \quad h_4 : z_1 z_4 - 10 = 0, \\
& \quad h_5 : z_4^2 + z_5^2 - 2 = 0, \\
& \quad h_6 : z_1 z_4 z_5 - 8 = 0, \\
& \quad h_7 : z_1 z_2 z_4^2 - z_5 z_3 z_2^2 = 0.
\end{align*}
\]
Variable $z_1$ and constraint $h_7$ prohibit separation of the problem into two smaller subproblems. One problem $P_1$ with variable partition $x_1 = [z_2, z_3]$, objective $f_1$ and constraint partition $h_1 = [h_1, h_2, h_3]$, and a second $P_2$ with variables $x_2 = [z_4, z_5]$, objective $f_2$ and constraints $h_2 = [h_4, h_5, h_6]$. Because variable $z_1$ prohibits full separation of the problem, it is defined to be a linking variable, or shared variable; $h_7$ is denoted a linking constraint, or coupling constraint, for the same reason. In general, linking variables and linking constraints prohibit full separation of an optimization problem. When linking variables are (temporarily) fixed and treated as parameters, and linking constraints are removed, a problem becomes fully separable.

For these particular partitions, variable subsets $x_1$ and $x_2$ are called local variables to problems $P_1$ and $P_2$ respectively, objectives $f_1$ and $f_2$ are local objectives, and $h_1$ and $h_2$ are local constraints. Define two sets of local functions $f_1 = [f_1, h_1]$ and $f_2 = [f_2, h_2]$. By additionally defining a linking variable partition $s = z_1$ and a linking constraints partition $h_C = h_7$ a compact FDT of problem (2.5) for these specific partitions can be constructed (see Figure 2.2(b)). The compact FDT has a non-zero entry at $(i, j)$ when at least one function of function subset $i$ depends on at least one variable of variable subset $j$; elements are non-zero otherwise. When removing shared variable columns and coupling constraint rows from the compact FDT, the full separability of the problem is represented by the remaining block diagonal structure of the compact FDT.

**General coordination strategy**

After partitioning the variables and functions based on the structure of the problem, most decomposition methods exploit one or more of the defining properties to define
2.3 Classification of decomposition methods

Decomposition methods can be categorized as **hierarchical** and **non-hierarchical**. Hierarchical methods can solve problems which are linked through a small number of either constraints or variables. Hierarchical methods also exist for solving problems with a mixture of both coupling constraints and linking variables, but the number of coupling constraints must be small. Hierarchical methods use dual variables associated with the linking constraints, primal variables, or a subset of both as coordinating variables. When temporarily fixing a subset of the primal and/or dual variables, subproblems exist which can be solved independently, i.e. an information hierarchy exists in the problem: information is communicated down from the master problem to multiple sub-problems and back upwards to the master problem. For problems with a multilevel hierarchy, elements are linked to elements at the next higher level, or at the next lower level, but never within a level; a clear information hierarchy exists in the problem.

Non-hierarchical methods are used in the decomposition of non-hierarchical problems with a large amount of coupling between elements. Elements are coupled through both
variables and constraints, and exchange information with many other elements; no clear information hierarchy exists. By introducing approximations of one element’s effect on others, elements can be decoupled from the problem hierarchy. Non-hierarchical decomposition methods then impose a strictly hierarchical coordination strategy over the elements decoupled from the non-hierarchical problem.

To illustrate the difference between hierarchical and non-hierarchical problems consider Figure 2.4. Within a structure one can speak of an element’s ancestors, parents, children and grand-children. For example in the hierarchical structure of Figure 2.4(a), element $e_0$ is both the parent of $e_{11}$, and the grand-parent of elements $e_{21}$. Elements $e_{21}$ and $e_{22}$ are both children of $e_{11}$, and grand-children of $e_0$. For hierarchical structures, elements only have connections with their parent and their children (See Figure 2.4(a)), while for non-hierarchical structures, elements may have connections with any other element within the hierarchy (See Figure 2.4(b)).

Both hierarchical and non-hierarchical decomposition methods have been intensively studied by both engineers and mathematicians. Sobieszczanski-Sobieski and Haftka [Sob97], and Kodiyalam and Sobieszczanski-Sobieski [Kod01] reviewed several of the methodologies with an engineering background; Lootsma [Loo89], Wagner [Wag93], and Nelson [Nel97] provide an overview of several decomposition strategies with a mathematical background. [Opt00] presents an extensive list of publications regarding decomposition methods with both engineering and mathematical backgrounds.
2.4 Dual hierarchical decomposition methods

First, hierarchical methods are discussed. Further distinction between hierarchical methods can be made: dual, feasible, and dual-feasible methods use dual, primal, or both dual and primal variables as coordinating variables, respectively. The additional defining properties additive separability, linearity/convexity and monotonicity allow the use of different methods. Several dual hierarchical methods and their required defining properties for the primal problem are discussed in this section.

Dual methods are used to decompose problems with linking functions. For these methods, the dual formulation of the primal problem, as illustrated in Section 2.1, allows the partition of the problem into a two level problem where dual variables \((\lambda, \mu)\) associated with linking constraints, are allocated to the master problem and primal variables \(x\) are allocated in sub-problems. For additive separable objective and constraints, the vector of optimization variables \(x\) can be partitioned into \(p\) vectors \(x_i\) for \(i = 1, \ldots, p\) with disjoint index sets for \(i \neq j\). Under these conditions, the Lagrangian becomes additive separable with respect to \(x_i\), and is therefore separable into \(p\) terms as well:

\[
L(x, \lambda, \mu) = \sum_{i=1}^{p} L_i(x_i, \lambda, \mu)
\] (2.6)

One master problem is defined to maximize the Lagrangian with respect to dual variables \(\lambda, \mu\) while treating primal variables \(x_i\) as fixed parameters. Additionally \(p\) sub-problems are defined to minimize the individual terms of the Lagrangian with respect to primal variable subsets \(x_i\). This max-min strategy, however without separation can also be seen in the dual problem formulation of an optimization problem (See Eq. (2.3)).

For most methods, an iterative max-min coordination strategy serves to solve the partitioned problem. Iteratively the master problem is solved for \(\lambda, \mu\) and optimal values \(\lambda^*, \mu^*\) are used in the solution of the sub-problems which in turn send their optimal solutions \(x_i^*\) back to the master problem which then is solved for \(\lambda, \mu\) again. This process is repeated until some convergence criterion is met. A generic algorithm for dual decomposition methods is displayed in Figure 2.5, and Figure 2.6 illustrates the dual coordination strategy.

Problem structure for dual decomposition methods

As mentioned before, problems for dual decomposition methods require the presence of coupling constraints. The compact FDTs required for dual decomposition methods are depicted in Figure 2.7, where \([g_C, h_C]\) is used to label the linking constraints. From the FDT of problems for dual methods, full separation with respect to \(x_i\) could be obtained.
Chapter 2. Optimization by decomposition

1. Initialize $k = 0$, and $(x_k, \lambda_k, \mu_k)$

2. Holding $(\lambda_k, \mu_k)$ constant, obtain $x_k^*$ by solving:
   \[
   \min_x L(x, \lambda_k, \mu_k)
   \]

3. Holding $x_k^*$ constant, obtain $(\lambda_k^*, \mu_k^*)$ by solving:
   \[
   \max_{\lambda, \mu} L(x_k^*, \lambda, \mu)
   \]

4. If converged stop; otherwise increment $k$, set $(x_k, \lambda_k, \mu_k) = (x_{k-1}^*, \lambda_{k-1}^*, \mu_{k-1}^*)$ and go to 2.

Figure 2.5: Algorithm for generic dual decomposition method [Wag93]

Figure 2.6: Two level structure of dual methods [Wag93]

if coupling constraints $h_C$ are removed. A dual decomposition realizes the separation with respect to $x_i$ by temporarily removing the coupling constraints and solving for the dual variables of these constraints in a master problem. Wagner [Wag93] gives an overview of dual decomposition methods, where each of the presented dual methods have different requirements regarding the remaining problem characteristics.

To summarize, all dual decomposition methods require coupling constraints, but the remaining characteristic properties of the primal problem differ for each method. Dual methods exploit separability of the Lagrangian for fixed dual variables.

Goal coordination methods

Problems with a coupled constraint set structure as illustrated in Figure 2.8(a) can be reformulated to a separable constraint set and a set of coupling constraints, required for the application of dual methods. This manipulation technique is illustrated in Fig-
2.4. Dual hierarchical decomposition methods

Figure 2.7: Compact FDTs for dual decomposition methods

Figure 2.8: Constraint set structure manipulation for goal coordination methods

Wismer and Chattergy proposed to extend this approach with the KKT conditions to derive a gradient algorithm for the master problem [Wis78]. Diaz and Belding proposed a similar method with a master problem using goal programming instead of the Lagrangian function [Dia91]. More recently, Lassiter et al. proposed a Lagrangian relaxation of the introduced coupling constraints, and computation of dual variables using an update mechanism [Las02]. For convergence, the terms of the additive separable objective must be bounded from below.
2.5 Feasible hierarchical decomposition methods

Where dual methods require linking functions, feasible methods require linking variables. In feasible decomposition methods, primal variables are partitioned into a set of global variables $s$, and a set of local variables $x$. Commonly, the linking variables to a problem are global and are therefore selected as coordination variables. The global variables $s$ are updated in a master problem, and optimal values $s^*$ are sent to the sub-problem. In turn, the sub-problem is solved for $x$, and optimal values $x^*$ are returned to the master problem. When it is possible to partition the vector of local variables $x$ into $p$ disjoint sets $x_i$, and local constraints into $p$ sets, with set $i$ only depending on variable set $x_i$, the sub-problem can be separated into $p$ independent sub-problems. Because the coordinating variables are primal variables, feasible decomposition methods are often named model coordination methods.

The two-level structure of feasible decomposition methods is illustrated in Figure 2.9. Feasible methods are widely applied in design problems because, even without convergence, the intermediate solutions are feasible with respect to sub-problem constraints and solutions are usually an improvement of the original design.

\[
\begin{align*}
\min_{s \in \mathbb{R}^n} & \quad f(s, x^*) \\
\text{subject to} & \quad h(s) = 0 \\
& \quad g(s) \leq 0
\end{align*}
\]

\[
\begin{align*}
\min_{x \in \mathbb{R}^n} & \quad f(s^*, x) \\
\text{subject to} & \quad h(s^*, x) = 0 \\
& \quad g(s^*, x) \leq 0
\end{align*}
\]

Figure 2.9: Two level structure of feasible methods [Wag93]

For a feasible method to have convergence properties, the dependence of local optimal variables $x^*$ on global variables $s$ must be accounted for [Wag93]. An explicit solution to this dependence is rarely obtained because of the complex nature of optimization problems, therefore approximations of $x^*(s)$ are required. For small dimensions of $s$ (three or less), $x^*(s)$ can be traced out [All90], and for larger dimensions of $s$, linear approximations can be constructed using sensitivity derivatives [Geo71].
2.5. Feasible hierarchical decomposition methods

Feasible separable methods

Problems with linking variables $s$, $p$ sets of local variables $x_i$ for $i = 1, \ldots, p$, and a primal form as in Eq. (2.7) can be solved with several feasible decomposition methods named feasible separable methods.

\[
\begin{align*}
\min_{s, x_1, \ldots, x_p} & \quad f_0(s) + \sum_{i=1}^{p} f_i(s, x_i) \\
\text{subject to} & \quad g(s) \leq 0 \\
& \quad h(s) = 0 \\
& \quad g_i(s, x_i) \leq 0 \quad \text{for} \quad i = 1, \ldots, p \\
& \quad h_i(s, x_i) = 0 \quad \text{for} \quad i = 1, \ldots, p
\end{align*}
\] (2.7)

The compact FDT for the problem of Eq. (2.7) is depicted in Figure 2.10. When fixing the linking variables the problem becomes fully separable in $x_i$. The Lagrangian for the primal problem is separable with respect to local primal variables $x_i$ and dual variables $\lambda_i, \mu_i$ for $i = 1, \ldots, p$, as illustrated in Eq. (2.8). For fixed $s$, the $p + 1$ terms of the Lagrangian can be minimized independently.

\[
L(s, x, \lambda, \mu) = L_0(s, \lambda_0, \mu_0) + \sum_{i=1}^{p} L_i(s, x_i, \lambda_i, \mu_i)
\] (2.8)

Figure 2.10: Compact FDT for feasible separable methods

Wagner [Wag93] stated that Kirsch’s decomposition method describes the most generic algorithm for solving the partitioned problem [Kir81], this generic algorithm is displayed in Figure 2.11.

Wagner [Wag93] also gives an overview of special implementations of Kirsch’s algorithm as developed by Rosen [Ros63], Benders [Ben62], and Azarm and Li [Aza88]. Rosen’s method [Ros63] can be applied to a special structure of problem functions of
1. Initialize \( k = 0 \), and choose vector \( s_0 \).

2. Holding \( s_k \) constant, solve the \( p \) independent sub-problems with respect to \( x_i \) to obtain \( x_i^* \).

3. Modify the value of \( s_k \) to \( s_{k+1} \) to reduce the objective in the feasible domain by a sufficient amount \( \delta \),

\[
\sum_{i=1}^{p} f_i(s_{k+1}, x_i) - \sum_{i=1}^{p} f_i(s_k, x_i) \leq \delta
\]

4. If converged stop; otherwise set \( k = k + 1 \) and return to step 2.

Figure 2.11: Algorithm for Kirsch’s generic feasible separable decomposition method

Eq. (2.7). Sub-problems are solved independently and a linear approximation of \( x_i(s) \) is constructed in the sub-problems. The master problem is formulated strictly in terms of \( s \). The algorithm converges for problems with a separable objective, and convex and separable constraints.

Benders decomposition [Ben62] is constructed for problems with constraints linear in \( x \), but nonlinear with respect to \( s \). No assumptions are made with respect to continuity of \( s \), nor its feasible domain. The \( s \)-dependence of the constraint set is exploited to formulate a master problem relating \( s \) and \( x \). The linearity with respect to \( x \) is exploited to formulate dual sub-problems in terms of extreme points.

Azarm and Li’s method [Aza88] can be applied to problems of the form of (2.7) when no equality constraints exist and the problem’s functions are monotonic with respect to \( x_i \). The algorithm exploits monotonic properties of the sub-problems when \( s \) is held fixed. Furthermore the number of active constraints in a sub-problem must be equal to the dimension of \( x_i \).

### Engineering-based feasible separable methods

Many of the engineering-based multi-disciplinary optimization (MDO) approaches deal with problems linked through a set of shared variables. Several of these methods are presented here. All methods are based on a multidisciplinary decomposition of the original AAO design problem. For each discipline, an optimization sub-problem is defined that uses the variables of other disciplines as fixed parameters. Over these disciplinary sub-problems, a coordination problem is superimposed to account for the coupling between disciplines.

Sobieski [Sob82a] developed an algorithm for solving problems of the form of Eq. (2.7),
but with an objective function only depending on the global variables $s$. Since the objective only depends on $s$, it only appears in the master problem. For each sub-problem, a penalty-like objective function is formed, minimizing a measure of constraint violations. Linear approximations of sub-problems are included in the master problem by computing sensitivities of a sub-problem’s objective function and of local variables with respect to $s$. No proof of convergence is given, but move limits are suggested to keep the linear approximations valid.

Sobieski’s method can be classified as a Lagrangian penalty decomposition: all inequality constraints are included in the objective using Lagrangian penalty functions. The Lagrangian of the reformulated problem becomes additively separable with respect to $x_i$. The master problem minimizes the Lagrangian with respect to $s$, and sub-problems minimize each term in the Lagrangian independently with respect to local variables $x_i$.

Braun et al. [Bra96, Tap97] developed a method called Collaborative Optimization. Although the underlying ideas for this method are intuitively appealing, the method suffers from analytical and computational difficulties, as proven by [Ale02]. Sobieszczanski-Sobieski et al. developed a bilevel approach BLISS that uses optimum sensitivity data computed at the disciplinary sub-problems in the master problem. Although no convergence difficulties are reported, convergence for this method has not been proven. More recently, Haftka and Watson [Haf04] have presented a MDO method and supplied convergence proof.
2.6 Dual-feasible hierarchical decomposition methods

Where dual or feasible decomposition methods allow only linking through either variables or through constraints. Dual-feasible methods allow a mixture of both. Figure 2.12 illustrates the FDT structure of problems for dual-feasible decomposition methods. Using both dual and feasible methods, a three level approach can be developed: linking functions are solved for by a dual method, and the dual sub-problem is further decomposed using a feasible method accounting for linking variables.

The optimization problem for the dual master problem is expressed in dual variables \((\mu, \lambda)\) associated with coupling constraints, while fixing all primal variables \((s, x)\). The feasible master problem has feasible coordinating variables \(s\) as optimization variables, while treating \((\mu, \lambda, x)\) as a set of parameters. Finally, for the bottom-level subproblems, optimization problems are formulated with local variables \(x\) as variables, and \((\mu, \lambda, s)\) as parameters. A generic algorithm for the dual-feasible method is presented in Figure 2.14 and the three level coordination structure is illustrated in Figure 2.13. Ritter [Rit67] proposed a dual-feasible method which requires problems to be linear in \(x_i\), and convergence is guaranteed only under explicit conditions.

![Figure 2.12: Compact FDT for dual-feasible methods](image)

![Figure 2.13: Three level dual-feasible method](image)
2.7 Non-hierarchical decomposition methods

The hierarchical methods are characterized by independent sub-problems, and linking variables or linking functions are solved for by a single master problem, dual-feasible methods presented a first step towards non-hierarchical methods. Like dual-feasible methods, non-hierarchical methods deal with problems which have both linking functions and linking variables. In contrast to dual-feasible methods which allow only a small number of coupling constraints, coupling for non-hierarchical methods may be very dense. Problems with both linking variables and densely coupled constraints have the compact FDT structure as displayed in Figure 2.15.

A dual-feasible method may not suffice to solve the densely coupled non-hierarchical problem, however independent sub-problems can be constructed with approximations of one sub-problems effect on another. Wagner [Wag93] gives an overview of non-hierarchical methods proposed by Wismer and Chattergy [Wis78], Sobieski [Sob88], Pan and Diaz [Pan90], Wu [Wu91], and Unger et al. [Ung92]. A summary of some of these methods is presented below.

Wismer and Chattergy [Wis78] suggested a relaxation strategy for densely coupled problems with an additive separable objective with respect to \( x_i \). A set of variable copies \( s \) is introduced and constraints \( h_C = s - x = 0 \) force the copies to match the originals. From the KKT conditions and sensitivity information, a relationship for the update of associated dual variables is computed. Coupling between sub-problems is approximated through the required sensitivity of local constraints of sub-problem \( j \) with respect to copy variables \( s_i \) of sub-problem \( i \).

Sobieski [Sob88, Sob90] presented methods for decoupling constraint sets using Global

---

Figure 2.14: Algorithm for generic dual-feasible decomposition method

---

1. Initialize \( k = 0 \), and \((x_k, s_k, \lambda_k, \mu_k)\)
2. Holding \((s_k, \lambda_k, \mu_k)\) constant, solve the \( p \) independent sub-problems with respect to \( x_i \) to obtain \( x_i^* \).
3. Holding \((x_i^*, \lambda_k, \mu_k)\) constant, obtain \( s_k^* \) by solving the feasible master problem for \( s_k \).
4. Holding \((x_i^*, s_k^*)\) constant, obtain \((\lambda_k^*, \mu_k^*)\) by solving the dual master problem for \((\lambda_k, \mu_k)\).
5. If converged stop; otherwise increment \( k \),
   set \((x_k, s_k, \lambda_k, \mu_k) = (x_{k-1}, s_{k-1}, \lambda_{k-1}, \mu_{k-1})\) and return to step 2.
Sensitivity Equations [Sob90]. The method can take advantage of disciplinary dependent optimization strategies when using an aspect partition.

Unger et al. [Ung92] propose the use of variable complexity models to decompose the analysis sequence in a multi-disciplinary design problem. Sub-problems at the higher levels of the hierarchy have fairly superficial models, whereas lower level problems have very detailed models. The amount of model detail increases when moving down the levels of the hierarchy. The method is a model reduction method profiting from variable levels of detail in the iteration sequence.
2.8 Summary

This chapter presented several decomposition strategies. Each decomposition strategy consists of two steps: (1) partitioning the original problem, and formulating a master problem and the associated sub-problems, and (2) coordinating the solution of the problems.

Each strategy is applicable to problems with specific properties. The five defining problem properties are: linking variables, linking functions, additive separability, linearity/convexity, and monotonicity.

Hierarchical methods are applicable to problems with linking variables (feasible methods) or linking functions (dual methods) or a sparse combination of both (dual-feasible methods). Non-hierarchical methods also deal with both linking variables and linking equations, however the coupling of objectives and constraints is dense. Additional defining properties determine what algorithms can be applied, or are required to assure convergence of the method.

Linking variables and linking functions can be identified from a problem’s functional dependence table (FDT), which is a matrix representation of a problem’s structure. Figure 2.16 depicts several compact FDTs and their appropriate decomposition method.

Figure 2.16: Summary of FDT structure for several decomposition methods
Chapter 2. Optimization by decomposition
Chapter 3

Analytical target cascading

Analytical target cascading is a methodology inspired by the design of large systems and follows a natural decomposition of the engineering system. In this chapter, the ATC methodology as defined in [Kim01a, Kim03, Mic03], is reviewed and classified following Chapter 2.

Previous work on ATC did not present a method for partitioning a large optimization problem into a hierarchy of ATC sub-problems. This is one of the reasons that only a few analytical test problems have been formulated and solved using the ATC methodology. [Kim01a, Etm04, Mic04a] used only one example problem to illustrate the ATC methodology. [Hul03] presented in his M.Sc. thesis two additional example problems.

This chapter presents ATC as a method for decomposition-based optimization. Section 3.1 briefly describes the ATC methodology, illustrating the methodology’s background and concepts. In Section 3.2, the ATC optimization sub-problem formulation is presented, from which several characteristics of problems solvable with ATC are derived. Based on these characteristics, a decomposition approach for ATC is presented. Section 3.3 presents the general all-at-once (AAO) formulation of optimization problems for ATC. Section 3.4 describes the relaxation of several of the constraints of the AAO problem. Section 3.5 discusses the partitioning process for ATC, and Section 3.6 presents the ATC sub-problem formulation. Section 3.7 illustrates the partitioning approach by an example problem, and Section 3.8 discusses the solution coordination for ATC. Finally, a discussion on various subjects addressed in this chapter is presented in Section 3.9.

The decomposition method presented in this chapter has been applied to a number of classic nonlinear programming problems from [Hoc81, Sch87, Flo90], which are decomposed and formulated as ATC sub-problems. This new set of ATC test examples is enclosed in Appendix A.
3.1 The ATC methodology

Analytical target cascading (ATC) is a methodology for the design of large engineering systems in the conceptual design phase [Kim01a]. ATC as design methodology was motivated originally by design cases from automotive industry. The ATC design process tries to mimic multidisciplinary vehicle design in practice. In the design of a vehicle, requirements (targets) of the vehicle are assigned at the top level. These targets are propagated throughout the rest of the systems, the sub-systems, and the components of the vehicle (Figure 3.1 illustrates the systems of a vehicle positioned in a hierarchy). For each discipline, a design task is executed. To account for the interactions between discipline, this design task is repetitively conducted until a feasible and consistent design is found.

Figure 3.1: Partial object partition of a vehicle

Figure 3.2: Two approaches to optimal design (adapted from [Kim00])
Two conceptual approaches to this design problem exist: an all-at-once (AAO) approach and a decomposition approach. The AAO approach develops one big optimization model integrating the design models of all disciplines at the various levels. In a decomposition approach, targets are cascaded down level by level and are rebalanced up, based on the lower level designs (see Figure 3.2). The AAO approach requires the solution of a very large design problem, while the decomposition approach requires the sequential solution of a number of smaller problems. The reduction of the single problem size for the decomposed approach must compensate for the additional required coordination analyses.

Analytical target cascading is such a decomposition method. It entails the following steps [Kim01a]: (1) specification of overall targets, (2) propagation of these targets to system, sub-system and component targets, (3) design of the decomposed systems, sub-systems and components to achieve their respective sub-targets, and (4) verification of the final design with respect to the overall targets. Figure 3.3 illustrates these four steps.

![Figure 3.3: The four steps of the ATC process, and the sub-steps of step two](image)

The target setting in the second step is mathematically supported by a multilevel optimization formulation. For this purpose, four sub-steps have to be taken: (2a) appropriate models have to be developed in order to formulate the optimal design problem as an optimization problem, (2b) the optimization problem has to be partitioned, (2c) ATC sub-problems have to be formulated, and (2d) the partitioned problem has to be solved using a coordination strategy. These four sub-steps are briefly discussed below.

**Model development**

The original design problem can be reformulated as a mathematical multilevel optimization problem. For this purpose, appropriate quantitative models for the design objective and constraints are required. With these models, the optimal design problem can be expressed as an optimization problem in the primal form of Eq. (2.1). If a model is not available, it should be developed. Models can for example be analytical,
simulation-based, or finite element-based. Computationally expensive models should be avoided, because of the frequent consultation of the models in target cascading, inherent to ATC’s iterative approach. To reduce computational cost, expensive models can be replaced by cheaper surrogate models. Throughout this report, the availability of appropriate models is assumed.

System partitioning

Target cascading in vehicle design applications [Kim01b, Mic01, Kim02] partitions the original problem into sub-problems associated with vehicle components. Generally, ATC follows an object partition, such as shown in Figure 3.4. The amount of model detail becomes greater for elements at lower levels in the hierarchy. When model functions have been identified, a model-based partition can be obtained. However this model-based partition may not coincide with the human organization of design teams involved in the optimal design process. In that case usually the decomposition based on the human organization is followed.

In Figure 3.4, the object partitioning of a system (left) into a hierarchy of connected optimization sub-problems (right) is illustrated; sub-problems are labelled $P$. The downward flows in the hierarchy represents the propagation of targets throughout the system. The upward flows represent the responses of the disciplinary elements and sub-problems with respect to the received targets. Ideally, these responses are equal to the targets, expressing that the sub-system design can meet its specifications. However, it is possible that targets are unattainable due to local constraints in the sub-problem. In that case, a sub-problem’s response defines the best (locally) feasible solution of the sub-problem.

As explained in Chapter 2, partitioning a problem reveals the problem’s structure. For ATC, a strict hierarchical problem structure is required: elements may only be connected to their parent and their children. In this case, a parent and a child are linked through so called response targets. One form of non-hierarchical coupling is accounted
for by ATC: coupling of elements with the same parent. This non-hierarchical coupling in ATC is reformulated to hierarchical coupling by coordinating the linking between children by their parent (e.g. problem $P_0$ coordinates linking between problems $P_{11}$ and $P_{12}$). For this type of child-child coupling, the children are linked through the parent by linking targets. Other non-hierarchical coupling however is not allowed.

**Sub-problem formulation**

For each disciplinary element in the system, an optimization sub-problem is formulated. Each sub-problem computes targets which are sent to its children and responses for its parent. Each sub-problem tries to match the parent-level targets and child-level responses it receives, while satisfying local design constraints. The sub-problem formulation is discussed in greater detail in Section 3.2.

**Solution coordination**

To solve the decomposed problem, a coordination strategy defines in what order sub-problems are solved and when data is exchanged between sub-problems. Figure 3.4 illustrates the location of and communications between elements. After a successful target cascading process, optimal values for all design variables are specified. In design practice, optimal values for the targets are often used as inputs for more detailed models.

### 3.2 Sub-problem formulation

As mentioned in the previous section, an optimization sub-problem is defined for each element of the system. Each element receives targets from a parent and sends responses back to the parent, based on the received targets. Also, each element computes targets for its children and receives responses of the children to these targets.

Figure 3.5 depicts the flows in and out of an ATC sub-problem. Response target variables are labelled $R$ and $r$, and linking target variables are labelled $Y$ and $y$. Downstream targets are indicated by capital symbols $R$ and $Y$, whereas upstream responses are indicated by lowercase symbols $r$ and $y$. Each symbol has two indices. The first index is used to indicate the level at which each target or response is computed, and the second denotes the sub-problem index to which targets are sent or responses are received from. The sub-problem $P_{ij}$ of element $e_{ij}$ at level $i$, receives targets $R_{(i-1)j}$ and $Y_{(i-1)j}$ computed at the parent level $i-1$. Problem $P_{ij}$ also receives responses $r_{(i+1)k_1}, y_{(i+1)k_1}, \ldots, r_{(i+1)k_{c_{ij}}}, y_{(i+1)k_{c_{ij}}}$ from its children $P_{(i+1)k_1}, \ldots, P_{(i+1)k_{c_{ij}}}$, where $c_{ij}$ denotes the number of children of sub-problem $P_{ij}$. Furthermore, $P_{ij}$ computes responses $r_{ij}$ and $y_{ij}$ for its parent, and targets $R_{ik_1}, Y_{ik_1}, \ldots, R_{ik_{c_{ij}}}, Y_{ik_{c_{ij}}}$ for its children.
Define a set of local design variables \( x_{ij} \) associated exclusively to element \( e_{ij} \). The local objective of the element is denoted by \( f_{ij} \), and vectors \( g_{ij} \) and \( h_{ij} \) are used to denote respectively the local inequality constraints and the local equality constraints associated exclusively to element \( e_{ij} \). Assumed is that each element \( e_{ij} \) computes its target responses \( r_{ij} \) through a set of response analysis functions \( a_{ij} \). These analysis functions often are considered to be "black-box" functions. Furthermore, the objective \( f_{ij} \), constraints \( g_{ij} \) and \( h_{ij} \), and analysis functions \( a_{ij} \) do not depend on received targets or responses, but only on local variables and computed targets and responses. The elementary sub-problem \( P_{ij} \) is formulated as:

\[
\begin{align*}
\text{minimize} & \quad f_{ij}(x_{ij}, y_{ij}, r_{ik_1}, \ldots, r_{ik_{c_{ij}}}) + \\
& \quad \|w^R_{i-1}(r_{(i-1)j} - r_{ij})\|^2 + \|w^Y_{ij}(y_{(i-1)j} - y_{ij})\|^2 + \\
& \quad \sum_{k \in \mathcal{C}_{ij}} \|w^R_{ik}(r_{ik} - r_{(i+1)k})\|^2 + \\
& \quad \sum_{k \in \mathcal{C}_{ij}} \|w^Y_{ik}(y_{ik} - y_{(i+1)k})\|^2,
\end{align*}
\]

with respect to \( x_{ij}, r_{ij}, y_{ij}, r_{ik_1}, Y_{ik_1}, \ldots, r_{ik_{c_{ij}}}, Y_{ik_{c_{ij}}} \) \( (3.1) \)

subject to

\[
\begin{align*}
& g_{ij}(x_{ij}, y_{ij}, r_{ik_1}, \ldots, r_{ik_{c_{ij}}}) \leq 0, \\
& h_{ij}(x_{ij}, y_{ij}, r_{ik_1}, \ldots, r_{ik_{c_{ij}}}) = 0, \\
& r_{ij} - a_{ij}(x_{ij}, y_{ij}, r_{ik_1}, \ldots, r_{ik_{c_{ij}}}) = 0, \\
& Y_{ik_1} - Y_{ik} = 0, \\
& \{k \in \mathcal{C}_{ij} | k \neq k_1 \}
\end{align*}
\]

where operator \( \circ \) denotes an element by element multiplication of two vectors (e.g.
3.2. Sub-problem formulation

\[ \mathbf{x} \circ \mathbf{y} = [x_k]_{k \times 1} \circ [y_k]_{k \times 1} = [x_k y_k]_{k \times 1} \]. Furthermore (to summarize):

- \( \mathbf{x}_{ij} \) is the vector of local variables, exclusively associated with the element,
- \( \mathbf{r}_{ij} \) is the vector of target responses of the element,
- \( \mathbf{y}_{ij} \) is the vector of linking responses of the element,
- \( \mathbf{R}_{ik} \) is the vector of response targets for the \( k \)-th child of the element,
- \( \mathbf{Y}_{ik} \) is the vector of linking targets for the \( k \)-th child of the element,
- \( \mathbf{g}_{ij} \) is the vector of inequality constraints of the element,
- \( \mathbf{h}_{ij} \) is the vector of equality constraints of the element,
- \( \mathbf{R}_{(i-1)j} \) is the vector of parent level response targets received from element \( e_{ij} \)’s parent,
- \( \mathbf{Y}_{(i-1)j} \) is the vector of parent level linking targets received from element \( e_{ij} \)’s parent,
- \( C_{ij} = \{k_1, \ldots, k_{c_{ij}}\} \) is the set of children of the element, with \( c_{ij} \) the number of child elements,
- \( \mathbf{r}_{(i+1)k} \) is the vector of target responses received from the \( k \)-th child of element \( e_{ij} \),
- \( \mathbf{y}_{(i+1)k} \) is the vector of linking responses received from the \( k \)-th child of element \( e_{ij} \),
- \( \mathbf{w}_R^{ij} \) and \( \mathbf{w}_Y^{ij} \) are weighting vectors associated with respectively response and linking targets of element \( e_{ij} \) and its parent,
- \( \mathbf{w}_R^{(i+1)k} \) and \( \mathbf{w}_Y^{(i+1)k} \) are weighting vectors associated with respectively response and linking targets of element \( e_{ij} \) and its \( k \)-th child.

This notation is slightly different from earlier presented work on ATC. Some of these differences are introduced for the elegance of notation in the derivation of the sensitivity-based coordination strategies in Chapter 4. Appendix C discusses the notational differences in greater detail.

Kokkolaras et al. [Kok02] proposed a slightly different formulation for the design of product families. Their formulation allows multiple parents per element, which is a form of non-hierarchical coupling.

The objective of sub-problem \( P_{ij} \) is composite. It is the sum of both a local objective \( f_{ij} \) and a number of terms penalizing inconsistencies (the terms with \( \| \cdot \| \)), with inconsistencies defined as the difference between targets and their associated responses. In the composite objective, weights \( \mathbf{w} \) are used as scaling parameters associated with
Chapter 3. Analytical target cascading

the inconsistencies. When weights are chosen large, the objective tries to minimize inconsistencies before even regarding the local objective $f_{ij}$. For smaller weights however, the composite objective aims at finding a balance between minimizing inconsistencies and minimizing the local objective.

Two types of constraints can be identified. Local constraints $g_{ij}$, $h_{ij}$, and $r_{ij} - a_{ij}(x_{ij}, y_{ij}, R_{ik_1}, \ldots, R_{ik_{c_{ij}}}) = 0$, and consistency constraints $Y_{ik_1} - Y_{ik} = 0$. The consistency constraints make sure that all children receive identical linking targets. In a sub-problem, the local constraints are always satisfied, and therefore local design solutions are always feasible. A shorthand form of the ATC optimization sub-problem (3.1) is:

$$
\begin{align*}
\min_{\mathbf{x}_{ij}} & \quad f_{ij}(\mathbf{x}_{ij}) + \\
& \quad \|w^{R}_{ij} \circ (R_{(i-1)j} - r_{ij})\|_2^2 + \|w^{Y}_{ij} \circ (y_{(i-1)j} - y_{ij})\|_2^2 + \\
& \quad \sum_{k \in C_{ij}} \|w^{R}_{(i+1)k} \circ (R_{ik} - r_{(i+1)k})\|_2^2 + \\
& \quad \sum_{k \in C_{ij}} \|w^{Y}_{(i+1)k} \circ (Y_{ik} - y_{(i+1)k})\|_2^2, \\
\text{subject to} & \quad g_{ij}(\mathbf{x}_{ij}) \leq 0, \\
& \quad h_{ij}(\mathbf{x}_{ij}) = 0,
\end{align*}
$$

with $\mathbf{x}_{ij} = [x_{ij}, y_{ij}, R_{ik_1}, Y_{ik_1}, \ldots, R_{ik_{c_{ij}}}, Y_{ik_{c_{ij}}}]$, the vector of all optimization variables of sub-problem $P_{ij}$. Furthermore, response analysis function constraints $r_{ij} - a_{ij}(\mathbf{x}_{ij}) = 0$, and consistency constraints $Y_{ik_1} - Y_{ik} = 0$ are added to the set of local equality constraints $h_{ij}(\mathbf{x}_{ij}) := [h_{ij}(\mathbf{x}_{ij}), r_{ij} - a_{ij}(\mathbf{x}_{ij}), Y_{ik_1} - Y_{ik_{c_{ij}}}, \ldots, Y_{ik_1} - Y_{ik_{c_{ij}}}].$

A similar compact notation is used in the papers of Kim, Papalambros, and co-workers, as the starting point for the ATC decomposition method. However, the process of decomposing an all-at-once problem into a set of ATC sub-problems has not been described. For this reason, only a small number of analytical test problems is available. The next sections show how an original all-at-once (AAO) optimization problem can be manipulated and decomposed into ATC form as given by Eq. (3.2).

3.3 Original AAO problem

This section discusses several aspects of the AAO optimization problem formulation. First, general properties of the AAO problem required for ATC are derived from Eq. (3.1), after which the identification of structure in AAO problems is discussed. The manipulations of the problem structure required for decomposition are also presented.
3.3. Original AAO problem

AAO problem characteristics for ATC

Similar to the decomposed optimization methods presented in Chapter 2, ATC is applicable to AAO problems with specific characteristics. Recall the five problem characteristics relevant for decomposition methods: (1) linking variables, (2) linking constraints, (3) additive separability, (4) linearity and convexity, and (5) monotonicity. The required characteristics for ATC can be derived from the individual sub-problem statement of Eq. (3.1).

Clearly, the ATC decomposition method deals with linking variables, which for ATC are targets and responses. The sub-problem may also have a set of local variables $x_{ij}$ associated exclusively to the element in the hierarchy. Only local constraints are allowed for ATC, since each sub-problem has its individual set of constraints $g_{ij}$ and $h_{ij}$. The objective of an ATC sub-problem is composite, one term is a local objective $f_{ij}$ depending on both local variables and targets and responses, whereas the additional penalty terms introduced by relaxation of the problem only depend on targets and responses. Before relaxation, the penalty terms are not included in the objective. The objective $f$ of the original AAO problem before relaxation can be defined as the sum of all local objectives $f_{ij}$. Therefore, the objective of the AAO problem needs to be additively separable for ATC. Furthermore, ATC has a convergence proof under convexity assumptions [Mic03]. Monotonicity or linearity however are not required for ATC.

To summarize, the characteristics of the original AAO problem for ATC are:

1. Variables link sub-problems
2. Constraints are only local
3. The objective is partially separable
4. Convexity of functions is required for the convergence proof
5. Monotonicity is not required

When comparing these characteristics of ATC to classic decomposition methods as presented in Chapter 2, properties 1 and 2 imply that ATC is a feasible separable decomposition method. Similar to feasible separable methods, ATC has linking variables, and only local constraints. From the discussed feasible methods, ATC is similar to the engineering-based decomposition methods. ATC is a MDO method as well, however convergence proof exists for ATC. Several of the engineering-based MDO approaches do not have a convergence proof, or even have analytical and numerical problems for convergence [Ale02].

The possibility to prove convergence makes ATC favorable over the other feasible decomposition methods, which do not have convergence proof. Many other feasible separable
methods have been developed during the last decades. It has however never been investigated whether ATC outperforms these methods from an analytical and computational point of view. A comparison study for several methods may provide more insights in the performance of ATC with respect to alternative methods.

AAO problem structure identification

From the defining characteristics for ATC, it is clear that the AAO problems have an underlying structure. The AAO problems have to be partitioned into a number of subproblems, connected through a set of linking variables. Automated model-based algorithms are available for partitioning an optimal design problem [Wag93, Kri97, Mic97]. For ATC an object-based or aspect-based partition may also be used to define connected sub-problems. For the example problems presented in this report, an arbitrary partition is selected after rearranging the rows and columns of the FDT of the original AAO problem. Section A.1 discusses the identification of a multilevel ATC hierarchy from the rearranged FDT. For now, assume the availability of a partition of all original variables into a number of sets of local variables, and one set of shared variables. Furthermore assume that constraints and objectives are also partitioned into a number of local constraint sets and objectives.

AAO problem statement

From the characteristic properties for ATC, a general problem formulation can be constructed (3.3). In this statement, the underlying problem structure is assumed typical for ATC. Vector $\mathbf{s}$ contains all shared variables $s$ (all response and linking targets), while vectors $\mathbf{x}_{ij}$ contain the local variables of element $e_{ij}$. Functions $f_{ij}(\mathbf{s}, \mathbf{x}_{ij})$ represents terms of the additive separable objective. The constraint subsets $g_{ij}(\mathbf{s}, \mathbf{x}_{ij})$ and $h_{ij}(\mathbf{s}, \mathbf{x}_{ij})$ contain all local constraints depending on the shared and one set of local variables. The set of equality constraints $h_{ij}$ also contains the response analysis function constraints $R_{(i-1)j} - a_{ij}(\mathbf{s}, \mathbf{x}_{ij})$ defining the response $a_{ij}(\mathbf{s}, \mathbf{x}_{ij})$ to a target $R_{(i-1)j}$. Note that the response $R_{(i-1)j}$ is a subset of shared variables $s$. Figure 3.6 displays the compact FDT for a feasible separable problem of $N + 1$ levels. The formulation of an $N + 1$ level optimization AAO problem for ATC is given by:

$$\begin{align*}
\min_{s, \mathbf{x}_0, \mathbf{x}_{11}, \ldots, \mathbf{x}_{ij}} & \quad \sum_{i=0}^{N} \sum_{j \in \mathcal{E}_i} f_{ij}(\mathbf{s}, \mathbf{x}_{ij}), \\
\text{subject to} & \quad g_{ij}(\mathbf{s}, \mathbf{x}_{ij}) \leq 0, \\
& \quad h_{ij}(\mathbf{s}, \mathbf{x}_{ij}) = 0, \\
& \quad j \in \mathcal{E}_i, \quad i = 0, 1, \ldots, N, \\
\end{align*}$$

(3.3)
where $\mathcal{E}_i$ denotes the set of elements at level $i$. Levels indices range from $i = 0, \ldots, N$, and element indices range form $j = 1, \ldots, p_i$, where parameter $p_i$ denotes the number of elements at level $i$. Top-level variables $\mathbf{x}_0$ and functions $f_0$, $h_0$, and $g_0$ should therefore formally have the sub-script 01, but because there is only one top-level problem for ATC, the index is abbreviated by 0.

<table>
<thead>
<tr>
<th>$f_0$</th>
<th>$f_{11}$</th>
<th>$\vdots$</th>
<th>$f_{N_{PN}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathbf{s}$</td>
<td>$\mathbf{x}_0$</td>
<td>$\mathbf{x}_{11}$</td>
<td>$\mathbf{x}<em>{N</em>{PN}}$</td>
</tr>
</tbody>
</table>

![Figure 3.6: General FDT schematic for ATC (Eq. (3.3))](image)

Elements $e_{ij}$ within any problem hierarchy are linked through shared variables $\mathbf{s}$. ATC distinguishes between two types of shared variables: response targets and linking targets. Response targets are variables shared by elements at two succeeding levels of the problem hierarchy, and linking targets are variables shared by elements at the same level, under the restriction that elements have the same parent. An example problem hierarchy for ATC is displayed in Figure 3.7. Response targets are indicated with solid lines (e.g. between elements $e_{0}$ and $e_{11}$), while linking targets are represented by dashed lines (e.g. between elements $e_{11}$ and $e_{12}$).

![Figure 3.7: Example hierarchy with location of response targets (solid lines) and linking targets (dashed lines)](image)
The analytical target cascading methodology does not allow coupling between elements at non-consecutive levels (e.g. between elements $e_0$ and $e_{21}$) or between elements at the same level without a mutual parent (e.g. between elements $e_{21}$ and $e_{23}$). Reformulation of the AAO problem can account for both cases to fit the ATC hierarchy. In the first case, linking between non-consecutive levels, the introduction of variable copies at the intermediate level(s) manipulates non-allowed coupling. Consider e.g. the introduction of a variable copy at the intermediate level element $e_{11}$ for linking between $e_0$ and $e_{21}$. In the second case, communication between elements at the same level with different parents, linking can be coordinated by a mutual ancestor instead of a parent (e.g. $e_0$ coordinates linking targets between $e_{21}$ and $e_{23}$ by introducing variable copies at the intermediary elements, $e_{11}$ and $e_{12}$).

AAO problem modification

To prepare for the decomposition of the original AAO problem, copies of the shared variables $s$ are introduced. In the original problem, response targets $R_{(i-1)j}$ are shared by both the parent and the child. Child level copies $r_{ij}$ of these response targets are introduced, and additional equality constraints force the copies to match: $R_{(i-1)j} = r_{ij}$. For linking between elements at a single level, $c_{ij}$ copies of this linking variable are introduced at the parent level: $Y_{(i-1)k_1}, \ldots, Y_{(i-1)k_{c_{ij}}}$. All of these copies are forced equal through a set of consistency constraints: $Y_{(i-1)k_1} = Y_{(i-1)k_2} = \ldots = Y_{(i-1)k_{c_{ij}}}$. At the child level, $c_{ij}$ copies are introduced as well: $y_{ik_1}, \ldots, y_{ik_{c_{ij}}}$. Additionally, each combination of child $k$ copies and parent level copies are forced to match through: $Y_{(i-1)k} = y_{ik}$.

Figure 3.8 depicts the result of introducing copies of shared variables and the associated constraints for an example problem. The figure on the left represents the structure of the AAO problem without additional copies, and the figure on the right shows the problem structure after introduction of copies and constraints.

Originally, the sub-problems depended on local variables $x_{ij}$, and shared variables $s$. As a result of the introduction of the copies, each sub-problem depends on local variables $x_{ij}$ and a number of shared variables, or copies of these variables, associated exclusively with that element. In general, the additional variables for element $e_{ij}$ are: target and linking response variables $r_{ij}$, and $y_{ij}$ for coupling with its parent, and response and linking target variables $R_{ik_1}, \ldots, R_{ik_{c_{ij}}}$ and $Y_{ik_1}, \ldots, Y_{ik_{c_{ij}}}$ for linking with its children. After the modification, the sub-problems are no longer coupled through a set of shared variables, but through a set of coupling constraints.
3.3. Original AAO problem

The manipulated AAO problem statement after introduction of the variable copies now becomes:

\[
\begin{align*}
\text{minimize} \\
\sum_{i=0}^{N} \sum_{j \in E} f_{ij}(x_{ij}, y_{ij}, R_{ik1}, \ldots, R_{ikc_{ij}}), \\
\text{with respect to} \\
x_{ij}, & \quad j \in \mathcal{E}, i = 0, 1, \ldots, N \\
y_{ij}, & \quad j \in \mathcal{E}, i = 0, 1, \ldots, N \\
R_{ik1}, \ldots, R_{ikc_{ij}}, Y_{ik1}, \ldots, Y_{ikc_{ij}}, & \quad j \in \mathcal{E}, i = 1, 2, \ldots, N - 1
\end{align*}
\]

subject to

\[
\begin{align*}
g_{ij}(x_{ij}, y_{ij}, R_{ik1}, \ldots, R_{ikc_{ij}}) & \leq 0, & j \in \mathcal{E}, i = 0, 1, \ldots, N \\
h_{ij}(x_{ij}, y_{ij}, R_{ik1}, \ldots, R_{ikc_{ij}}) & = 0, & j \in \mathcal{E}, i = 0, 1, \ldots, N \\
R_{i(-1)j} - r_{ij} & = 0, & j \in \mathcal{E}, i = 1, 2, \ldots, N \\
Y_{i(-1)j} - y_{ij} & = 0, & j \in \mathcal{E}, i = 1, 2, \ldots, N \\
Y_{ik1} - Y_{ik} & = 0, & \{k \in C_{ij} | k \neq k_1\}, j \in \mathcal{E}, i = 0, 1, \ldots, N - 1
\end{align*}
\]

Note that the response analysis model constraints \(r_{ij} - a_{ij}(x_{ij}, y_{ij}, R_{ik1}, \ldots, R_{ikc_{ij}}) = 0\) have been included in the set of local constraints \(h_{ij}\). A shorthand form of the modified optimization problem (3.4) is:
\[
\min_{\bar{x}_0, \bar{x}_{i1}, \ldots, \bar{x}_{NPN}} \sum_{i=0}^{N} \sum_{j \in \mathcal{E}_i} f_{ij}(\bar{x}_{ij}),
\]
subject to
\[
\begin{align*}
    g_{ij}(\bar{x}_{ij}) &\leq 0, & j &\in \mathcal{E}_i, i = 0, \ldots, N \\
    h_{ij}(\bar{x}_{ij}) & = 0, & j &\in \mathcal{E}_i, i = 0, \ldots, N \\
    h_C(\bar{x}_0, \bar{x}_{i1}, \ldots, \bar{x}_{NPN}) & = 0,
\end{align*}
\]

with the elementary optimization variables given by \( \bar{x}_{ij} = [x_{ij}, r_{ij}, y_{ij}, R_{ik1}, Y_{ik1}, \ldots, R_{ikcij}, Y_{ikcij}] \). Furthermore, consistency constraints forcing the parent level linking variable copies of element \( e_{ij} \) to match \( Y_{ik1} - Y_{ik} = 0 \) are added to the set of local constraints of that element \( h_{ij} \). The constraints \( R_{(i-1)j} - r_{ij} = 0 \) and \( Y_{(i-1)j} - y_{ij} = 0 \) are grouped into a set of coupling constraints \( h_C \).

The compact FDT for shorthand problem (3.5) is depicted in Figure 3.9. Recall that the compact FDT of the original AAO problem (3.3) was coupled through shared variables \( s \). The modified AAO problem (3.5) is coupled through coupling constraints \( h_C \) instead. By the introduction of copies of shared variables, sub-problems can be separated with respect to the optimization variables \( \bar{x}_{ij} \). The coupling is represented by coupling constraints \( h_C \). These constraints are often called the consistency constraints, since the values of these constraints equal the inconsistencies between the original variable and its associated copy.

The reformulation of AAO problems for ATC resembles goal coordination methods (see Section 2.4), however there are some differences. A first difference is that the ATC methodology requires response analysis functions to define a sub-problem’s response. Response functions that link two elements have to be equality constraints in the original AAO problem (3.3); if elements are linked through inequality constraints, the connection would be lost when the constraint becomes inactive. Goal coordination however
allows for coupling through inequality constraints, making it more general than ATC. Secondly, optimization problems for ATC are restricted to the ATC hierarchy, illustrated in Figure 3.7. Introducing variable copies and additional constraints to manipulate a problem’s hierarchy to fit the ATC structure increases the problem’s dimensionality, which often increases the computational effort required for solving the problem.

As another alternative to ATC, the manipulated problem (3.5) can be solved with dual methods, developed to decompose problems with linking constraints. A comparison study of the analytical and numerical properties of ATC and dual methods may provide improvement of the ATC method or perhaps provide a more successful approach to solving problems of the form of Eq. (3.5).

### 3.4 Relaxation of coupling constraints of AAO problem

The modified AAO problem (3.5) is still linked through the coupling constraints. It is possible to separate the modified AAO problem directly. However, the coupling constraints force responses of a sub-problem to match the targets exactly. When targets are unattainable with respect to a sub-problem’s feasible domain, the optimization sub-problem would not have a feasible solution which for obvious reasons is not desired.

To separate the modified AAO problem of ATC, a penalty relaxation of the coupling constraints is suggested [Kim01a]. With this relaxation, optimization sub-problems always have a feasible solution. An external penalty function \( \pi = \| h_C \| \) is added to the objective, with \( \| \cdot \| \) a vector norm and \( h_C \) the set of coupling constraints. The penalty terms allow for non-zero values of the coupling constraints, i.e. deviations between originally coupled variables is allowed. The relaxed AAO problem is defined as:

\[
\min_{\bar{x}_0, \bar{x}_{i1}, \ldots, \bar{x}_{iN}} \sum_{i=0}^{N} \sum_{j \in E_i} f_{ij}(\bar{x}_{ij}) + \| h_C \|,
\]

subject to

\[
\begin{align*}
g_{ij}(\bar{x}_{ij}) & \leq 0, & j \in E_i, i = 0, 1, \ldots, N \\
h_{ij}(\bar{x}_{ij}) & = 0, & j \in E_i, i = 0, 1, \ldots, N
\end{align*}
\]

Michalek and Papalambros [Mic04a] proposed a weighted squared \( l_2 \) norm. Weights are used as scaling factors to indicate a relative importance of each of the coupling constraints. As mentioned before, the coupling constraint values are equal to the inconsistencies of a variable and its copy. By increasing the weights, the optimization problem puts a larger emphasis on minimizing the inconsistencies.

According to [Mic04b], the squared \( l_2 \) norm is not a true norm, however, the direct
use of the $l_1$, $l_2$, and $l_\infty$ norms results in derivative discontinuities and numerical difficulties. They also present general requirements of the penalty function, which should be fully continuous and differential over its domain, and monotonically increasing in all directions away from $h_C = 0$. The squared $l_2$ norm satisfies all these requirements. Additionally, the $l_2$ norm penalty function is separable with respect to each individual target and response. Therefore, the penalty function using a weighted $l_2$ norm can be written as:

$$\pi = \|w_hC\|_2^2 = \sum_{i=1}^{N} \sum_{j \in E_i} \|w_{ij}^R \circ (R_{(i-1)j} - r_{ij})\|_2^2 + \sum_{i=1}^{N} \sum_{j \in E_i} \|w_{ij}^Y \circ (Y_{(i-1)j} - y_{ij})\|_2^2, \quad (3.7)$$

with $w_{ij}^R$ the weighting vectors associated with response targets of sub-problem $ij$, $w_{ij}^Y$ the set of weighting vectors associated with linking targets of sub-problem $ij$, and $w = [w_{11}^R, w_{11}^Y, \ldots, w_{N_{p,p}}^R, w_{N_{p,p}}^Y]$ denoting the vector of all weights.

With the penalty function as defined in (3.7), the relaxed AAO optimization problem for ATC can be formulated as:

$$\min \bar{x} \quad \sum_{i=0}^{N} \sum_{j \in E_i} f_{ij}(\bar{x}_{ij}) + \sum_{i=1}^{N} \sum_{j \in E_i} \|w_{ij}^R \circ (R_{(i-1)j} - r_{ij})\|_2^2 + \sum_{i=1}^{N} \sum_{j \in E_i} \|w_{ij}^Y \circ (Y_{(i-1)j} - y_{ij})\|_2^2,$$  

subject to

$$g_{ij}(\bar{x}_{ij}) \leq 0 \quad j \in E_i, i = 0, 1, \ldots, N$$

$$h_{ij}(\bar{x}_{ij}) = 0 \quad j \in E_i, i = 0, 1, \ldots, N$$

where $\bar{x} = [\bar{x}_0, \bar{x}_{11}, \ldots, \bar{x}_{N_{p,p}}]$. Variables $\bar{x}$ are the *primal* variables of the relaxed optimization problem of Eq. (3.8), and weights $w$ are the *dual* variables (see Section 2.1).

Figure 3.10 illustrates the relaxation of the problem, and its effect on the problem structure. The left figure shows the modified problem statement of (3.5), where elements are coupled through consistency constraints, and the figure on the right shows the structure of the relaxed problem (3.8). Sub-problems in the relaxed problem formulation are only coupled through the penalty function $\pi$. Note that the penalty function is the sum of terms, each term associated with one coupling constraint. Each of the coupling constraints in Figure 3.10(a) is relaxed, and each coupling constraint link can be replaced by a penalty term. The sum of all these terms form the penalty function
3.4. Relaxation of coupling constraints of AAO problem

π depends on all shared optimization variables. The linking between the bottom level problems in Figure 3.10(b) is introduced by the summation of terms of the penalty function. If individual terms of the penalty function were displayed in Figure 3.10(b), the bottom level problems would not be linked.

Figure 3.10: Relaxation of the modified ATC problem

Impact of relaxation

The solution to the relaxed problem is not equal to the solution of the original problem, due to the allowed non-zero inconsistencies. Michalek and Papalambros showed that the solutions cannot be equal for finite weights [Mic04a]. For increasing weights, the solution of the relaxed problem becomes more accurate, motivating to simply use large weights. Numerical behavior of optimization algorithms however puts an upper bound on the size of the weights: for increasing weights, the relaxed problem becomes harder to solve. If weights are chosen too large, the optimization algorithm finds optimal solutions different from the original solution, because it only puts effort into minimizing deviations, neglecting the original objective function. Hulshof et al. observed this behavior for the ATC decomposition as well. They concluded and illustrated that optimal weights exist for which the error between the solution of ATC and the original problem is minimal [Tze03, Hul03]. Chapter 5 elaborates further on the parameters that determine the optimal weights.
3.5 Variable allocation

Partitioning primal variables $\bar{x}$ into a set of shared variables $s$ containing all targets and responses $s = [R_{01}, Y_{01}, \ldots, R_{(N-1)p_N}, Y_{(N-1)p_N}, r_{11}, y_{11}, \ldots, r_{Np_N}, y_{Np_N}]$, and a set of local variables $x = [x_0, x_{11}, \ldots, x_{Np_N}]$, results in the FDT of the relaxed problem depicted in Figure 3.11. The FDT illustrates that for fixed dual variables $w$ and shared variables $s$, the relaxed problem is fully separable.

Figure 3.11 shows that the relaxed ATC formulation of Eq. (3.8) is coupled through both shared and dual variables by the penalty function. Note that the vector of shared variables $s$ is different from the one used in Eq. (3.3) and Figure 3.6. In Figure 3.11 it contains all original shared variables and their introduced copies. As illustrated in the previous chapter, a dual-feasible decomposition method accounts for linking through primal and dual variables. A dual decomposition is used to account for the linking penalty function, and the dual sub-problem is separated further into a hierarchy of coupled sub-problems to account for linking variables.

![Figure 3.11: Compact FDT schematic for relaxed problem of ATC (Eq. (3.8))](image)

Analogous to dual-feasible methods, a dual master problem for ATC formulated in dual variables $w$ can be defined while treating primal variables $\bar{x}$ as parameters. For the feasible decomposition part of ATC, a hierarchy of sub-problems is formulated. The primal variables $\bar{x}$ are partitioned into multiple sets of sub-problem variables $[\bar{x}_0, \bar{x}_{11}, \ldots, \bar{x}_{Np_N}]$. For this partition, each sub-problem is solved for its set of local variables and its set of shared variable copies, while fixing the variables of all other sub-problems. A problem hierarchy is formed that iteratively solves sub-problems after which targets and responses computed in the sub-problems are updated in the other sub-problems.

The hierarchical dual-feasible approach for ATC is illustrated in Figure 3.12, where
the information exchanges within the dual-feasible decomposition are depicted. In this figure, downstream targets computed at level $i$ are labelled $T_i^*$, and upstream responses computed at level $i$ are labelled $t_i^*$. Furthermore, the vector $x^* = [x_0^*, x_1^*, \ldots, x_{N_p}^*]$ is used to denote the optimal solutions of all local variables.

![Diagram of dual-feasible decomposition for ATC](image)

Figure 3.12: Hierarchical dual-feasible decomposition for ATC

The feasible master problem for ATC differs from alternative master problems for feasible decomposition strategies, as presented in Section 2.5. Classic feasible master problems are usually defined only in shared variables $s$. For the hierarchical ATC decomposition method, there is not one master problem since all ATC sub-problems coordinate part of the linking. For classic methods however, the top-level ATC problem is merely another sub-problem, and a master problem is constructed artificially. Because the ATC problem uses part of the original problem as a master problem, convergence can be proven, in contrast to most of the classic feasible methods.

The original ATC approach proposed in [Kim01a] only consisted of the feasible part of the decomposition method. Recent efforts by Michalek and Papalambros also include the dual part [Mic04a]. Results for the dual-feasible approach are promising. The dual master problem, as proposed in [Mic04a], is a weight update mechanism; the updates are computed from deviations between originals primal variables and their associated copies. Possibly, an ATC dual master problem can be formulated in one of the more classic methods presented in [Wag93, Loo89].
3.6 Formulation of ATC sub-problems

Variables have been assigned to different elements of the problem hierarchy, and the relaxed AAO problem can be decomposed into a number of sub-problems. Consider the relaxed optimization problem of Eq. (3.8), with fixed dual variables \( w \).

A sub-problem is defined as the relaxed AAO problem, which is solved for only a subset of variables, instead of all optimization variables. Sub-problem \( P_{ij} \) is only solved for variables allocated to that problem, \( \bar{x}_{ij} \), while all other variables are treated as static variables. Because the problem is only solved for \( \bar{x}_{ij} \), only local constraints \( g_{ij} \) and \( h_{ij} \) are of concern, the remaining constraints do not depend on \( \bar{x}_{ij} \) and are therefore not relevant to sub-problem \( P_{ij} \). Similar to the constraints, only the local objective \( f_{ij} \) that depends on variables \( \bar{x}_{ij} \) has to be included in the sub-problem. The remaining local objectives do not depend on \( \bar{x}_{ij} \) and therefore do not have to be considered in the formulation of sub-problem \( P_{ij} \). Likewise, only terms of the penalty function \( \pi \) that depend on \( \bar{x}_{ij} \) have to be included in the objective of sub-problem \( P_{ij} \), the remaining terms do not depend on \( \bar{x}_{ij} \) and therefore do not contribute to the optimization sub-problem of \( P_{ij} \). By including only the terms relevant to \( P_{ij} \), the generic problem formulation of an ATC sub-problem \( P_{ij} \) is:

\[
\begin{align*}
\min_{\bar{x}_{ij}} & \quad f_{ij}(\bar{x}_{ij}) + \\
& \quad \| w^R_{ij} \odot (R_{(i-1)j} - r_{ij}) \|_2^2 + \| w^Y_{ij} \odot (y_{(i-1)j} - y_{ij}) \|_2^2 + \\
& \quad \sum_{k \in C_{ij}} \| w^R_{ik} \odot (R_{ik} - r_{(i+1)k}) \|_2^2 + \\
& \quad \sum_{k \in C_{ij}} \| w^Y_{ik} \odot (Y_{ik} - y_{(i+1)k}) \|_2^2, \\
\text{subject to} & \quad g_{ij}(\bar{x}_{ij}) \leq 0, \\
& \quad h_{ij}(\bar{x}_{ij}) = 0,
\end{align*}
\]

(3.9)

where (to summarize):

- \( \bar{x}_{ij} \) is the vector of local variables,
- \( r_{ij} \) is the vector of target responses of the element,
- \( y_{ij} \) is the vector of responses of associated with linking between elements at level \( i \) that share the parent of element \( e_{ij} \),
- \( R_{ik} \) is the vector of response targets for the \( k \)-th child of the element,
- \( Y_{ik} \) is the vector of linking targets for the \( k \)-th child of the element,
- \( \bar{x}_{ij} = [x_{ij}, r_{ij}, y_{ij}, R_{ik1}, Y_{ik1}, \ldots, R_{ik_{kij}}, Y_{ik_{kij}}] \) is the vector of optimization variables associated with the element,
3.6. Formulation of ATC sub-problems

- $g_{ij}$ is the vector of local inequality constraints of the element,
- $h_{ij}$ is the vector of local equality constraints of the element, which also includes $r_{ij} = \mathbf{a}_{ij}(x_{ij}), \ Y_{ik} = \mathbf{y}_{ik}, \ \{k \neq k_1|k \in C_{ij}\},$
- $\mathbf{a}_{ij}$ is the vector of response analysis functions of the element,
- $\mathbf{R}_{(i-1)j}$ is the vector of parent level response targets received from element $e_{ij}$’s parent,
- $\mathbf{Y}_{(i-1)j}$ is the vector of parent level linking targets received from element $e_{ij}$’s parent,
- $C_{ij} = \{k_1, \ldots, k_{c_{ij}}\}$ is the set of children of element $e_{ij}$, with $c_{ij}$ the number of child elements,
- $\mathcal{E}_i$ is the set of elements at level $i$,
- $r_{(i+1)k}$ is the vector of target responses of the $k$-th child of element $e_{ij}$,
- $y_{(i+1)k}$ is the vector of linking responses of the $k$-th child of element $e_{ij}$,
- $w^R_{ij}$ and $w^Y_{ij}$ are weighting vectors associated with respectively response and linking targets for element $e_{ij}$ received from its parent,
- $w^R_{(i+1)k}$ and $w^Y_{(i+1)k}$ are weighting vectors associated with respectively response and linking targets for element $e_{ij}$ sent to its $k$-th child.

The final step of decomposition of the AAO problem is depicted in Figure 3.13. The left figure shows the modified AAO problem structure obtained after the introduction of copies and consistency constraints, and the figure on the right depicts the problem structure after relaxation and variable allocation. The sub-problems are separated and the original AAO problem has been decomposed into a number of smaller sub-problems. The sub-problems however still need information from each other. For the solution of one sub-problem, the shared variables solved for by other sub-problems are treated as fixed parameters, and vice versa. After a sub-problem is solved, its optimal solution must be updated in the other sub-problems, where the solution is used as a fixed parameter.

The coordination strategy defines the order in which sub-problems are solved. Various strategies for ATC are discussed in Section 3.8, but first the decomposition approach is illustrated with an example.
Chapter 3. Analytical target cascading

3.7 Problem formulation example

To illustrate the partitioning approach leading to target cascading sub-problem formulations, consider the geometric programming problem of [Kim01a]:

\[
\begin{align*}
\min_{z_1, z_2, \ldots, z_{14}} & \quad z_1^2 + z_2^2 \\
\text{subject to} & \quad g_1 : z_3^2 + z_4^2 - z_5^2 \leq 0, \\
& \quad g_2 : z_6^2 + z_7^2 - z_8^2 \leq 0, \\
& \quad g_3 : z_9^2 + z_{10}^2 - z_{11}^2 \leq 0, \\
& \quad g_4 : z_2^2 + z_{12}^2 - z_{13}^2 \leq 0, \\
& \quad g_5 : z_{11}^2 + z_{12}^2 - z_{13}^2 \leq 0, \\
& \quad g_6 : z_{11}^2 + z_{12}^2 - z_{14}^2 \leq 0, \\
& \quad h_1 : z_1^2 - z_3^2 - z_4^2 - z_5^2 = 0, \\
& \quad h_2 : z_2^2 - z_5^2 - z_6^2 - z_7^2 = 0, \\
& \quad h_3 : z_3^2 - z_6^2 - z_9^2 - z_{10}^2 - z_{11}^2 = 0, \\
& \quad h_4 : z_4^2 - z_7^2 - z_{12}^2 - z_{13}^2 - z_{14}^2 = 0, \\
& \quad z_1, z_2, \ldots, z_{14} \geq 0.
\end{align*}
\]  

(3.10)

A three level hierarchy can be identified from the rearranged FDT of the original problem. This three-level partition is not the only possible partition of the problem, but the selected three-level partition serves to illustrate the decomposition approach for ATC. Appendix A presents the partition presented here and an alternative partition.

The FDT containing the associated variable and function partitions is depicted in Figure 3.14(a). Variables are partitioned into five vectors: local vectors \(x_{11} = [z_4]\), \(x_{12} = [z_7]\), \(x_{21} = [z_8, z_9, z_{10}]\), \(x_{22} = [z_{12}, z_{13}, z_{14}]\), and shared variable vector \(s = [z_1, z_2, z_3, z_5, z_6, z_{11}]\). For the top level element \(e_0\), the vector of local optimization vari-

---

Figure 3.13: Decomposition of the relaxed ATC problem
3.7. Problem formulation example

variables is empty. Constraints are partitioned into response analysis functions: $a_{11} : [h_1]$, $a_{12} : [h_2]$, $a_{21} : [h_3]$, and $a_{22} : [h_4]$; and local constraints $g_{11} = [g_1]$, $g_{12} = [g_2]$, $g_{21} = [g_3, g_4]$, and $g_{22} = [g_5, g_6]$. The objective function is allocated to the top-level problem $f_0 = f$.

Figure 3.14: Structure identification of Problem (3.10)

Figure 3.14(b) illustrates the location of elements in the hierarchy, and the allocation of variables, local and shared, as well as functions. Three levels can be distinguished. The top level element $e_0$ shares variable $z_1$ with element $e_{11}$, and $z_2$ with $e_{12}$. At the intermediate level, elements $e_{11}$ and $e_{12}$ share variable $z_5$. Variables $z_3$ and $z_6$ are shared by $e_{11}$ and $e_{21}$, and $e_{12}$ and $e_{22}$, respectively. At the bottom level, $e_{21}$ and $e_{22}$ share variable $z_{11}$.

Because the bottom level elements share variable $z_{11}$, but do not have a mutual parent, the connection at the bottom level must be redirected. By introducing two copies $z_{15}$ and $z_{16}$ and equality constraints $h_5 : z_{11} - z_{15} = 0$ and $h_6 : z_{11} - z_{16} = 0$, linking with variable $z_{11}$ can be redirected to the intermediate level, where linked elements do have a mutual parent. The introduced equality constraints are added to the vector of response analysis functions for the bottom level problems: $a_{21} : [h_3, h_5]$ and $a_{22} : [h_4, h_6]$. The hierarchy of the manipulated problem is depicted in Figure 3.15.
The manipulated primal problem of Eq. (3.10), can be reformulated to the modified ATC formulation of Eq. (3.5) by introducing copies of response variables $R$ and $r$, and linking variables $Y$ and $y$:

$$
\begin{align*}
\min & \quad f_0(\bar{x}_0), \\
\text{subject to} & \quad g_{ij}(\bar{x}_{ij}) \leq 0, \\
& \quad h_{ij}(\bar{x}_{ij}) \leq 0, \\
& \quad R_{(i-1)j} - r_{ij} = 0, \\
& \quad Y_{(i-1)j} - y_{ij} = 0,
\end{align*}
$$

(3.11)

Where:

- $\bar{x}_0 = [], \bar{x}_{11} = [z_4], \bar{x}_{12} = [z_7], \bar{x}_{21} = [z_8, z_9, z_{10}, z_{15}], \bar{x}_{22} = [z_{12}, z_{13}, z_{14}, z_{16}],$
- $\bar{r}_0 = [], \bar{r}_{11} = [z_1], \bar{r}_{12} = [z_2], \bar{r}_{21} = [z_3, z_{11}], \bar{r}_{22} = [z_5, z_{11}],$
- $\bar{y}_0 = \bar{y}_{21} = \bar{y}_{22} = [],$ and $\bar{y}_{11} = \bar{y}_{12} = [z_5, z_{11}],$
- $\bar{R}_{01} = [z_1], \bar{R}_{02} = [z_2], \bar{R}_{11} = [z_3, z_{11}], \bar{R}_{12} = [z_5, z_{11}],$
- $\bar{Y}_{01} = [z_5, z_{11}], \bar{Y}_{02} = [z_5, z_{11}], \bar{Y}_{11} = \bar{Y}_{12} = [],$
- $\bar{x}_{ij} = [x_{ij}, r_{ij}, y_{ij}, R_{ik_1}, Y_{ik_1}, \ldots, R_{ik_{ij}}, Y_{ik_{ij}}],$ e.g. $\bar{x}_{11} = [x_{11}, r_{11}, y_{11}, R_{11}] = [z_4, z_1, z_5, z_{11}, z_6, z_{11}],$
- $f_0(\bar{x}_0) = z_1^2 + z_2^2, f_{11}(\bar{x}_{11}) = f_{12}(\bar{x}_{12}) = f_{21}(\bar{x}_{21}) = f_{22}(\bar{x}_{22}) = 0,$
3.7. Problem formulation example

- \( a_{11}(\bar{x}_{11}) = \sqrt{z_3^2 + z_4^2 + z_5^2}, \) \( a_{12}(\bar{x}_{12}) = \sqrt{z_6^2 + z_7^2}, \)
  \( a_{21}(\bar{x}_{21}) = [\sqrt{z_8^2 + z_9^2 + z_{10}^2 + z_{11}^2 + z_{12}^2}, z_{13}, z_{14}], \)
  \( a_{22}(\bar{x}_{22}) = [\sqrt{z_{15}^2 + z_{16}^2 + z_{17}^2 + z_{18}^2}, z_{19}], \)

- \( g_0 = [], \)
  \( g_{11}(\bar{x}_{11}) = [z_2^2 + z_8^2 - z_0^2], \)
  \( g_{12}(\bar{x}_{12}) = [z_3^2 + z_6^2 - z_1^2], \)
  \( g_{21}(\bar{x}_{21}) = [z_4^2 + z_9^2 - z_{10}^2 + z_{11}^2 - z_{12}^2], \)
  \( g_{22}(\bar{x}_{22}) = [z_{13}^2 + z_{14}^2 + z_{15}^2 + z_{16}^2 + z_{17}^2], \)

- \( h_0 = [Y_{01} - Y_{02}], \)
  \( h_{11}(\bar{x}_{11}) = [r_1 - a_{11}(\bar{x}_{11}), (y_{11})_2 - (R_{11})_2], \)
  \( h_{12}(\bar{x}_{12}) = [r_2 - a_{12}(\bar{x}_{12}), (y_{12})_2 - (R_{12})_2], \)
  \( h_{21}(\bar{x}_{21}) = [r_2 - a_{21}(\bar{x}_{21})], \)
  \( h_{22}(\bar{x}_{22}) = [r_2 - a_{22}(\bar{x}_{22})], \)

- \( \mathcal{E}_j \) is the set of elements at level \( j \).

The selection operator \( \langle \mathbf{a} \rangle_i \) selects element \( i \) from vector \( \mathbf{a} \).

After the introduction of a penalty function for the coupling constraints of linking variables and response targets, the relaxed problem can be separated into a top level optimization problem \( P_0 \), two intermediate level problems \( P_{11} \) and \( P_{12} \), and two bottom level problems \( P_{21} \) and \( P_{22} \). The top level problem \( P_0 \) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_0} \quad & f_0(\bar{x}_0) + \\
& \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 + \|w_{12}^R \circ (R_{02} - r_{12})\|_2^2 + \\
& \|w_{11}^Y \circ (Y_{01} - y_{11})\|_2^2 + \|w_{12}^Y \circ (Y_{02} - y_{12})\|_2^2,
\end{align*}
\] (3.12)

subject to \( h(\bar{x}_0) = 0 \).

The intermediate level problem \( P_{11} \) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_{11}} \quad & \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 + \|w_{11}^Y \circ (Y_{01} - y_{11})\|_2^2 + \\
& \|w_{21}^R \circ (R_{11} - r_{21})\|_2^2,
\end{align*}
\] (3.13)

subject to \( g_{11}(\bar{x}_{11}) \leq 0, \)
\( h_{11}(\bar{x}_{11}) = 0. \)

Intermediate level problem \( P_{12} \) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_{12}} \quad & \|w_{12}^R \circ (R_{02} - r_{12})\|_2^2 + \|w_{12}^Y \circ (Y_{02} - y_{12})\|_2^2 + \\
& \|w_{22}^R \circ (R_{12} - r_{22})\|_2^2,
\end{align*}
\] (3.14)

subject to \( g_{12}(\bar{x}_{12}) \leq 0, \)
\( h_{12}(\bar{x}_{12}) = 0. \)
The bottom level problem $P_{21}$ is formulated as:

$$\min_{\bar{x}_{21}} \| w_{21}^R \circ (R_{11} - r_{21}) \|_2^2,$$
subject to
$$g_{21}(\bar{x}_{21}) \leq 0,$$
$$h_{21}(\bar{x}_{21}) = 0.$$ \hfill (3.15)

The bottom level problem $P_{22}$ is formulated as:

$$\min_{\bar{x}_{22}} \| w_{22}^R \circ (R_{12} - r_{22}) \|_2^2,$$
subject to
$$g_{22}(\bar{x}_{22}) \leq 0,$$
$$h_{22}(\bar{x}_{22}) \leq 0.$$ \hfill (3.16)

### 3.8 Coordination method

The previous sections presented the partitioning process leading to the formulation of ATC sub-problems. The next step of the ATC decomposition method is solving the partitioned problem. So far, the ATC literature defines the coordination strategy as an iterative process that determines the order in which the sub-problems optimizations are carried out. Only the feasible coordination part is discussed. The dual part of the coordination has received very little attention, but is an important part of the coordination strategy. The new coordination method developed in Chapter 4 is a feasible coordination strategy and therefore only available feasible strategies are discussed in this section. However, additional dual strategies may improve the behavior of the ATC solution process.

#### Existing coordination strategies for ATC

Feasible coordination strategies developed for ATC so far focused on defining a solution sequence. Figure 3.16 shows several sequence schemes for a three level hierarchy. Scheme I, II, and III were presented by [Etm02], and scheme IV was introduced by [Hul03]. For all coordination schemes, the convergence of the ATC process is determined after solving the top-level problem.

Scheme I describes a loop-like coordination sequence: starting from the top level, all levels in the hierarchy are solved from high to low. After solving the bottom level problems, the top problem is solved again and convergence is checked. If the process has not converged, the iteration loop is repeated.
3.8. Coordination method

Scheme II prescribes a similar loop-like approach, but instead of jumping to the top-level after solving the bottom level problems, sub-problems at intermediate levels are solved starting at the bottom and climbing up through the hierarchy.

Scheme III prescribes a nested coordination sequence, with one inner loop nested in the outer loop. The scheme is similar to scheme II, but instead of directly jumping up one level in the bottom-up sequence, first convergence of the lowest two levels must be checked, before the sequence is allowed to progress to level 0.

Scheme IV prescribes another nested sequence, and is similar to scheme III. Now the two top-level problems are treated as nested in the outer loop with level 2.

![Diagram of ATC solution sequences for a three level hierarchy](image)

Figure 3.16: ATC solution sequences for a three level hierarchy

Technically, scheme II is a sub-class of schemes III and IV. If the convergence criterion for the inner loop of schemes III and IV is always satisfied, the sequence is always allowed to move to the next level in the hierarchy, resulting in scheme II.

For schemes II through IV, convergence proof is available [Mic03]. For this convergence proof, convexity is assumed and the top-level objective must be of the form $f_0 = ||R - r||_2^2$ and all other local objectives zero, $f_{ij} = 0$. The nested property is the key element for the convergence proof, and therefore convergence for scheme I has not been proven. In numerical examples, schemes II, III and IV provide similar numerical behavior, but scheme I is instable in some cases [Hul03]. With the convergence proof, five different converging schemes for a four-level hierarchy are presented in [Mic03]. These schemes are depicted in Figure 3.17.

The sequences shown in Figure 3.17(a) contain two nested inner iterations. Level 1 can only be solved when both levels 2 and 3 have converged, and level 0 can only be solved when the lower 3 levels have converged. The sequence illustrated in Figure 3.17(b) first solves levels 0 through 2, but only proceeds down to level 3 when levels 1 and 2 have converged. In the bottom up sequence, level 0 may only be solved when levels 1 through 3 have converged. In Figure 3.17(c) the bottom-down sequence may
only proceed to level 2 after levels 0 and 1 have converged. Similarly, the bottom-up sequence may only proceed to level 1 after levels 2 and 3 have converged. The sequence illustrated in Figure 3.17(d) is the mirror sequence of the scheme in Figure 3.17(b); the top-down sequence may only move to level 3 after levels 0 through 2 have converged, and the bottom-up sequence may only progress to level 0 after levels 1 and 2 have converged. Figure 3.17(e) depicts the mirror sequence of Figure 3.17(a): the top down sequence may only progress to level 2 after levels 0 and 1 have converged, the sequence may only proceed to level 3 after all other levels have converged. After solving the bottom level problems, the sequence moves up to level 0 in a bottom-up fashion.

As mentioned above, current coordination strategies of the coordination process for ATC are concerned with defining a solution sequence of the sub-problems. Problems are solved in a predetermined sequence and target and response values are communicated when they become available.

**Coordination in the FDT**

The coordination process can be related to the structure of a problem’s FDT. Consider the example four-level problem with one element at each level (Figure 3.18). Block A and B respectively contain local and coupling functions. Local functions in block A only depend on one subset of variables $\bar{x}_i$, while coupling constraints $h_C$ in block B depend on two subsets of consecutive levels. Sub-problems in block A could be solved in parallel if block B was deleted.

By fixing a subset of variables, sub-problems can be solved for both local and coupling
3.8. Coordination method

functions. Current hierarchical sequences solve the problem level-by-level. Consider e.g. the top-down bottom-up sequence (Scheme II). First level 0 is solved for \( x_0 \) while fixing all other problem variables. By fixing all other variables \([x_1, x_2, x_3]\), constraints in both blocks A and B associated with the first level column can be solved for \( x_0 \). After solving level 0, the next level in the sequence is selected: e.g. level 1. The optimization sub-problems in level 1 are solved for variables \( x_1 \) while fixing the variables \([x_0, x_2, x_3]\) of the other levels. Level 2 is selected and solved for \( x_2 \), etc. The bottom-up sequence is defined using a similar approach.

Alternate coordination sequences

The coordination process is based on solving sub-problems of one level while fixing all variables of sub-problems at connected levels (e.g. solving level 1 while fixing variables associated with levels 0 and 2). Alternatives to the existent level-by-level approaches are possible.

Semi-parallel coordination strategy

For ATC, it is possible to define an iterative sequence that first solves all sub-problems at the even levels for fixed variables of the odd levels, after which all sub-problems at the odd levels are solved for fixed variables of the even levels. This semi-parallel odd-even sequence is motivated from the fact that ATC sub-problems are only connected to sub-problems that are parents or children. Figure 3.19 displays the coordination algorithm for the semi-parallel coordination sequence, and Figure 3.20 illustrates this approach. For two-level problems, the semi-parallel approach is identical to the existing level-by-level approaches.
Chapter 3. Analytical target cascading

1. Set \( k = 0 \), choose initial variables \( x^{0}_{ij} \).

2. Solve all sub-problems at even levels \( P_{ij|i=\text{even}} \) for fixed \( x^{k}_{ij|i=\text{odd}} \), to obtain \( x^{k+1}_{ij|i=\text{even}} \).

3. Solve all sub-problems at odd levels \( P_{ij|i=\text{odd}} \) for fixed \( x^{k}_{ij|i=\text{even}} \), to obtain \( x^{k+1}_{ij|i=\text{odd}} \).

4. Check convergence.

5. If step 3 is successful, stop; otherwise set \( x^{k+1}_{ij|i=\text{even}} = x^{k}_{ij|i=\text{even}} \) and \( x^{k+1}_{ij|i=\text{odd}} = x^{k}_{ij|i=\text{odd}} \), \( k = k + 1 \), and return to step 2.

Figure 3.19: Semi-parallel coordination for ATC

![Figure 3.19: Semi-parallel coordination for ATC](image)

Figure 3.20: Semi-parallel solution sequence for ATC

To illustrate the new sequence, consider the example problem of Figure 3.18. It is possible to solve unconnected levels in parallel by fixing the variables of sub-problems at connected levels. For instance, levels 0 and 2 can be solved in parallel for \( x_0 \) and \( x_2 \) when fixing variables associated with levels 1 and 3, \( x_1 \) and \( x_3 \). As a second step, levels 1 and 3 are solved for \( x_1 \) and \( x_3 \), while fixing \( x_0 \) and \( x_2 \).

All-parallel coordination strategy

A second, hierarchy independent, all-parallel sequence can be defined. In this all-parallel sequence, all sub-problems are solved in parallel. But, instead of fixing variables of connected sub-problems at values computed at the current iterate, variables are fixed at optimal values of the previous iterate. Coupling constraints of block B in the FDT of Figure 3.18 are decoupled completely and all sub-problems can be solved in parallel. After all sub-problems have been solved, the optimal values are updated for each sub-problem. The all-parallel coordination algorithm is presented in Figure 3.21, and the
3.8. Coordination method

sequence is illustrated in Figure 3.22.

1. Set $k = 1$, choose initial variables $\bar{x}_{ij}^0$.
2. Solve all sub-problems $P_{ij}$, while fixing the remaining variables $\bar{x}_{lm|lm \neq ij}^k$ to $\bar{x}_{lm|lm \neq ij}^{k-1}$, to obtain $\bar{x}_{ij}^k$.
3. Check convergence.
4. If step 3 is successful, stop; otherwise set $k = k + 1$, and return to step 2.

When the all-parallel scheme is applied to hierarchical problems, it solves two semi-parallel schemes at once, one of them started at the odd levels, the other on the even levels. To illustrate consider Figure 3.23, where two semi-parallel schemes A and B are depicted as well as an all-parallel scheme. In one iteration scheme A solves all sub-problems of the even levels in sub-iteration (1a), and in sub-iteration (1b), sub-problems at all odd levels are solved. Scheme B follows a mirrored sequence. First sub-problems at all odd levels are solved (1a), after which the sub-problems at all even levels are solved (1b). If both sub-iterations (1a) and (1b) of both schemes A and B were performed in parallel, the all-parallel coordination strategy is obtained. At sub-iteration (1a) of schemes A and B, all sub-problems at both even and odd levels are solved, after which all sub-problems at odd and even levels are solved again for sub-iteration (1b). Similarly for the all-parallel coordination scheme, at iteration (1) all sub-problems at all levels are solved, and for iteration (2) again all sub-problems at all levels are solved.

The mirrored semi-parallel approach, and the all-parallel approach are very similar, however differences between the approaches exist. In a single iteration, the mirrored semi-parallel approach solves every sub-problem twice. For the all-parallel approach,
each sub-problem is solved only once every iteration. Therefore, one iteration of the mirrored semi-parallel approach can be compared to two iterations of the all-parallel approach. Another important difference is that the all-parallel approach does not necessarily require a hierarchic problem structure, in contrast to the hierarchy based semi-parallel approach. The application of the all-parallel approach could provide a step towards a non-hierarchical ATC formulation.

Note that the convergence criterion for the all-parallel approach must be adapted to account for the underlying mirrored semi-parallel behavior for hierarchical problems. The comparison of two consecutive all-parallel iterations should be avoided, since the two underlying semi-parallel strategies are independent of one another. To account for this behavior, convergence should be determined by considering each semi-parallel scheme separately.
3.9 Discussion

The previous sections provided new insights in the ATC methodology. The stepwise approach of decomposing an optimization problem into a multilevel hierarchy of coupled ATC sub-problems provides a basis for discussion and further research.

Hierarchical vs. non-hierarchical ATC

The problem structure for ATC currently must be hierarchical. However, a non-hierarchical version of ATC is possible. The current restriction to hierarchical ATC only holds for the coordination strategy. The proposed all-parallel strategy is hierarchy-independent and allows for non-hierarchical coupling of elements. One of the major benefits of the current hierarchical coordination strategies however is the availability of convergence proof, whereas for non-hierarchical ATC convergence has not been proven.

Inexact external vs. exact penalty functions

ATC currently uses an inexact external penalty relaxation of the coupling constraints. The solution to the relaxed problem for this external penalty function is not equal to the solution to the original problem. The use of an exact penalty formulation does not alter the solution to the problem. The use of exact penalty functions may however have undesired consequences with respect to the numerical behavior of the ATC process.

An augmented Lagrangian penalty function has been suggested for ATC [Pap02]. The ATC decomposition method and the formulation of sub-problems under with an augmented Lagrangian penalty relaxation is presented in Appendix D. The augmented Lagrangian penalty function however is only exact when the penalty parameters are equal to the optimal values of the Lagrange multipliers associated with the coupling constraints relaxed with the penalty function. To determine these optimal values, an iterative process has to be defined. This additional iterative process may experience poor convergence behavior or other undesired numerical difficulties. Whether the decomposition method benefits from the application of an exact penalty relaxation of the coupling constraints however remains to be investigated.

Artificially constructed master problem

The partitioning process for ATC is based on a problem’s hierarchy. Variables and functions are allocated to elements within the hierarchy, and a feasible master problem is defined to determine the solution sequence. However, alternative feasible master problems can be defined by using elements of feasible decomposition methods presented in Chapter 2. The sensitivity-based coordination approach presented in the next chapter is an artificially constructed master problem.
Notation and formulation for ATC

Notations and formulations of ATC sub-problems have changed a number of times after the initial introduction of ATC by Kim [Kim01a]. Alternatives are presented by Michelen et al. [Mic03], Etman et al. [Etm04], and Michalek et al. [Mic04b]. The notation presented in this report differs from those presented previously. The differences here were needed for elegance of notation of the coordination presented in the next chapter.

Objective formulation of original AAO problem

The characteristics of the objective of the original AAO problem formulation of problems decomposable with ATC has not been defined unanimously. Some work on ATC only allowed the objective to depend on top-level variables, and that the objective can be expressed as deviations from overall targets [Kim01a, Mic03, Mic04a]. Later work on ATC allowed a more general additive separable objective, but only when the individual terms are expressible as deviations from 'local' targets only associated with one individual element [Etm04, Kok02, Hul03, Tze03].

Work of [Kim01a] and [Etm02, Etm04] on a geometric optimization problem (presented in Section 3.7) shows another discrepancy: where [Kim01a] uses the original objective \( f \) as local objective for the top-level problem \( f_0 = f \), [Etm04] uses the original objective \( f \) as a response function for the top-level problem. The objective for the top-level [Etm02, Etm04] is defined as the deviation of this response function from an arbitrary target \( T_f \): \( f_0 = \| T_f - f \| \). The latter formulation should be rejected, because it defines an objective for the ATC problem which differs from the objective of the original problem; clearly this is not desired in decomposed optimization: the objective of the decomposed optimization problem must remain identical to the objective of the original problem. As a second disadvantage, the approach of [Etm04] requires the definition of an arbitrary target \( T_f \). The goal of the ATC problem has shifted from minimizing the original objective to finding a value of the objective that minimizes deviations between the objective function value and an arbitrary target \( T_f \). When target \( T_f \) is chosen poorly, it is possible that the objective can be reduced below \( T_f \), but their ATC formulation restricts this. See Appendix B for a paper further elaborating on this discussion. The conclusion of this paper is that the approach introduced by [Kim01b] (copying the original objective) should be preferred over the approach followed by [Etm02, Etm04].

Inclusion of responses \( r_{ij} \) in \( x_{ij} \)

Previous work on ATC suggested two possibilities for computing response variables \( r_{ij} \). Responses can be computed by including them in the vector of sub-problem optimization variables, or they can be computed with an embedded definition \( r_{ij} = a_{ij}(x_{ij}) \). By including responses \( r_{ij} \) in \( x_{ij} \) it is no longer necessary for parents and children to be coupled through a response analysis function \( a_{ij}(x_{ij}) \), but coupling through inequality constraints is allowed as well. The connection will be lost for inactive inequality
constraints, but the problem partition is still valid. By including responses in the optimization variables of a sub-problem, elements may be coupled by both types of constraints, similar to goal-coordination methods of Section 2.4.

**Coordination strategy**

Currently, ATC coordination strategies only exchange responses and targets computed in the sub-problems. By storing computed targets and responses of sub-problems, it may be possible to approximate the behavior of a sub-problem. By approximating how changes in sub-problems affect other sub-problems, it may be possible to generate better targets.

The dual part of the coordination method has received little attention. The formulation of a dual problem in the line of classical dual decomposition methods (presented in Chapter 2) may result in a faster convergence of the ATC process. Michalek et al. presented a dual coordination process that acts as a weighting update method (WUM) [Mic04a]. With the WUM, computational effort required for solving the decomposed ATC problem is decreased. Another benefit of the WUM is that the values for the weights do not have to be estimated by the user, but are updated during the process.

**Alternative decomposition methods to ATC**

As indicated throughout the chapter, alternatives for solving problem formulations typical to ATC are available. The analytical and numerical comparison of ATC to these methods may provide an indication to favor ATC over these methods, or to use the alternatives instead of ATC. It may also be possible to incorporate techniques from alternative methods in ATC in order to improve it both analytically and numerically. A comparison study would probably provide a great deal of insight in the characteristics of ATC and its alternatives.
Chapter 4

Sensitivity based coordination

The previous chapter presented the ATC decomposition method and its current coordination alternatives. This chapter investigates whether sensitivity information of sub-problem responses with respect to targets is helpful to accelerate the coordination process. By communicating sensitivity information in addition to responses and targets, ‘smarter’ target selection may be possible. If not only information is available on how responses react to targets, but also on how sensitive these reactions are, better targets may be computed. Several feasible decomposition methods use similar approaches to approximate one sub-problems effect on the other (e.g. Rosen’s method).

The following sections will discuss a sensitivity-based coordination strategy for ATC. First, the concept of the method is illustrated. Second, the computation of the sensitivities and their characteristics for ATC are discussed. Based on these characteristics, two safeguarded coordination algorithms are proposed as well as a method without safeguards. Numerical experiments for all three alternatives are discussed in Chapter 5, where they are also compared to existing coordination strategies.

4.1 Concept of sensitivity-based coordination

The objective of the coordination process is to iteratively solve the elementary sub-problems of the ATC hierarchy, until the solution converges to an optimum. In this optimal solution, all interactions between elements are specified. Elements in an ATC hierarchy interact through response and linking targets and their associated responses. Consider the generic ATC element of Figure 4.1(a). This element \( e_{ij} \) receives targets \([R_{(i-1)j}, Y_{(i-1)j}]\) from its parent, and responses \([r_{(i+1)k_1}, y_{(i+1)k_1}], \ldots, [r_{(i+1)k_{c_{ij}}}, y_{(i+1)k_{c_{ij}}}]\) from its \( c_{ij} \) children. The element computes responses \([r_{ij}, y_{ij}]\) for its parent, and targets \([R_{(i+1)k_1}, Y_{(i+1)k_1}], \ldots, [R_{(i+1)k_{c_{ij}}}, Y_{(i+1)k_{c_{ij}}}]\) for its children.

For element \( e_{ij} \), an input vector \( q_{ij} \) can be defined, which contains all the targets and
responses the element receives. Similarly, an output vector $Q_{ij}$ is defined, which contains all the targets and responses an element computes and are sent to other elements. This concept is illustrated in Figure 4.1(b). For element $e_{ij}$, the input and output vectors are defined by:

\[
q_{ij} = [R_{(i-1)j}, Y_{(i-1)j}, r_{(i+1)k_1}, y_{(i+1)k_1}, \ldots, r_{(i+1)k_{c_{ij}}}, y_{(i+1)k_{c_{ij}}}],
\]

\[
Q_{ij} = [r_{ij}, y_{ij}, R_{ik_1}, Y_{ik_1}, \ldots, R_{ik_{c_{ij}}}, Y_{ik_{c_{ij}}}],
\]

### Input-output equations

Outputs $Q_{ij}$ of element $e_{ij}$ are computed by solving the elementary sub-problem $P_{ij}$. The outputs $Q_{ij}$ of an element can be obtained directly from the optimal solution to an ATC sub-problem, if the sub-problem is formulated in the general formulation of Chapter 3. The output of an element can also be called the output of a sub-problem, because the outputs of an element are a subset of the optimization variables. Both the terms sub-problem outputs and element outputs have identical meanings in this Chapter. Elements however are defined as a part of the hierarchy, whereas a sub-problem is the optimization problem defined for that particular element. Output vector $Q_{ij}$ is a sub-vector of the optimization variables $\tilde{x}_{ij}$ of sub-problem $P_{ij}$ as defined by:
4.1. Concept of sensitivity-based coordination

\[ \bar{x}_{ij} = [x_{ij}, Q_{ij}], \quad (4.3) \]

with \( x_{ij} \) the vector of local variables of element \( e_{ij} \). The solution to the sub-problem \( P_{ij} \) depends on the set of inputs \( q_{ij} \) the element receives. The dependence of an element’s outputs \( Q_{ij} \) on inputs \( q_{ij} \) can be represented by a vector function \( F_{ij} \):

\[ Q_{ij} = F_{ij}(q_{ij}). \quad (4.4) \]

This vector function \( F_{ij} \) computes outputs \( Q_{ij} \) from a set of inputs \( q_{ij} \). Since outputs \( Q_{ij} \) depend on the optimal solution of sub-problem \( P_{ij} \), function \( F_{ij} \) defines how the optimal solution of sub-problem \( P_{ij} \) depends on the inputs \( q_{ij} \) is receives. In general, the function \( F_{ij} \) is unknown, and is therefore considered to be a black-box function.

Consider an \( N + 1 \) level ATC-hierarchy with \( p_N \) problems at the bottom level. For all elements in this ATC hierarchy, an input-output equation (4.4) can be constructed:

\[
\begin{bmatrix}
  Q_0 \\
  Q_{11} \\
  \vdots \\
  Q_{Np_N}
\end{bmatrix} =
\begin{bmatrix}
  F_0(q_0) \\
  F_{11}(q_{11}) \\
  \vdots \\
  F_{Np_N}(q_{Np_N})
\end{bmatrix}.
\]

(4.5)

By defining a global output vector \( Q = [Q_0, Q_{11}, \ldots, Q_{Np_N}] \), a global input vector \( q = [q_0, q_{11}, \ldots, q_{Np_N}] \), and a global vector function \( F = [F_0, F_{11}, \ldots, F_{Np_N}] \), global input-output relations are defined by:

\[ Q = F(q). \quad (4.6) \]

Note that the dependency notation of \( F \) suggests that each function in \( F \) can depend on each input of \( q \), whereas in ATC each subset of functions only depends on a subset of variables. The functional dependence table of functions \( F \) is block angular (see Figure 4.2). The global input-output relations (4.6) define how targets and responses computed in one sub-problem depend on the targets and responses computed at other sub-problems.
Chapter 4. Sensitivity based coordination

Figure 4.2: FDT schematic of global input-output function $F$

Hierarchy equations

By grouping inputs and outputs of elements, information of the problem hierarchy is lost. In ATC, each output of one element serves as an input for another element. It is possible to capture this hierarchy in a connection matrix $A$, where element $a_{(k,l)}$ of this matrix is one if input $k$ is connected to output $l$, and zero otherwise:

$$a_{(k,l)} = \begin{cases} 
1 & \text{if input } k \text{ is connected to output } l, \\
0 & \text{otherwise.} 
\end{cases} \quad (4.7)$$

With this connection matrix, the vectors of global outputs and inputs are linked through a hierarchy equation:

$$q = AQ \quad (4.8)$$

The above equation defines the connections between inputs and outputs. The use of the connection matrix and Eq. (4.8) is illustrated by the example hierarchy depicted in Figure 4.3. The hierarchy consists of one parent level element $e_0$ with two children $e_{11}$ and $e_{12}$. Parent $e_0$ has child responses as inputs $q_0 = [r_{11}, r_{12}]$, and computed targets as outputs $Q_0 = [R_{11}, R_{12}]$. The children have $q_{11} = R_{11}$, $Q_{11} = r_{11}$ and $q_{12} = R_{12}$, $Q_{12} = r_{12}$ respectively. The hierarchy equation (4.8) for this example hierarchy with $q = [q_0, q_{11}, q_{12}]$, and $Q = [Q_0, Q_{11}, Q_{12}]$ is:

$$\begin{bmatrix} r_{11} \\ r_{12} \\ R_{11} \\ R_{12} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} R_{11} \\ R_{12} \\ r_{11} \\ r_{12} \end{bmatrix}. \quad (4.9)$$
A fixed-point/zero-search approach for ATC coordination

With the global input-output functions (4.6) and the hierarchy relation (4.8), all relations between the global input vector and the global output vector are defined. After substitution of the global input-output equations in the hierarchy equation, the following expression which only depends on input vector \( \mathbf{q} \) holds for a decomposed problem, formulated in ATC:

\[
\mathbf{q} = A\mathbf{F}(\mathbf{q}).
\]  (4.10)

Equation (4.10) can be reformulated to finding a zero of function \( A\mathbf{F}(\mathbf{q}) - \mathbf{q} \), which describes the inconsistencies \( \theta \) within the complete ATC hierarchy as a function of the inputs:

\[
\theta(\mathbf{q}) = A\mathbf{F}(\mathbf{q}) - \mathbf{q} = 0.
\]  (4.11)

Functions \( \theta(\mathbf{q}) \) will also be referred to as the inconsistency functions. Finding a consistent solution to the ATC process is finding a fixed point of Eq. (4.10) or finding a zero of Eq. (4.11). Current ATC coordination uses successive substitution to find a fixed point. This substitution method first computes a vector of outputs from a vector of inputs (\( \mathbf{Q}^k = \mathbf{F}(\mathbf{q}^k) \)) and then uses the outputs as inputs again (\( \mathbf{q}^{k+1} = A\mathbf{Q}^k \)). This process is repeated until a solution is found. The successive substitution of ATC coordination can be described by:

\[
\mathbf{q}^{k+1} = A\mathbf{F}(\mathbf{q}^k)
\]  (4.12)
Note that for hierarchical coordination this successive substitution is not applied as a whole, but only partly between two levels of the hierarchy. The proposed all-parallel coordination does apply successive substitution to the whole hierarchy. An alternative to this substitution approach is defining a gradient based method to find a zero of Eq. (4.14).

Iterative methods are available for finding fixed points and zeros in multiple dimensions [Hea97], e.g. the successive substitution, Secant gradient approximation method, and Newton’s gradient-based method. To compare performance of these methods, they are characterized by their rate of convergence. Denote the error at iteration $k$ by $\epsilon_k = x_k - x^*$, with $x_k$ the estimate of the solution at iteration $k$, and $x^*$ the true solution. A method is said to converge with rate $\rho$ if:

$$\lim_{k \to \infty} \frac{||\epsilon_{k+1}||}{||\epsilon_k||^\rho} = \gamma,$$

for some finite non-zero constant $\gamma$. Constant $\gamma$ must be smaller than 1 for the method to converge to the solution. When $\rho = 1$ the convergence rate is linear, when $\rho > 1$ the convergence rate is superlinear, and when $\rho = 2$ the convergence rate is quadratic. A linearly convergent sequence gains a fixed number of digits of accuracy after each iteration, whereas a superlinearly convergent sequence gains an increasing number of digits of accuracy.

Methods that use only function values are called zero-order methods, since they use a zero-order Taylor approximation of the function of interest. First-order methods also use gradient information to construct linear approximations of the function. Convergence rates of zero-order methods (e.g. simplex and Secant methods) range from linear to superlinear at best, and first-order gradient-based methods like Newton’s method converge quadratically, when started close enough to the solution.

ATC coordination currently only uses function values (outputs) and can therefore be classified as a zero-order method. Since it only passes values and does not e.g. compute an approximate of the gradients, it is comparable to successive substitution methods, which have a linear convergence rate. The ATC process simply passes the computed outputs to the associated elements for finding the fixed point of Eq. (4.10).

An alternative zero-order method for ATC coordination is for instance a Secant updating approach. Secant updating methods use information obtained from a series of evaluated points to construct or update an approximation of the Jacobian. The application of such a method to ATC sounds promising since no additional information is required, and (theoretical) convergence speed is increased. However, the behavior of such a method when applied to ATC has to be evaluated.
One of the most famous first-order methods is Newton’s method. It uses both the function value and function gradients to construct a linear approximation of the function. The zero of this approximation is used as the next iterate. When started close enough to the solution, Newton’s method has a quadratic convergence rate. The first-order Newton update for finding a zero of (4.11) for ATC is:

\[
q^{(k+1)} = q^{(k)} - [AJ(q^{(k)}) - I]^{-1}[AF(q^{(k)}) - q^{(k)}],
\]

with \( I \) the identity matrix, and \( J(q^{(k)}) \) the Jacobian matrix. The Jacobian matrix contains gradient information of functions \( F \). For every iteration \( k \) all ATC sub-problems are solved in parallel for one set of input vector \( q^{(k)} \). For this vector, the function value \( F(q^{(k)}) \) (outputs) and its gradients \( J(q^{(k)}) \) are computed.

Newton’s method only converges quadratically when started near a solution. When far from a solution, safeguarded methods like move limits or line searches are suggested to keep the linear approximations valid. The safeguards are used to prevent methods from computing unusually large steps. To evaluate the application of Newton’s method in ATC coordination, the global input-output functions \( F(q) \) are investigated in the following sections.

### 4.2 Computation of Jacobian matrix of sensitivities

A first-order Newton-like coordination method for ATC requires the computation of the Jacobian matrix of \( F(q) \). This matrix contains gradient information, or sensitivity information, of functions \( F \) with respect to \( q \):

\[
J = \left[ \frac{dF_k}{dq_l} \right]_{n^q \times n^q},
\]

with \( n^q \) the number of inputs \( q \). The global Jacobian can be constructed from the Jacobians \( J_{ij} \) of individual elements of the problem hierarchy, because elementary functions \( F_{ij} \) only depend on elementary inputs \( q_{ij} \), as illustrated in the FDT of the global functions \( F \) (See Figure 4.2). The global Jacobian therefore has a block diagonal structure identical to the structure of the global FDT, and can therefore be constructed by placing elementary Jacobians \( J_{ij} \) on the diagonal of the global Jacobian \( J \), as illustrated in Figure 4.4.
The elementary Jacobians, \( J_{ij} \), contain sensitivities of functions \( F_{ij} \) with respect to inputs \( q_{ij} \). Since the vector function \( F_{ij} \) defines the outputs of an element \( F_{ij} = Q_{ij} \), the elementary Jacobian contains all sensitivities of elementary outputs \( Q_{ij} \) with respect to inputs \( q_{ij} \). The elementary Jacobian \( J_{ij} \) is defined as (element indices \( (ij) \) are dropped for clarity):

\[
J = \frac{dF}{dq} = \frac{dQ}{dq} = \left[ \frac{dQ_k}{dq_l} \right]_{n^q \times n^q} = \begin{bmatrix}
\frac{dQ_1}{dq_1} & \frac{dQ_2}{dq_1} & \cdots & \frac{dQ_{ij}}{dq_1} \\
\frac{dQ_1}{dq_2} & \frac{dQ_2}{dq_2} & \cdots & \frac{dQ_{ij}}{dq_2} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{dQ_1}{dq_n} & \frac{dQ_2}{dq_n} & \cdots & \frac{dQ_{ij}}{dq_n}
\end{bmatrix},
\tag{4.16}
\]

where \( n^q \) is the number of in and outputs of the element. For ATC the Jacobian matrix is square, because the number of outputs equals the number of inputs. With the Jacobian matrix, the relevant sensitivities in an optimal point are defined.

The Jacobian matrix of an element is computed after solving the associated sub-problem for a specific input \( q \). Two techniques can be used to determine the Jacobian matrix \( J \), a finite difference approximation or a post-optimal analytical approach.

**Finite difference**

The finite difference approach [Hea97], approximates the gradient \( J \) in a point (in this case the optimal solution to an ATC sub-problem with a specific set of inputs), by the ratio between the change of the output \( \Delta Q \) and a change of the input \( \Delta q \). The ratio approaches the true gradient when the input change approaches zero.
4.2. Computation of Jacobian matrix of sensitivities

\[ J = \frac{dQ}{dq} = \left[ \frac{dQ_k}{dq} \right]_{n_q \times n_q} = \left[ \lim_{\Delta q_l \to 0} \frac{\Delta Q_k}{\Delta q_l} \right]_{n_q \times n_q}. \]  

(4.17)

In practice \( \Delta q \) never becomes entirely zero, therefore \( J \) is approximated by using small values for \( \Delta q \):

\[ J \approx \left[ \frac{\Delta Q_k}{\Delta q_l} \right]_{n_q \times n_q} = \begin{bmatrix} \frac{\Delta Q_1}{\Delta q_1} & \frac{\Delta Q_1}{\Delta q_2} & \cdots & \frac{\Delta Q_1}{\Delta q_n} \\ \frac{\Delta Q_2}{\Delta q_1} & \frac{\Delta Q_2}{\Delta q_2} & \cdots & \frac{\Delta Q_2}{\Delta q_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\Delta Q_n}{\Delta q_1} & \frac{\Delta Q_n}{\Delta q_2} & \cdots & \frac{\Delta Q_n}{\Delta q_n} \end{bmatrix}. \]

(4.18)

For each change in input \( \Delta q_l \), an additional optimization run is required to determine the change in outputs \( \Delta Q_1, \ldots, \Delta Q_n \). Drawback of the finite difference method is the large amount of computational effort required for solving the ATC sub-problem \( n_q \) additional times. In turn, these ATC sub-problems may be very hard to solve.

From the additional \( n_q \) optimization runs, it is also possible to determine sensitivities of for instance local variables, or Lagrange multipliers with little effort from the optimization runs already required for computation of the Jacobian. These additional sensitivities may provide additional information of an element’s behavior.

Analytical post-optimal sensitivity analysis

In the ATC optimization sub-problems, inputs \( q \) are treated as fixed parameters. Analytical techniques exist to compute the sensitivity of the optimal solution with respect to these fixed parameters. Identical methods are presented in both [Sob82b] and [Haf92]. The derivation of the method is described briefly below; functional relations are adapted to the formulation of the generic ATC sub-problem of Eq. (3.2).

Derivation of the method

Consider an optimization sub-problem for ATC, which is of the general form of Eq (E.1). Note that this problem has only equality constraints \( h \). When active, an inequality constraint becomes an equality constraint, and when not active, inequality constraints do not affect the location of the optimal solution, and can therefore be disregarded in the following analysis. If variables are restricted by simple bounds, these bounds should be reformulated to a negative-null form and included in the set of design constraints.
Chapter 4. Sensitivity based coordination

\[
\begin{align*}
\min_{z \in \mathbb{R}^n} & \quad f'(z, q), \\
\text{subject to} & \quad h(z) = 0,
\end{align*}
\]

(4.19)

Where \( n \) is the number of optimization variables. Additionally, \( m \) is used to denote the number of constraints \( h \). Constraints of the sub-problem only depend on a sub-problem’s optimization variables \( z \) and are therefore independent of input parameters \( q \). The sub-problem’s objective \( f'(z, q) \) consists of two parts (See Section 3.2 for details). The first part is the local objective depending on all variables of the element \( f(z) \), and the second part is a penalty term depending on inputs and outputs of an element \( \pi(Q, q) = \| w \circ (Q - q) \|_2^2 \). The sub-problem objective depends on all optimization variables \( z \) and input parameters \( q \): \( f'(z, q) = f(z) + \pi(Q, q) \). The objective also depends on output variables \( Q \), but this dependence does not need to be explicitly mentioned, because output variables \( Q \) are a subset of the elementary optimization variables \( z \).

The Lagrangian function \( L \) of problem (E.1) after dropping the dependency notation is:

\[
L = f' + \lambda^T h,
\]

(4.20)

with \( \lambda \) the column vector of \( m \) Lagrange multipliers associated with the \( m \) equality constraints \( h \). A solution to optimization problem (E.1) can be computed by solving the necessary Karush-Kuhn-Tucker (KKT) conditions for optimality at the solution \((z^*, \lambda^*)\) [Pap00]. These KKT-conditions are:

\[
\begin{align*}
\frac{\partial L}{\partial z} \bigg|_{z=z^*} &= \frac{\partial f'}{\partial z} \bigg|_{z=z^*} + \lambda^T \frac{\partial h}{\partial z} \bigg|_{z=z^*} = 0, \\
h \big|_{z=z^*} &= 0,
\end{align*}
\]

(4.21)

(4.22)

where for the objective gradient \( \frac{\partial f'}{\partial z} \) is a row vector of \( n \) elements: \( \frac{\partial f'}{\partial z} = [\frac{\partial f'}{\partial z_1}]_{1 \times n} \). The constraint gradient matrix \( \frac{\partial h}{\partial z} \) contains the derivatives of constraints \( h \) with respect to variables \( z \): \( \frac{\partial h}{\partial z} = [\frac{\partial h_k}{\partial z_l}]_{m \times n} \). Additionally \( \frac{\partial F}{\partial z} \bigg|_{z=z^*} \) stands for the partial derivative of function \( F \) evaluated at \( z^* \).

To determine the change of the optimal solution \((z^*, \lambda^*)\) with respect to a change
4.2. Computation of Jacobian matrix of sensitivities

in input parameters \( q \), the result of full differentiation of Eqs. (4.21) and (4.22) with respect to input vector \( q \) should equal zero. By setting the results of the differentiation to zero, the KKT optimality conditions are still satisfied for a (small) change in parameter value. Assumed is that the set of active constraints does not change. When respecting functional relationships and dropping the evaluation symbol \( |_{x = x^*} \), the full differentiation of the KKT conditions (4.21) and (4.22) with respect to input parameters \( q \) yields a system of \((n + m)\) coupled equations:

\[
\begin{bmatrix}
\frac{\partial^2 f'}{\partial z^2} + \sum_{i=1}^{m} \lambda_i^* \frac{\partial^2 h_i}{\partial z^2} \\
\frac{\partial h}{\partial z} \\
\frac{\partial h}{\partial q} \\
\end{bmatrix}
\begin{bmatrix}
\frac{\partial f'}{\partial q} \\
\frac{\partial f'}{\partial q} \\
\frac{\partial f'}{\partial q} \\
\end{bmatrix}
\begin{bmatrix}
\frac{\partial z}{\partial q} \\
\frac{\partial z}{\partial q} \\
\frac{\partial z}{\partial q} \\
\end{bmatrix}
\begin{bmatrix}
\frac{\partial x^*}{\partial q} \\
\frac{\partial x^*}{\partial q} \\
\frac{\partial x^*}{\partial q} \\
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
0 \\
\end{bmatrix}
\begin{bmatrix}
\frac{\partial q}{\partial q} \\
\frac{\partial q}{\partial q} \\
\frac{\partial q}{\partial q} \\
\end{bmatrix}
\begin{bmatrix}
0 \\
0 \\
0 \\
\end{bmatrix}
+ \begin{bmatrix}
\frac{\partial^2 f'}{\partial q^2} \\
\frac{\partial^2 f'}{\partial q^2} \\
\frac{\partial^2 f'}{\partial q^2} \\
\end{bmatrix}
\begin{bmatrix}
\frac{\partial x^*}{\partial q} \\
\frac{\partial x^*}{\partial q} \\
\frac{\partial x^*}{\partial q} \\
\end{bmatrix}
\begin{bmatrix}
\frac{\partial q}{\partial q} \\
\frac{\partial q}{\partial q} \\
\frac{\partial q}{\partial q} \\
\end{bmatrix}
\begin{bmatrix}
0 \\
0 \\
0 \\
\end{bmatrix}
\end{bmatrix}.
\]

(4.23)

System (4.23) can be solved either by explicit inversion of the matrix of coefficients \( C \) or by Gaussian elimination. Solvability of this system of \((n + m)\) coupled equations, depends on the properties of the matrix of coefficients \( C \). System (4.23) has a unique solution if matrix \( C \) is regular. If \( C \) is singular, matrix \( C \) cannot be inverted and no unique solution to system (4.23) exists. Gaussian elimination however may still provide useful results, e.g. a part of the solution which is unique, and information on how non-unique solutions relate to each other.

Computation of the Jacobian matrix

From the solution to system (4.23), the elementary Jacobian matrix can be obtained. Recall the vector of all optimization variables of an ATC sub-problem: \( z = [x, Q] \). The post-optimality approximation of the elementary Jacobian matrix of sensitivities \( J^P = \frac{dQ}{dq} \) is a sub-matrix of the sensitivity matrix \( \frac{dz}{dq} \) as given by:
The above analytical approach to computing sensitivities of outputs with respect to inputs is elegant. Much information of the sensitivity system (4.23) is often available as a by-product of the optimization routine. For instance, an SQP algorithm requires computation of constraint gradients and Lagrange multipliers; (estimates of) objective Hessians are often available. Constraint Hessians however are rarely available, and have to be computed after optimization, as well as the matrix $\frac{\partial^2 f}{\partial q \partial z}$. After computation of these additional terms, the system can be solved by Gaussian elimination.

The post-optimality sensitivity analysis requires only (relatively cheap) operations such as the computation of additional terms and the solution of the sensitivity system of equations, contrary to the finite difference approach which requires costly additional optimization runs. The lower computational effort of the sensitivity analysis suggests that this method should be preferred over the more expensive finite difference approach.
4.3 General sensitivity properties

sub-problems. For dependent outputs however, sensitivities are dependent as well. The
difference between independent and dependent outputs is discussed in greater detail in
Appendix F.

It is possible to eliminate dependent variables from the optimization problem, without
changing the problem’s properties. The number of optimization variables is decreased
by the number of dependent variables, and a number of equality constraints linking the
dependent variables are removed. Removing dependent variables from a problem is in
accordance with the notational changes with respect to ATC presented by [Mic04b],
which is also discussed in Appendix C. Independent variables cannot be removed from
the problem without changing it.

4.3 General sensitivity properties

To illustrate the characteristics of output sensitivities with respect to inputs, consider
the ATC sub-problem depicted in Figure 4.5. For this sub-problem \( z = \mathbf{x} = [x_1, x_2, r] \)
is the vector of optimization variables, \( R \) is a target input, \( r \) is the response of the sub-
problem, and \( a(x_1, x_2) \) is the analysis function of the sub-problem. Note that equality
constraints are labelled \( h \) and inequality constraints are labelled \( g \), whereas the post-
optimal sensitivity method used \( h \) for both equalities and active inequalities. For the
post-optimality analysis, a vector of active constraints \( h \) is constructed, which contains
all equality constraints, as well as all active inequality constraints. From the context it
becomes clear which inequality constraints are active and added to set \( h \).

![Figure 4.5: Example element with single in and single output](image)

Figure 4.5: Example element with single in and single output

The feasible range of \( x_1 \) and \( x_2 \) is limited to values between 0 and \((2 - x_2)\) for \( x_1 \), and
0 and \((2 - x_1)\) for \( x_2 \), and therefore the response variable \( r \) is bounded between 0 and
2, which are the extreme values of \( r = a(x_1, x_2) \) on the feasible sub-space of \( x_1 \) and \( x_2 \).
Figure 4.6(a) displays the response \( r^* \) as a function of targets \( R \).
Chapter 4. Sensitivity based coordination

\[ r^* = \begin{cases} 
0 & \text{if } R < 0 \\
R & \text{if } 0 \leq R \leq 2 \\
2 & \text{if } R > 2 
\end{cases} \]  \hfill (4.26)

Figure 4.6: Response behavior of SISO example problem of Figure (4.5)

In this figure, two important regions can be identified, one region for which inputs can be matched by outputs, and one for which they can’t be matched. Inputs \( R \) in the region \( 0 \leq R \leq 2 \) can be matched by the output \( r: R = r \). Inputs within this range are attainable, and the region \( 0 \leq R \leq 2 \) is called the attainable domain. In general, inputs that can be matched exactly by outputs \( (r = R) \), are called realizable inputs. The domain of inputs for which inputs are realizable is called the realizable domain of a sub-problem. The realizable domain of inputs is defined by the feasible sub-space of the outputs, which in turn is defined by a sub-problem’s constraints. The realizable domain and the feasible sub-space are identical under the ATC formulation of Chapter 3. Inputs \( R \) within the feasible sub-space of \( r \) are realized, while inputs \( R \) outside the feasible sub-space of response \( r \) are non-realizable, and have optimal solutions \( r^* \) with minimal distance to the input vector \( R \).

The sensitivity of response \( r^* \) with respect to target \( R \) as a function of \( R \) is very simple: in the realizable range, responses will follow targets, hence the sensitivity equals 1; outside the realizable range responses will not change for small changes in \( R \), hence the sensitivity is 0:

\[ \frac{dr^*}{dR} = \begin{cases} 
0 & \text{if } R < 0 \\
1 & \text{if } 0 \leq R \leq 2 \\
0 & \text{if } R > 2 
\end{cases} \]  \hfill (4.27)
4.3. General sensitivity properties

Figure 4.6(b) displays the sensitivity function. The sensitivity is discontinuous for \( R = 0 \) and \( R = 2 \), which can also be seen in the figure.

The above sensitivities are also computed using the post-optimal sensitivity analysis. The post-optimal analysis is applied to two example inputs, one realizable input, \( R = 1 \), is selected, and one non-realizable input \( R = -1 \).

**Realizable inputs**

Consider the realizable input \( R = 1 \). The feasible solution is every point \( z^* = (1, x_1^*, \sqrt{1-x_1^2}) \). The only constraint active in this point is \( h = [h_1] = [r - \sqrt{x_1^2 + x_2^2}] \). The associated Lagrange multiplier for this constraint is zero, \( \lambda^* = [\lambda_1^*] = 0 \), suggesting that it is redundant in the optimal point. The Lagrange multipliers are necessarily zero for optimality. For an optimal solution the KKT conditions (4.21) and (4.22) must be satisfied. For realizable inputs inconsistencies are zero, and therefore the sub-problem objective and its gradient are zero. Constraints gradients however are usually non-zero. From (4.21) follows that the Lagrange multipliers \( \lambda \) must therefore be zero, suggesting an unconstrained optimization problem for realizable inputs.

The sensitivity system 4.23 for e.g. the optimal point \( z_0^* = (1, \frac{1}{2} \sqrt{3}, \frac{1}{2}) \) is:

\[
\begin{bmatrix}
0 & 0 & 0 & -\frac{1}{2} \sqrt{3} \\
0 & 0 & 0 & -\frac{1}{2} \\
-\frac{1}{2} \sqrt{3} & -\frac{1}{2} & 1 & 0
\end{bmatrix}
\begin{bmatrix}
\frac{dx_1^*}{dR} \\
\frac{dx_2^*}{dR} \\
\frac{dr}{dR} \\
\frac{d\lambda_1^*}{dR}
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
2 \\
0
\end{bmatrix}
\]

or \( CU = B \). The matrix of coefficients \( C \) of this system is singular, and therefore the matrix of sensitivities \( U \) cannot be determined uniquely by \( U = C^{-1} B \). By Gaussian elimination however, output and Lagrange multiplier sensitivities can be computed uniquely: \( \frac{dx_1^*}{dR} = 1 \), and \( \frac{d\lambda_1^*}{dR} = 0 \).

The remaining two sensitivities with respect to local variables may take on any value, as long as \( \frac{1}{2} \sqrt{3} \frac{dx_1^*}{dR} + \frac{1}{2} \frac{dx_2^*}{dR} = 1 \) is satisfied.

The Jacobian does not predict constraint activity changes for inactive constraints, since it only describes local behavior. With an input in the realizable range, any change in input can be matched from a local point of view. With only the local sensitivity of
outputs, the range of realizable inputs seems unbounded. In reality however, realizable responses are constrained by the locally inactive constraints. Not being able to predict constraint activity changes is an important limitation of describing a sub-problem with only local sensitivities of outputs or Lagrange multipliers.

It is not possible to predict constraint activity by adding inactive constraints to the sensitivity system. Inactive constraints have Lagrange multipliers equal to zero, and locally the constraints will not become active for small changes in targets; hence Lagrange multipliers will be insensitive with respect to input changes. With an initial Lagrange multiplier equal to zero and a sensitivity of zero, no change in Lagrange multipliers (and thus constraint activity) can be predicted by using only output and Lagrange multiplier sensitivities.

Furthermore, the matrix of coefficients $C$ can become singular for an ATC sub-problem with a realizable input, as demonstrated. When the matrix $C$ is singular, sensitivities of the optimal solution $z^*, \lambda^*$, are non-unique. Fortunately, the optimal solution and sensitivities for both output and Lagrange multiplier are always unique. However, the optimal values and sensitivities of local variables $x = [x_1, x_2]$ are non-unique.

In general, one can say that ATC sub-problems have non-unique sensitivities if the optimal solution to the ATC problem is non-unique. The non-uniqueness of ATC solutions causes the matrix of coefficients to be singular, and therefore no unique solution to the sensitivity system exists. This situation occurs for instance when a problem has no local objective $f$, but an objective consisting only of the penalty term $f' = ||r - R||^2_2$, inputs are realizable ($r = R$), and the number of local variables is larger than the number of locally active constraints. The simultaneous occurrence of all these conditions is possible for ATC sub-problems. Under these conditions, degrees of freedom remain unrestricted by constraints or objective gradients.

Fortunately, the sensitivities of the output variables $Q$ are always unique, whereas only sensitivities with respect to local variables $x$ are non-unique. This fortunate property is caused by the fact that the solution of an ATC sub-problem is uniquely defined for the output variables $Q$ by the KKT-conditions, whereas optimal values for local variables are non-uniquely defined.

**Non-realizable inputs**

Again, consider the ATC sub-problem of Figure 4.6(a), but with non-realizable input $R = -1$, which is outside the realizable domain of $r$. The optimal solution is $z^* = [0, 0, 0]$, with all constraints except $g_1$ active and with Lagrange multipliers $\lambda = [\lambda_1^*, \mu_2^*, \mu_3^*] = [4, 4, -2]$, with $\lambda_1$ the Lagrange multiplier of $h_1$, and $\mu_2$ and $\mu_3$ the Lagrange multipliers of inequality constraints $g_2$ and $g_3$ respectively. The sensitivity system in this point is:
4.3. General sensitivity properties

\[
\begin{bmatrix}
0 & 0 & 0 & -2 & -1 & 0 \\
0 & 0 & 0 & 0 & -2 & -1 \\
0 & 0 & 2 & 0 & 1 & 0 \\
-2 & -2 & 1 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
x^*_1 \\
x^*_2 \\
x^*_3 \\
h_1^* \\
x^*_4 \\
x^*_5
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
2 \\
0 \\
0 \\
0
\end{bmatrix},
\]

(4.29)

with unique solution

\[
U = \begin{bmatrix}
dx^*_1/R \\
dx^*_2/R \\
x^*_3/R \\
h_1^*/R \\
x^*_4/R \\
x^*_5/R
\end{bmatrix}
T = \begin{bmatrix}
0 & 0 & 2 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}T.
\]

The sensitivity of output \(r^*\) is equal to 0, suggesting that any small change in input \(R\) does not change its solution. This prediction is accurate, since \(r = 0\) will still be the solution that minimizes the objective \(f' = ||r - R||^2_2\), within the feasible space of the sub-problem. For small changes in \(R\) sensitivities of local variables \(x^* = [x^*_1, x^*_2]\) are 0 as well, because equality constraint \(h_1\) will be satisfied only for \(x^* = [0, 0]\). Lagrange multipliers \(\lambda^*\) however are sensitive with respect to a change in input \(R\).

The output sensitivity can be used to estimate the realizable domain of inputs. A realizable input \(R' = R + \Delta R\) has a matching output \(r^* = R^*\). The change in input for which this condition holds can be computed from linear approximations of \(r^*\) around \(R\) by using sensitivity information. To find the realizable input the linear approximation of \(r^*\) must equal the new target \(R' = R + \Delta R\). The change in input \(\Delta R\) for which this happens is given by:

\[
r^* + \frac{dr^*}{dR} \Delta R = R + \Delta R.
\]

(4.30)

For the current input vector \(R = -1\), the solution to this equation is \(\Delta R = +1\), resulting in an estimated realizable input of \(R' = R + \Delta R = 0\), which is located exactly on the edge of the realizable domain of inputs. For MIMO sub-problems however, the linear approximations are not accurate (see Appendix F).

The Lagrange multiplier sensitivities can also be used to estimate the realizable domain of inputs. As observed for realizable inputs, all Lagrange multipliers are 0 when input \(R\) is realizable. It is possible to use Lagrange multiplier sensitivities to construct linear approximations of the optimal solutions of the Lagrange multipliers \(\lambda^*\). By setting these approximations equal to zero, an estimate of the edge of the realizable domain of inputs can be obtained. In other words, find a change in target \(\Delta R\) for which the approximations of Lagrange multipliers become zero, by solving:
\( \lambda_i' = \lambda_i^* + \frac{d\lambda_i^*}{dR} \Delta R = 0. \) (4.31)

When setting the input extrapolation \( R' \) equal to \( R' = R + \Delta R \), this extrapolated input \( R' \) is an estimate of the location of the realizable domain of inputs. For \( R = -1 \), this extrapolation suggests a change in target \( \Delta R = +1 \) to reach the edge of the realizable domain. The ‘new’ target should become \( R' = 0 \), for the target to be realizable, which equals the lower bound of realizable targets. Extrapolations for targets \( R \) greater than the upper bound of \( r \), also result in targets updates \( R' \) located exactly on the upper bound of \( r \). Again for MIMO sub-problems, the approximations however are not accurate.

Non-realizable inputs provide information on the location of (a subset of) inequality constraints, in contrast to realizable inputs where inequality constraint locations were not predictable. This difference is due to the fact that inequality constraints become active for non-realizable inputs, and are therefore included in the sensitivity equations. With this additional information, the constraint location can be determined by extrapolation of either the outputs, or the Lagrange multipliers.

**Generalization for SISO sub-problems**

For any sub-problem with one input \( q \) and one output \( Q \), the following can be said: with \( \mathcal{Q} \) the feasible sub-space of output \( Q \), and thus the realizable domain of input \( q \), the output \( Q \) with respect to input \( q \) is:

\[
Q = \begin{cases} 
\min(Q \mid Q \in \mathcal{Q}) & \text{if } q < \min(Q \mid Q \in \mathcal{Q}), \\
q & \text{if } q \in \mathcal{Q}, \\
\max(Q \mid Q \in \mathcal{Q}) & \text{if } q > \max(Q \mid Q \in \mathcal{Q}).
\end{cases}
\] (4.32)

Note that the minimum or maximum of \( \mathcal{Q} \) does not necessarily occur at edges of the realizable domain, but may be an unconstrained minimum or maximum within the domain. The sensitivity of output \( Q \) with respect to input \( q \) is:

\[
\frac{dQ}{dq} = \begin{cases} 
0 & \text{if } q < \min(Q \mid Q \in \mathcal{Q}), \\
1 & \text{if } q \in \mathcal{Q}, \\
0 & \text{if } q > \max(Q \mid Q \in \mathcal{Q}).
\end{cases}
\] (4.33)
Furthermore, the Lagrange multipliers of all constraints are zero when inputs are in
the realizable domain. Sensitivities of Lagrange multipliers of equality constraints are
zero as well. For non-realizable inputs, Lagrange multipliers are non-zero for equality
constraints and active inequality constraints. Sensitivities of Lagrange multipliers and
their optimal values can be used to predict the location of constraint boundaries.

Generalization of properties for ATC sub-problems

This section summarizes and generalizes properties of the Jacobian matrix of sensitivi-
ties and Lagrange multipliers for all ATC sub-problems.

For realizable input vectors, the following can be said with respect to sensitivities of
outputs and Lagrange multipliers associated with the problem’s active constraints:

- Independent outputs are only sensitive with respect to the associated output (e.g.
target and associated response). The outputs are insensitive to changes is other
inputs. For a problem with only independent outputs, the Jacobian matrix is the
identity matrix, implying that any change in target produces exactly the same
change in the output.

- Dependent outputs have linearly dependent sensitivities. The dependent outputs
are sensitive with respect to multiple coupled inputs (e.g. all computed linking
targets for an element’s children are sensitive to changes in a linking response of
one child). Dependent outputs can only match input changes when all associ-
ated inputs change by amounts satisfying the constraints coupling the dependent
outputs.

- All Lagrange multipliers are zero (including equality constraint multipliers). Sensi-
tivities of non-coupling equality constraints are zero, whereas equality constraints
coupling dependent output variables are sensitive with respect to input changes.
The Lagrange multiplier extrapolations are only equal when dependent input
changes satisfy the coupling constraint.

- Activity changes of inequality constraints are not predicted by sensitivity infor-
mation, because only active inequality constraints are considered in the post-
optimality analysis.

For non-realizable input vectors, the following can be said with respect to sensitivities
of outputs and Lagrange multipliers:

- All outputs have either linearly dependent sensitivities or outputs are completely
insensitive to changes in the input. For non-realizable input vectors, inequality
constraints become active and create dependencies between independent variables. All independent variables become dependent and will all have dependent sensitivities.

- All equality constraint Lagrange multipliers are non-zero, as well as multipliers associated with active inequality constraints. All Lagrange multipliers are sensitive with respect to input changes. Constraints with linear dependent gradients have linear dependent sensitivities of the Lagrange multipliers.

- Output and Lagrange multiplier sensitivity can be used to approximate the location of active inequality constraints.

For problems with \( n^i \) inputs \( q_i \), \( n^Q \) dependent outputs \( Q_i \), and \( n^{Q'} \) independent outputs, and \( Q \) denoting the feasible domain of outputs \( Q_i \), the Jacobian of output sensitivities is of the following rank:

\[
\text{rank}(J) \begin{cases} 
= n^{Q'} & \text{if } q_i \in Q, \\
< n^{Q'} & \text{if } q_i \notin Q. 
\end{cases}
\] (4.34)

Dependent output variables have dependent sensitivities and therefore do not contribute to the rank of the Jacobian matrix. The rank of the Jacobian also decreases for non-realizable input vectors. In this case, inequality constraints become active and can be regarded as equality constraints that link the degrees of freedom that are not coupled through the equality constraints. Independent output variables become coupled, with dependent sensitivities, decreasing the rank of the Jacobian.

### 4.4 Sensitivity based coordination algorithms

The previous sections discussed the characteristics of output sensitivities for analytical target cascading. This section will present three coordination strategies that use sensitivity information of outputs with respect to inputs. The first is a pure Newton-like coordination strategy, using only sensitivity information without any safeguards. The remaining two have safeguards to keep the linear approximations constructed with output sensitivities valid. The safeguarded methods are developed to account for the unboundedness of outputs implied by locally computed output sensitivities.

#### Finding ATC solutions with minimal inconsistency

In Section 4.1 the ATC coordination process for finding solutions with zero inconsistencies was reformulated to a zero search (4.11), recall:
4.4. Sensitivity based coordination algorithms

\[ \theta(q) = AF(q) - q = 0. \]

However, for ATC it is possible that targets are unattainable, and the inconsistencies can become non-zero. A possible source of non-zero inconsistencies is the use of ‘stretch’ targets (e.g. to maximize profits). Usually, unattainable ‘stretch’ targets are located at the top-level and included in the top-level local objective function and are not used as inputs or outputs. However, when one target cannot be met, none of the targets can be met, and as a result all inconsistencies are non-zero, as proven by [Mic04a].

For non-zero inconsistencies Eq. (4.11) does not hold. One suggestion to account for non-zero inconsistencies is to minimize the (weighted) inconsistencies:

\[ \min_w \|w \circ \theta\|_2^2 = \min_q \|w \circ (AF(q) - q)\|_2^2, \]  
(4.35)

where weighting coefficients \( w \) express the relative importance of inconsistencies, and also serve as scaling parameters. These weighting coefficients should be chosen identical to those used for the relaxation of the all-at-once formulation. If not chosen equal, the optimal solutions of the coordination process and the ATC process are not identical, resulting in a coordination process that does not guide the ATC process to its solution, but pushes it away towards a different solution. Note that if a consistent solution exists (i.e. \( \theta = 0 \)), this point is the optimal solution to optimization problem (4.35). Within the optimization problem (4.35) it is possible to add lower and upper bounds on the input variables, the bounds corresponding with bounds on the original problem variables.

For this problem (4.35) zero-order and first-order approximations of functions \( F(q) \) can be constructed. With these approximations, an iterative coordination strategy can be defined. The approximations are computed for each iteration \( k \), where all ATC sub-problems are solved in parallel. For this parallel run, function values \( Q^{(k)} = F(q^{(k)}) \) are evaluated and the Jacobian \( J \) is constructed. With the approximations constructed, the coordination algorithm computes a new vector of inputs \( q^{(k+1)} \).

Zero-order algorithms use only the function value to approximate the global function. Successive substitution like algorithms use the function values \( F(q^{(k)}) \) computed for iteration \( k \) to construct zero-order Taylor approximations of the input-output functions: \( F(q^{(k+1)}) \approx Q^{(k)} = F(q^{(k)}) \). The optimization problem of (4.35), which is solved for the updated input vector \( q^{(k+1)} \) for zero-order methods is given by:

\[ \min_{q^{(k+1)}} \|w \circ (AF(q^{(k)}) - q^{(k+1)})\|_2^2, \]  
(4.36)
The solution to this problem is \( q^{(k+1)} = A Q^{(k)} \). This solution indicates that for a zero-order ATC coordination method, input updates are equal to the outputs computed with the previous input vector, and outputs are simply sent to the associated elements. The result of the iterative zero-order strategy that minimizes inconsistencies is equal to the all-parallel coordination strategy for ATC of Section 3.8.

A first-order approach uses linear approximations of functions \( F \) around \( q^{(k)} \). This first order Taylor approximation of \( F \) around \( q^{(k)} \) is given by:

\[
F(q^{(k+1)}) \approx F(q^{(k)}) + J(q^{(k)})(q^{(k+1)} - q^{(k)}). \quad (4.37)
\]

After substitution of these linear approximations in problem (4.35), the following optimization problem is obtained for an iterative first-order coordination strategy for ATC:

\[
\min_{q^{(k+1)}} ||w \circ (A F(q^{(k)}) + A J(q^{(k)})(q^{(k+1)} - q^{(k)}) - q^{(k+1)})||^2_2, \quad (4.38)
\]

which is solved for input update \( q^{(k+1)} \). After solving the coordination problem, updated inputs are sent to the elements in the ATC hierarchy. For this set of inputs, outputs and their sensitivities are computed by solving all ATC sub-problems and performing the post-optimality sensitivity analyses. From these individual outputs and sensitivities, the global vector of outputs and the global Jacobian is constructed. The global outputs and Jacobian are used as parameters in the first-order coordination problem (4.38). This process iterates until some convergence criterion is met, e.g. the absolute inconsistencies are smaller than some predefined tolerance. The algorithm for non-safeguarded sensitivity-based coordination for ATC is depicted in Figure 4.7. The two-level structure of the coordination strategy for this feasible coordination method is displayed in Figure 4.8.

Note that when all inputs are realizable, all inconsistencies are zero, and the sensitivity-based coordination no longer updates the inputs based on output sensitivities. Therefore, when all inputs are realizable, the sensitivity-based coordination method reduces to the zero-order all-parallel coordination strategy presented in Section 3.8.
4.4. Sensitivity based coordination algorithms

1. Set \( k = 0 \), choose initial global input vector \( \mathbf{q}^0 \).

2. Solve all ATC sub-problems for fixed inputs \( \mathbf{q}^k \), to obtain global outputs \( \mathbf{Q}^k \), and compute global Jacobian \( J^k \) with the post-optimality sensitivity equations (4.23).


4. If step 3 fails, solve master-problem (4.38):

\[
\min_{\mathbf{q}^{k+1}} || \mathbf{w} \circ ((A\mathbf{Q}^k + AJ^k)(\mathbf{q}^{k+1} - \mathbf{q}^k) - \mathbf{q}^{k+1}) ||^2_2,
\]

then set \( k = k + 1 \), and return to step 2; otherwise stop.

Figure 4.7: Non-safeguarded sensitivity-based coordination for ATC

\[
\min_{\mathbf{q}^{k+1}} || \mathbf{w} \circ ((A\mathbf{Q}^k + AJ^k)(\mathbf{q}^{k+1} - \mathbf{q}^k) - \mathbf{q}^{k+1}) ||^2_2,
\]

\( \mathbf{q}^* \) \( \downarrow \) \( \mathbf{Q}^*, J \)

Figure 4.8: Two level structure of non-safeguarded sensitivity-based coordination

Safeguarded methods

Newton’s method for finding zeros of functions in multiple dimensions only converges when started close enough to the solution. In general, move limits are used to keep the linearisations valid. With these move limits, a sort of trust region is placed around the point for which the functions are approximated. If the next iterate based on the Newton step is outside of this trust region, the iterate is moved to the point within the trust region, closest to the computed Newton step. If an iterate is within the trust region, the approximation model is reliable and the iterate can be trusted.

For the ATC coordination problem, suppose \( \mathbf{m} \) is a vector of positive move limits, where element \( i \) of \( \mathbf{m} \) holds the move limits for input \( i \). The move limits are implemented in the coordination method as simple bounds around the input vector \( \mathbf{q}^{(k)} \) used...
Chapter 4. Sensitivity based coordination

to compute outputs and their sensitivities. The first-order optimization problem with move limits for ATC coordination is:

\[
\begin{align*}
\min_{q^{(k+1)}} & \quad \|\mathbf{w} \circ (\mathbf{A}F(q^{(k)}) + \mathbf{A}J(q^{(k)})(q^{(k+1)} - q^{(k)}) - q^{(k+1)})\|_2^2, \\
\text{subject to} & \quad q^{(k)} - \mathbf{m} \leq q^{(k+1)} \leq q^{(k)} + \mathbf{m}.
\end{align*}
\] (4.39)

The optimal solution for input updates \(q^{(k+1)}\) lies in the interval of \(\pm \mathbf{m}\) around the evaluation inputs \(q^{(k)}\). The algorithm for non-safeguarded sensitivity-based coordination with move limits is depicted in Figure 4.9. The two-level structure of this coordination strategy with move limits is displayed in Figure 4.10.

1. Set \(k = 0\), set move limit vector \(\mathbf{m}\), choose initial global input vector \(q^0\).

2. Solve all sub-problems for fixed inputs \(q^k\), to obtain global outputs \(Q^k\), and compute global Jacobian \(J^k\) with the post-optimality sensitivity equations (4.23).


4. If step 3 fails, solve master-problem (4.39):

\[
\begin{align*}
\min_{q^{k+1}} & \quad \|w \circ (AQ^k + AJ^k(q^{k+1} - q^k) - q^{k+1})\|_2^2, \\
\text{subject to} & \quad q^k - s \leq q^{k+1} \leq q^k + s,
\end{align*}
\]

to obtain optimal vector of global inputs \(q^{k+1}\), set \(k = k + 1\), and return to step 2; otherwise stop.

Figure 4.9: Sensitivity-based coordination for ATC with move limits

A second, more ATC specific, safeguard can be considered. In the previous sections, the characteristics of the individual Jacobians were discussed, and for realizable inputs was observed that the linear approximations cannot predict constraint activity changes. Because of the unboundedness suggested by the Jacobian, it is possible that the coordination process computes input updates outside of the realizable range of the element. To account for this unboundedness, a hybrid coordination strategy is proposed. This hybrid method switches between a first-order procedure and a zero-order method. The first-order approach is used when Jacobian information provides useful information on constraint locations, thus for non-realizable inputs, and the zero-order approach is used when the Jacobian predicts unboundedness, thus for realizable inputs. This hybrid
4.4. Sensitivity based coordination algorithms

\[
\min_{\mathbf{q}^{k+1}} \left\| \mathbf{w} \circ (A\mathbf{Q}^k + AJ^k(\mathbf{q}^{k+1} - \mathbf{q}^k) - \mathbf{q}^{k+1}) \right\|^2_2,
\]
subject to \( \mathbf{q}^k - \mathbf{m} \leq \mathbf{q}^{k+1} \leq \mathbf{q}^k + \mathbf{m}. \)

Figure 4.10: Two level structure of sensitivity-based coordination with move limits

The implementation of this hybrid method is rather straightforward: add a constraint to problem (4.38), for each input that was found to be realizable. This constraint should force the coordination process to keep input updates equal to the realizable inputs, computed from the previous iterate. The algorithm for the hybrid coordination strategy is depicted in Figure 4.11. The two-level structure of this hybrid strategy is displayed in Figure 4.12.

1. Set \( k = 0 \), choose initial global input vector \( \mathbf{q}^0 \).

2. Solve all sub-problems for fixed inputs \( \mathbf{q}^k \), to obtain global outputs \( \mathbf{Q}^k \), and compute global Jacobian \( J^k \) with the post-optimality sensitivity equations (4.23).


4. If step 3 is successful, stop; otherwise instantiate an empty equality constraint set \( \mathbf{c} = [] \), and for each realizable input \( \mathbf{q}^k_i \), add \( \mathbf{q}_i^{k+1} - \mathbf{q}_i^k = 0 \) to constraint set \( \mathbf{c} \), and solve the master-problem:

\[
\min_{\mathbf{q}^{k+1}} \left\| \mathbf{w} \circ (A\mathbf{Q}^k + AJ^k(\mathbf{q}^{k+1} - \mathbf{q}^k) - \mathbf{q}^{k+1}) \right\|^2_2,
\]
subject to \( \mathbf{c}(\mathbf{q}^{k+1}) = 0. \)

to obtain optimal vector of global inputs \( \mathbf{q}^{k+1} \), set \( k = k + 1 \), and return to step 2.

Figure 4.11: Hybrid coordination for ATC
The following chapter discusses the comparison of the proposed methods to existing ATC coordination strategies. Coordination strategies are compared with respect to convergence behavior and computational costs. Three example problems are used to compare all methods.

4.5 Discussion

Alternative feasible master problems

Three sensitivity-based Newton-like feasible master problems are proposed for the solving the ATC decomposed sub-problem. This may however not be the only way to use output sensitivity information in the coordination process. It is possible to define alternative master problems in terms of inputs $q$.

One alternative is to define a zero-order feasible master problem based on a Secant algorithm for minimizing the inconsistencies. In this Secant-like approach, approximations of the gradients of the inconsistency functions can be constructed from the outputs values of sub-problems, without performing the post-optimality analyses.

Another alternative is to define a coordination method that uses the sensitivities of outputs or Lagrange multipliers to construct an approximation of the realizable domain of inputs $q$. The master problem then uses this approximation as a constraint, estimating the realizable domain of inputs. The master problem may have an objective $f$, which is defined only as a function of inputs $q$ by using sensitivities of both outputs and local variables. With these sensitivities a linear approximation of the objective $f$ as a function of inputs $q$ can be defined:

$$f(q + \Delta q) \approx f^*(q) + \frac{df}{dq} \Delta q = f^*(q) + \left( \frac{\partial f}{\partial q} + \frac{\partial f}{\partial x} \frac{dx}{dq} \right) \Delta q$$

(4.40)
It is also possible to linearize constraints in a similar manner:

\[
\begin{align*}
g(q + \Delta q) & \approx g^*(q) + \frac{dg}{dq} \Delta q = g^*(q) + \left( \frac{\partial g}{\partial q} + \frac{\partial g}{\partial x} \frac{dx}{dq} \right) \Delta q \\
h(q + \Delta q) & \approx h^*(q) + \frac{dh}{dq} \Delta q = h^*(q) + \left( \frac{\partial h}{\partial q} + \frac{\partial h}{\partial x} \frac{dx}{dq} \right) \Delta q
\end{align*}
\]

(4.41) (4.42)

By defining a master problem only in inputs \( q \), a reduced form of the original AAO problem is obtained. The master problem is defined only in input variables \( q \), instead of all optimization variables, which are not only inputs, but also outputs and local variables.

With this reduced problem formulation, a set of realizable inputs is searched, that minimizes the original objective.

**Non-hierarchical sensitivity-based coordination for ATC**

Just as the all-parallel coordination proposal of Section 3.8, sensitivity-based coordination does not exclude non-hierarchical problem structures. The coordination method is only concerned with elements that have inputs and outputs and in what way they are connected. Elements do not necessarily need to placed in a hierarchical framework, but may be coupled non-hierarchically. For non-hierarchical problems, the distinction between response and linking variables is no longer required, because linking variables no longer need to be communicated through a parent, but can be communicated directly between two children. An ATC like problem formulation would only have to distinguish fixed input parameters and variable output parameters. A revised problem statement for element \( e_{ij} \) for a non-hierarchical problem could be of the form:

\[
\begin{align*}
\min_{\bar{x}_{ij}} & \quad f_{ij}(\bar{x}_{ij}) + \|w_{ij} \circ (q_{ij} - Q_{ij})\|_2^2, \\
\text{subject to} & \quad g_{ij}(\bar{x}_{ij}) \leq 0, \\
& \quad h_{ij}(\bar{x}_{ij}) = 0, \\
& \quad \text{where } \bar{x}_{ij} = [x_{ij}, Q_{ij}].
\end{align*}
\]

(4.43)

Both the non-hierarchical problem structure and the elegant problem formulation provide an easy to implement feasible decomposition method. The behavior of such a non-hierarchical ATC method has to be investigated further to determine the usefulness of such a method.
Chapter 4. Sensitivity based coordination
Chapter 5

Coordination performance comparison

In the previous chapter three new coordination algorithms for ATC using output sensitivities have been proposed. It is expected that these new sensitivity-based algorithms will outperform the other algorithms with respect to required computational costs and convergence speed. The goal of the new coordination method is to reduce the number of ATC iterations needed for convergence, and therefore, possibly, computational effort. This chapter presents numerical experiments for five different coordination strategies and discusses the obtained results.

5.1 Description of the numerical experiments

Evaluated coordination algorithms

Two zero-order coordination methods are evaluated: hierarchical scheme III and the all-parallel coordination scheme, both presented in Section 3.8. Scheme III is used as a state-of-the-art reference of ATC coordination, and the all-parallel coordination is evaluated as an alternative to investigate the possibility of applying ATC to non-hierarchical problems. Three first-order approaches are evaluated as well: the non-safeguarded sensitivity-based method, and two safeguarded sensitivity methods, one with move limits, and the other is the hybrid method. All first-order algorithms are presented in Section 4.4.

Note that the actual state-of-the-art for ATC actually is the weighting update method (WUM) of [Mic04a] combined with hierarchical coordination. The WUM however addresses the dual part of the coordination of ATC, whereas the hierarchical and sensitivity-based methods only address the feasible coordination. The WUM can be superimposed over all of the feasible coordination strategies to form a dual-feasible co-
ordination approach to ATC. The proposed coordination methods are not to be used instead of the WUM, but in combination with it. Therefore only feasible coordination strategies are included in the comparison.

Since the dual coordination is not considered, the feasible coordination uses fixed weights. The behavior of the proposed coordination methods should be evaluated for a range of weights, similar to the numerical experiments presented by Hulshof [Hul03]. In his work, experiments with hierarchical coordination were conducted to find an optimal weighting factor $w$, that minimizes the error between the solution to the ATC process, and the true solution for a number of ATC termination tolerances. Small errors however required many iterations. In the current examples, the numerical behavior of all coordination methods is evaluated for a range of weights and a number of ATC termination tolerances, similar to the work of Hulshof. Appendix H contains all Matlab files of the implementation of all five coordination algorithms for all three example problems.

Evaluated optimization problems

The numerical behavior is investigated for several example problems presented in Appendix A. Three geometric optimization problems are selected: the geometric optimization problem 1 (5.6), and two three-level decompositions of the geometric optimization problem 2 used also as an example problem in [Kim01a, Etm02, Etm04, Tze03, Hul03, Mic04a]. The problem decompositions differ in the tackling of unallowed coupling of elements at the bottom level that do not share a parent. In the first decomposition, the shared variable is used as a fixed parameter with its known optimal value, and in the second, the unallowed linking is redirected through the first mutual ancestor. The latter decomposition is the decomposition presented in Section 3.7. Numerical behavior is investigated for each example under the five different ATC coordination algorithms.

Performance indicators

Three different performance indicators are used to compare the five evaluated coordination strategies. The first indicator is the error of the ATC solution with respect to the true optimum. The second indicator is the number of function evaluations required for the ATC process to converge, and the third indicator is the convergence behavior of the algorithm. These indicators are evaluated for each weight-tolerance combination.

The error $\epsilon_k$ of the ATC solution estimate $z_k$ at iteration $k$ is defined by:

$$
\epsilon_k = ||z_k - z^*||_2^2,
$$

(5.1)
with \( \mathbf{z}^* \) the solution to the original problem. The number of function evaluations, \( N_f \) is used as an indicator for the computational cost of the different methods. Computational costs comparison by means of required computation time is possible if all experiments are conducted on identical (single user) computers. However the experiments presented in this report are solved on dual-pentium Linux servers with different capacities and possibly multiple users. In previous work, the number of ATC iterations, \( K \), was used as a performance indicator. However the number of iterations may not provide a clear measure because some iterations may involve more costly optimizations than other iteration steps. Consider e.g. the all-parallel scheme and the sensitivity-based coordination. One iteration for these schemes is defined as solving all ATC sub-problems and computing new targets for all sub-problems. The target update for the sensitivity-based method is however much more computationally expensive since it incorporates solving the optimization problem of the coordination master. To account for these additional costs, the total number of function evaluations of all optimization problems is used as a performance measure.

Another important characteristic of numerical behavior of a method, is the reduction of the error throughout the ATC process, or convergence rate. For successive substitution numerical methods (in this case hierarchical and all-parallel ATC coordination) the convergence rate is expected to be linear. Each iteration decreases the error with a fixed fraction of the previous error. The slope to a semi-logarithmic plot of the error vs the iteration number is constant throughout the process, resulting in a straight line. For a superlinear and quadratic convergence rate, the negative slope will increase during the process, which is expected for the sensitivity-based coordination strategies. The line in the semi-logarithmic plot will drop rapidly as the iterations increase. Coordination strategies can be compared with respect to convergence speed by evaluating these semi-logarithmic plots.

**Convergence criteria**

The ATC coordination algorithms are assumed to have converged if the maximum change between two consecutive global output vector estimates \( \mathbf{Q}^k \) differs within a user defined termination tolerance \( \tau \). For scheme III and the three sensitivity-based coordination methods, the final two iteration results are compared to determine convergence. The ATC algorithms are said to have converged when the following condition is satisfied:

\[
\|\mathbf{Q}^k - \mathbf{Q}^{k-1}\|_\infty < \tau
\]  

(5.2)
Chapter 5. Coordination performance comparison

The infinity norm is used to determine convergence for all ATC algorithms. The infinity vector norm determines the value of the largest element of the vector, whereas e.g. the $l_2$ vector norm equals the root of the sum of squares of all elements. The $l_2$ norm grows larger as the vector contains more elements, whereas the infinity norm only depends on the largest value of the vector. To illustrate, for $a = [1]_{1 \times 2}$ and $b = [1]_{1 \times 10}$, $\|a\|_\infty = \|b\|_\infty = 1$ and $\|a\|_2 = \sqrt{2}$, and $\|b\|_2 = \sqrt{10}$. For ATC, the infinity norm is used in the convergence criterion, since it does not depend on the number of elements in vector $Q$, and the termination criterion (5.2) does not become harder to satisfy for problems, with large global output vectors $q$.

Furthermore, convergence is based only on output vectors, and not on all optimization vectors, including local variables. As demonstrated in Chapter 4, solutions to ATC sub-problems may become non-unique with respect to local variables. With a convergence criterion based on the local variables as well, the ATC algorithm may fail to terminate because non-unique optimal solutions have not converged to a unique solution.

Note that [Tze03, Hul03] use a squared $l_2$ norm and a convergence criterion based on all optimization variables. Fortunately, the test problem(s) considered in their work have sub-problems with unique solutions. In [Mic04a], a convergence criterion is defined that is satisfied when some level of inconsistency is reached. For practical applications however, this level of inconsistency may not be realizable at all, and therefore it is possible that the ATC algorithm reaches a solution with minimal inconsistencies but is not terminated at this solution.

Figure 5.1: Two mirrored semi-parallel solution sequences located in the all-parallel strategy

As discussed in Section 3.8, the all-parallel approach requires a convergence criterion different from (5.2). Due to its two underlying independent semi-parallel sequences,
5.1. Description of the numerical experiments

The convergence criterion should be satisfied when one of the semi-parallel sequences has converged. Consider the sequence illustrated in Figure 5.1. Denote the two semi-parallel schemes solved in parallel A and B. At iteration $k$ of the all-parallel sequence, sub-problems at all odd levels are solved for semi-parallel scheme A, and sub-problems at all even for scheme B. For iteration $k+1$, sub-problems all even levels are solved for scheme A, and sub-problems at all odd levels for scheme B. At iteration $k+2$ the cycle is completed, because all odd levels are again solved for scheme A, and even levels for scheme B. See Figure 5.1 for an illustration of this sequence.

The global output vector for scheme A, $Q_A$, is only partially update after each iteration, since either problems at odd or at even levels are solved. A full update of the global vector of outputs at iteration $k+2$ for scheme A can be obtained by combining updates from iteration $k+1$ and iteration $k+2$:

$$Q_A^{k+2} = \left[ [Q_{\text{odd}}^{k+2}]_{1 \times n_{\text{odd}}} \ , [Q_{\text{even}}^{k+1}]_{1 \times n_{\text{even}}} \right], \quad (5.3)$$

where $n_{\text{odd}}$ and $n_{\text{even}}$ denotes the number of outputs computed at respectively odd and even levels. For scheme B, the vector of output variables $Q_B^{k+2}$ is similarly defined as:

$$Q_B^{k+2} = \left[ [Q_{\text{even}}^{k+2}]_{1 \times n_{\text{odd}}} \ , [Q_{\text{odd}}^{k+1}]_{1 \times n_{\text{even}}} \right], \quad (5.4)$$

Convergence of one of the schemes is based on comparing two consecutive global vectors of scheme A or scheme B. When one of the schemes has converged, the all-parallel algorithm is said to have converged. Note that one iteration of a semi-parallel scheme requires two iterations in the all-parallel strategy, as explained in Section 3.8. Therefore, the all-parallel algorithm is said to have converged when the following condition is satisfied:

$$\min(\|Q_A^k - Q_A^{k-2}\|_{\infty}, \|Q_B^k - Q_B^{k-2}\|_{\infty}) < \tau. \quad (5.5)$$

Depending on which scheme has converged, the optimal solution obtained by ATC is the solution computed with the converged scheme.
Additional settings

In all cases, the Matlab [Mat02] 6.5.0 SQP algorithm fmincon is used to solve the subproblems and coordination problem, optimizer tolerances are set to TolCon = TolFun = TolX = 10^{-10}. Furthermore, constraint and objective gradients and Hessians are computed analytically. Note that the ATC problem suffers from numerical problems when finite difference gradients computed by MatLab are used. For some settings, the ATC process does not converge at all when using these finite difference gradients.

Finally, sensitivity-based coordination with move limits uses move limits equal to 0.1 for all example problems, \( s = 0.1 \). These move limits are chosen arbitrarily based on the (known) optimal solutions to the original problems, which are in the order of 1. If move limits are chosen too small, the coordination method is slowed down by these move limits. If move limits are chosen too large however, the linear approximations will not be accurate and the coordination method may compute unusually large steps. An optimal move limit size is expected, however this has not been investigated in this research.

5.2 Geometric optimization problem 1

First, consider geometric optimization problem (5.6), which is test example A.10 from Appendix A. The problem has seven optimization variables \( z_1, \ldots, z_7 \), two equality constraints \( h_1 \) and \( h_2 \), and two inequality constraints \( g_1 \) and \( g_2 \). The optimal solution to problem (5.6) is \( z^* = [2.15, 2.08, 1.32, 0.76, 1.07, 1.00, 1.47] \). Note that the final error used in Eq. (5.1) is computed with the optimal solution obtained from the AAO implementation of (5.6) in Matlab with TolCon = TolFun = TolX = 10^{-10} used in fmincon.

\[
\begin{align*}
\min_{z_1, \ldots, z_7} & \quad f = z_1^2 + z_2^2, \\
\text{subject to} & \quad g_1 : z_3^2 - z_4^2 + z_5^2 - z_6^2 - z_7^2 \leq 0, \\
& \quad g_2 : z_5^2 + z_6^2 - z_7^2 \leq 0, \\
& \quad h_1 : z_3^2 - z_4^2 - z_5^2 - z_6^2 = 0, \\
& \quad h_2 : z_5^2 - z_4^2 - z_6^2 - z_7^2 = 0, \\
& \quad z_1, z_2, \ldots, z_7 \geq 0.
\end{align*}
\]

Following Appendix A, the problem can be decomposed into a two level hierarchy with one top-level element \( e_0 \) with two children \( e_{11} \) and \( e_{12} \). The objective \( f \) is allocated to \( e_0 \), local constraints \( g_1 \) and \( h_1 \) are assigned to \( e_{11} \), and \( g_2 \) and \( h_2 \) are allocated to \( e_{12} \). Parent \( e_0 \) is linked to its children \( e_{11} \) and \( e_{12} \) through response variables \( R_{01} = r_{11} = z_1 \) and \( R_{02} = r_{12} = z_2 \), respectively. Children \( e_{11} \) and \( e_{12} \) share linking variable \( Y_{01} = Y_{02} = y_{11} = y_{12} = z_5 \). Variables \( z_3 \) and \( z_4 \) are local variables in \( e_{11} \), and \( z_6 \)
5.2. Geometric optimization problem 1

and \( z_7 \) are local in \( e_{12} \). The decomposed problem statement is presented in Section A.10.

The ATC sub-problems are solved for equal weights \( w_{11}^R = w_{12}^R = w_{11}^Y = w_{12}^Y = w \), in accordance with \([Hul03]\). For the experiments presented in this chapter, 15 weights ranging from \( 10^0 \) to \( 10^3 \) are used. Evaluated ATC termination tolerances are: \( \tau_1 = 10^{-4} \), \( \tau_2 = 10^{-6} \), \( \tau_3 = 10^{-8} \), and \( \tau_4 = 10^{-10} \). For each ATC optimization run, the initial infeasible estimate of the optimal solution is \( z_0 = [1]_{1 \times 7} \), and the initial non-realizable vector of global inputs is \( q_0 = [1]_{1 \times 8} \). The error of the solution found by ATC, \( \epsilon \), is computed for each ATC run, as well as the total number of function evaluations \( N_f \). All results are listed in Section G.1.

**Final error**

Figure 5.2 depicts the error of the ATC solution with respect to the true solution, for several weights and the five coordination strategies. Each line in the figure represents the results of the ATC process for one specific tolerance. The error characteristics of scheme III (Figure 5.2(a)) are similar to those presented in \([Hul03]\). Figures 5.2(a), 5.2(b), 5.2(c), 5.2(d), and 5.2(e) depict the error behavior for respectively all-parallel coordination, non-safeguarded sensitivity-based coordination, sensitivity-based coordination with move limits, and the hybrid coordination strategy.

When comparing all five error characteristics, the non-safeguarded sensitivity-based coordination method outperforms all other strategies. Strangely, the error of the solution for both sensitivity-based methods is insensitive with respect to the termination tolerance. Experiments with even looser tolerances show that the error only becomes sensitive for very loose tolerances (\( \tau \approx 10^{-1} \)). The two safeguarded methods were designed to avoid numerical problems, but for all weights, the sensitivity-based coordination does not suffer from numerical problems, making the safeguards superfluous.

The sensitivity-based method with move limits shows a similar high performance, and the error is nearly insensitive to the termination tolerance as well. Both zero-order approaches are comparable with respect to their error behavior. The hybrid approach shows behavior similar to the two zero-order approaches.

For all five strategies, a downward slope of the error for small weights can be identified. This downward slope indicates that for small weights \( w \), the error \( \epsilon \) decreases when \( w \) increases. This error behavior is caused by the weighted penalty relaxation of the all-at-once problem. Inconsistencies between variable copies are allowed after relaxation, whereas inconsistencies are zero in the original problem. As discussed in Section 3.2, the inconsistencies can become relatively large for small weights, because they are not heavily penalized; the inconsistencies become smaller when weights are increased. The most important effect of relaxation is the alteration of the solution to
the problem. The solution to the relaxed problem is not equal to the true solution to the original problem. Since ATC is a method for solving the relaxed problem, the same trends are observed; as weights $w$ increase, the inconsistencies decrease, and therefore the error $\epsilon$ decreases.

To remove the relaxation behavior from the error characteristics of Figure 5.2, the error $\epsilon'_k$ is defined as the error between the solution ATC finds, and the solution to the relaxed problem:

$$
\epsilon'_k = \| z_{rel}^* - z^k \|_2^2,
$$

(5.7)

where $z_{rel}^*$ denotes the solution to the relaxed problem. Figure 5.3 depicts the error characteristics for all five coordination strategies, for the alternative error definition of (5.7). Note that the additional runs for the two sensitivity-based methods for very loose tolerances show an influence of the termination tolerance on the error characteristics.
For small weights, the solution ATC finds is very close to the relaxed solution, but as weights increase, so does the error. Apparently, all five ATC coordination strategies have difficulties for finding the relaxed solution for larger weights. This behavior is caused by the decomposition. Because of the decomposed formulation, each element aims at finding outputs close to the inputs it has received. The change between two consecutively computed outputs can be seen as a step taken by the algorithm toward the relaxed solution. The larger the weights, the closer the computed outputs will be to the received inputs, and the smaller the steps are. When the step size becomes smaller than the termination tolerance $\tau$, the ATC coordination process is terminated. Apparently, large weights slow down the ATC solution algorithms. For larger weights, step sizes are smaller, and the coordination process it terminated earlier, resulting in a larger solution error.

If weight are increased beyond a certain value, the error becomes constant for all five strategies. The constant value of the error depends on the selection coordination strategy and the initial estimates $z_0$ and $q_0$. For each coordination strategy, the weight for which the error becomes constant, depends on the termination tolerance $\tau$. For these large weights, inconsistencies are heavily penalized and a consistent solution is found rapidly. However this consistent solution will not be optimal. Because weights are very
large, the first step computed with this consistent solution is smaller than the termination tolerance, and the coordination process is terminated. Therefore, the error of the solution found with large weights is determined by the first consistent design found by a coordination strategy.

The use of the alternative error definition provides more insight in the ATC coordination process. However the true error, as defined in (5.1), will be used because the goal of the ATC decomposition method is to find an optimal solution for the original all-at-once problem, not the relaxed problem.

Furthermore, Figure 5.2(c) suggested that the error obtained with sensitivity-based methods does not depend on the termination tolerance $\tau$, however Figure 5.3(c) shows that the solution error does depend on tolerance $\tau$. Two additional curves for $\tau = 10^{-2}$ and $\tau = 10^0$ are displayed for the sensitivity-based algorithms. For all five strategies, the solution ATC finds becomes more accurate when the termination tolerance is tightened.

**Computational costs**

As a second measure of performance, the number of function evaluations required for an ATC run is compared for all five coordination schemes. Figure 5.4 depicts the number of function evaluations, $N_f$, required for solving the ATC process for a range of weights and 4 different termination tolerances.

Comparison of all five strategies shows a large reduction of computational costs when applying the two sensitivity-based coordination strategies instead of the traditional hierarchical scheme III. The all-parallel and hybrid methods however required a similar or larger computational effort for convergence, when compared to the scheme III.

A general trend for small weights can be observed for all five coordination strategies: the number of function evaluations $N_f$ increases as the weight $w$ increases. For increasing weights, the relaxed all-at-once problem becomes harder to solve, which consequently also holds for the ATC sub-problems. When problems are harder to solve, more function evaluations are required for convergence of the solution algorithm. Moreover, large weights allow only small steps. To reach the relaxed solutions, a greater number of steps is required for larger weights, which increases the computational costs for solving the decomposed problem.

For zero-order coordination strategies and the hybrid approach, a sudden drop is observed for large weights. The weight for which this happens is equal to the weight for which the ATC process is stopped as soon as a consistent but non-optimal solution is found. The weight for which the number of function evaluations drops depends on the termination tolerance $\tau$. Note that numerical behavior of the hybrid method is unpredictable for weights after the drop-off point.
5.2. Geometric optimization problem 1

The rate of error decrease during the solution process is compared for all five schemes as a third performance indicator. The rate of decrease is monitored for one single solution run with specific settings. Figure 5.5(a) shows the error $\varepsilon_k$ at iteration $k$ for all five coordination strategies. For these particular runs, a weight of $w = 20$ is used, and a termination tolerance of $\tau = 10^{-10}$.

Obviously, the two sensitivity-based coordination methods outperform the zero-order schemes as well as the hybrid approach. For both sensitivity-based methods, the error decreases rapidly in a few iterations. For the zero-order and hybrid strategies, an initial drop is observed. During this initial drop, the zero-order and hybrid methods rapidly move towards a consistent solution. After this drop, the ATC process slowly moves towards the final solution.

As expected, the slope of the curves for zero-order and hybrid coordination is constant for the intermediate part of the process. This constant slope implies that the zero-order and hybrid method are linearly convergent algorithms. Quadratic behavior of the first-order coordination method can be observed in the first 40 steps. The convergence
Chapter 5. Coordination performance comparison

Figure 5.5: Convergence behavior for the first example problem

The convergence behavior for the first 40 iterations is depicted in Figure 5.5(b). For the sensitivity-based method with move limits, the shape of the error reduction curve is quadratic, implying quadratic convergence for the first 15 iterations. For the non-safeguarded method this quadratic shape is only observable for the first 5 steps, after which the error becomes constant.

The all-parallel and the hybrid coordination have slopes only half as steep as the hierarchical scheme III. As explained in Section 3.8, the all-parallel strategy actually solves two semi-parallel schemes in parallel. For two-level problems, the semi-parallel schemes are identical to schemes III, and therefore the all-parallel strategy solves two schemes III in parallel for this two-level problem. Because the all-parallel strategy requires twice as many iterations as the semi-parallel strategy (see Section 3.8), the slope of the error reduction curve is half as steep for the all-parallel approach. Since the hybrid approach switches to the all-parallel approach for realizable inputs, it has a similar slope.

As the number of iterations increases, a decrease in convergence rate is observed for all five strategies. This decrease of convergence rate is caused by the choice of error definition which in this case is defined as the difference between the true solution and the ATC estimate (5.1). If the error was defined to be the difference between the relaxed solution and the ATC estimate, following (5.7), the convergence rate would not deteriorate as the number of iterations increases. The linear convergence behavior of this altered error definition is depicted in Figure 5.6.

The convergence rate for the method with move limits is quadratic throughout the solution run. For the non-safeguarded sensitivity-based algorithm however, the con-
5.2. Geometric optimization problem 1

Figure 5.6: Convergence behavior for alternative error definition

Convergence rate becomes linear after 5 iterations. At iteration 5, a consistent solution is found. As mentioned in Section 4.4, the sensitivity-based methods switch to a zero-order all-parallel scheme with a linear convergence rate. This behavior can be observed for iterations 5-38 of the sensitivity-based coordination strategy.

Comparison of coordination strategies

When comparing the five coordination strategies, the non-safeguarded sensitivity-based method outperforms all other strategies for all performance indicators. The method gives smaller solution errors for all ATC termination tolerances $\tau$, and it requires far less computational costs due to its observed quadratic convergence behavior. The results motivate to simply use a loose termination tolerance $\tau$ and relatively large weights $w$ for optimal performance. For these settings, an accurate solution is found at low computational costs.

The two safeguarded methods were designed to account for possible numerical problems of the sensitivity-based method. For this example problem, these numerical problems were not observed, and the safeguards only deteriorate the performance of the coordination algorithm. For other optimization problems, the safeguards may be necessary for the method to converge.

The all-parallel zero-order coordination does not outperform the hierarchical scheme III, but this was not expected. However, the all-parallel approach does provide a pos-
sibility for non-hierarchical ATC. For non-hierarchical ATC, linking targets between bottom-level problems possibly do not have to be coordinated through the parent anymore, but can be exchanged directly between the bottom-level problems. The direct communication of targets decrease the dimensionality of the problem and the number of sub-problems involved in the linking between two non-hierarchically coupled elements.

5.3 Geometric optimization problem 2, partition 1

Consider the example geometric optimization problem 2 (5.8) from Section A.14 of Appendix A. The problem has fourteen optimization variables \( z_1, \ldots, z_{14} \), four equality constraints \( h_1, \ldots, h_4 \), and six inequality constraints \( g_1, \ldots, g_6 \). The optimal solution to problem (5.8) is \( z^* = [2.84, 3.09, 2.36, 0.76, 0.87, 2.81, 0.94, 0.97, 0.86, 0.80, 1.30, 0.84, 1.76, 1.55] \). Again, the true solution used in 5.1 is obtained with \( \text{fmincon} \) and \( \text{TolCon} = \text{TolCon} = \text{TolX} = 10^{-10} \).

\[
\begin{align*}
\min_{z_1, z_2, \ldots, z_{14}} & \quad z_1^2 + z_2^2 \\
\text{subject to} & \quad g_1 : z_3^2 + z_4^2 - z_5^2 \leq 0, \quad g_2 : z_6^2 + z_7^2 - z_8^2 \leq 0, \\
& \quad g_3 : z_8^2 + z_9^2 - z_{11}^2 \leq 0, \quad g_4 : z_8^2 + z_{10}^2 - z_{11}^2 \leq 0, \\
& \quad g_5 : z_{11}^2 + z_{12}^2 - z_{13}^2 \leq 0, \quad g_6 : z_{11}^2 + z_{12}^2 - z_{14}^2 \leq 0, \\
& \quad h_1 : z_1^2 - z_3^2 - z_4^2 - z_5^2 = 0, \\
& \quad h_2 : z_2^2 - z_5^2 - z_6^2 - z_7^2 = 0, \\
& \quad h_3 : z_3^2 - z_6^2 - z_9^2 - z_{10}^2 - z_{11}^2 = 0, \\
& \quad h_4 : z_6^2 - z_{11}^2 - z_{12}^2 - z_{13}^2 - z_{14}^2 = 0, \\
& \quad z_1, z_2, \ldots, z_{14} \geq 0.
\end{align*}
\]

Several partitions of this problem are possible. Kim presented a two-level decomposition in [Kim01a]. Etman et al. [Etm02, Etm04], Hulshof [Hul03] and Michalek [Mic04a] present a three-level decomposition. In this section, the same three level decomposition is used to analyze the performance of the five coordination strategies. In Section 5.4 an alternative three-level decomposition is analyzed.

The three-level decomposition is as follows. The partition has one top-level element \( e_0 \), which has two children \( e_{11} \) and \( e_{12} \). Each of these elements has one child, \( e_{21} \) and \( e_{22} \), respectively. Variable \( z_1 \) is a response target variable shared by element \( e_0 \) and its child \( e_{11} \). Variable \( z_2 \) is the response target variable between \( e_0 \) and \( e_{12} \). Element
5.3. Geometric optimization problem 2, partition 1

e_{11} shares response target \( z_3 \) with element \( e_{21} \), and \( z_6 \) is shared by \( e_{12} \) and \( e_{22} \). Furthermore, elements \( e_{11} \) and \( e_{12} \) share linking variable \( z_5 \). Elements \( e_{21} \) and \( e_{22} \) share variable \( z_{11} \), but because these elements do not have the same parent, this variable is treated as a fixed parameter with the value of the known optimal value \( z_{11} = p = 1.30 \). ATC in its original form only allows linking variables shared between children of the same parent. The objective function \( f \) is allocated to element \( e_0 \), and inequality constraints \( g_1, g_2, g_3, g_4, g_5, g_6 \) are allocated to \( e_{11}, e_{12}, e_{21}, e_{22}, e_{22} \) respectively. Equality constraints \( h_1, h_2, h_3, h_4 \) are used to define the response analysis functions of elements \( e_{11}, e_{12}, e_{21}, e_{22} \) respectively. The decomposition and decomposed problem statement are presented in Section A.14.

The ATC sub-problems are again solved for equal weights \( w_{R_{11}} = w_{R_{12}} = w_{R_{21}} = w_{R_{22}} = w_Y = w_{Y_{11}} = w_{Y_{12}} = w \). For each ATC optimization run, the initial estimate of the optimal solution is \( z_0 = [1]_{1 \times 14} \), which is infeasible, and non-realizable input vector \( q_0 = [1]_{1 \times 12} \). The error of the solution found by ATC, \( \epsilon \), is computed for each ATC run, as well as the total number of function evaluations \( N_f \). All results are listed in Section G.2.

Final error

Figure 5.7 depicts the error of the ATC solution with respect to the true solution, for several weights and the five coordination strategies. Again, the 15 selected weights \( w \) range from \( 10^0 \) to \( 10^3 \), and termination tolerances are \( \tau_1 = 10^{-4}, \tau_2 = 10^{-6}, \tau_3 = 10^{-8}, \) and \( \tau_4 = 10^{-10} \). Each line in the figure represents the results of the ATC process for one specific tolerance. The error characteristics of scheme III (Figure 5.7(a)) are identical to those presented in [Hul03]. Figures 5.7(a), 5.7(b), 5.7(c), 5.7(d), and 5.7(e) depict the error behavior for respectively all-parallel coordination, non-safeguarded sensitivity-based coordination, sensitivity-based coordination with move limits, and the hybrid coordination strategy.

The error characteristics for the ATC process for this example problem are comparable to results obtained with the previous example problem. Again, the non-safeguarded sensitivity-based coordination outperforms all other strategies. Furthermore, the final error is insensitive with respect to termination tolerances \( \tau \) for both sensitivity-based methods, which was also observed for the previous example. Numerical problems for the sensitivity-based method for \( w = 10^3 \) are observed. For this large weight, the non-safeguarded ATC algorithm did not converge for any of the selected termination tolerance.

Again, both zero-order approaches are comparable with respect to their error behavior. The hybrid approach shows behavior similar to both the zero-order approaches.

The results presented in this report differ from results on ATC implementations of this decomposition presented by Hulshof [Hul03]. In the implementation by Hulshof, the coordination algorithm was programmed in \( \chi \) [Kle01], a concurrent programming
Chapter 5. Coordination performance comparison

Figure 5.7: Error of the solution for the selected coordination strategies

language. MatLab [Mat02] was used to solve the optimization sub-problems, and Python [Hof01] was used for communicating data between $\chi$ and MatLab. The implementation of Hulshof required truncation of communicated data for the ATC process to behave predictable. The Matlab implementation of Appendix H however, reaches identical results without truncation. Why the truncation is required for the $\chi$-Python-Matlab implementation still remains to be investigated.

Computational costs

Figure 5.8 depicts the number of function evaluations, $N_f$, required for solving the ATC process. Again, a large reduction in computational costs is observed for the two sensitivity-based coordination strategies. Note that the all-parallel and the nested scheme III require a similar amount of function evaluations before convergence is reached. In the previous two-level example, scheme III outperformed the all-parallel scheme, because the all-parallel scheme solved two schemes III at the same time. But for this three-level problem, this is no longer the case. The nested bottom-up sequence for scheme III requires the bottom two levels to converge before the top-level problem can be solved. The all-parallel coordination however solves the top-level problem every time new responses from intermediate levels are available. For the all-parallel scheme,
the top-level problem is solved more often than for scheme III. Because the top-level problem is solved more often, its targets are cascaded down the hierarchy more often, allowing all sub-problems to adjust to these new top-level targets. On the other hand, two semi-parallel schemes are still solved at the same time for all-parallel coordination, increasing the required computational costs. The frequent communication of targets however reduces the computational cost to a level equal to that of the nested scheme III strategy. These findings imply that a semi-parallel scheme has even lower computational costs, and therefore may outperform scheme III with respect to computational cost.

Figure 5.8: Number of function evaluations for the selected coordination strategies

**Error reduction during ATC process**

The rate of error decrease during the solution process is compared for all five schemes as a third performance indicator. The rate of decrease is monitored for one single solution run with $w = 20$ and $\tau = 10^{-10}$. Figure 5.9(a) shows the error $\epsilon_k$ at iteration $k$ for all five coordination strategies.

Again, the two sensitivity-based coordination methods outperform the zero-order schemes as well as the hybrid approach. For the zero-order and hybrid strategies, the initial drop
Chapter 5. Coordination performance comparison

is large. Apparently, an initial consistent solution is found very close to the optimal solution, resulting in the large initial drop. From this initial consistent estimate, the ATC process moves towards the final solution with a linear convergence rate again.

As expected, the zero-order methods and hybrid coordination are linearly convergent. For the second example problem, the slope for hierarchical scheme III is twice as steep compared to the all-parallel and hybrid approaches, which was also observed for the first example problem. Quadratic behavior of the first-order coordination methods can again be observed in the first iterations. The convergence behavior for the first 50 iterations is depicted in Figure 5.9(b). For the sensitivity-based method with move limits, the shape of the error curve is quadratic, implying quadratic convergence for the first 50 iterations. For the non-safeguarded method this quadratic shape is only observable for the first 5 steps, after which the convergence rate is no longer quadratic because a consistent solution is found.

At the end of the solution process, a decrease in convergence rate is observed for all five processes. Again, this decrease of convergence rate is caused by the choice of error definition. When the error $\epsilon_k$ was defined to be the difference between the relaxed solution and the ATC estimate, the convergence rate does not deteriorate at the end of the process.
5.4. Geometric optimization problem 2, partition 2

Comparison of coordination strategies

When comparing the five coordination strategies for this example problem, the non-safeguarded sensitivity-based method again outperforms all other strategies for all performance indicators. The solution errors are smaller for all ATC termination tolerances $\tau$, and the process requires far less computational costs due to its observed quadratic convergence behavior. The results from this example too motivate to simply use the sensitivity-based method with relatively large weights and a loose termination tolerance.

An interesting observation is that the computational cost for the all-parallel approach was similar to the cost for scheme III. The low computational cost for the all-parallel method motivates to use a semi-parallel strategy. The semi-parallel scheme is expected to require half of the computational cost of the all-parallel approach, because the all-parallel algorithm solves two semi-parallel schemes at once. By only solving one semi-parallel scheme, the computational cost may be cut in half with respect to the all-parallel approach, making the semi-parallel scheme computationally less expensive compared to scheme III. Further investigations however are required to compare scheme III to the semi-parallel scheme.

The two safeguarded methods were designed to account for possible numerical problems of the sensitivity-based method. For this second convex example problem, again these numerical problems were not observed, and the safeguards only deteriorate all three performance indicators of the coordination algorithms.

5.4 Geometric optimization problem 2, partition 2

Again, consider the geometric optimization problem 2 (5.8). An alternative to the three-level partition of [Etm02, Etm04, Hul03, Mic04a] is possible. This alternative partition is presented as an example in Section 3.7, and is enclosed in Appendix A.

This alternative three-level decomposition again partitions the problem into a three-level hierarchy. The partition is identical to the one presented in [Etm02, Etm04, Hul03, Mic04a], however it differs in the tackling of the prohibited linking of the two bottom level elements $e_{21}$ and $e_{22}$. Instead of fixing the linked variable at its optimal value, it is coordination by the top-level element $e_0$. This coordination is facilitated by the introduction of two variables and two equality constraints. Each of the elements $e_{11}$ and $e_{12}$ receives one of the two support variables, and the linking is moved to the mid-level sub-problems. The decomposition and decomposed problem statement is presented in Section A.14.

Again, all weights are set equal: $w^{R}_{11} = w^{R}_{12} = w^{R}_{21} = w^{R}_{22} = w^{Y}_{11} = w^{Y}_{12} = w$. For each ATC optimization run, the initial estimate of the optimal solution is $z_0 = [1]_{1 \times 16}$,
which is infeasible, and $q_0 = [1]_{1\times 20}$ which is non-realizable. The error of the solution found by ATC, $\epsilon$, is computed for each ATC run, as well as the total number of function evaluations $N_f$. All results are listed in Section G.3.

**Final error**

Figure 5.10 depicts the error of the ATC solution with respect to the true solution, for several weights and the five coordination strategies. Again, the weights $w$ range from $10^0$ to $10^3$, and the termination tolerances are $\tau_1 = 10^{-4}$, $\tau_2 = 10^{-6}$, $\tau_3 = 10^{-8}$, and $\tau_4 = 10^{-10}$.

![Figure 5.10: Error of the solution for the selected coordination strategies](image)

The characteristics of the final error are very similar for all five coordination strategies, in contrast to the previous two examples. None of the methods clearly outperforms the others for this example problem. The final error depends on the termination tolerance $\tau$ for all coordination strategies, also for the two sensitivity-based methods.

Furthermore, for larger weights, scheme III and the all-parallel coordination suffer from
numerical difficulties, resulting in ATC processes that did not terminate. For \( w = 10^3 \),
the sensitivity-based coordination did not terminate as well, and the sensitivity-based
method with move limits shows unpredictable large errors for this specific weight. For
the sensitivity-based coordination methods, these numerical problems were observed in
the previous example as well. However, for the zero-order methods, this is the first
problem for which they occur. It is possible that these numerical difficulties occur be-
cause of the size of the problem, or the large number of input variables (20), but this
remains to be investigated. Both the safeguarded methods however do converge for
large weights, where the other strategies do not terminate at all.

Computational costs

Figure 5.11 depicts the number of function evaluations, \( N_f \), required for solving the
ATC process for a range of weights and 4 different termination tolerances, \( \tau_1 = 10^{-4} \),
\( \tau_2 = 10^{-6} \), \( \tau_3 = 10^{-8} \), and \( \tau_4 = 10^{-10} \).

Figure 5.11: Number of function evaluations for the selected coordination strategies
The five strategies only differ for small weights, however in the range \( 20 < w < 60 \)
where the error is minimal, all five strategies show similar characteristics. The two sensitivity-based methods no longer outperform the remaining three strategies. None of the coordination strategies outperforms the other with respect to computational effort.

**Error reduction during ATC process**

The rate of error decrease during the solution process is compared for all five schemes as a third performance indicator. The rate of decrease is monitored for one single solution run with \( w = 20 \) and \( \tau = 10^{-10} \). Figure 5.12(a) shows the error \( \epsilon_k \) at iteration \( k \) for all five coordination strategies.

The sensitivity-based coordination methods again do not outperform scheme III. The convergence rate of these three methods are very similar. The expected quadratic convergence of the sensitivity-based methods is not observed in the first iterations (See Figure 5.12(b) for the convergence behavior for the first 40 iterations). Only for the method with move limits, an initial quadratic convergence rate is observed for the first 20 iterations, but after that, the rate decreases to linear.

As explained in Section 4.4, the sensitivity-based method reduces to the zero-order all-parallel method when all inputs are realizable. Apparently, the sensitivity-based ATC solution process finds a realizable input vector within the first few iterations, after which it switches to the zero-order all-parallel scheme, with a linear convergence rate. However, the error for the sensitivity-based method decreases twice as fast as the error.
for the all-parallel scheme. Apparently, the sensitivity-based approach accelerates the all-parallel scheme. This acceleration is probably caused by the fact that some sub-problems compute non-realizable outputs for other sub-problems, when given realizable inputs. With these non-realizable inputs, sensitivities can again be computed, and the sensitivity-based coordination can be used again. In turn, the first-order coordination computes realizable inputs, forcing the coordination strategy to switch to a zero-order approach for the next iteration. This switch between zero and first-order coordination increases the convergence rate of the sensitivity-based methods compared to pure all-parallel coordination.

Furthermore, the slope for the all-parallel and the hybrid strategies are again half as steep as the slope for scheme III, similar to the results from the other two test examples.

From Figure 5.12(b), a typical result for the all-parallel strategy can be observed. The error shows chattering behavior, caused by the solution of two semi-parallel schemes at once. For two consecutive iterations, the error is computed for two different semi-parallel schemes. Because it takes two all-parallel iterations to solve one iteration of a semi-parallel scheme, all errors for odd iterations belong to one semi-parallel scheme, and all errors for even iterations belong to the other semi-parallel scheme. It is very likely that these two separate schemes have different intermediate solutions, with different errors. Due to this two-sided behavior of the all-parallel schemes, the error shows this chattering behavior.

Comparison of coordination strategies

For this example, none of the coordination strategies clearly outperforms the others with respect to final error and required computational costs. However, numerical problems are more frequent for the zero-order strategies, whereas the first-order and hybrid methods appear to have less numerical difficulties. What causes these numerical difficulties for the zero-order schemes however remains to be investigated.

With respect to the convergence rate, the sensitivity-based methods and scheme III outperform the all-parallel and the hybrid approaches. The required number of iterations for the first-order methods and scheme III is only half the number required for the all-parallel and the hybrid method. This reduction of required iterations however does not result in a reduction of the required computational costs, and therefore may depend on the definition of an ”iteration”.
5.5 Overall performance comparison

Three example problems are used to compare the performance of the five coordination strategies. For the first two problems, the sensitivity-based coordination strategies clearly outperformed the remaining zero-order and hybrid schemes. For the third problem however, none of the coordination methods outperforms the others. For these three examples, the sensitivity-based methods have better or at least equal performance, compared to the state-of-the-art coordination strategy for ATC, scheme III. The conditions for which performance is better however are not clearly identified.

A general trend however can be observed: for non-realizable input vectors, the convergence rate of the sensitivity-based methods is quadratic. For realizable input vectors however, the convergence rate decreases to linear. This linear convergence rate is equal to the convergence rate of the state-of-the-art strategy, scheme III.

5.6 Discussion

Differences between second and third example problem

The results for the second and third example show very different behavior, even though both problems have a three-level decomposition of the same problem. The difference in the partitioning approach is the coordination of the unallowed linking between the bottom level problems. In the second example problem, the linking is tackled by treating the linking variable as a fixed parameter with the known optimal value of the linking variable. The third example coordinates the linking variable through the top-level element instead of fixing it at its optimal value.

By fixing the linking variable for the second example problem, the two bottom level problems have a realizable input domain that consists of a single point, the optimal solution. As discussed in Section 4.4, the sensitivity-based coordination method reduces to an all-parallel coordination strategy when the global vector of inputs is realizable. The switch from sensitivity-based to all-parallel coordination however does not occur before the optimal solution is found. The reason this switch does not occur is that the realizable sub-spaces for the bottom-level problems are single points, equal to the optimal solution. Because of this property, the sensitivity-based method has a quadratic convergence rate until the solution is found.

For the third example however, both bottom-level problems have a larger realizable input domain. For this larger realizable domain, the sensitivity-based ATC solution process finds a realizable input vector within the first few iterations, after which it switches to the zero-order all-parallel scheme, with a linear convergence rate.
Influence of weights on the numerical behavior of ATC

The inexact penalty relaxation of the coupling constraints has major effects on the performance of the ATC method. As penalty weights are increased, the solution to the relaxed problem becomes a more accurate estimate of the solution to the original problem. However, the solution to the relaxed problem is only equal to the solution of the original all-at-once problem for infinite weights. As mentioned before, ATC is an algorithm that searches for the solution to the relaxed problem. Therefore, for finite weights, the solution to the ATC problem is not equal to the solution of the original problem.

The error of the ATC solution can be decreased by increasing the weights, however if weights become too large, the ATC algorithm suffers from numerical problems. Due to these numerical problems, the algorithm is terminated before it reaches the relaxed solution, hence a sub-optimal solution is obtained. Furthermore, the step-sizes of the ATC algorithms are reduced when weights are increased. Because step-sizes are smaller, the ATC process requires more of these small steps to reach the relaxed solution, which was observed for all three example problems. Therefore, the convergence speed of the ATC algorithm becomes smaller for larger weights.

The accuracy of the solution estimates found with ATC motivates to use large weights. However, the numerical behavior of the ATC algorithm motivates to use small weights. These conflicting trends impose a limit to the solution error obtained with ATC. For all zero-order coordination strategies, this error limit depends on the termination tolerance of the ATC process, whereas the first two examples do not show this dependency for the sensitivity-based coordination strategies. For the third example however, this dependency is observed for all strategies.

The great influence of weights on the ATC process was also observed by Michalek and Papalambros [Mic04a], who developed a method for gradually increasing the weights as the ATC iterations increase. The weights are updated based on the size of inconsistencies between targets and responses. The method showed a reduction of computational costs compared to ATC without the updating mechanism.

The error limit observed in the experiments may be reduced by using an exact penalty relaxation. For an exact penalty relaxation, the relaxed solutions are equal to the true solution of the original problem for finite weights, and possibly the ATC algorithms are able to reach this exact relaxed solution. However, the numerical behavior of ATC coordination algorithms under such an exact penalty relaxation remains to be investigated.
Chapter 5. Coordination performance comparison
Chapter 6

Conclusions and recommendations

Analytical target cascading (ATC) is a decomposition methodology for the design of large and complex engineering systems [Kim01a, Kim03]. The method defines how overall design targets are propagated throughout the elements making up the hierarchy of the design problem. ATC has been applied to a number of practical cases [Kim01b, Mic01, Kim02, Kok02, Kim03], and the convergence properties have been theoretically investigated [Mic03]. The numerical behavior of the method has also been investigated [Tze03, Hul03, Mic04b], showing that the ATC solution process is a computationally expensive process. A coordination strategy defines the solution algorithm to solve the decomposed problem. At present, ATC coordination strategies only exchange target and response values between sub-problems. With post-optimality analysis, sensitivities of responses with respect to targets can be obtained without much effort. By using these sensitivities in the coordination strategy for ATC, ‘smarter’ targets may be computed, and the (rate of) convergence may be improved. The objective of this research is to investigate the possibility of using sensitivities of responses with respect to these targets in the coordination process of ATC.

6.1 Review and classification of ATC

In order to meet the objective, the ATC decomposition methodology is first reviewed and classified in the field of both mathematical and engineering-based decomposition methods. Earlier work on ATC did not present a method for partitioning a large all-at-once (AAO) optimization problem into a hierarchy of multiple connected ATC sub-problems, and neither describes the problem structure (FDT) it has to adhere to. This report however presents a method for partitioning AAO problems for the ATC methodology. By comparing the partitioning method to classical decomposition methods, ATC can be classified as a dual-feasible decomposition method. Previously, ATC had only partially been recognized as a dual-feasible method for decomposition. With the partitioning method however, the full dual-feasible framework is exposed, providing
a basis for further improvement of the method.

The ATC dual-feasible decomposition method comprises of four steps, which are all formalized in this report. Previous work on ATC indicated a need for more test problems for ATC. The decomposition process is applied to a number of optimization test problems from literature [Hoc81, Sch87, Flo90], providing a large number of test problems formulated as ATC decomposed problems. The problems and their decompositions are presented in Appendix A.

The four steps of the ATC decomposition method are: (1) problem structure identification and manipulation, (2) variable partitioning, (3) sub-problem formulation, and (4) solution coordination. These four steps are briefly discussed below.

**Problem structure identification and manipulation**

The ATC decomposition method is only applicable to AAO optimization problems with a hierarchical structure linked through a set of shared variables. Constraints may only be local to the elements of the structure. The hierarchical structure for ATC can be obtained by partitioning the AAO problem using an object or aspect-based approach, or an automated model-based partitioning method based on matrix representations of the AAO optimization problem structure [Wag93, Mic97, Kri97].

ATC introduces copies of the shared variables for each sub-problem. The linking of sub-problems is accounted for by defining coupling constraints forcing the copies to match. For decomposition, these coupling constraints are relaxed using an external penalty function, which is added to the objective of the AAO problem. For each coupling constraint, one penalty term is introduced. The weights associated with each term can be seen as the dual variables of the problem.

The relaxed ATC decomposed problem has a solution different from the solution to the original AAO problem. The error between both solutions only becomes zero when the weights approach infinity. Errors are large for small weights, but become smaller for larger weights. In practice, setting very large weights introduces numerical difficulties in the ATC solution algorithms, and therefore the accuracy of the solution ATC finds is limited.

**Variable allocation**

The decomposition of the relaxed problem is obtained by defining optimization sub-problems that are only solved for their own set of local variables and linking variable copies. During the solution of a sub-problem, all variables associated with other sub-problems are fixed.
6.2. Sensitivity-based coordination

Master and sub-problem formulation

Because sub-problems are solved for their specific set of variables, only the terms of the objective depending on this set are included in a sub-problems’ objective. Similarly, only local constraints associated with the sub-problem variables have to be included in the sub-problem formulation.

Where sub-problems for ATC are defined in earlier work, the feasible and dual master problems have received very little attention. Currently, the feasible master problem is only used to update the values of responses and targets computed in the individual elements. One possible dual master problem has been proposed and acts as a mechanism for slowly increasing the weights throughout the ATC process [Mic04a]. As a result, the computation effort required to solve the problem has been reduced.

Solution coordination

A coordination strategy is an algorithm that defines how the decomposed relaxed problem is solved. The research presented in this report only considers the feasible part of the decomposition method. The feasible part of the coordination strategy currently consists of defining a sequence in which sub-problems are solved. The current solution sequences are based on the underlying problem hierarchy.

Two alternative feasible coordination strategies are presented in this report. The first embodies a semi-parallel sequence of solving sub-problems. This hierarchy-based strategy iterates between solving sub-problems at all even levels and all odd levels, whereas previous coordination strategies only solve sub-problems at one specific level. An all-parallel coordination solution algorithm has also been presented. This hierarchy independent strategy simply solves all sub-problems at once, and then updates all targets and responses for all sub-problems.

6.2 Sensitivity-based coordination

Sensitivities of responses with respect to targets can be computed with post-optimality techniques. A coordination method that uses these sensitivities has been proposed.

Concept of the method

The method follows from an alternative point of view on the solution to the decomposed problem. The decomposed problem is regarded as a hierarchy of connected elements, and each element is seen as a black-box function that computes outputs from inputs. The outputs of an element are the computed targets and responses, and inputs are targets and responses received from other elements. Two relations between inputs and outputs exist: the black-box input-output functions defining how outputs of an element depend on the inputs that the element receives, and a hierarchy relation defining what
output is connected to what input. Note that the hierarchy relation is not restricted to a pure hierarchical problem structure, but also allows non-hierarchical problems.

These two relations can be combined to define a vector function that describes how the inconsistencies between sub-problems depend on a global vector of inputs. Finding a consistent solution to ATC can be reformulated to finding a zero of this inconsistency function. However, it is possible that the solution to the decomposed problem is not consistent. This motivates not to search for a zero, but to minimize the inconsistencies.

The all-parallel coordination strategy presented in this report actually is a zero-order iterative algorithm for minimizing the inconsistencies. In the algorithm, zero-order Taylor approximations of the inconsistency functions are used to compute new targets, which in the field of fixed-point algorithms is called successive substitution. The function values of the input-output relations are computed by the sub-problems, and the coordination strategy passes the values to the other sub-problems in the hierarchy.

A first order solution algorithm is also proposed. The algorithm is motivated by Newton’s gradient-based method for finding a zero in multiple dimensions. A feasible master optimization problem is defined to minimize inconsistencies, by using first-order Taylor approximations of the inconsistency functions. The gradients of the inconsistency functions used in this algorithm, can be computed from the sensitivities of a sub-problem’s outputs with respect to its inputs.

In theory, zero-order successive substitution algorithms have a linear convergence rate, whereas the first-order algorithms converge quadratically. Therefore, the first-order sensitivity-based method for ATC is expected to outperform the slower zero-order methods, and to reduce the computational effort required to solve the decomposed problem. First-order methods however often require safeguards for stability. ATC sub-problem behavior is analyzed to find appropriate safeguards.

**Sensitivities for ATC**

Sensitivities of outputs with respect to inputs can be computed with post-optimality techniques. Sensitivity properties have been analyzed for ATC optimization sub-problems. For realizable inputs, the outputs are equal to the inputs, and no constraints are active. Because inputs can be matched, sensitivities are equal to 1, and since no constraints are active the sensitivity information suggests unboundedness of outputs. For many sub-problems however, the outputs are subjected to constraints. This constraint activity cannot be predicted with only sensitivities of outputs, and therefore unboundedness of outputs is implied for sensitivities computed from realizable inputs. For non-realizable inputs however, inputs cannot be matched by outputs, and sensitivities are not equal to 1. One or more constraint are active, and the output sensitivities can be used to predict the location of the realizable domain of inputs.
To account for the implied unboundedness of outputs, two safeguards are proposed. The first safeguarded method imposes move limits on the inputs computed in the master optimization problem. The second method is a hybrid approach that fixes realizable inputs in the optimization problem while using the non-realizable inputs as optimization variables. In this way, output sensitivities are only used when they can be trusted. Furthermore, any of the sensitivity-based algorithm switches to the all-parallel approach when all inputs are realizable.

Performance comparison

Five coordination strategies are compared on three different ATC problems. The available hierarchical scheme III is used as a benchmark strategy, and furthermore the all-parallel scheme is evaluated, as well as three first-order sensitivity-based methods. These three methods are respectively the non-safeguarded algorithm, and the two safeguarded methods.

Two test examples clearly show that the non-safeguarded method outperforms all other strategies with respect to computational cost and convergence rate. The observed convergence rate for the first-order algorithms is indeed found to be quadratic, whereas the zero-order methods converge linearly. For the third test example however, all coordination strategies are comparable and are linearly convergent. What decreases the performance of the sensitivity-based approaches for this last example however remains to be investigated. However, for all three examples, sensitivity-based coordination performs better or similar to the benchmark coordination strategy. Numerical deterioration of the algorithms are observed for large weights for all five coordination schemes and all three problems.

Additional numerical observations

The numerical behavior of ATC is analyzed from the numerical experiments conducted for the evaluation of the coordination strategies. The solution found by the ATC algorithms is compared to the solution to the AAO relaxed problem. The accuracy of the solution found by ATC is higher when the ATC termination tolerance is increased. Furthermore, increasing weights has a negative effect on the ATC algorithms. The accuracy of the solution decreases as weights increase, because the steps taken by the ATC algorithms are smaller for larger weights. As the ATC solution estimate moves closer to the relaxed solution, step-sizes become even smaller. When step-sizes are smaller than the termination tolerance, the ATC algorithm terminates. Therefore, by increasing the weights, the ATC solution becomes less accurate. At a certain weight depending on the termination tolerance and the coordination algorithm, the accuracy no longer decreases because the ATC algorithms terminate at the first realizable solution.
6.3 Recommendations

The research presented in this report revealed the dual-feasible decomposition background of ATC. From this dual-feasible point of view, several suggestions for future research can be made.

**Exact penalty relaxation**

The solution to the relaxed problem is not equal to the solution to the original problem, due to the use of an inexact penalty function. For larger weights, this error becomes smaller, however the solution process of ATC becomes slower and more computationally expensive. The use of exact penalty formulations can be considered for ATC. With an exact penalty function, smaller weights can be used for a higher solution accuracy. With these smaller weights, the numerical difficulties observed for ATC may be reduced. However, the use of these exact relaxations may cause new numerical problems. More research is required to investigate the numerical behavior of exact penalty functions for ATC.

**Alternative dual master problems**

By revealing the dual-feasible strategy of ATC, the possibility of a dual coordination superimposed over the feasible decomposition is revealed. Only one dual method has been presented sofar, however many other dual master problems can be formulated. These new dual methods may again decrease the computational effort required to solve the the decomposed problem and improve the convergence of ATC algorithms.

**Alternative feasible master problems**

Similar to the dual master problem, feasible master problems for ATC have received very little attention. In this report, only a few suggestions are made for feasible master problems. Alternative master problems however may be formulated. An alternative is to define a zero-order feasible master problem based on a Secant algorithm for minimizing the inconsistency functions. In this Secant-like approach, approximations of the gradients of the inconsistency functions can be constructed from the outputs values of sub-problems.

Another sensitivity-based alternative is to use the sensitivities of all variables of an optimization sub-problem with respect to its inputs. These sensitivities are also available from the post-optimality sensitivity analysis. Local objectives and constraints approximations defined as functions of inputs only can be constructed with the sensitivities of the optimization variables of a sub-problem. A feasible master problem can be defined to find a vector of inputs that minimizes the approximate objective under the approximate constraints.
Non-hierarchical ATC

The proposed all-parallel and sensitivity-based feasible master problems were not restricted to the hierarchy required for the available hierarchy-based coordination strategies. The hierarchy for ATC requires the introduction of more copies than strictly necessary to account for coupling of two elements (e.g. consider the coordination of linking variables by the parent). A non-hierarchic decomposition of the original problem may reduce the dimensionality of the ATC sub-problems, and therefore possibly reduce the computational costs required to solve the decomposed problem.

Comparison of ATC to other decomposition methods

Many decomposition methods are available to decompose and solve the problems to which ATC is applied. However, very little information is available of the numerical performance of ATC in comparison with other available decomposition strategies. A performance comparison may provide insights in what the strong and weak aspects of the ATC decomposition method are. Also, ATC may be further improved by implementing techniques from other decomposition methods.
Chapter 6. Conclusions and recommendations
Bibliography


Bibliography


Appendix A

ATC-formulation for several problems

This appendix provides guidelines for identifying a hierarchic problem structure from the associated FDT. Using this approach, several optimization problems from [Hoc81], [Sch87], [Kim00], and [Flo90] are partitioned by hand (their dimensions were small enough to perform a manual FDT partitioning). The final problem was constructed with the geometric optimization problem of Section 3.7 as a basis. Constraints and variables were added to arrive at a problem with 26 variables and 20 constraints partitioned into a four-level hierarchy with 9 sub-problems.
A.1 Identifying problem hierarchy in the FDT

This section presents guidelines for the identification of a hierarchical problem structure from the functional dependence table. The approach is illustrated using the example geometric optimization problem of Section 3.7. This problem provides enough complexity to clearly illustrate the guidelines, but is small enough to allow a manual partitioning. The approach is applicable to large-scale problems, but automated partitioning methods as in [Wag93, Mic97, Kri97] are required.

Identifying multiple levels in a hierarchy

In this section, the identification of multiple levels from the FDT is illustrated. A multilevel hierarchy for ATC consists of a number of sub-problems (e.g. one at each level) connected only to adjacent levels (i.e. the next higher and/or the next lower level).

Consider the FDT of a four element partially separable problem, depicted in Figure A.1(a). As mentioned, elements may only share variables with adjacent levels. Figure A.1(b) depicts the structure of a four-level hierarchy of the problem. The linking between adjacent levels can be identified from the column associated with the shared variables $s$: element 0 shares variables with element 1, element 1 with 2, and 2 with 3. The four-level hierarchy which can be obtained from the FDT is depicted in Figure A.1(c).

Identifying multiple elements with one parent

In this section, the identification of multiple elements with one parent is illustrated. A parent-multiple children hierarchy consists of a number of sub-problems (one master at the top level and multiple children at a bottom level). All children must share variables with the parent.

Again, consider the FDT of a four element partially separable problem, depicted in Figure A.2(a). Figure A.2(b) depicts the structure of a parent-children two-level hierarchy of the problem. Note that it is possible that each sub-problem $i$ is linked to the parent with a unique shared variable $s_i$; in that case, the column associated with shared variables $s$ would only consist of a smaller number of non-zero entries. The two-level hierarchy which can be obtained from the FDT is depicted in Figure A.2(c).
A.1. Identifying problem hierarchy in the FDT

Identifying non-hierarchical ATC coupling

ATC allows a small degree of non-hierarchical coupling: children of the same parent may share variables amongst each other. In other words: all children must share variables with the parent, and children may share variables with each other.

Again, consider the FDT of a four element partially separable problem, depicted in Figure A.3(a). As mentioned, children must share variables with the parent (response target variables $R$) and may share variables with each other (linking target variables $y$). Figure A.3(b) depicts the structure of a parent-children two-level hierarchy of the problem. Response targets $R$ (indicated with solid lines) are common to all sub-problems, and linking target variables $y$ (indicated with dashed lines) are only shared by the children. The set of response targets and linking targets forms the set of shared variables: $s = \{R, y\}$. Note that it is possible that each sub-problem $i$ is linked to the parent with a unique response target variable $R_i$; in that case, the column associated with response target variables $R$ would only consist of a small number of non-zero entries. The two-level hierarchy which can be obtained from the FDT is depicted in Figure A.3(c).
Appendix A. ATC-formulation for several problems

Figure A.2: Structure for partially separable 4-element 2-level problem

(a) General FDT

(b) 2-level FDT

(c) Structure

Figure A.3: Structure for partially separable 4-element ATC 2-level problem

(a) General FDT

(b) 2-level FDT for ATC

(c) Structure
Example structure identification

Consider the geometric optimization problem first presented in [Wis78], and used in the context of ATC by [Kim01a, Etm02, Etm04, Hul03, Tze03, Mic04a]:

\[
\begin{align*}
\min_{z_1, z_2, \ldots, z_{14}} & \quad z_1^2 + z_2^2 \\
\text{subject to} & \quad g_1 : (z_3^2 + z_4^2)z_5^2 - 1 \leq 0, \quad g_2 : (z_5^2 + z_6^2)z_7^2 - 1 \leq 0, \\
& \quad g_3 : (z_8^2 + z_9^2)z_{11}^2 - 1 \leq 0, \quad g_4 : (z_8^2 + z_{10}^2)z_{11}^2 - 1 \leq 0, \\
& \quad g_5 : (z_{11}^2 + z_{12}^2)z_{13}^2 - 1 \leq 0, \quad g_6 : (z_{11}^2 + z_{12}^2)z_{14}^2 - 1 \leq 0, \\
& \quad h_1 : z_1^2 - z_3^2 - z_4^2 - z_5^2 = 0, \\
& \quad h_2 : z_2^2 - z_5^2 - z_6^2 - z_7^2 = 0, \\
& \quad h_3 : z_3^2 - z_8^2 - z_9^2 - z_{10}^2 - z_{11}^2 = 0, \\
& \quad h_4 : z_6^2 - z_{11}^2 - z_{12}^2 - z_{13}^2 - z_{14}^2 = 0, \\
& \quad z_1, z_2, \ldots, z_{14} \geq 0. \\
\end{align*}
\]  

(A.1)

The FDT for this problem is depicted in Figure A.4(a). By declaring response target variables \( R = [z_1, z_2, z_3, z_6] \), and linking target variables \( y = [z_5, z_{11}] \), and the remainder as local variables \( x = [z_4, z_7, z_8, z_9, z_{10}, z_{12}, z_{13}, z_{14}] \), the FDT can be manually partitioned by rearranging rows and columns. The partitioned FDT is depicted in Figure A.4(b). From the FDT, one can identify the top problem \( P_0 \). Problems \( P_{11} \) and \( P_{12} \) are linked as children of \( P_1 \) through response target variables \( z_1 \) and \( z_2 \) respectively. Problems \( P_{21} \) and \( P_{22} \) are children of \( P_{11} \) and \( P_{12} \) respectively, and are linked through response target variables \( z_{3} \) and \( z_{6} \) respectively. \( P_{11} \) and \( P_{12} \) are linked through linking variable target \( z_5 \). Elements \( P_{21} \) and \( P_{22} \) are coupled through variable \( z_{11} \), but do not have the same parent. This last form of coupling is not allowed in the ATC hierarchy and the coupling must be redirected to elements with the same parent (as illustrated in Section 3.7). The non-hierarchical structure of Problem (A.1) is depicted in Figure A.4(c).

Note that parent-child connections are equality constraints, which is necessary for ATC to define response functions for each child. In this case, the response functions for problems \( P_{11}, P_{12}, P_{21} \) and \( P_{22} \) are formed from \( h_1, h_2, h_3 \) and \( h_4 \), respectively. If inequality constraints couple sub-problems, extra variables and equality constraints can be introduced (e.g. if \( h_1 \) was an inequality constraint, introduce \( z_1^* \) as a copy of response variable \( z_1 \), and define an additional equality constraint \( h_5 = z_1^* - z_1 = 0 \), which becomes the response function of \( P_{11} \).
A.2 Example problems

In this appendix \( z^* \) holds the values for design variables in the optimum, \( f(z^*) \), the objective value in the optimum, and \( g_a \) holds the collection of indices of active inequality constraints in the optimum. Note that simple bounds are only mentioned in the primal problem statement, however they apply to all ATC sub-problems as well. The final problem is constructed by the author with the knowledge of target cascading structures. The problem itself is convex and therefore has a unique global optimum.
A.3 Problem 1

Problem 6 of [Hoc81]:

\[
\begin{align*}
\min_{z_1, z_2} & \quad f : (1 - z_1)^2 \\
\text{subject to} & \quad h_1 : 10(z_2 - z_1^2) = 0.
\end{align*}
\] (A.2)

Optimal solution: \(z^* = [1, 1], f(z^*) = 0\).

The partitioned FDT is displayed in Figure A.5(a), and a possible partition is depicted in Figure A.5(b), with \(x_{11} = z_2\), and \(a_{11} : h_1\).

Figure A.5: Partitioning of Problem (A.2)

The top-level problem \(P_0\) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_0} & \quad f_0(\bar{x}_0) + \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 \\
\text{subject to} & \quad h_{11}(\bar{x}_{11}) = 0.
\end{align*}
\] (A.3)

Where \(x_0 = [], R_{01} = [z_1], \bar{x}_0 = [R_{01}],\) and \(f_0(\bar{x}_0) = (1 - z_1)^2\). There is one bottom-level problem. Problem \(P_{11}\) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_{11}} & \quad \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 \\
\text{subject to} & \quad h_{11}(\bar{x}_{11}) = 0
\end{align*}
\] (A.4)

Where \(x_{11} = [z_2], r_{11} = [z_2], \bar{x}_{11} = [x_{11}, r_{11}], h_{11}(\bar{x}_{11}) = [r_{11} - a_{11}(\bar{x}_{11})],\) and \(a_{11}(\bar{x}_{11}) = \sqrt{z_2}\).
A.4 Problem 2

Problem 220 of [Sch87]:

\[
\begin{align*}
\min_{z_1, z_2} & \quad f : z_1 \\
\text{subject to} & \quad h_1 : (z_1 - 1)^3 - z_2 = 0, \\
& \quad 1 \leq z_1, \quad 0 \leq z_2.
\end{align*}
\] (A.5)

Optimal solution: \( z^* = [1, 0], f(z^*) = 1. \)

The partitioned FDT is displayed in Figure A.6(a), and a possible partition is depicted in Figure A.6(b), with \( x_{11} = z_1 \), and \( a_{11} : h_1 \).

The top-level problem \( P_0 \) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_0} & \quad f_0(\bar{x}_0) + \|w_{11}^R \circ (R_{01} - r_{11})\|^2_2 \\
\text{subject to} & \quad h_{11}(\bar{x}_{11}) = 0
\end{align*}
\] (A.6)

Where \( x_0 = [], \; R_{01} = [z_1], \; \bar{x}_0 = [R_{01}], \; \) and \( f_0(\bar{x}_0) = z_1. \) There is one bottom-level problem. Problem \( P_{11} \) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_{11}} & \quad \|w_{11}^R \circ (R_{01} - r_{11})\|^2_2 \\
\text{subject to} & \quad h_{11}(\bar{x}_{11}) = 0
\end{align*}
\] (A.7)

Where \( x_{11} = [z_2], \; r_{11} = [z_1], \; \bar{x}_{11} = [x_{11}, r_{11}], \; h_{11}(\bar{x}_{11}) = [r_{11} - a_{11}(\bar{x}_{11})], \) and \( a_{11}(\bar{x}_{11}) = \sqrt{z_2} - 1. \)
A.5 Problem 3

Problem 27 of [Hoc81]:

\[
\begin{align*}
\min_{z_1, z_2, z_3} & \quad f : 0.01(z_1 - 1)^2 + (z_2 - z_1^2)^2 \\
\text{subject to} & \quad h_1 : z_1 + z_3^2 + 1 = 0.
\end{align*}
\] (A.8)

Optimal solution: \( z^* = [-1, 1, 0] \), \( f(z^*) = 0 \).

The partitioned FDT is displayed in Figure A.7(a), and a possible partition is depicted in Figure A.7(b), with \( x_0 = z_2 \), \( x_{11} = z_3 \), and \( a_{11} : h_1 \).

![Partitioning of Problem (A.8)](image)

The top-level problem \( P_0 \) is formulated as:

\[
\begin{align*}
\min_{\tilde{x}_0} & \quad f_0(\tilde{x}_0) + \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 \\
\text{subject to} & \quad h_{11}(\tilde{x}_{11}) = 0
\end{align*}
\] (A.9)

Where \( x_0 = [z_2] \), \( R_{01} = [z_1] \), \( \tilde{x}_0 = [x_0, R_{01}] \), and \( f_0(\tilde{x}_0) = 0.01(z_1 - 1)^2 + (z_2 - z_1^2)^2 \).

There is one bottom-level problem. Problem \( P_{11} \) is formulated as:

\[
\begin{align*}
\min_{\tilde{x}_{11}} & \quad \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 \\
\text{subject to} & \quad h_{11}(\tilde{x}_{11}) = 0
\end{align*}
\] (A.10)

Where \( x_{11} = [z_3] \), \( r_{11} = [z_1] \), \( \tilde{x}_{11} = [x_{11}, r_{11}] \), \( h_{11}(\tilde{x}_{11}) = [r_{11} - a_{11}(\tilde{x}_{11})] \), and
\( a_{11}(\bar{x}_{11}) = -z_3^2 - 1. \)
A.6 Problem 4

Problem 254 of [Sch87]:

\[
\begin{align*}
\min_{z_1, z_2, z_3} & \quad f : \log(z_3) - z_2 \\
\text{subject to} & \quad h_1 : z_2^2 + z_3^2 - 4 = 0, \\
& \quad h_2 : z_3 - 1 - z_1^2 = 0.
\end{align*}
\]  

(A.11)

Optimal solution: \( z^* = [0, \sqrt{3}, 1] \), \( f(z^*) = -\sqrt{3} \).

The partitioned FDT is displayed in Figure A.8(a), and a possible partition is depicted in Figure A.8(b), with \( x_0 = z_2 \), \( x_{11} = z_1 \), \( h_0 : h_1 \), and \( a_{11} : h_2 \).

Figure A.8: Partitioning of Problem (A.11)

The top-level problem \( P_0 \) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_0} & \quad f_0(\bar{x}_0) + \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 \\
\text{subject to} & \quad h_{0}(\bar{x}_0) = 0
\end{align*}
\]  

(A.12)

Where \( \bar{x}_0 = [z_2] \), \( R_{01} = [z_1] \), \( \bar{x}_0 = [x_0, R_{01}] \), \( h_0 = z_2^2 + z_3^2 - 4 \) and \( f_0(\bar{x}_0) = \log(z_3) - z_2 \).

There is one bottom-level problem. Problem \( P_{11} \) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_{11}} & \quad \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 \\
\text{subject to} & \quad h_{11}(\bar{x}_{11}) = 0
\end{align*}
\]  

(A.13)

Where \( \bar{x}_{11} = [z_3] \), \( r_{11} = [z_1] \), \( \bar{x}_{11} = [x_{11}, r_{11}] \), \( h_{11}(\bar{x}_{11}) = [r_{11} - a_{11}(\bar{x}_{11})] \), and \( a_{11}(\bar{x}_{11}) = z_1^2 + 1 \).
A.7 Problem 5

Problem 39 of [Hoc81]:

\[ \begin{align*}
& \min_{z_1, z_2, z_3} f : -z_1 \\
& \text{subject to } h_1: z_2 - z_1^3 - z_3^2 = 0, \\
& \quad h_2: z_1^2 - z_2 - z_4^2 = 0. \\
\end{align*} \] (A.14)

Optimal solution: \( z^* = [1, 1, 0, 0], f(z^*) = -1. \)

The partitioned FDT is displayed in Figure A.9(a), and a possible partition is depicted in Figure A.9(b), with \( x_{11} = z_3, x_{12} = z_4, a_{11} : h_1, \) and \( a_{12} : h_2. \)

![Diagram](image-url)

**Figure A.9: Partitioning of Problem (A.14)**

The top-level problem \( P_0 \) is formulated as:

\[ \begin{align*}
& \min_{x_0} f_0(x_0) + \\
& \quad \| \mathbf{w}_{11}^R \circ (\mathbf{R}_{01} - \mathbf{r}_{11}) \|_2^2 + \| \mathbf{w}_{12}^R \circ (\mathbf{R}_{02} - \mathbf{r}_{12}) \|_2^2 + \\
& \quad \| \mathbf{w}_{11}^Y \circ (\mathbf{Y}_{01} - \mathbf{y}_{11}) \|_2^2 + \| \mathbf{w}_{12}^Y \circ (\mathbf{Y}_{02} - \mathbf{y}_{12}) \|_2^2 \\
& \text{subject to } h_0(x_0) = 0 \\
\end{align*} \] (A.15)

Where \( x_0 = [], R_{01} = [z_1], R_{02} = [z_1], Y_{01} = [z_2], Y_{02} = [z_2], \) \( x_0 = [R_{01}, Y_{01}, R_{02}, Y_{02}], f_0(x_0) = -z_1, \) and \( h_0(x_0) = [R_{01} - R_{02}, Y_{01} - Y_{02}]. \) There are two bottom-level problems. Problem \( P_{11} \) is formulated as:
\[ \min_{\bar{x}_{11}} \| \mathbf{w}_1^R \circ (\mathbf{R}_{01} - \mathbf{r}_{11}) \|_2^2 + \| \mathbf{w}_1^Y \circ (\mathbf{Y}_{01} - \mathbf{y}_{11}) \|_2^2 \quad (A.16) \]

subject to \( h_{11}(\bar{x}_{11}) = 0 \)

Where \( x_{11} = [z_3], \mathbf{r}_{11} = [z_1], \mathbf{y}_{11} = [z_2], \bar{x}_{11} = [x_{11}, \mathbf{r}_{11}, \mathbf{y}_{11}], h_{11}(\bar{x}_{11}) = [\mathbf{r}_{11} - \mathbf{a}_{11}(\bar{x}_{11})], \mathbf{a}_{11}(\bar{x}_{11}) = \sqrt{z_2 - z_3^2}. \) Problem \( P_{12} \) is formulated as:

\[ \min_{\bar{x}_{12}} \| \mathbf{w}_1^R \circ (\mathbf{R}_{02} - \mathbf{r}_{12}) \|_2^2 + \| \mathbf{w}_1^Y \circ (\mathbf{Y}_{01} - \mathbf{y}_{12}) \|_2^2 \quad (A.17) \]

subject to \( h_{12}(\bar{x}_{12}) = 0 \)

Where \( x_{12} = [z_4], \mathbf{r}_{12} = [z_1], \mathbf{y}_{12} = [z_2], \bar{x}_{12} = [x_{12}, \mathbf{r}_{12}, \mathbf{y}_{12}], h_{12}(\bar{x}_{12}) = [\mathbf{r}_{12} - \mathbf{a}_{12}(\bar{x}_{12})], \mathbf{a}_{12}(\bar{x}_{12}) = \sqrt{z_2 + z_4^2}. \)
Appendix A. ATC-formulation for several problems

A.8 Problem 6 (Colville No.6)

Problem 87 of [Hoc81]:

\[
\begin{align*}
\min_{z_1, z_2, \ldots, z_6} & \quad f : f_1(z) + f_2(z) \\
\text{subject to} & \quad h_1 : 300 - z_1 - \frac{1}{a} z_3 z_4 \cos(b - z_6) + \frac{c}{a} d z_3^2 = 0, \\
& \quad h_2 : -z_2 - \frac{1}{a} z_3 z_4 \cos(b + z_6) + \frac{c}{a} z_3^2 = 0, \\
& \quad h_3 : -z_5 - \frac{1}{a} z_3 z_4 \cos(b + z_6) + \frac{c}{a} z_4^2 = 0, \\
& \quad h_4 : 200 - \frac{1}{a} z_3 z_4 \cos(b - z_6) + \frac{c}{a} d z_3^2 = 0, \\
& \quad 0 \leq z_1 \leq 400 \quad 340 \leq z_3 \leq 420 \quad -1000 \leq z_5 \leq 10000 \\
& \quad 0 \leq z_2 \leq 1000 \quad 340 \leq z_4 \leq 420 \quad 0 \leq z_6 \leq .5236 \\
& \quad a = 131.078, \quad b = 1.48577, \quad c = .909798, \\
& \quad d = \cos(1.47588), \quad e = \sin(1.47588).
\end{align*}
\]

Optimal solution: \( z^* = \begin{bmatrix} 107.8119 \\ 196.3186 \\ 373.8307 \\ 420 \\ 213.0713 \\ 0.1532920 \end{bmatrix} \), \( f(z^*) = 8927.5977 \).

The partitioned FDT is displayed in Figure A.10(a), and a possible partition is depicted in Figure A.10(b), with \( x_{11} = z_5, h_{11} = [h_3, h_4], a_{11} : h_1, \) and \( a_{12} : h_2 \).

The top-level problem \( P_0 \) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_0} & \quad f_0(\bar{x}_0) + \\
& \quad \|w^{R}_{11} \circ (R_{01} - r_{11})\|_2^2 + \|w^{R}_{12} \circ (R_{02} - r_{12})\|_2^2 + \\
& \quad \|w^{R}_{11} \circ (Y_{01} - y_{11})\|_2^2 + \|w^{R}_{12} \circ (Y_{02} - y_{12})\|_2^2 \\
\text{subject to} & \quad h(\bar{x}_0) = 0
\end{align*}
\]

Where \( x_0 = [\cdot], R_{01} = [z_1], R_{02} = [z_2], Y_{01} = [z_3, z_4, z_5], Y_{02} = [z_3, z_4, z_5], \bar{x}_0 = [R_{01}, Y_{01}, R_{02}, Y_{02}], f_0(\bar{x}_0) = f_1(\bar{x}_0) + f_2(\bar{x}_0), \) and \( h(\bar{x}_0) = [R_{01} - R_{02}, Y_{01} - Y_{02}] \).
There are two bottom-level problems. Problem $P_{11}$ is formulated as:

$$\begin{align*}
\min_{\bar{x}_{11}} \quad & \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 + \|w_{11}^Y \circ (Y_{01} - y_{11})\|_2^2 \\
\text{subject to} \quad & h_{11}(\bar{x}_{11}) = 0
\end{align*}$$

(A.20)

Where $x_{11} = [z_5]$, $r_{11} = [z_1]$, $y_{11} = [z_3, z_4, z_6]$, $\bar{x}_{11} = [x_{11}, r_{11}, y_{11}]$, $h_{11}(\bar{x}_{11}) = [r_{11} - a_{11}(\bar{x}_{11}), h_3, h_4]$, and $a_{11}(\bar{x}_{11}) = 300 - \frac{1}{a} z_3 z_4 \cos(b - z_6) + \frac{c}{a} dz_4^2$. Problem $P_{12}$ is formulated as:

$$\begin{align*}
\min_{\bar{x}_{12}} \quad & \|w_{12}^R \circ (R_{02} - r_{12})\|_2^2 + \|w_{12}^Y \circ (Y_{02} - y_{12})\|_2^2 \\
\text{subject to} \quad & h_{12}(\bar{x}_{12}) = 0
\end{align*}$$

(A.21)

Where $x_{12} = [\,]$, $r_{12} = [z_2]$, $y_{12} = [z_3, z_4, z_6]$, $\bar{x}_{12} = [x_{12}, r_{12}, y_{12}]$, $h_{11}(\bar{x}_{11}) = [r_{11} - a_{12}(\bar{x}_{12})]$, and $a_{12}(\bar{x}_{12}) = -\frac{1}{a} z_3 z_4 \cos(b + z_6) + \frac{c}{a} dz_4^2$. 

\[\text{Figure A.10: Partitioning of Problem (A.18)}\]
Appendix A. ATC-formulation for several problems

A.9 Problem 7

Problem 56 of [Hoc81]:

\[
\begin{align*}
\min_{z_1, z_2, \ldots, z_7} & \quad f : -z_1 z_2 z_3 \\
\text{subject to} & \quad h_1 : z_1 - 4.2 \sin^2 z_4 = 0, \\
& \quad h_2 : z_2 - 4.2 \sin^2 z_5 = 0, \\
& \quad h_3 : z_3 - 4.2 \sin^2 z_6 = 0, \\
& \quad h_4 : z_1 + 2 z_2 + 2 z_3 - 7.2 \sin^2 z_7 = 0.
\end{align*}
\] (A.22)

Optimal solution: 
\[z^* = \begin{bmatrix} 2.4 & 1.2 & 1.2 & \pm e \mod \pi & \pm d \mod \pi & \pm d \mod \pi & \pi/2 \mod \pi \end{bmatrix}, \]
\[f(z^*) = -3.456.\]

The partitioned FDT is displayed in Figure A.11(a), and a possible partition is depicted in Figure A.11(b), with 
\[x_0 = z_7, \ x_{11} = z_4, \ x_{12} = z_5, \ x_{13} = z_6, \ h_0 = h_4, \ a_{11} : h_1, \ a_{12} : h_2, \text{ and } a_{13} : h_3.\]

The top-level problem \(P_0\) is formulated as:

\[
\begin{align*}
\min_{x_0} & \quad f_0(x_0) + \|w_{11}^R \circ (R_{01} - r_{11}(x_{11}))\|_2^2 + \\
& \quad \|w_{12}^R \circ (R_{02} - r_{12}(x_{12}))\|_2^2 + \|w_{13}^R \circ (R_{03} - r_{13}(x_{13}))\|_2^2 \\
\text{subject to} & \quad h_0(x_0) = 0
\end{align*}
\] (A.23)

Where \(x_0 = [z_7], \ R_{01} = [z_1], \ R_{02} = [z_2], \ R_{03} = [z_3], \ x_0 = [x_0, R_{01}, R_{02}, R_{03}], \ h_0(x_0) = [h_4], \text{ and } f_0(x_0) = -R_{01} R_{02} R_{03}.\)
A.9. Problem 7

There are three bottom-level problems. Problem $P_{11}$ is formulated as:

$$\begin{align*}
\min_{\bar{x}_{11}} & \quad \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 \\
\text{subject to} & \quad h_{11}(\bar{x}_{11}) = 0
\end{align*}$$ (A.24)

Where $x_{11} = [z_4]$, $r_{11} = [z_1]$, $\bar{x}_{11} = [x_{11}, r_{11}]$, $h_{11}(\bar{x}_{11}) = [r_{11} - a_{11}(\bar{x}_{11})]$, and $a_{11}(\bar{x}_{11}) = 4.2 \sin^2 z_4$.

Problem $P_{12}$ is formulated as:

$$\begin{align*}
\min_{\bar{x}_{12}} & \quad \|w_{12}^R \circ (R_{02} - r_{12})\|_2^2 \\
\text{subject to} & \quad h_{12}(\bar{x}_{12}) = 0
\end{align*}$$ (A.25)

Where $x_{12} = [z_5]$, $r_{12} = [z_2]$, $\bar{x}_{12} = [x_{12}, r_{12}]$, $h_{12}(\bar{x}_{12}) = [r_{12} - a_{12}(\bar{x}_{12})]$, and $a_{12}(\bar{x}_{12}) = 4.2 \sin^2 z_5$.

Problem $P_{13}$ is formulated as:

$$\begin{align*}
\min_{\bar{x}_{13}} & \quad \|w_{13}^R \circ (R_{03} - r_{13})\|_2^2 \\
\text{subject to} & \quad h_{13}(\bar{x}_{13}) = 0
\end{align*}$$ (A.26)

Where $x_{13} = [z_6]$, $r_{13} = [z_3]$, $\bar{x}_{13} = [x_{13}, r_{13}]$, $h_{13}(\bar{x}_{13}) = [r_{13} - a_{13}(\bar{x}_{13})]$, and $a_{13}(\bar{x}_{13}) = 4.2 \sin^2 z_5$. 
A.10 Problem 8 (Geometric optimization problem 1)

This problem is a reduced version of the example geometric optimization problem of [Kim00], which is presented in Section A.14. Two equality constraints and four inequality constraints are removed, as well as 7 variables. The reduced problem statement is:

\[
\begin{align*}
\min_{z_1, \ldots, z_7} & \quad f = z_1^2 + z_2^2, \\
\text{subject to} & \quad g_1 : z_3^2 + z_4^2 - z_5^2 \leq 0, \\
& \quad g_2 : z_5^2 + z_6^2 - z_7^2 \leq 0, \\
& \quad h_1 : z_1^2 + z_2^2 - z_4^2 - z_5^2 = 0, \\
& \quad h_2 : z_2^2 + z_5^2 - z_6^2 - z_7^2 = 0, \\
& \quad z_1, z_2, \ldots, z_7 \geq 0. 
\end{align*}
\]

(A.27)

Optimal solution: \( z^* = \begin{bmatrix} 2.1491 \\ 2.0759 \\ 1.3161 \end{bmatrix}, f(z^*) = 8.9282. \)

The partitioned FDT is displayed in Figure A.12(a), and a possible partition is depicted in Figure A.12(b), with \( x_{11} = [z_3, z_4], x_{12} = [z_6, z_7], a_{11} : h_1, g_{11} = [g_1], a_{12} : h_2, \) and \( g_{12} = [g_2] \). The top-level problem \( P_0 \) is formulated as:

Figure A.12: Partitioning of Problem (A.27)
A.10. Problem 8 (Geometric optimization problem 1)

\[
\begin{align*}
\min_{\mathbf{x}_0} & \quad f_0(\mathbf{x}_0) + \\
& \quad \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 + \|w_{12}^R \circ (R_{02} - r_{12})\|_2^2 + \\
& \quad \|w_{11}^Y \circ (Y_{01} - y_{11})\|_2^2 + \|w_{12}^Y \circ (Y_{02} - y_{12})\|_2^2
\end{align*}
\]

subject to \( h_0(\mathbf{x}_0) = 0 \) \hspace{1cm} (A.28)

Where \( \mathbf{x}_0 = [], \quad R_{01} = [z_1], \quad R_{02} = [z_2], \quad Y_{01} = Y_{02} = [z_3], \quad \mathbf{x}_0 = [R_{01}, Y_{01}, R_{02}, Y_{02}], \quad h_0(\mathbf{x}_0) = Y_{01} - Y_{02}, \) and \( f_0(\mathbf{x}_0) = z_1^2 + z_2^2. \) There are two bottom level problems. Problem \( P_{11} \) is:

\[
\begin{align*}
\min_{\mathbf{x}_{11}} & \quad \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 + \|w_{11}^Y \circ (Y_{01} - y_{11})\|_2^2 \\
\text{subject to } & \quad h_{11}(\mathbf{x}_{11}) = 0 \\
& \quad g_{11}(\mathbf{x}_{11}) \leq 0
\end{align*}
\]

\hspace{1cm} (A.29)

Where \( \mathbf{x}_{11} = [z_3, z_4], \quad r_{11} = [z_1], \quad y_{11} = [z_5], \quad \mathbf{x}_{11} = [\mathbf{x}_{11}, r_{11}, y_{11}], \quad h_{11}(\mathbf{x}_{11}) = [r_{11} - a_{11}(\mathbf{x}_{11})], \quad a_{11}(\mathbf{x}_{11}) = \sqrt{z_3^2 + z_4^2 + z_5^2}, \) and \( g_{11}(\mathbf{x}_{11}) = z_3^2 + z_4^2 - z_5^2. \) Problem \( P_{12} \) is:

\[
\begin{align*}
\min_{\mathbf{x}_{12}} & \quad \|w_{12}^R \circ (R_{02} - r_{12})\|_2^2 + \|w_{12}^Y \circ (Y_{02} - y_{12})\|_2^2 \\
\text{subject to } & \quad h_{12}(\mathbf{x}_{12}) = 0 \\
& \quad g_{12}(\mathbf{x}_{12}) \leq 0
\end{align*}
\]

\hspace{1cm} (A.30)

Where \( \mathbf{x}_{12} = [z_6, z_7], \quad r_{12} = [z_2], \quad y_{12} = [z_5], \quad \mathbf{x}_{12} = [\mathbf{x}_{12}, r_{12}, y_{12}], \quad h_{12}(\mathbf{x}_{12}) = [r_{12} - a_{12}(\mathbf{x}_{12})], \quad a_{12}(\mathbf{x}_{12}) = \sqrt{z_6^2 + z_7^2 + z_5^2}, \) and \( g_{12}(\mathbf{x}_{12}) = z_5^2 + z_6^2 - z_7^2. \)
A.11 Problem 9 (Static Power Scheduling)

Problem 107 of [Hoc81]:

\[ \min_{z_1, z_2, \ldots, z_9} f : 3000z_1 + 1000z_1^2 + 2000z_2 + 666.667z_2^2 \]  \hspace{1cm} (A.31)

subject to

\[ h_1 : 4 - z_1 + 2c z_1^2 - z_5 z_6 (du_1 + cu_2) - z_5 z_7 (du_3 + cu_4) = 0, \]
\[ h_2 : 4 - z_2 + 2c z_2^2 + z_5 z_6 (du_1 - cu_2) + z_6 z_7 (du_5 - cu_6) = 0, \]
\[ h_3 : 8 + 2c z_3^2 + z_5 z_7 (du_3 - cu_4) - z_6 z_7 (du_5 + cu_6) = 0, \]
\[ h_4 : 2 - z_3 + 2d z_3^2 + z_5 z_6 (cu_1 - cu_2) + z_5 z_7 (cu_3 - du_4) = 0, \]
\[ h_5 : 2 - z_4 + 2d z_4^2 - z_5 z_6 (cu_1 + cu_2) - z_6 z_7 (cu_5 + du_6) = 0, \]
\[ h_6 : -3.37 + 2d z_6^2 - z_5 z_7 (cu_3 + du_4) + z_6 z_7 (cu_5 - du_6) = 0, \]
\[ 0 \leq z_i, i = 1, 2, \quad 0.90909 \leq x_i \leq 1.0909, i = 5, 6, 7, \]
\[ u_1 = \sin z_8, \quad u_2 = \cos z_8, \quad u_3 = \sin z_9, \]
\[ u_4 = \cos z_9, \quad u_5 = \sin (z_8 - z_9), \quad u_6 = \cos (z_8 - z_9), \]
\[ c = (48.4/50.176) \sin .25, \quad d = (48.4/50.176) \cos .25. \]

Optimal solution: \( z^* = \begin{bmatrix} 0.6670095 \\ 1.0224 \\ 0.2283 \\ 0.1848 \\ 0.01099 \\ 1.0909 \\ 1.0690 \\ 0.1066 \\ -0.3388 \end{bmatrix} \), \( f(z^*) = 5055. \)

The partitioned FDT is displayed in Figure A.13(a), and a possible partition is depicted in Figure A.14(b), with \( x_{11} = [z_3, \ldots, z_9], h_{11} = [h_3, \ldots, h_6], \) and \( a_{11} : [h_1, h_2]. \)

The top-level problem \( P_0 \) is formulated as:

\[ \min_{x_0} f_0(x_0) + \|w_{\mathbf{R}_{01}^1}^R \circ (\mathbf{R}_{01} - \mathbf{r}_{11})\|^2_2 \]  \hspace{1cm} (A.32)

Where \( x_0 = [], R_{01} = [z_1, z_2], x_0 = [x_0, R_{01}], \) and \( f_0(x_0) = 3000z_1 + 1000z_1^2 + 2000z_2 + 666.667z_2^2. \) There is one bottom-level problem. Problem \( P_{11} \) is formulated as:
A.11. Problem 9 (Static Power Scheduling)

149

Figure A.13: Partitioning of Problem (A.31)

\[
\begin{align*}
\min_{x_{11}} & \quad \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 \\
\text{subject to} & \quad h_{11}(x_{11}) = 0
\end{align*}
\]

(A.33)

Where \(x_{11} = [z_3, \ldots, z_9]\), \(r_{11} = [z_1, z_2]\), \(\bar{x}_{11} = [x_{11}, r_{11}]\), \(h_{11}(\bar{x}_{11}) = [r_{11} - a_{11}(\bar{x}_{11}), h_3, \ldots, h_6]\), and \(a_{11}(\bar{x}_{11}) = [0.4 + 2cz_5^2 - z_5z_6(du_1 + cu_2) - z_5z_7(du_3 + cu_4), 0.4 + 2cz_6^2 + z_5z_6(du_1 - cu_2) + z_5z_7(du_5 - cu_6)]\).
Appendix A. ATC-formulation for several problems

A.12 Problem 10 (Heat exchanger design I)
Problem 4.8 of [Flo90]:

\[
\begin{align*}
\min_{z_1, z_2, \ldots, z_7} & \quad f : 35 \left( \frac{z_1}{200z_3} \right)^{0.6} + 35 \left( \frac{z_2}{200z_4} \right)^{0.6} \\
\text{subject to} & \quad h_1 : z_1 - 10^4(z_7 - 100) = 0, \\
& \quad h_2 : z_2 - 10^4(300 - z_7) = 0, \\
& \quad h_3 : z_1 - 10^4(600 - z_5) = 0, \\
& \quad h_4 : z_2 - 10^4(900 - z_6) = 0, \\
& \quad h_5 : z_3 = \frac{z_5 + z_7 - 700}{\ln \frac{z_5 - 100}{600 - z_7}} = 0, \\
& \quad h_6 : z_4 = \frac{z_6 - z_7 - 600}{\ln \frac{z_6 - z_7}{600}} = 0, \\
& \quad z_5 \leq 600, z_6 \leq 900, 100 \leq z_7 \leq 300 \\
\end{align*}
\]  

(A.34)

Best known solution: \( z^* = \begin{bmatrix} 600 \\ 700 \\ 100 \end{bmatrix} \), \( f(z^*) = 189.3 \).

The partitioned FDT is displayed in Figure A.14(a), and a possible partition is depicted in Figure A.14(b), with \( x_0 = [z_1, z_3, z_5] \), \( x_{11} = [z_2, z_4, z_6, z_7] \), \( f_0 = 35 \left( \frac{z_1}{200z_3} \right)^{0.6} \), \( f_{11} = 35 \left( \frac{z_2}{200z_4} \right)^{0.6} \), \( h_0 = [h_1, h_3, h_5] \), and \( h_{11} = [h_2, h_4, h_6] \). A copy of variable \( z_7 \) is introduced \( h_7 : z_7' = z_7 \).

The top-level problem \( P_0 \) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_0} & \quad f_0(\bar{x}_0) + \left\| w_{11}^T \circ (R_{01} - r_{11}) \right\|_2^2 \\
\text{subject to} & \quad h_0(\bar{x}_0) = 0 
\end{align*}
\]  

(A.35)
Where $x_0 = [z_1, z_3, z_5], R_{01} = z_7', \bar{x}_0 = [x_0, R_{01}], h_0 = [h_1, h_3, h_5], \text{ and } f_0 = 35 \left( \frac{z_1}{200} \right)^{0.6}.$

There is one bottom-level problem. Problem $P_{11}$ is formulated as:

$$\begin{align*}
\min_{\bar{x}_{11}} & \quad f_{11}(\bar{x}_{11}) + \|w_{11}^R \circ (R_{01} - r_{11})\|^2_2 \\
\text{subject to} & \quad h_{11}(\bar{x}_{11}) = 0
\end{align*}$$  \hspace{1cm} (A.36)

Where $x_{11} = [z_2, z_4, z_6, z_7], r_{11} = [z_7'], \bar{x}_{11} = [x_{11}, r_{11}], h_{11} = [r_{11} - a_{11}(\bar{x}_{11}), h_2, h_4, h_6], a_{11}(\bar{x}_{11}) = z_7, \text{ and } f_{11} = 35 \left( \frac{z_2}{200} \right)^{0.6}.$
Appendix A. ATC-formulation for several problems

A.13 Problem 11 (Heat exchanger design II)

Problem 4.9 of [Flo90]:

\[
\begin{align*}
\min_{z_1, z_2, \ldots, z_{11}} & \quad \left( \frac{z_1}{120z_4} \right)^{0.6} + \left( \frac{z_2}{80z_5} \right)^{0.6} + \left( \frac{z_3}{40z_6} \right)^{0.6} \\
\text{subject to} & \quad h_1 : z_1 - 10^5(z_7 - 100) = 0, \\
& \quad h_2 : z_2 - 10^5(z_8 - z_7) = 0, \\
& \quad h_3 : z_3 - 10^5(500 - z_8) = 0, \\
& \quad h_4 : z_1 - 10^5(300 - z_9) = 0, \\
& \quad h_5 : z_2 - 10^5(400 - z_{10}) = 0, \\
& \quad h_6 : z_3 - 10^5(600 - z_{11}) = 0, \\
& \quad h_7 : z_4 - \frac{z_7 + z_9 - 400}{\ln \frac{z_9 - 100}{300 - z_7}} = 0, \\
& \quad h_8 : z_5 - \frac{z_8 + z_{10} - z_7 - 400}{\ln \frac{z_{10} - z_7}{300 - z_8}} = 0, \\
& \quad h_9 : z_6 - \frac{z_8 - 100}{\ln \frac{z_{11} - z_8}{100}} = 0, \\
& \quad 100 \leq z_7, z_8 \leq 500, z_9 \leq 300, z_{10} \leq 400, z_{11} \leq 600.
\end{align*}
\]  

(A.37)

Best known solution: \( z^* = \begin{bmatrix} 1.8900 \cdot 10^6 \\ 1.1360 \cdot 10^7 \\ 2.0450 \cdot 10^7 \\ 181.9 \\ 295.0 \\ 281.1 \\ 286.4 \\ 395.5 \end{bmatrix} \), \( f(z^*) = 259.9 \).

The partitioned FDT is displayed in Figure A.15(a), and a possible partition is depicted in Figure A.15(b), with \( x_0 = [z_1, z_4, z_9] \), \( x_{11} = [z_2, z_5, z_7, z_{10}] \), \( x_{21} = [z_3, z_6, z_8, z_{11}] \), \( f_0 = \left( \frac{z_1}{120z_4} \right)^{0.6} \), \( f_{11} = \left( \frac{z_2}{80z_5} \right)^{0.6} \), \( f_{21} = \left( \frac{z_3}{40z_6} \right)^{0.6} \), \( h_0 = [h_1, h_4, h_7] \), \( h_{11} = [h_2, h_5, h_8] \), and \( h_{21} = [h_3, h_6, h_9] \). Two variable copies and constraints are introduced: \( h_{10} : z'_7 = z_7 \), and \( h_{11} : z'_8 = z_8 \).

The top-level problem \( P_0 \) is formulated as:
Problem 11 (Heat exchanger design II)

\[
\min_{x_0} f_0(x_0) + \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2
\]
\[
\text{subject to } h_0(x_0) = 0
\]  

(A.38)

Where \(x_0 = [z_1, z_4, z_9]\), \(R_{01} = z_7\), \(x_0 = [x_0, R_{01}]\), \(h_0 = [h_1, h_4, h_7]\), and \(f_0 = \left(\frac{z_1}{120z_4}\right)^{0.6}\).

There is one intermediate-level problem. Problem \(P_{11}\) is formulated as:

\[
\min_{x_{11}} f_{11}(\bar{x}_{11}) + \|w_{11}^R \circ (R_{01} - r_{11})\|_2^2 + \|w_{21}^R \circ (R_{11} - r_{21})\|_2^2
\]
\[
\text{subject to } h_{11}(\bar{x}_{11}) = 0
\]  

(A.39)

Where \(x_{11} = [z_2, z_5, z_7, z_{10}]\), \(r_{11} = [z_7]\), \(R_{11} = z_8\), \(\bar{x}_{11} = [x_{11}, r_{11}, R_{11}]\), \(h_{11} = [r_{11} - a_{11}(\bar{x}_{11}), h_2, h_5, h_8]\), \(a_{11}(\bar{x}_{11}) = z_7\), and \(f_{11} = \left(\frac{z_2}{80z_7}\right)^{0.6}\).

There is one bottom-level problem. Problem \(P_{21}\) is formulated as:
Appendix A. ATC-formulation for several problems

\[
\min_{\bar{x}_{21}} f_{21}(\bar{x}_{21}) + \| w_{21}^R \odot (R_{21} - r_{21}(\bar{x}_{21})) \|_2^2 \\
\text{subject to } h_{21}(\bar{x}_{21}) = 0
\]  
\text{(A.40)}

Where \( x_{21} = [z_3, z_6, z_8, z_11], r_{21} = z_8', \bar{x}_{21} = [x_{11}, r_{21}], h_{21} = [r_{11} - a_{21}(\bar{x}_{21}), h_3, h_6, h_9], \) 
\( a_{21}(\bar{x}_{21}) = z_8, \) and \( f_{21} = \left( \frac{z_3}{4056} \right)^{0.6} \).
A.14 Problem 12 (Geometric optimization problem 2)

Example problem of [Kim00]:

$$\begin{align*}
\min_{z_1, z_2, \ldots, z_{14}} & \quad z_1^2 + z_2^2 \\
\text{subject to} & \quad g_1 : z_3^2 + z_4^2 - z_5^2 \leq 0, \quad g_2 : z_5^2 + z_6^2 - z_7^2 \leq 0, \\
& \quad g_3 : z_8^2 + z_9^2 - z_{11}^2 \leq 0, \quad g_4 : z_8^2 - z_{10}^2 - z_{11}^2 \leq 0, \\
& \quad g_5 : z_{12}^2 + z_{13}^2 - z_{14}^2 \leq 0, \quad g_6 : z_{11}^2 + z_{12}^2 - z_{14}^2 \leq 0, \\
& \quad h_1 : z_1^2 - z_3^2 - z_4^2 - z_5^2 = 0, \\
& \quad h_2 : z_2^2 - z_3^2 - z_6^2 - z_7^2 = 0, \\
& \quad h_3 : z_2^2 - z_8^2 - z_9^2 - z_{10}^2 - z_{11}^2 = 0, \\
& \quad h_4 : z_6^2 - z_{11}^2 - z_{12}^2 - z_{13}^2 - z_{14}^2 = 0, \\
& \quad z_1, z_2, \ldots, z_{14} \geq 0.
\end{align*}$$

(A.41)

Optimal solution: $\mathbf{z}^* = [2.8365, 3.0892, 2.3567, 0.7591, 0.8697, 0.8644, 0.9719, 0.8644, 0.7957, 1.3007, 0.8434, 1.7601, 1.5502]$, $f(\mathbf{z}^*) = 17.5888$, $g_a = [1, 2, 3, 4, 5, 6]$.

Partition 1

A three-level partition of the problem is presented in [Kim01a, Etm04, Hul03, Mic04a]. In this partition, variable $z_{11}$ is treated as a static parameter with its known optimal value $z_{11} = p = z_{11}^*$. The partition is depicted in Figure A.16(b), with $x_{11} = [z_1]$, $x_{12} = [z_7]$, $x_{21} = [z_8, z_9, z_{10}]$, $x_{22} = [z_{12}, z_{13}, z_{14}]$, $a_{11} : [h_1]$, $a_{12} : [h_2]$, $a_{21} : [h_3]$, $a_{22} : [h_4]$, $g_{11} = [g_1]$, $g_{12} = [g_2]$, $g_{21} = [g_3, g_4]$, and $g_{22} = [g_5, g_6]$. The partitioned FDT is displayed in Figure A.16(a). Top level problem $P_0$ can be formulated as:

$$\begin{align*}
\min_{\mathbf{x}_0} & \quad f_0(\mathbf{x}_0) + \\
& \quad \|w_{11}^r \circ (R_1 - r_{11})\|^2 + \|w_{12}^r \circ (R_2 - r_{12})\|^2 + \\
& \quad \|w_{21}^y \circ (Y_{01} - y_{11})\|^2 + \|w_{22}^y \circ (Y_{02} - y_{12})\|^2 \\
\text{subject to} & \quad h_0(\mathbf{x}_0) = 0
\end{align*}$$

(A.42)

Where $\mathbf{x}_0 = [], R_{01} = [z_1]$, $Y_{01} = [z_5]$, $R_{02} = [z_2]$, $Y_{02} = [z_5]$, $\mathbf{x}_0 = [\mathbf{x}_0, R_{01}, Y_{01}, R_{02}, Y_{02}]$, $h_0(\mathbf{x}_0) = [Y_{01} - Y_{02}]$, and $f_0(\mathbf{x}_0) = z_1^2 + z_2^2$. There are two intermediate level problems. Problem $P_{11}$ is formulated as:
Appendix A. ATC-formulation for several problems

Figure A.16: Partitioning of Problem (A.41)

\[
\begin{align*}
\min_{x_{11}} \quad & \|w_{11}^R \circ (R_{11} - r_{11})\|_2^2 + \|w_{11}^Y \circ (Y_{11} - y_{11})\|_2^2 + \|w_{11}^T \circ (R_{21} - r_{21})\|_2^2 \\
\text{subject to} \quad & g_{11}(\tilde{x}_{11}) \leq 0 \\
& h_{11}(\tilde{x}_{11}) = 0
\end{align*}
\] (A.43)

Where \(x_{11} = [z_4], \ r_{11} = [z_4], \ y_{11} = [z_5], \ R_{11} = [z_3], \ \tilde{x}_{11} = [x_{11}, r_{11}, y_{11}, R_{11}], \ g_{11}(\tilde{x}_{11}) = [g_1], \ h_{11}(\tilde{x}_{11}) = [r_{11} - a_{11}(\tilde{x}_{11})], \) and \(a_{11}(\tilde{x}_{11}) = \sqrt{z_4^2 + z_4^{-2} + z_5^2}.

Intermediate level problem \(P_{12}\) is formulated as:

\[
\begin{align*}
\min_{x_{12}} \quad & \|w_{12}^R \circ (R_{12} - r_{12})\|_2^2 + \|w_{12}^Y \circ (Y_{12} - y_{12})\|_2^2 + \|w_{12}^T \circ (R_{22} - r_{22})\|_2^2 \\
\text{subject to} \quad & g_{12}(\tilde{x}_{12}) \leq 0 \\
& h_{12}(\tilde{x}_{12}) = 0
\end{align*}
\] (A.44)

Where \(x_{12} = [z_7], \ r_{12} = [z_7], \ y_{12} = [z_5], \ R_{12} = [z_6], \ \tilde{x}_{12} = [x_{12}, r_{12}, y_{12}, R_{12}], \ g_{12}(\tilde{x}_{12}) = [g_2], \ h_{12}(\tilde{x}_{12}) = [r_{12} - a_{12}(\tilde{x}_{12})], \) and \(a_{12}(\tilde{x}_{12}) = \sqrt{z_5^2 + z_6^2 + z_7^2}.

There are two bottom level problems. Problem \(P_{21}\) is formulated as:

\[
\begin{align*}
\min_{x_{21}} \quad & \|w_{21}^R \circ (R_{11} - r_{21})\|_2^2 \\
\text{subject to} \quad & g_{21}(\tilde{x}_{21}) \leq 0 \\
& h_{21}(\tilde{x}_{21}) = 0
\end{align*}
\] (A.45)
Where $\mathbf{x}_{21} = [z_8, z_9, z_{16}], \mathbf{r}_{21} = [z_3], \mathbf{x}_{21} = [x_{21}, \mathbf{r}_{21}], \mathbf{g}_{21}(\mathbf{x}_{21}) = [g_3, g_4], h_{21}(\mathbf{x}_{21}) = [r_{21} - a_{21}(\mathbf{x}_{21})], \text{and } a_{21}(\mathbf{x}_{21}) = \sqrt{z_8^2 + z_9^2 + z_{16}^2 + p^2}$. Bottom level problem $P_{22}$ is formulated as:

$$
\min_{\mathbf{x}_{22}} \|w_{22}^R \circ (\mathbf{R}_{12} - \mathbf{r}_{22})\|_2^2 \\
\text{subject to } \mathbf{g}_{22}(\mathbf{x}_{22}) \leq 0 \\
h_{22}(\mathbf{x}_{22}) = 0
$$

(A.46)

Where $\mathbf{x}_{22} = [z_{12}, z_{13}, z_{14}], \mathbf{r}_{22} = [z_6], \mathbf{x}_{22} = [x_{21}, \mathbf{r}_{22}], \mathbf{g}_{22}(\mathbf{x}_{22}) = [g_5, g_6], h_{22}(\mathbf{x}_{22}) = [r_{22} - a_{22}(\mathbf{x}_{22})], \text{and } a_{22}(\mathbf{x}_{22}) = \sqrt{z_{12}^2 + z_{13}^2 + z_{14}^2 + p^2}$. 

**Partition 2**

After introducing two copies $z_{15}$ and $z_{16}$ and equality constraints $h_5 : z_{11} - z_{15} = 0$ and $h_6 : z_{11} - z_{16} = 0$, a possible three-level partition is depicted in Figure A.17(b), with $\mathbf{x}_{11} = [z_4], \mathbf{x}_{12} = [z_7], \mathbf{x}_{21} = [z_8, z_9, z_{10}, z_{15}], \mathbf{x}_{22} = [z_{12}, z_{13}, z_{14}, z_{16}], r_{11} : [h_1], r_{12} : [h_2], r_{21} : [h_3, h_5], r_{22} : [h_4, h_6], g_{11} = [g_1], g_{12} = [g_2], g_{21} = [g_3, g_4], \text{and } g_{22} = [g_5, g_6].$ The partitioned FDT is displayed in Figure A.17(a).

Top level problem $P_0$ can be formulated as:

![Figure A.17: Partitioning of Problem (A.41)](image-url)
min \( x_0 \) \( f_0(x_0) + \) \( \|w^R_1 \circ (R_{11} - r_{11})\|_2^2 + \|w^R_{12} \circ (R_{02} - r_{12})\|_2^2 + \|w^Y_1 \circ (Y_{01} - y_{11})\|_2^2 + \|w^Y_{12} \circ (Y_{02} - y_{12})\|_2^2 \) \hspace{1cm} (A.47)

subject to \( h_0(x_0) = 0 \)

Where \( x_0 = \{1\}, R_{01} = [z_1], Y_{01} = [z_5, z_{11}], R_{02} = [z_2], Y_{02} = [z_5, z_{11}], x_0 = [x_0, R_{01}, Y_{01}, R_{02}, Y_{02}], h_0(x_0) = [Y_{01} - Y_{02}], \) and \( f_0(x_0) = z_1^2 + z_2^2. \) There are two intermediate level problems. Problem \( P_{11} \) is formulated as:

min \( \bar{x}_{11} \) \( \|w^R_1 \circ (R_{11} - r_{11})\|_2^2 + \|w^Y_1 \circ (Y_{01} - y_{11})\|_2^2 + \) \( \|w^R_{12} \circ (R_{21} - r_{21})\|_2^2 \) \hspace{1cm} (A.48)

subject to \( g_{11}(\bar{x}_{11}) \leq 0 \)

\( h_{11}(\bar{x}_{11}) = 0 \)

Where \( \bar{x}_{11} = [z_4], r_{11} = [z_1], y_{11} = [z_5, z_{11}], R_{11} = [z_3, z_{11}], \bar{x}_{11} = [x_{11}, r_{11}, y_{11}, R_{11}], \)

\( g_{11}(\bar{x}_{11}) = [g_1], h_{11}(\bar{x}_{11}) = [r_{11} - a_{11}(\bar{x}_{11})] \), and \( a_{11}(\bar{x}_{11}) = \sqrt{z_4^2 + z_2^2 + z_5^2}. \)

Intermediate level problem \( P_{12} \) is formulated as:

min \( \bar{x}_{12} \) \( \|w^R_2 \circ (R_{02} - r_{12})\|_2^2 + \|w^Y_2 \circ (Y_{02} - y_{12})\|_2^2 + \) \( \|w^R_{22} \circ (R_{21} - r_{22})\|_2^2 \) \hspace{1cm} (A.49)

subject to \( g_{12}(\bar{x}_{12}) \leq 0 \)

\( h_{12}(\bar{x}_{12}) = 0 \)

Where \( \bar{x}_{12} = [z_7], r_{12} = [z_2], y_{12} = [z_5, z_{11}], R_{12} = [z_6, z_{11}], \bar{x}_{12} = [x_{12}, r_{12}, y_{12}, R_{12}], \)

\( g_{12}(\bar{x}_{12}) = [g_2], h_{12}(\bar{x}_{12}) = [r_{12} - a_{12}(\bar{x}_{12})] \), and \( a_{12}(\bar{x}_{12}) = \sqrt{z_3^2 + z_4^2 + z_5^2}. \) There are two bottom level problems. Problem \( P_{21} \) is formulated as:

min \( \bar{x}_{21} \) \( \|w^R_1 \circ (R_{11} - r_{21})\|_2^2 \) \hspace{1cm} (A.50)

subject to \( g_{21}(\bar{x}_{21}) \leq 0 \)

\( h_{21}(\bar{x}_{21}) = 0 \)

Where \( \bar{x}_{21} = [z_8, z_9, z_{10}, z_{15}], r_{21} = [z_3, z_{11}], \bar{x}_{21} = [x_{21}, r_{21}], g_{21}(\bar{x}_{21}) = [g_3, g_4], \)

\( h_{21}(\bar{x}_{21}) = [r_{21} - a_{21}(\bar{x}_{21})] \), and \( a_{21}(\bar{x}_{21}) = [\sqrt{z_8^2 + z_9^2 + z_{10}^2 + z_{15}^2}, z_{15}]. \) Bottom level problem \( P_{22} \) is formulated as:

min \( \bar{x}_{22} \) \( \|w^R_2 \circ (R_{12} - r_{22})\|_2^2 \) \hspace{1cm} (A.51)

subject to \( g_{22}(\bar{x}_{22}) \leq 0 \)

\( h_{22}(\bar{x}_{22}) = 0 \)
Where $x_{22} = [z_{12}, z_{13}, z_{14}, z_{16}]$, $r_{22} = [z_{6}, z_{11}]$, $\bar{x}_{22} = [x_{21}, r_{22}]$, $g_{22}(\bar{x}_{22}) = [g_5, g_6]$, $h_{22}(\bar{x}_{22}) = [r_{22} - a_{22}(\bar{x}_{22})]$, and $a_{22}(\bar{x}_{22}) = [\sqrt{z_{12}^2 + z_{13}^2 + z_{14}^2 + z_{16}^2}, z_{16}]$. 
A.15 Problem 13 (Geometric optimization problem 3)

This problem is constructed by the author with the geometric programming problem of [Kim00] as a starting point. Six nonlinear inequality constraints and five nonlinear equality constraints have been added. The constraints were constructed after a partitioned problem structure was selected, constraints were tailored to fit the problem's ATC hierarchy. Depending on the choice of parameters $a_i$, local inequality constraints in the sub-problems become active.

Primal problem definition:

$$
\min_{z_1, z_2, \ldots, z_{26}} \quad z_1^2 + z_2^2
$$

subject to

\begin{align*}
g_1 & : z_3^2 + z_4^2 - a_1 z_5^2 \leq 0, \\
g_2 & : z_5^2 + z_6^2 - a_2 z_7^2 \leq 0, \\
g_3 & : z_8^2 + z_9^2 - a_3 z_{11}^2 \leq 0, \\
g_4 & : z_8^2 + z_{10}^2 - a_4 z_{11}^2 \leq 0, \\
g_5 & : z_{12}^2 + z_{15}^2 - a_5 z_{13}^2 \leq 0, \\
g_6 & : z_{12}^2 + z_{15}^2 - a_6 z_{14}^2 \leq 0, \\
g_7 & : z_{17}^2 - a_7 z_{18}^2 \leq 0, \\
g_8 & : z_{16}^2 - a_8 z_{17}^2 \leq 0, \\
g_9 & : z_{16}^2 + z_{17}^2 - a_9 z_{19}^2 \leq 0, \\
g_{10} & : z_{20}^2 + z_{23}^2 - a_{10} z_{22}^2 \leq 0, \\
g_{11} & : z_{21}^2 + z_{22}^2 - a_{11} z_{20}^2 \leq 0, \\
g_{12} & : z_{25}^2 + z_{20}^2 - a_{12} z_{24}^2 \leq 0, \\
h_1 & : z_1^2 - z_3^2 - z_4^2 - z_5^2 = 0, \\
h_2 & : z_2^2 - z_5^2 - z_6^2 - z_7^2 = 0, \\
h_3 & : z_3^2 - z_8^2 - z_9^2 - z_{10}^2 - z_{11}^2 = 0, \\
h_4 & : z_6^2 - z_8^2 - z_{12}^2 - z_{13}^2 - z_{14}^2 - z_{15}^2 = 0, \\
h_5 & : z_8^2 + z_9^2 - z_{16}^2 - z_{17}^2 - z_{18}^2, \\
h_6 & : z_8^2 - z_9^2 - z_{16}^2 - z_{18}^2, \\
h_7 & : z_{11}^2 - z_{16}^2 - z_{19}^2, \\
h_8 & : z_{13}^2 - z_{20}^2 - z_{21}^2 - z_{22}^2, \\
h_9 & : z_{15}^2 - z_{20}^2 - z_{24}^2 - z_{26}^2, \\
z_1, z_2, \ldots, z_{26} & \geq 0.
\end{align*}
Optimal solution:

Case 1: \( a = [20, 20, 20, 20, 20, 20, 20, 20, 20, 20] \): \( z^* = [1.6522, 0.8749, 1.4937, 1.7686, 0.4228, 0.6867, 0.3391, 1.0609, 0.9997, 0.0000, 0.3260, 0.4670, 0.1073, 0.4794, 0.1098, 0.1581, 1.4142, 0.3162, 0.2850, 0.1061, 0.0155, 2.5732, 0.8795, 0.0281, 0.0231, 0.0000] \), \( f(z^*) = 3.4951 \), \( g_a = [1, 2, 4, 5, 6] \).

Case 2: \( a = [10, 10, 10, 10, 20, 1, 1, 1, 1, 1] \): \( z^* = [2.5871, 0.9781, 2.4495, 1.4772, 0.4846, 0.7052, 0.4739, 1.8708, 1.0000, 0.0000, 1.2247, 0.4491, 0.1502, 0.4991, 0.1548, 0.7071, 1.3142, 1.4142, 1.0000, 0.1094, 0.1029, 26.8114, 22.6610, 0.1094, 0.0001, 0.0000] \), \( f(z^*) = 7.6499 \), \( g_a = [1, 2, 5, 6, 7, 8, 9, 11, 12] \).

Case 3: \( a = [10, 10, 3, 10, 10, 20, 1, 1, 1, 1, 1] \): \( z^* = [2.5871, 0.9790, 2.4495, 1.4775, 0.4847, 0.7070, 0.4728, 1.8708, 1.0000, 0.0000, 1.2247, 0.4493, 0.1510, 0.4990, 0.1616, 0.7071, 1.4142, 1.4142, 1.0000, 0.1143, 0.0987, 17.3357, 5.4796, 0.1143, 0.0000, 0.0001] \), \( f(z^*) = 7.6514 \), \( g_a = [1, 2, 3, 5, 6, 7, 8, 9, 11, 12] \).

The partitioned FDT is displayed in Figure A.18, and a possible partition is depicted in Figure A.19, with \( x_{11} = [z_4], x_{12} = [z_7], x_{21} = [z_{10}], x_{22} = [z_{12}, z_{14}], x_{31} = [z_8, z_9, z_{18}], x_{32} = z_{19}, x_{33} = [z_{21}, z_{22}, z_{23}], x_{34} = [z_{24}, z_{25}, z_{26}], a_{11} : [h_1], a_{12} : [h_2], a_{21} : [h_3], a_{22} : [h_4], a_{32} : [h_7], a_{33} : [h_8], a_{34} : [h_9], g_{11} = [g_1], g_{12} = [g_2], g_{21} = [g_3, g_4], g_{22} = [g_5, g_6], g_{31} = [g_7, g_8], g_{32} = [g_9], g_{33} = [g_{10}, g_{11}], g_{34} = [g_{12}], and b_{31} = [h_5, h_6].

Two variable copies are introduced: \( h_{10} : z'_8 = z_8 \), and \( h_{11} : z'_9 = z_9 \).

Top level problem \( P_0 \) can be formulated as:

\[
\begin{align*}
\min_{x_0} \quad & f_0(x_0) + \\
& \|w_{b1}^0 \circ (R_{01} - r_{11})\|_2^2 + \|w_{b2}^0 \circ (R_{02} - r_{12})\|_2^2 + \|w_{b1}^1 \circ (Y_{01} - y_{11})\|_2^2 + \|w_{b2}^1 \circ (Y_{02} - y_{12})\|_2^2 \\
\text{subject to} \quad & h_0(x_0) = 0
\end{align*}
\]  

(A.53)

Where \( x_0 = [], R_{01} = [z_1], R_{02} = [z_2], Y_{01} = Y_{02} = [z_5], x_0 = [R_{01}, Y_{01}, R_{02}, Y_{02}], y_0 = z_5, \) and \( h_0(x_0) = (Y_{01} - Y_{02}) \), and \( f_0(x_0) = z_1^2 + z_2^2 \). There are two first level problems. Problem \( P_{11} \) is formulated as:

\[
\begin{align*}
\min_{x_{11}} \quad & \|w_{b1}^0 \circ (R_{11} - r_{11})\|_2^2 + \|w_{b1}^1 \circ (Y_{01} - y_{11})\|_2^2 + \\
& \|w_{b1}^1 \circ (R_{11} - r_{12})\|_2^2 \\
\text{subject to} \quad & g_{11}(x_{11}) \leq 0 \\
& h_{11}(x_{11}) = 0
\end{align*}
\]  

(A.54)

Where \( x_{11} = [z_4], r_{11} = [z_1], y_{11} = [z_5], R_{11} = [z_3], x_{11} = [x_{11}, r_{11}, y_{11}, R_{11}], \)
Figure A.18: Partitioned FDT for Problem (A.52)

\[ g_{11} (x_{11}) = [g_1], \quad h_{11} (x_{11}) = [r_{11} - a_{11} (x_{11})], \quad a_{11} (x_{11}) = \sqrt{z_3^2 + z_4^2 + z_5^2}. \]

First level problem \( P_{12} \) is formulated as:

\[
\begin{align*}
\min_{x_{12}} & \quad \|w_{12}^R \circ (R_{02} - r_{12})\|_2^2 + \|w_{12}^Y \circ (y_0 - y_{12})\|_2^2 + \\
& \|w_{22}^R \circ (R_{12} - r_{22})\|_2^2 \\
\text{subject to} & \quad g_{12} (x_{12}) \leq 0 \\
& \quad h_{12} (x_{12}) = 0
\end{align*}
\] (A.55)

Where \( x_{12} = [z_7], \quad r_{12} = [z_2], \quad y_{12} = [z_5], \quad R_{12} = [z_6], \quad x_{12} = [x_{12}, r_{12}, y_{12}, R_{12}], \quad g_{12} (x_{12}) = [g_2], \quad h_{12} (x_{12}) = [r_{12} - a_{12} (x_{12})], \quad \text{and} \quad r_{12} (x_{12}) = \sqrt{z_3^2 + z_4^2 + z_5^2}. \]

There are two second level problems. Problem \( P_{21} \) is formulated as:

\[
\begin{align*}
\min_{x_{21}} & \quad \|w_{21}^R \circ (R_{11} - r_{21})\|_2^2 + \|w_{21}^R \circ (R_{21} - r_{31})\|_2^2 + \|w_{32}^R \circ (R_{22} - r_{32})\|_2^2 + \\
& \|w_{33}^R \circ (R_{22} - r_{32})\|_2^2 + \|w_{32}^R \circ (R_{22} - r_{32})\|_2^2 \\
\text{subject to} & \quad g_{21} (x_{21}) \leq 0 \\
& \quad h_{21} (x_{21}) = 0
\end{align*}
\] (A.56)
Where $x_1 = [z_{10}], r_1 = [z_3], R_1 = [z_4], Z_2 = [z_{11}], Y_1 = Y_2 = [z_{16}, z_{17}], \bar{x}_1 = [x_1, r_1, R_1, Y_1, R_2, Y_2], g_1(x_1) = [g_3, g_4], h_1(x_1) = [r_1 - a_1(x_1), Y_1 - Y_2], \quad \text{and} \quad a_1(x_1) = \sqrt{z_{10}^2 + z_{12}^2} + z_{11}^2 + z_{13}^2.

Second level problem $P_22$ is formulated as:

$$
\begin{align*}
\min_{x_{22}} & \quad \|w_2^R \circ (R_{12} - r_{22})\|_2^2 +
\|w_3^R \circ (R_{23} - r_{33})\|_2^2 + \|w_3^Y \circ (Y_{23} - y_{33})\|_2^2 + \|w_4^Y \circ (Y_{24} - y_{34})\|_2^2 \\
\text{subject to} & \quad g_{22}(\bar{x}_{22}) \leq 0 \\
& \quad h_{22}(\bar{x}_{22}) = 0
\end{align*}
$$

(A.57)

Where $x_2 = [z_{12}, z_{14}], r_2 = [z_6], R_2 = [z_{13}], R_24 = [z_{15}], Y_23 = Y_24 = [z_6], \bar{x}_2 = [x_2, R_{23}, Y_{23}, R_{24}, Y_{24}], g_2(x_2) = [g_5, g_6], h_2(x_2) = [r_2 - a_2(x_2), Y_{23} - Y_{24}], \quad \text{and} \quad a_2(x_2) = \sqrt{z_{15}^2 + z_{13}^2} + z_{14}^2 + z_{15}^2. \quad \text{There are four bottom level problems. Problem $P_{31}$ is formulated as:}

$$
\begin{align*}
\min_{x_{31}} & \quad \|w_{31}^R \circ (R_{21} - r_{31})\|_2^2 + \|w_{31}^Y \circ (Y_{21} - y_{31})\|_2^2 \\
\text{subject to} & \quad g_{31}(\bar{x}_{31}) \leq 0 \\
& \quad h_{31}(\bar{x}_{31}) = 0
\end{align*}
$$

(A.58)
Appendix A. ATC-formulation for several problems

Where \( x_{31} = [z_{18}], r_{31} = [z_6, z_5], y_{31} = [z_{16}, z_{17}], x_{31} = [x_{31}, r_{31}, y_{31}], g_{31}(\bar{x}_{31}) = [g_7, g_8], h_{31}(\bar{x}_{31}) = [r_{31} - a_{31}(\bar{x}_{31}), h_{5}, h_{6}], \) and \( a_{31}(\bar{x}_{31}) = [z_8, z_9] \). Problem \( P_{32} \) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_{32}} & \quad \| w^R_{32} \circ (R_{22} - r_{32}) \|_2^2 + \| w^Y_{32} \circ (Y_{22} - y_{32}) \|_2^2 \\
\text{subject to} & \quad g_{32}(\bar{x}_{32}) \leq 0 \\
& \quad h_{32}(\bar{x}_{32}) = 0
\end{align*}
\] (A.59)

Where \( x_{32} = [z_{19}], r_{32} = [z_{11}], y_{32} = [z_{16}, z_{17}], \bar{x}_{32} = [x_{32}, r_{32}, y_{32}], g_{32}(\bar{x}_{32}) = [g_9], h_{32}(\bar{x}_{32}) = [r_{32} - a_{32}(\bar{x}_{32})] \), and \( a_{32}(\bar{x}_{32}) = \sqrt{z_{16}^2 + z_{19}^2} \). Problem \( P_{33} \) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_{33}} & \quad \| w^R_{33} \circ (R_{23} - r_{33}) \|_2^2 + \| w^Y_{33} \circ (Y_{23} - y_{33}) \|_2^2 \\
\text{subject to} & \quad g_{33}(\bar{x}_{33}) \leq 0 \\
& \quad h_{33}(\bar{x}_{33}) = 0
\end{align*}
\] (A.60)

Where \( x_{33} = [z_{21}, z_{22}, z_{23}], r_{33} = [z_{13}], y_{33} = [z_{20}], \bar{x}_{33} = [x_{33}, r_{33}, y_{33}], g_{33}(\bar{x}_{33}) = [g_{10}, g_{11}], h_{33}(\bar{x}_{33}) = [r_{33} - a_{33}(\bar{x}_{33})] \), and \( a_{33}(\bar{x}_{33}) = \sqrt{z_{20}^2 + z_{21}^2 + z_{23}^2} \). Problem \( P_{34} \) is formulated as:

\[
\begin{align*}
\min_{\bar{x}_{34}} & \quad \| w^R_{34} \circ (R_{24} - r_{34}) \|_2^2 + \| w^Y_{34} \circ (Y_{24} - y_{34}) \|_2^2 \\
\text{subject to} & \quad g_{34}(\bar{x}_{34}) \leq 0 \\
& \quad h_{34}(\bar{x}_{34}) = 0
\end{align*}
\] (A.61)

Where \( x_{34} = [z_{24}, z_{25}, z_{26}], r_{34} = [z_{15}], y_{34} = [z_{20}], \bar{x}_{34} = [x_{34}, r_{34}, y_{34}], g_{34}(\bar{x}_{34}) = [g_{12}], h_{34}(\bar{x}_{34}) = [r_{34} - a_{34}(\bar{x}_{34})] \), and \( a_{34}(\bar{x}_{34}) = \sqrt{z_{20}^2 + z_{24}^2 + z_{26}^2} \).
Appendix B

Discussion paper on objective function properties for ATC

B.1 Introduction

This paper discusses the translation of an original non-decomposed problem’s objective function to the objective function of the top-level subproblem in Analytical Target Cascading (ATC) [Kim01a]. The discussion arose during the investigation of two different implementations of ATC on the geometric optimization problem illustrated in [Kim01a], where a two-level ATC decomposition of the original optimization problem was proposed. [Etm02] produces a three-level hierarchy from the same problem. [Tze03] conducts further research on convergence difficulties on this problem observed by [Etm02]. One striking difference between both implementations can be seen, besides the obvious decomposition related issues. The top problem has a different response function for each implementation. This paper proposes a more generic formulation for the top sub-problem in ATC to tackle this issue.

B.2 Differences between [Kim01a] and [Etm02]

Recall the objective function of the original geometric optimization problem of [Kim01a]:
\[ f = z_1^2 + z_2^2. \]
Both [Kim01a] and [Etm02] (and therefore [Tze03]) have similar ATC implementations, except for the definition of the response function of the top problem. [Kim01a] treats variables \( z_1 \) and \( z_2 \) as individual responses in the top-level problem both with targets set to zero: \( R_0 = [z_1, z_2]^T \) and \( T_0 = [0, 0]^T \). Both [Etm02] and [Tze03] assign the objective function of the original problem as the response function of the top-level sub-problem with zero as a target: \( R_0 = z_1^2 + z_2^2 \) and \( T_0 = 0 \).
The impact of these differences can be observed from the local objective function of the top problem. Using a squared \( l_2 \)-norm and a scaling parameter \( s \), as proposed in [Etm02], the objective function for the first case becomes:
\[ f_0 = z_1^2 + z_2^2 + s e_0^y + s e_0^R, \]
while for the second, \( f_0 = (s_1^2 + s_2^2) + s_0 + s_0^R \) is obtained.

In general, the objective functions for each case can be defined as: \( f_0 = f + \varepsilon_0^y + s_0^R \), and \( f_0 = (f - T_f)^2 + s_0^y + s_0^R \) respectively, with \( T_f \) an arbitrary target for the objective function value of the original problem. The difference between both implementations is that [Kim01a] recognizes the need for an objective function of the original problem that expresses deviations from targets, while [Etm02] uses the original objective function as a response function, which in turn has to match an arbitrary target \( T_f \).

To my opinion, the first implementation is to be preferred over the second. The objective of the ATC decomposition is equal to the original objective for a consistent design \( (\varepsilon_0^y = \varepsilon_0^R = 0) \), while for the second ATC objective always differs from the original.

The first approach however limits the set of decomposable problems that can be tackled with ATC because of the required shape of the original objective. Another drawback of the second method is that a poor choice for \( T_f \) implies a limit on the original objective function, since an improvement beyond \( T_f \) is penalized.

### B.3 Proposed top-objective formulation

To find the correct top-objective, we return to the general formulation of an optimization problems with design variables \( \mathbf{x} \), an objective only depending on a selection of design variables \( f(\bar{x}_0) \) (common in ATC), inequality constraints \( \mathbf{g}(\mathbf{x}) \), and equality constraints \( \mathbf{h}(\mathbf{x}) \):

\[
\min_{\mathbf{x}} \quad f(\bar{x}_0) \\
\text{subject to} \quad \mathbf{g}(\mathbf{x}) \leq 0, \quad \mathbf{h}(\mathbf{x}) = 0. \quad (B.1)
\]

With a scaling \( s \) parameter as defined in [Etm02], the ATC relaxation of the problem above is [Kim01a], [Mic03]:

\[
\begin{align*}
\min_{\bar{x}, \mathbf{y}, \varepsilon^R, \varepsilon^y} & \quad f(\bar{x}_0) + \sum_{i=0}^{N-1} \sum_{j \in E_i} s^R_{ij} \varepsilon^R_{ij} + \sum_{i=0}^{N-1} \sum_{j \in E_i} s^y_{ij} \varepsilon^y_{ij} \\
\text{subject to} & \quad \begin{cases}
\sum_{k \in C_{ij}} \| \mathbf{R}^j_{(i+1)k} - \mathbf{R}^j_{(i+1)k} \| \leq \varepsilon^R_{ij} \\
\sum_{k \in C_{ij}} \| \mathbf{S}^j_{ik} \mathbf{y}^j_{(i+1)k} - \mathbf{y}^j_{(i+1)k} \| \leq \varepsilon^y_{ij}
\end{cases} \\
\text{where} & \quad \begin{cases}
\mathbf{g}_{ij}(\bar{x}_{ij}) \leq 0 \\
\mathbf{h}_{ij}(\bar{x}_{ij}) = 0 \\
\mathbf{R}^j_{ij} = \mathbf{r}_{ij}(\bar{x}_{ij}) \\
\bar{x}_{ij} = [\bar{x}_{ij}, \mathbf{y}_{ij}, \mathbf{R}^j_{(i+1)k_1}, \ldots, \mathbf{R}^j_{(i+1)k_r}]^T
\end{cases} \\
\forall j \in E_i, i = 0, 1, \ldots, N - 1
\end{align*} \quad (B.2)
\]

Where \( \bar{x}, \mathbf{y}, \varepsilon^R \) and \( \varepsilon^y \) are the collections of local and linking variables and inconsistencies with respect to response targets and linking targets: \( \bar{x} = [\bar{x}_0, \ldots, \bar{x}_{(N-1)E_{N-1}}]^T \), \( \mathbf{y} = [\mathbf{y}_1, \ldots, \mathbf{y}_{(N-1)E_{N-1}}]^T \), \( \varepsilon^R = [\varepsilon^R_0, \ldots, \varepsilon^R_{(N-1)E_{N-1}}]^T \), and \( \varepsilon^y = [\varepsilon^y_0, \ldots, \varepsilon^y_{(N-1)E_{N-1}}]^T \).
**B.4. Example Problem**

\( C_{ij} \) is the set of children of element \( e_{ij} \), \( E_i \) is the number of elements at level \( i \), and \( N \) is the number of levels in the problem hierarchy. The above can be separated into target cascading sub-problems. The decomposition of the relaxed AAO problem has a top-problem defined by:

\[
\min_{\bar{x}, y} f(\bar{x}) + s_R^{\bar{x}} + s_y y \\
\text{subject to} \\
\sum_{k \in C_0} \| R^0_{(1)k} - R^1_{(1)k} \| \leq \varepsilon_R^0 \\
\sum_{k \in C_0} \| S_k y^0_1 - y^1_{(1)k} \| \leq \varepsilon_y^R \\
g_0(\bar{x}) \leq 0 \\
h_0(\bar{x}) = 0
\tag{B.3}
\]

where \( \bar{x} = [x_0, R^0_{(1)k_1}, \ldots, R^0_{(1)c_0}] \)

A general expression for the top-level objective in ATC can be defined. With \( s = s_R^0 = s_y^0 \), the top-problem objective becomes:

\[
f_0 = f + s_0^{\bar{x}} + s_0^{y} \tag{B.4}
\]

Where [Kim01a] only allowed distance like functions for \( f \), any function may be inserted in this formulation, making it applicable to a broader set of objective functions. [Etm02], and [Tze03] however implemented an improper top-objective function.

### B.4 Example Problem

To illustrate the approach presented in the previous section and its differences with the approaches by [Kim01a], [Etm02], and [Tze03] consider problem 39 from [Hoc81]:

\[
\min_{z_1, z_2, z_3} -z_1 \\
\text{subject to} \\
h_1 : z_2 - z_1^3 - z_3^2 = 0, \\
h_2 : z_1^2 - z_2 - z_3^2 = 0
\tag{B.5}
\]

with \( z^* = [1, 1, 0, 0]^T \), \( f(z^*) = -1 \).

This problem can be decomposed into a two-level ATC problem with two bottom-level problems with \( z_1 \) as target and sharing variable \( z_2 \). \( h_1 \) is the response function of one bottom problem and \( h_2 \) defines the response of the other. Bottom problem one has \( z_3 \) as local variable, while the second has \( z_4 \) as a local variable. The top problem coordinates the targets and linking variable while maximizing \( z_1 \).

In contrast to the example problems illustrated in [Kim01a], this problem’s objective is not a target match, instead it aims to maximize \( z_1 \). Using the general ATC sub-problem
formulation from [Kim01a], we would not be able to define a top-problem, while the second approach requires the definition of $T_f$. To avoid limitation of the objective, $T_f$ should be chosen to be $-\infty$. This choice causes convergence difficulties of the top-problem, since the inconsistency terms in the objective become negligible with respect to minimizing the deviations from top level targets.

Using the suggested approach the top-problem can be defined as:

$$\min_{R_1, y_1, \varepsilon_0^R, \varepsilon_0^y} - R_1 + s\varepsilon_0^R + s\varepsilon_0^y$$

subject to

$$\|y_1 - y_{11}\| + \|y_1 - y_{12}\| \leq \varepsilon_0^y$$

$$\|R_1 - R_{11}\| + \|R_1 - R_{12}\| \leq \varepsilon_0^R$$

where $R_1 := z_1$, $y_1 := z_2$.

The first bottom problems is defined as:

$$\min_{x_{11}, y_{11}} \|R_{11} - R_{11}^U\| + \|y_{11} - y_{11}^U\|$$

where $x_{11} := z_3$, $y_{11} := z_2$, $R_{11} = r_{11}(x_{11}, y_{11}) = \sqrt{z_2 - z_3^2}$.

The second problem is:

$$\min_{x_{12}, y_{12}} \|R_{12} - R_{12}^U\| + \|y_{12} - y_{12}^U\|$$

where $x_{12} := z_4, y_{12} := z_2$, $R_{12} = r_{12}(x_{12}, y_{12}) = \sqrt{z_2 + z_3^2}$.

### B.5 Geometric Optimization Problem

Consider a second problem, the geometric optimization problem of [Kim01a], [Etm02], and [Tze03]. The results presented in [Tze03] on this geometric optimization problem are based on the assumption of the top-level response being the original objective function, with a target for this objective function $T_f = 0$. An implementation in MatLab of the scheme III (both bottom branches are solved in a nested fashion) ATC coordination process for this problem is constructed. Both objective functions are analyzed in experiments similar to [Tze03]. The first case uses $f_0 = (z_1^2 + z_2^2)^2 + s\varepsilon_0^y + s\varepsilon_0^R$ as top-level objective, and the second analyzed case uses the proposed objective $f_0 = z_1^2 + z_2^2 + s\varepsilon_0^y + s\varepsilon_0^R$. Scaling factors for the first are set similar to the values investigated in [Tze03] and range from $10^2$ to $10^8$.

For the proposed top level objective, we try to find a similar scaling between the objective $f = z_1^2 + z_2^2$ and the inconsistency terms. The objective in this case is a factor $z_1^2 + z_2^2$ smaller than the objective of the first case ($f = (z_1^2 + z_2^2)^2$). At the AAO optimum,
the value for $f = z_1^2 + z_2^2$ is approximately 17.6. To have comparable scaling between the objective and inconsistencies, scaling factors $s$ for the experiments with the new objective are chosen 17.6 times smaller than factors used in the first set of experiments for clearer comparison: $10^2/17.6$ to $10^3/17.6$.

In both cases, the proposed truncation of communicated values is implemented with $\text{trunc} = 6$. The ATC process was said to have converged when the top-problem has converged: $\|x_k^i - x_{k-1}^i\| < \text{tol}$. Three different termination tolerances ($\text{tol}$) have been analyzed: $10^{-4}$, $10^{-6}$ and $10^{-8}$. Optimizer tolerance is set to $10^{-7}$. Sub-problems are formulated in the general form presented in [Mic03], thus substituting the $\varepsilon$’s in the objective function by the norms of the difference between computed child targets and their realizations. Realizations to the parent are treated as decision variables, and the response functions are added to the set of equality constraints.

As performance measures, again the accuracy of the final ATC solution is regarded, as well as the number of ATC iterations $K$. Results with both formulations are displayed in Figure B.1. Numerical results are presented in the final section of this paper. The only major difference is the predicted horizontal shift for both the accuracy and computational cost graphs. For some values of the scaling parameter, both cases differ in their convergence behavior. Some solutions are more accurate for one of the two cases at the expense of higher computational cost. For small scaling factors ($10^2$ and $10^3$ in case 1) accuracy of the ATC solutions for both cases differ systematically, possibly due to a different ratio $(z_1^2 + z_2^2)$ between the two scaling factors. This difference is caused by the relaxation implied by the ATC approach. For smaller scaling factors (greater relaxation), the values for $z_1$ and $z_2$ are not equal to the optimal values of the AAO solution, rendering a different value for the scaling ratio $(z_1^2 + z_2^2)$.

The observations presented in [Tze03] are still valid. The theoretical optimal solution of the ATC problem is still the same as the AAO solution. However, my MatLab implementation of ATC does not have convergence difficulties. Experiments without truncation at an optimizer tolerance of $10^{-7}$ show no significant differences with truncated results from [Tze03]. The MatLab implementation does not have the convergence difficulties observed in the $\chi$-Python-MatLab implementation. When altering the MatLab-implementation to match [Tze03]’s formulation, no convergence difficulties are observed. It is possible that the observed convergence issues are caused by the $\chi$-Python-MatLab setting, instead of the sensitivity of the ATC process to small numerical inaccuracies. Numerical results are presented in the final section.
B.6 Conclusions and Discussion

The ATC top-objective as proposed by [Etm02], and later [Tze03], implies a difference between the objective of the original problem and the objective of a consistent ATC design. A proposal for the formulation of the top-objective that does match the original objective is presented. Comparison of both approaches by means of experiments on the geometric programming test example of [Etm02] does not show a difference in obtained accuracy, but only a smaller optimal value for the scaling parameter. A second observation is the insensitivity of the ATC process implemented only in MatLab with respect to numerical inaccuracies. Where the $\chi$-Python-MatLab implementation of [Etm02] showed a sensitivity of the convergence to small numerical inaccuracies ([Tze03]), my own MatLab implementation is insensitive to these numerical inaccuracies, suggesting that the $\chi$-Python-MatLab causes the observed convergence difficulties. This needs to be further investigated.
This paper’s aim is to propose two discussions. The first on whether it is possible to copy the original objective into the ATC top-problem instead of defining new targets and changing the objective. What impact would these changes have on the ATC process considering the top-level objective no longer is quadratic, but takes on the characteristics of the original objective? (N.B. Technically the approach of [Etm02] also brings characteristics of the original objective into the ATC top-problem by defining the top-level response as the original objective.)

Secondly the results in this paper do not concur with the convergence difficulties observed in [Tze03]. The accuracy of the ATC process implemented in MatLab does not appear to be sensitive to small numerical inaccuracies. Are these sensitivities caused by the χ-Python-MatLab implementation?

B.7 Appendix: Results for the Geometric Optimization Problem

This section states results obtained with the MatLab implementation of the geometric optimization problem. As parameters, the scaling parameter $s$ and termination tolerance $tol$ are used. Performance of the ATC algorithm is expressed in an accuracy measure, and the number of required ATC iterations $K$ as an indicator for computational cost. The accuracy measure is defined as the squared $l_2$-norm of the difference between the known optimum and the optimum computed with ATC:

$$\text{acc} = \|x^K - x^*\|_2^2 \quad (B.9)$$

The first element in the cells of the following tables contains the value for the solution accuracy $acc$, and the second the number of ATC iterations $K$.

### Geometric Problem, scheme III, MatLab, formulation [Mic03], trunc=6

<table>
<thead>
<tr>
<th>Objective as [Etm02], [Tze03]</th>
<th>Proposed objective</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>termination tolerance $tol$</strong></td>
<td><strong>termination tolerance $tol$</strong></td>
</tr>
<tr>
<td>$s$</td>
<td>1e-4</td>
</tr>
<tr>
<td>1e2</td>
<td>1.57094</td>
</tr>
<tr>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td>1e3</td>
<td>0.09390</td>
</tr>
<tr>
<td>10</td>
<td>26</td>
</tr>
<tr>
<td>1e4</td>
<td>0.02824</td>
</tr>
<tr>
<td>10</td>
<td>59</td>
</tr>
<tr>
<td>1e5</td>
<td>0.03130</td>
</tr>
<tr>
<td>10</td>
<td>14</td>
</tr>
<tr>
<td>1e6</td>
<td>0.03179</td>
</tr>
<tr>
<td>10</td>
<td>14</td>
</tr>
<tr>
<td>1e7</td>
<td>0.03183</td>
</tr>
<tr>
<td>10</td>
<td>14</td>
</tr>
</tbody>
</table>
### Geometric Problem, ATC scheme III, formulation [Etm02]

\( \chi \)-Mat.-Py. [Etm02]/[Tze03], \( \text{trunc}=6 \)

<table>
<thead>
<tr>
<th>( s )</th>
<th>( 1 \times 10^{-4} )</th>
<th>( 1 \times 10^{-6} )</th>
<th>( 1 \times 10^{-8} )</th>
<th>( 1 \times 10^{-10} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 1 \times 10^{2} )</td>
<td>0.87287</td>
<td>0.87507</td>
<td>0.87564</td>
<td>0.87566</td>
</tr>
<tr>
<td>( 1 \times 10^{3} )</td>
<td>0.04929</td>
<td>0.03826</td>
<td>0.03727</td>
<td>0.03720</td>
</tr>
<tr>
<td>( 1 \times 10^{4} )</td>
<td>0.03198</td>
<td>0.00617</td>
<td>0.00671</td>
<td>( 1 \times 10^{-5} )</td>
</tr>
<tr>
<td>( 1 \times 10^{5} )</td>
<td>0.03506</td>
<td>0.03465</td>
<td>0.00582</td>
<td>0.00054</td>
</tr>
<tr>
<td>( 1 \times 10^{6} )</td>
<td>0.03540</td>
<td>0.03524</td>
<td>0.03514</td>
<td>( 1 \times 10^{-10} )</td>
</tr>
<tr>
<td>( 1 \times 10^{7} )</td>
<td>0.03520</td>
<td>0.03525</td>
<td>0.03515</td>
<td>( 1 \times 10^{-10} )</td>
</tr>
<tr>
<td>( 1 \times 10^{8} )</td>
<td>0.03540</td>
<td>0.03524</td>
<td>0.03515</td>
<td>( 1 \times 10^{-10} )</td>
</tr>
</tbody>
</table>

#### MatLab, no truncation

<table>
<thead>
<tr>
<th>( s )</th>
<th>( 1 \times 10^{-4} )</th>
<th>( 1 \times 10^{-6} )</th>
<th>( 1 \times 10^{-8} )</th>
<th>( 1 \times 10^{-10} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 1 \times 10^{2} )</td>
<td>0.87289</td>
<td>0.87512</td>
<td>0.87564</td>
<td>0.87570</td>
</tr>
<tr>
<td>( 1 \times 10^{3} )</td>
<td>0.04933</td>
<td>0.03828</td>
<td>0.03728</td>
<td>0.03719</td>
</tr>
<tr>
<td>( 1 \times 10^{4} )</td>
<td>0.03212</td>
<td>0.00464</td>
<td>0.00067</td>
<td>0.00055</td>
</tr>
<tr>
<td>( 1 \times 10^{5} )</td>
<td>0.03513</td>
<td>0.03473</td>
<td>0.00282</td>
<td>0.00015</td>
</tr>
<tr>
<td>( 1 \times 10^{6} )</td>
<td>0.03545</td>
<td>0.03529</td>
<td>0.03523</td>
<td>0.00366*</td>
</tr>
<tr>
<td>( 1 \times 10^{7} )</td>
<td>0.03554</td>
<td>0.03529</td>
<td>0.03519</td>
<td>0.03544*</td>
</tr>
<tr>
<td>( 1 \times 10^{8} )</td>
<td>0.03545</td>
<td>0.03529</td>
<td>0.03519</td>
<td>0.03545*</td>
</tr>
</tbody>
</table>

Elements indicated with a (*) required an optimizer tolerance of \( 10^{-9} \) for the ATC process to converge.
Appendix C

Notational differences

The optimization problem notation presented in this report is different to previous formulations of the ATC process. Consider the AAO problem statement for ATC:

$$\begin{align*}
\min_{\bar{x}_0, \ldots, \bar{x}_{1p}} & \quad \sum_{i=0}^1 \sum_{j \in E_i} f_{ij}(\bar{x}_{ij}), \\
\text{subject to} & \quad g_{ij}(\bar{x}_{ij}) \leq 0, \\
& \quad h_{ij}(\bar{x}_{ij}) = 0, \\
& \quad r_{ij} - a_{1j}(\bar{x}_{ij}) = 0, \\
& \quad Y_{0j} - Y_{1j} = 0, \\
& \quad Y_{01} - Y_{0j} = 0, \\
& \quad R_{0j} - r_{1j} = 0, \\
\forall i, j \in E_i,
\end{align*}$$

(C.1)

where $\bar{x}_0 = [x_0, R_{01}, Y_{01}, \ldots, R_{0p}, Y_{0p}]$, $\bar{x}_{1j} = [x_{1j}, r_{1j}, y_{1j}]$. Furthermore, $E_i$ is the set of elements at level $i$ ($E_0 = 0$ and $E_1 = [1, \ldots, p]$). Each element has a set of local constraints $[g_{ij}, h_{ij}]$ and a response analysis functions $a_{1j}$ with which responses $r_{ij}$ to top-level targets $R_{0j}$ are computed ($R_{0j} = r_{ij} = a_{1j}(\bar{x}_{ij})$). Additionally copies of linking targets are introduced $Y_{01} = y_{11}, \ldots, Y_{0p} = y_{1p}$, as well as coupling constraints $Y_{01} = Y_{0j}$ forcing the top-level copies to match.

This notation is slightly different from the notations used in previous work on ATC. Only the differences with the most recent notation presented in [Mic04b] are discussed. The most recent problem formulation of the manipulated optimization problem is (C.2).
\[
\min_{y_{10}^0, \bar{x}_0, \ldots, \bar{x}_{1p}} \|R^0_0 - T\|,
\]
subject to
\[
g_{ij}(\bar{x}_{ij}) \leq 0, \quad \forall i, j \in \mathcal{E}_i,
\]
\[
h_{ij}(\bar{x}_{ij}) = 0, \quad \forall i, j \in \mathcal{E}_i,
\]
\[
S^y_{ij}y^0_{10} - y^1_{1j} = 0, \quad j \in \mathcal{E}_1,
\]
\[
R^0_{ij} - R^1_{1j} = 0, \quad j \in \mathcal{E}_1,
\]
(C.2)

where \(R^i_{ij} = r_{ij}(\bar{x}_{ij})\), \(\bar{x}_0 = [x^0_{ij}, R^0_{ij}, \ldots, R^0_{1p}]\), and \(\bar{x}_{ij} = [x^1_{ij}, y^1_{ij}]\). In their case, response analysis functions \(r_{ij}\) define the response \(R^i_{ij}\) of each element \(e_{ij}\) to a target \(R^{(i-1)}_{ij}\) received from the parent, and one master copy of linking targets \(y_{10}\), and a number of child copies of linking targets \(y^1_{1j}, \ldots, y^1_{1j}\) are introduced. With a binary selection matrix \(S^y_{ij}\) defining what part of the master linking variable copy is relevant to element \(e_{1j}\), additional coupling constraints forcing the linking variable copies equal are \(S^y_{ij}y^0_{10} = y^1_{1j}\).

Another obvious difference is the use of super-scripts differentiating between two copies of the same variable. The new notation makes this distinction by differentiating between upper and lower case symbols and the (first) level index of the subscript. This last distinction is not necessary, but provides insight in the allocation of variables during partitioning.

The new notation also allows any separable objective, where the previous notation only allowed an objective for the top-level element of the form \(\|R^0_0 - T\|\), which is a subclass of separable objectives. For this special sub-class of objectives, convergence has been proven in [Mic03]. Other sources on ATC allow the use of local targets for each element, these local targets are included in the objective as separable squared deviation norms [Hul03, Tze03, Kok02], convergence has not been proven for their notation, nor for the new notation. However, it is expected that the ATC process converges under convexity assumptions for problems with separable objectives because the manipulated problem (C.1) is also convex. Furthermore, separable objectives can be reformulated to local target norms, and these local targets can be redirected through the ATC hierarchy and coordinated by the top-level element. This top level element receives all local targets and sends them throughout the hierarchy. If all local targets are coordinated by the top element, the problem with initially a separable objective is reformulated to a problem with an augmented top-level objective coordinating both global and local targets.

Another difference is the introduction of additional copies of linking variables. The notation of [Mic04b] only introduces one master vector of linking variables \(y^0_{10}\), and uses selection matrices \(S^y_{ij}\) to indicate what components of this vector are relevant to element \(e_{1j}\). The new notation introduces multiple master copies, one for each sub-problem, \(\bar{Y}_0 = [Y_{01}, \ldots, Y_{0p}]\), and forcing these to match. The difference between
both approaches is that dimensionality is added by the new approach, both more variables and constraints are introduced. Theoretically the previous notation should be preferred, but the new coordination method, presented in Chapter 4 can be derived more elegantly by introducing these extra variable copies. Note that it is possible to adapt the coordination method to the formulation of [Mic04b], this however adds complexity to the formulation of the coordination method.

An additional difference also clarifies the notation of the coordination method: response variable copies $r_{1j}$ are introduced and used as optimization variables, whereas [Mic04b] suggests to use them as embedded definitions.
Appendix D

Augmented Lagrangian relaxation for ATC

D.1 General augmented Lagrangian penalty function

Consider a general optimization problem of the form:

$$\begin{align*}
\min_{x \in X} & \quad f(x), \\
\text{subject to} & \quad h_i(x) = 0 \quad \forall i,
\end{align*}$$

(D.1)

with $x \in \mathbb{R}^n$ the vector of design variables, $\mathbb{R}^n$ the $n$-dimensional real space, $X$ is the feasible space of $x$, $f$ the problem’s objective function and $h_i(x)$ the $i$-th equality constraint, $i = 1, \ldots, N$. The equality constraint can be rewritten to an exact augmented Lagrangian penalty function [Ber95] which is added to the original objective. The function to be minimized $L$ is called the augmented Lagrangian of Problem (D.1). The relaxed optimization problem can be defined as:

$$\min_{x,v,w} L(x, v, w) = f(x) - \sum_{i=1}^{N} v_i h_i(x) + 2 \sum_{i=1}^{N} w_i h_i(x)^t h_i(x),$$

(D.2)

with $v_i$ the Lagrange multiplier associated with equality constraint $h_i(x)$, $w_i$ the positive penalty parameter associated with constraint $h_i$, $v = [v_1, \ldots, v_N]$, and $w = [w_1, \ldots, w_N]$. Instead of solving the problem all at once, an iterative strategy is proposed to solve the minimization problem [Ber95]. A minimization problem is solved for $x$ while treating values of both Lagrange multipliers $v$ and penalty parameters $w$ as constants. After the minimization of the Lagrangian, convergence is checked. If the sequence has not converged, updates are computed for $v$ and $w$. The algorithm is presented in Figure D.1.
Appendix D. Augmented Lagrangian relaxation for ATC

1. Set $k = 1$, $v^1$, $w^1$.

2. While fixing $v^k$ and $w^k$, obtain $x^k$ by solving:
   \[
   \min_{x^k \in \mathcal{X}} \ f(x^k) - \sum_{i=1}^{N} v^k_i h_i(x^k) + 2 \sum_{i=1}^{N} w^k_i h_i(x^k)^t h_i(x^k)
   \]

3. Check convergence. If converged, set $K = k$ and stop.
   Optimal solution: $(x^*, v^*) = (x^K, v^K)$.
   Otherwise increase counter $k = k + 1$, update $v^k$ and $w^k$, and return to step 2.

Figure D.1: Algorithm for solving augmented Lagrangian Problem (D.2)

Update strategies for both the Lagrange multiplies $v$ and penalty parameters $w$ exist. Both a linear and second order update of the Lagrange multipliers $v$ are available from [Ber95]:

linear: \[
v^{k+1} = v^k + 4(w^k)^t h(x^k),
\]

second order: \[
v^{k+1} = v^k + (B^k)^{-1} h(x^k),
\]

with $B^k = \nabla h(x^k)^t \{\nabla^2_{xx} L_{w^k}(x^k, v^k)\}^{-1} \nabla h(x)$, and $\nabla^2_{xx} L_{w^k}(x^k, v^k)$ denoting the Hessian with respect to $x$ of the Lagrangian function $L_{w^k}$ with fixed $w^k$. Note that the linear update strategy can be separated for each equality constraint $h_i$: $v^{k+1}_i = v^k_i + 4w^k_i h_i(x^k)$. The second order approach does not allow a decoupling of the update strategy.

For the penalty parameters $w$, Bertsekas proposes the following [Ber95]:

\[
w^{k+1} = \begin{cases} 
\beta w^k & \text{if } ||h(x^k)|| > \gamma ||h(x^{k-1})||, \\
w^k & \text{if } ||h(x^k)|| \leq \gamma ||h(x^{k-1})||,
\end{cases}
\]

with $5 \leq \beta \leq 10$ and typically $\gamma = 0.25$. Note that the above update can be separated with respect to $w^{k+1}_i$:

\[
w^{k+1}_i = \begin{cases} 
\beta_i w^k_i & \text{if } |h_i(x^k)| > \gamma_i |h_i(x^{k-1})|, \\
w^k_i & \text{if } |h_i(x^k)| \leq \gamma_i |h_i(x^{k-1})|.
\end{cases}
\]
D.2 Augmented Lagrangian penalty function for ATC

Consider a $N + 1$ level hierarchic feasible separable optimization problem solvable with ATC (Eqn. (3.3)) with $p_i$ elements on level $i$:

$$
\min_{s,x} \sum_{i=1}^{p} f_{ij}(s, x_{ij}),
$$

subject to

$$
g_{ij}(s, x_{ij}) \leq 0, $$
$$h_{ij}(s, x_{ij}) = 0, $$
$$\forall i, j \in E_i,$$

with $x = [x_0, \ldots, x_{Np_i}]$. Problem (D.7) can be rewritten in terms of local functions $[f_{ij}(x_{ij}), g_{ij}(x_{ij}), h_{ij}(x_{ij})]$, and a set of coupling constraints regarding response functions $R_{ik} = r_{(i+1)k}$, and linking variables $Y_{ik} = y_{(i+1)k}$:

$$
\min_{x} \sum_{i=0}^{N} \sum_{j \in E_i} f_{ij}(\bar{x}_{ij}),
$$

subject to

$$
g_{ij}(\bar{x}_{ij}) \leq 0, $$
$$h_{ij}(\bar{x}_{ij}) = 0, $$
$$Y_{ik} - y_{(i+1)k} = 0, $$
$$R_{ik} - r_{(i+1)k} = 0, $$
$$\forall i, j \in E_i, k \in C_{ij},$$

Coupling constraints in Equation (D.8) can be relaxed by adding an augmented Lagrangian penalty function to the objective, similar to Equation (D.2):

$$
\min_{x,v,w} \sum_{i=0}^{N} \sum_{j \in E_i} f_{ij}(\bar{x}_{ij}) +
\sum_{i=0}^{N} \sum_{j \in E_i} -v_{ij}^R \circ (R_{(i+1)j} - r_{ij}) + \sum_{i=1}^{N} \sum_{j \in E_i} -v_{ij}^Y \circ (Y_{(i+1)j} - y_{ij}) +
2 \sum_{i=1}^{N} \sum_{j \in E_i} \|w_{ij}^R \circ (R_{ik} - r_{(i+1)k})\|_2^2 + 2 \sum_{i=1}^{N} \sum_{j \in E_i} \|w_{ij}^Y \circ (Y_{ik} - y_{(i+1)k})\|_2^2,
$$

subject to

$$
g_{ij}(\bar{x}_{ij}) \leq 0, $$
$$h_{ij}(\bar{x}_{ij}) = 0, $$
$$\forall i, j \in E_i,$$

with primal variables $x = [x_0, \ldots, x_{Np_i}]$, and dual variables $v = [v_{11}^R, \ldots, v_{Np_i}^R, v_{11}^Y, \ldots, v_{Np_i}^Y]$, and $w = [w_{11}^R, \ldots, w_{Np_i}^R, w_{11}^Y, \ldots, w_{Np_i}^Y]$. Similar to the present ATC partitions, the relaxed problem of Equation (D.10) can
Appendix D. Augmented Lagrangian relaxation for ATC

be separated into a dual-feasible structure. Dual variables $\mathbf{v}$ and $\mathbf{w}$ are assigned to the dual master problem, while the dual bottom problem is defined as a multi-level feasible decomposition common to ATC. The dual master problem computes updates of the dual variables using update mechanisms of Equations (D.3), (D.4), or (D.6) while treating primal variables as parameters.

The feasible decomposition is obtained by fixing subsets of primal variables associated with a sub-problems parent or children. ATC sub-problem $P_{ij}$ for the augmented Lagrangian relaxation is defined as:

$$
\begin{align*}
\min_{\mathbf{x}_{ij}} & \quad f_{ij}(\mathbf{x}_{ij}) + \mathbf{v}_{ij}^R \cdot \mathbf{r}_{ij} + \mathbf{v}_{ij}^Y \cdot \mathbf{y}_{ij} + \\
& \quad \left\| \mathbf{w}_{ij}^R \circ (\mathbf{R}_{(i-1)j} - \mathbf{r}_{ij}) \right\|_2^2 + \left\| \mathbf{w}_{ij}^Y \circ (\mathbf{y}_{(i-1)j} - \mathbf{y}_{ij}) \right\|_2^2 + \\
& \quad \sum_{k \in C_{ij}} \left\| \mathbf{w}_{(i+1)k}^R \circ (\mathbf{r}_{ik} - \mathbf{r}_{(i+1)k}) \right\|_2^2 + \\
& \quad \sum_{k \in C_{ij}} \left\| \mathbf{w}_{(i+1)k}^Y \circ (\mathbf{y}_{ik} - \mathbf{y}_{(i+1)k}) \right\|_2^2,
\end{align*}
$$

$$
P_{ij}:
\begin{align*}
\text{subject to} & \quad \mathbf{g}_{ij}(\mathbf{x}_{ij}) \leq \mathbf{0}, \\
& \quad \mathbf{h}_{ij}(\mathbf{x}_{ij}) = \mathbf{0}, \\
& \quad \forall i, j \in \mathcal{E}_i.
\end{align*}
$$

The difference between the augmented Lagrangian formulation and the external penalty formulation can be observed in the linear terms associated with dual variables $\mathbf{v}$. The remainder of the augmented Lagrangian sub-problem formulation is identical to the external penalty formulation.

An algorithm for solving problems with the augmented Lagrangian ATC formulation is similar to the algorithm of Figure D.1, except step 2 is replaced by solving the feasible decomposed ATC problem.
Appendix E

Derivation of sensitivity equations

Consider an optimization problem with \( n \) optimization variables \( z \), and \( n^q \) input parameters \( q \). The problem has an objective \( f' \) that depends on both variables and parameters: \( f' = f'(z, q) \). The problem is only subjected to a set of equality constraints which are represented by a column vector \( h \). The equality constraints only depend on optimization variables, and not on parameters: \( h = h(z) \). The problem in its primal form is:

\[
\begin{align*}
\min_{z \in \mathbb{R}^n} & \quad f'(z, q), \\
\text{subject to} & \quad h(z) = 0.
\end{align*}
\] (E.1)

For problems with inequality constraints, active inequality constraints should be included in \( h \), and the inactive should be (temporarily) removed from the problem. The Lagrangian function \( L \) of Problem (E.1) after dropping the dependency notation is:

\[
L = f' + \lambda^T h,
\] (E.2)

with \( \lambda \) the column vector of \( m \) Lagrange multipliers associated with the \( m \) equality constraints \( h \). A solution to optimization problem (E.1) can be computed by solving the necessary Karush-Kuhn-Tucker (KKT) conditions for optimality at the solution \((z^*, \lambda^*)\) [Pap00]. These KKT-conditions are:

\[
\left. \frac{\partial L}{\partial z} \right|_{z=z^*} = \left. \frac{\partial f'}{\partial z} \right|_{z=z^*} + \lambda^* \left. \frac{\partial h}{\partial z} \right|_{z=z^*} = 0,
\] (E.3)

\[
\left. g \right|_{z=z^*} = 0,
\] (E.4)

where \( \frac{\partial f'}{\partial z} = \left[ \frac{\partial f'}{\partial z_l} \right]_{1 \times n} \), \( \frac{\partial h}{\partial z} = \left[ \frac{\partial h_k}{\partial z_l} \right]_{m \times n} \), and \( \left. \frac{\partial F}{\partial z} \right|_{z=z^*} \) stands for the partial derivative of function \( F \) evaluated at \( z^* \).
These conditions must remain satisfied, for any change in parameters to remain feasible. The result of full differentiation of the KKT-conditions with respect to parameters $q$ should equal zero, under the assumption that the set of active inequality constraints does not change. Therefore:

$$\frac{d}{dq} \frac{\partial L}{\partial z} = \frac{\partial f'}{\partial q} \frac{\partial L}{\partial z} + \frac{\partial}{\partial q} \left( \lambda^T \frac{\partial h}{\partial z} \right) = 0,$$

(E.5)

$$\frac{dh}{dq} = 0,$$

(E.6)

with:

$$\frac{d}{dq} \frac{\partial f'}{\partial z} = \frac{\partial^2 f'}{\partial q \partial z} + \frac{\partial^2 f'}{\partial z^2} \frac{dz}{dq},$$

(E.7)

$$\frac{d}{dq} \left( \lambda^T \frac{\partial h}{\partial z} \right) = d\lambda^T \frac{\partial h}{\partial q} + \lambda^T \frac{d}{dq} \frac{\partial h}{\partial z}$$

$$= \frac{\partial h^T}{\partial z} \frac{d\lambda}{dq} + \lambda^T \left( \frac{\partial^2 h}{\partial q \partial z} \right) \frac{dz}{dq}$$

$$= \frac{\partial h^T}{\partial z} \frac{d\lambda}{dq} + \sum_{i=1}^{m} \lambda_i \frac{\partial^2 h_i}{\partial q \partial z} \frac{dz}{dq} + \left( \sum_{i=1}^{m} \lambda_i \frac{\partial^2 h_i}{\partial z^2} \right) \frac{dz}{dq},$$

(E.8)

$$\frac{dh}{dq} = \frac{\partial h}{\partial q} + \frac{\partial h}{\partial z} \frac{dz}{dq}. $$

(E.9)

All partial derivatives of constraints $h$ with respect to input parameters $q$ are zero, because the constraints do not depend on these parameters. Substitution of the above relations in (E.5) and (E.6) yields:

$$\frac{\partial^2 f'}{\partial q \partial z} + \frac{\partial^2 f'}{\partial z^2} \frac{dz}{dq} + \frac{\partial h^T}{\partial z} \frac{d\lambda}{dq} + \left( \sum_{i=1}^{m} \lambda_i \frac{\partial^2 h_i}{\partial z^2} \right) \frac{dz}{dq} = 0,$$

(E.10)

$$\frac{\partial h}{\partial z} \frac{dz}{dq} = 0.$$

(E.11)

With these expressions and by defining $\frac{dz}{dq}_{|z=z^*} = \frac{dx^*}{dq}$, the following system of $(n + m)$ equations can be constructed (matrix dimensions are inscribed for clarity):

$$\begin{bmatrix}
    \frac{\partial^2 f'}{\partial z^2} + \sum_{i=1}^{m} \lambda_i \frac{\partial^2 h_i}{\partial z^2} \\
    \frac{\partial h}{\partial z} \\
    \frac{d\lambda}{dq} \\
\end{bmatrix}_{n \times (n+m)} \begin{bmatrix}
    \frac{dx^*}{dq} \\
    \frac{dz^*}{dq} \\
\end{bmatrix}_{m \times n} = \begin{bmatrix}
    \frac{\partial^2 f'}{\partial q \partial z} \\
    \frac{\partial h}{\partial q} \\
    \frac{d\lambda^*}{dq} \\
\end{bmatrix}_{m \times (n+m)}. $$(E.12)
Appendix F

Sensitivities for MIMO sub-problems

In Section 4.3, the output sensitivity properties for SISO sub-problems were generalized to properties for any ATC sub-problem. This Appendix presents two additional examples of MIMO sub-problems on which this generalization is based. The first example discusses a MIMIO sub-problem, and the second discusses a MIMDO sub-problem.

F.1 Sensitivities for MIMIO sub-problems

This section discusses the sensitivity characteristics for multiple input, multiple independent output sub-problems. Consider the sub-problem of Figure F.1 with one response target $R$ and one linking target $Y$ received from its parent. The input vector is $\mathbf{q} = [R, Y]$. The element computes responses $r$ and $y$ for its parent. The output vector is $\mathbf{Q} = [r, y]$. In the figure we have local variables $x_1$ and $x_2$, analysis function $a(x_1, x_2, y) = \sqrt{x_1^2 + x_2^2 + y^2}$, and weights $w_R = w_Y = 1$.

Realizable inputs

For realizable input $\mathbf{q} = [2.5, 1.5]$, the optimal solution for output variables is $\mathbf{Q}^* = [2.5, 1.5]$, and solutions for local variables $x_1^*, x_2^*$ are non-unique, but must satisfy $x_1^2 + x_2^2 = R^2 - Y^2$. Only equality constraint $h_1$ is active $\mathbf{h} = [h_1]$, but with Lagrange multiplier $\mathbf{\lambda}^* = [\lambda_1^*] = 0$. The sensitivity system for e.g. $\mathbf{z}^* = [1.0, 0.58, 2.5, 1.5]$ is:
Appendix F. Sensitivities for MIMO sub-problems

$$\begin{align*}
\text{min } \quad & (R - r)^2 + (Y - y)^2, \\
\text{subject to } \quad & h_1 = r_{11} - a_{11} (x_{11}, R_{21}, R_{22}) = 0, \\
& g_1 = x_1^2 - x_2^2 - 1 \leq 0, \\
& g_2 : x_1 \geq 0, \\
& g_3 : x_2 \geq 0, \\
\text{where } \quad & a(x_1, x_2, y) = \sqrt{x_1^2 + x_2^2 + y^2}, \\
\mathbf{z} = [x_1, x_2, r, y], \\
\end{align*}$$

Figure F.1: Example problem with multiple inputs and multiple dependent outputs

$$\begin{bmatrix}
0 & 0 & 0 & 0 & -0.8 \\
0 & 0 & 0 & 0 & 4.16 \\
0 & 0 & 2 & 0 & 1 \\
-0.8 & 4.16 & 1 & -1.2 & 0 \\
\end{bmatrix}
\begin{bmatrix}
\frac{dx_1}{dq} \\
\frac{dx_2}{dq} \\
\frac{J}{dq} \\
\frac{d\lambda_1}{dq} \\
\end{bmatrix}
= \begin{bmatrix}
0 & 0 \\
0 & 0 \\
2 & 0 \\
0 & 0 \\
\end{bmatrix}, \quad (F.1)
$$

with $\frac{dx_1}{dq}$ and $\frac{dx_2}{dq}$ the vector of sensitivities of local variables $x_1$ and $x_2$ with respect to inputs $q$, $J$ the Jacobian matrix containing sensitivities of outputs $Q$, and $\frac{d\lambda_1}{dq}$ the vector of sensitivities of the Lagrange multiplier. Only sensitivities of outputs $Q$ and the Lagrange multiplier $\lambda_1$ can be computed uniquely by Gaussian elimination. The sensitivities in $\mathbf{z}^*$ are:

$$J = \begin{bmatrix}
1 & 0 \\
0 & 1 \\
\end{bmatrix}, \quad \frac{d\lambda_1}{dq} = [0, 0],$$

Four observations are similar to SISO problems. Firstly, the Jacobian is the identity matrix for MIMIO sub-problems with a realizable input vector. The Jacobian predicts that for realizable inputs, any change in input $q$ can be followed by both outputs $Q$. Secondly, the active equality constraint has Lagrange multiplier 0, and the multiplier is insensitive to input changes, which was also observed for SISO problems. Third, the sensitivities of local variables are non-unique, which again was observed for SISO problems. Unfortunately, similar to SISO sub-problems, constraint activity changes cannot be predicted with the sensitivity information.
Non-realizable inputs

Consider the same problem, but with non-realizable input \( q = [1, 1] \). The optimal solution is \( z^* = [1.10, 0.91, 2.02, 1.29] \) and constraints \( h_1 \) and \( g_1 \) are active, with Lagrange multipliers \( \lambda^* = [\lambda_1^*, \mu_1^*] = [2.03, 1.21] \). The sensitivity system in \( z^* \) is:

\[
\begin{bmatrix}
2.31 & -0.36 & 0 & 2.07 & -0.91 & -0.54 \\
-0.36 & -2.48 & 0 & 1.64 & 1.10 & 0.65 \\
0 & 0 & 2 & 0 & 0 & 1 \\
2.07 & 1.64 & 0 & 5.78 & -1.55 & -0.64 \\
-0.91 & 1.10 & 0 & -1.55 & 0 & 0 \\
-0.54 & 0.65 & 1 & -0.64 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\frac{dx_1^*}{dq} \\
\frac{dx_2^*}{dq} \\
\frac{d\lambda_1^*}{dq} \\
\frac{d\mu_1^*}{dq}
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
2 \\
0 \\
0 \\
0
\end{bmatrix}, \quad (F.2)
\]

with \( \frac{dx_1^*}{dq} \) and \( \frac{dx_2^*}{dq} \) the vector of sensitivities of local variables \( x_1 \) and \( x_2 \) with respect to inputs \( q \). \( J \) the Jacobian matrix containing sensitivities of outputs \( Q \), and \( \frac{d\lambda_1^*}{dq} \) and \( \frac{d\mu_1^*}{dq} \) the vectors of sensitivities of the Lagrange multipliers. The unique solutions to the sensitivity system are:

\[
\frac{dx_1^*}{dq} = [0.18, -0.62], \quad \frac{dx_2^*}{dq} = [-0.13, 0.18],
\]

\[
J = \begin{bmatrix}
0.06 & -0.14 \\
-0.20 & 0.49
\end{bmatrix}, \quad \frac{d\lambda_1^*}{dq} = [-1.07, -0.69], \quad \frac{d\mu_1^*}{dq} = [1.89, 0.28],
\]

Again, parallels with single output problems can be observed. For non-realizable outputs, outputs changes cannot be matched. However, in this case, outputs \( Q \) are sensitive to changes in inputs, as are local variables \( x \) and Lagrange multipliers \( \lambda \). Because the Jacobian is not the identity matrix, a change in inputs \( q \) cannot be followed exactly, but still generates a change in the optimal solution \( z^* = [x^*, Q^*] \) of the ATC sub-problem. The optimal solution moves on the edge of the inequality constraint \( g_1 \), providing a feasible solution minimizing the objective function \( f' = ||Q - q||_2^2 \).

As a second observation, typical for MIMO sub-problems, constraint activity causes outputs \( Q \) to be linearly dependent for non-realizable inputs, indicated by the linearly dependent rows of the Jacobian. For non-realizable inputs, constraints become active (Lagrange multipliers are non-zero). Active constraints define how design variables depend on one another, and therefore link the design variables of the problem. The active constraints impose a dependency between \( r \) and \( y \), and therefore their sensitivities.
Appendix F. Sensitivities for MIMO sub-problems

An estimate of the location of the edge of the realizable domain can be computed, with sensitivities of outputs and Lagrange multipliers. This concept was explained for non-realizable input vectors for SISO sub-problems in Section 4.3. This approximation method however is not exact for the MIMO sub-problems. For SISO sub-problems, the realizable domain was simply an interval which was simply bounded. However for MIMO problems, the realizable domain of inputs is generally a non-linear $n^q$ dimensional space with a non-linear bounding surface. For this bounding surface, linear approximation are not exact.

Generalization for multiple input multiple independent output problems

For any sub-problem with $n^q$ inputs $q$ and $n^q$ independent outputs $Q$, the following can be said: with $Q$ the feasible sub-space of outputs $Q$, and thus the realizable domain of inputs $q$, the rank of the Jacobian matrix equals:

$$\text{rank}(J) = \begin{cases} n^q & \text{if } q \in Q, \\ 1 & \text{if } q \notin Q. \end{cases}$$ (F.3)

Furthermore, the Jacobian equals the identity matrix when inputs are within the realizable domain of outputs. For realizable inputs, all Lagrange multipliers are zero and insensitive to changes is inputs. For non-realizable inputs, Lagrange multipliers are non-zero for equality constraints and active inequality constraints.

F.2 Sensitivities for MIMDO sub-problems

This section discusses the sensitivity characteristics for multiple input, multiple dependent output (MIMDO) sub-problems. Consider the sub-problem of Figure (F.2) with one response target received from its parent $R_{11}$, and two target responses received from its children $r_{21}$ and $r_{22}$. The input vector is $q_{11} = [R_{11}, r_{21}, r_{22}]$. The element computes a response $r_{11}$ for the parent, and response targets for its children $R_{21}$ and $R_{22}$. The output vector is $Q_{11} = [r_{11}, R_{21}, R_{22}]$. In this problem we have local variable $x_{11}$, analysis function $a_{11}(x_{11}, R_{21}, R_{22}) = x_{11} + R_{21}$, and weights $w_{R_{11}} = 1$, $w_{R_{21}} = 1$, and $w_{R_{22}} = 1$.

Note that outputs $R_{21}$ and $R_{22}$ are dependent, and the coupling constraint is equality constraint $h_2$. It is possible to modify the problem by replacing $R_{22}$ by $R_{21}$ in the objective and constraints, making constraint $h_2$ redundant, thus removable.
F.2. Sensitivities for MIMDO sub-problems

\[
\begin{align*}
\min_{\mathbf{z}_{11}} \quad & (R_{11} - r_{11})^2 + (R_{21} - r_{21})^2 + (R_{22} - r_{22})^2, \\
\text{subject to} \quad & h_1 = r_{11} - a_{11}(x_{11}, R_{21}, R_{22}) = 0, \\
& h_2 = R_{21} - R_{22} = 0, \\
& g_1 = x_{11}^2 + R_{21}^2 - 9 \leq 0, \\
& g_2 : 0 \leq x_{11}, \\
& g_3 : 0 \leq R_{21}, \\
& g_4 : 0 \leq R_{22}, \\
\text{where} \quad & a_{11}(x_{11}, R_{21}, R_{22}) = x_{11} + R_{21}; \\
\mathbf{z}_{11} &= [x_{11}, r_{11}, R_{21}, R_{22}].
\end{align*}
\]

Figure F.2: Example problem with multiple inputs and multiple dependent outputs

Realizable inputs

For realizable input \( \mathbf{q}_{11} = [2, 1, 1] \), the optimal solution is \( \mathbf{z}^*_1 = [1, 2, 1, 1] \) and equality constraints \( h_1 \) and \( h_2 \) are active, but with Lagrange multipliers \( \mathbf{\lambda}^* = [\lambda_1^*, \lambda_2^*] = \mathbf{0} \), similar to the single input single output sub-problem. The sensitivity system becomes:

\[
\begin{bmatrix}
0 & 0 & 0 & 0 & -1 & 0 \\
0 & 2 & 0 & 0 & 1 & 0 \\
0 & 0 & 2 & 0 & -1 & 1 \\
0 & 0 & 0 & 2 & 0 & -1 \\
-1 & 1 & -1 & 0 & 0 & 0 \\
0 & 0 & 1 & -1 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\frac{d\mathbf{x}_{11}}{d\mathbf{q}_{11}} \\
\frac{d\mathbf{J}_{11}}{d\mathbf{q}_{11}} \\
\frac{d\lambda_1^*}{d\mathbf{q}_{11}} \\
\frac{d\lambda_2^*}{d\mathbf{q}_{11}}
\end{bmatrix}
= \begin{bmatrix}
0 & 0 & 0 \\
2 & 0 & 0 \\
0 & 2 & 0 \\
0 & 0 & 2 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{bmatrix}, \quad \text{(F.4)}
\]

with \( \frac{d\mathbf{x}_{11}}{d\mathbf{q}_{11}} \) the vector of sensitivities of local variable \( x_{11} \) with respect to inputs \( \mathbf{q}_{11} \), \( \mathbf{J} \) the Jacobian matrix containing sensitivities of outputs \( \mathbf{Q}_{11} \), and \( \frac{d\lambda_1^*}{d\mathbf{q}_{11}} \) and \( \frac{d\lambda_2^*}{d\mathbf{q}_{11}} \) vectors of sensitivities of Lagrange multipliers. The unique solutions to the sensitivity system are:
Appendix F. Sensitivities for MIMO sub-problems

\[
\frac{d\bar{x}^r}{dq_{11}} = [1, -0.5, -0.5], \quad J_{11} = \begin{bmatrix}
1 & 0 & 0 \\
0 & 0.5 & 0.5 \\
0 & 0.5 & 0.5
\end{bmatrix},
\]

\[
\frac{d\lambda^r}{dq_{11}} = \begin{bmatrix}
0 & 0 & 0 \\
0 & 1 & -1
\end{bmatrix}.
\]

The rank of the Jacobian matrix is 2, which is equal to the number of independent outputs of the sub-problem. Output \( r_{11} \) is independent of outputs \( R_{21} \) and \( R_{22} \). However, outputs \( R_{21} \) and \( R_{22} \) are linked through the equality constraint \( h_2 = R_{21} = R_{22} = 0 \) forcing them to be equal; outputs \( R_{21} \) and \( R_{22} \) are linearly dependent, and therefore their sensitivities, decreasing the rank of the Jacobian by 1. For any ATC problem \( P_{ij} \) that computes linking targets for its children \( C_{ij} \), all linking variable copies are forced to match and are therefore dependent and decrease the rank of the Jacobian by the number of additional copies, which is equal to the number of children minus 1 \( (c_{ij} - 1) \). For realizable inputs in general, the rank of the Jacobian is equal to the number of independent outputs, which is also true for single and multiple independent realizable outputs.

For the realizable input vector, several similarities exist with the single input single output sub-problems. First, the sensitivity of response \( r_{11} \) (row 1 of \( J_{11} \)) with respect to target \( R_{11} \) is 1 (element (1,1) of \( J_{11} \)), i.e. the response will locally match any change in target. The Lagrange multiplier associated with the analysis model constraint is insensitive to changes in inputs (row 1 of \( \frac{d\lambda^r}{dq_{11}} \) is \([0, 0, 0]\)). Furthermore, constraint activity changes are not predicted with sensitivity information.

Other observations can be explained. The second and third row of the Jacobian are equal. Note that \( R_{21} \) and \( R_{22} \) are not sensitive to changes in \( R_{11} \). Identical sensitivities suggest that the second and third output \( (R_{21}^r \text{ and } R_{22}^r) \) have identical behavior with respect to changes in inputs \( (\Delta r_{21} \text{ and } \Delta r_{22}) \), which is in accordance with the optimization problem which forces \( R_{21}^r \) and \( R_{22}^r \) to be equal (constraint \( h_2 \)). The equality of output targets is therefore accurately predicted by the Jacobian matrix.

Moreover, the sensitivities of \( R_{21} \) and \( R_{22} \) with respect to inputs \( r_{21} \) and \( r_{22} \) respectively are smaller than 1. If for instance only \( r_{21} \) is increased by \( \Delta r_{21} \) while \( r_{22} \) is fixed, then the outputs \( R_{21} \) will not be able to follow \( \Delta r_{21} \) exactly; progress of \( R_{21} \) is slowed down by the unchanged \( r_{22} \). The formulation of the ATC problem provides information why progress is slowed down. The penalty function ‘attracts’ the outputs \( R_{21} \) and \( R_{22} \) when they move away from inputs \( r_{21} \) and \( r_{22} \). The weights determine the ‘force of attraction’. Since \( R_{21} \) and \( R_{22} \) are equal, the resulting output location of \( R_{21} = R_{22} \) will be a weighted average of inputs \( r_{21} \) and \( r_{22} \). Hence, by changing only one input while fixing the other, equal outputs \( R_{21} = R_{22} \) will only move a part of the change of
input, the part depending on weights assigned to each input. Since the change can only be followed partially, sensitivities are smaller than 1.

A third observation regarding outputs $R_{21}^*$ and $R_{22}^*$ can be explained. Consider the linear extrapolations of output changes $\Delta R_{21}$ and $\Delta R_{22}$ with respect to input changes $\Delta r_{21}$ and $\Delta r_{22}$:

\[
\begin{align*}
\Delta R_{21} &= 0.5\Delta r_{21} + 0.5\Delta r_{22} \\
\Delta R_{22} &= 0.5\Delta r_{21} + 0.5\Delta r_{22}
\end{align*}
\] (F.5)

Since input responses are realizable $r_{21} = r_{22} = R_{21}^* = R_{22}^*$, new designs will be realizable when $\Delta r_{21} = \Delta R_{21}$ and $\Delta r_{22} = \Delta R_{22}$. From the extrapolation equations (F.5), these conditions are only satisfied when $\Delta r_{21} = \Delta r_{22}$. Therefore, the Jacobian accurately predicts that any change in responses $r_{21}$ and $r_{22}$, can only be realizable if the changes are identical, and thus $r_{21}' = r_{22}'$.

A fourth observation predicts identical behavior. Consider the Lagrange multiplier sensitivities. The two opposite non-zero terms in row 2 in $\frac{d\lambda}{dq_{11}}$ are associated with Lagrange multiplier changes of constraint $R_{21} = R_{22}$. The optimal value of the Lagrange multiplier was $\lambda_1^* = 0$, and $r_{21} = r_{22} = R_{21}^* = R_{22}^*$. For a change $\Delta r_{21}$ and $\Delta r_{22}$ to remain feasible, the change in Lagrange multiplier $\lambda_2$ must be zero:

\[
\Delta \lambda_2 = \Delta r_{21} - \Delta r_{22} = 0
\] (F.6)

The change will only be zero if $\Delta r_{21} = \Delta r_{22}$. Hence, the sensitivities of Lagrange multipliers also suggest that for realizable and equal initial inputs for dependent outputs, $\Delta r_{21} = \Delta r_{22}$ is demanded for any change in inputs to be realizable.

For multiple realizable inputs, the dependency of variables is described accurately by the Jacobian and Lagrange multiplier sensitivities. For any new set of inputs to be realizable, the Jacobian suggests that coupled outputs should be equal. Locations of inactive constraints are again not predicted by the Jacobian, and new inputs may take on any value as long as the coupling constraints of the problem are satisfied.

**Non-realizable inputs**

Consider the same problem, but with non-realizable input $q_{11} = [0, 0, 1]$. The optimal solution to this problem is $z_{11}^* = [0, 0.33, 0.33, 0.33]$, $Q_{11}^* = [0.33, 0.33, 0.33]$, constraints
$h = [h_1, h_2, g_1]$ are active with associated Lagrange multipliers $\lambda^* = [\lambda_2^*, \lambda_3^*, \mu_1^*] = [0.67, 1.33, 0.67]$. The sensitivity system in $z_{11}$ is:

$$
\begin{bmatrix}
0 & 0 & 0 & 0 & -1 & 0 & -1 \\
0 & 2 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 2 & 0 & -1 & 1 & 0 \\
0 & 0 & 0 & 2 & 0 & -1 & 0 \\
-1 & 1 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & -1 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\frac{dx_{11}}{dq_{11}} \\
\frac{d\lambda_2^*}{dq_{11}} \\
\frac{d\lambda_3^*}{dq_{11}} \\
\frac{d\mu_1^*}{dq_{11}}
\end{bmatrix}
= 
\begin{bmatrix}
0 & 0 & 0 \\
0 & 2 & 0 \\
0 & 0 & 2 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{bmatrix}, \quad (F.7)
$$

with unique solutions:

$$
\frac{dx_{11}}{dq_{11}} = [0, 0, 0], \quad J_{11} = \begin{bmatrix}
0.33 & 0.33 & 0.33 \\
0.33 & 0.33 & 0.33 \\
0.33 & 0.33 & 0.33
\end{bmatrix},
$$

$$
\frac{d\lambda^*}{dq_{11}} = 
\begin{bmatrix}
1.33 & -0.67 & -0.67 \\
0.67 & 0.67 & -1.33 \\
-1.33 & 0.67 & 0.67
\end{bmatrix}.
$$

Sensitivities of local variable $x_{11}$ are all 0, indicating that any change of input will not locally change the optimal solution of $x_{11}$.

The rank of the Jacobian equals 1, in this case because all rows of the Jacobian are equal. The rows of the Jacobian are equal, because all output variables are equal and local variable $x_{11}$ is zero. With $x_{11} = 0$, constraint $g_2$ reduces to $r_{11} - R_{21} = 0$, forcing $r_{11} = R_{21}$, while constraint $g_4$ forces $R_{21} = R_{22}$. Since no change in $x_{11} = 0$ is predicted (sensitivities equal to 0), all output variables will remain equal and have identical sensitivities in $z_{11}$. For all non-realizable input vectors, the rank of the Jacobian equals 1. It is possible that $x_{11}$ is sensitive with respect to inputs (e.g. with input vector $q_{11} = [6, \frac{3}{2}\sqrt{2}, \frac{3}{2}\sqrt{2}]$, and $z_{11} = \frac{3}{2}\sqrt{2}[1, 2, 1, 1]$), but in that case one or more outputs are insensitive to input changes, resulting in the same rank of the Jacobian. For a fully insensitive solution (e.g. with $q_{11} = [-1, -1, -1]$, and $z_{11}^* = [0, 0, 0, 0]$), the Jacobian is of rank 0. For non-realizable inputs in general, the rank of the Jacobian is smaller than the number of independent outputs, which is also true for multiple independent outputs with non-realizable inputs.

Similar to problems with independent outputs, locations of inequality constraints can
be predicted with Lagrange multipliers and their sensitivities, as well as output sensitivities. For dependent outputs, the condition that input extrapolations $r_{21}'$ and $r_{22}'$ are equal must be satisfied for any input update to be realizable.

**Generalization for multiple input multiple dependent output problems**

For any sub-problem with $n^q$ multiple inputs $q$ and $n^q$ multiple dependent outputs $Q$, and $n^Q$ independent outputs, the following can be said: with $Q$ the feasible domain of outputs $Q$, the rank of the Jacobian matrix equals:

$$\text{rank}(J) \begin{cases} = n^Q & \text{if } q \in Q, \\ < n^Q & \text{if } q \notin Q \end{cases}$$ (F.8)

Furthermore, for realizable inputs all Lagrange multipliers are zero and insensitive to changes in inputs, except for constraints linking the dependent variables. The Lagrange multipliers of these linking constraints are sensitive to changes in inputs, and the multipliers can be extrapolated to zero, for which the constraint itself remains satisfied. In other words, the Lagrange multipliers of linking constraints are zero for realizable inputs, and can only remain zero for input changes satisfying the linking constraints.

For non-realizable inputs, Lagrange multipliers of equality constraints and active inequality constraints are non-zero. The Lagrange multipliers of inequality constraints and their sensitivities with respect to input changes can be used to approximate the realizable domain of inputs. The approximations for a single active inequality constraint become more accurate when inputs are closer to the realizable domain. However, no constraint approximation method is proposed in this research.
Appendix F. Sensitivities for MIMO sub-problems
Appendix G

Results

All numerical results are presented for all of the experiments conducted for Chapter 5. For each setting of weights $w$, and termination tolerance $\tau$, the final error, $e_K$, and the number of required function evaluations, $N_f$, is presented.

G.1 Geometric optimization problem 1

The results presented here are computed for geometric programming problem 1 of Section 5.2. The decomposed problem formulation is presented in Section A.10. Results are presented for all five evaluated coordination strategies: hierarchical scheme III, all-parallel, sensitivity-based, sensitivity-based with move limits, and the hybrid strategy.

<table>
<thead>
<tr>
<th>Scheme III</th>
<th>All-parallel</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w$</td>
<td>$w$</td>
</tr>
<tr>
<td>$1.00e0$</td>
<td>$1.00e0$</td>
</tr>
<tr>
<td>$3.2556$</td>
<td>$2.3956$</td>
</tr>
<tr>
<td>$3.2558$</td>
<td>$2.3958$</td>
</tr>
<tr>
<td>$1.2351$</td>
<td>$1.2356$</td>
</tr>
<tr>
<td>$1.2397$</td>
<td>$1.2393$</td>
</tr>
<tr>
<td>$607$</td>
<td>$1678$</td>
</tr>
<tr>
<td>$507$</td>
<td>$1398$</td>
</tr>
<tr>
<td>$357$</td>
<td>$1062$</td>
</tr>
<tr>
<td>$207$</td>
<td>$838$</td>
</tr>
<tr>
<td>$1.64e0$</td>
<td>$1.64e0$</td>
</tr>
<tr>
<td>$1.3744$</td>
<td>$1.3743$</td>
</tr>
<tr>
<td>$1.3745$</td>
<td>$1.3742$</td>
</tr>
<tr>
<td>$1.3773$</td>
<td>$1.3729$</td>
</tr>
<tr>
<td>$1.3712$</td>
<td>$1.3812$</td>
</tr>
<tr>
<td>$650$</td>
<td>$2652$</td>
</tr>
<tr>
<td>$434$</td>
<td>$2172$</td>
</tr>
<tr>
<td>$272$</td>
<td>$1572$</td>
</tr>
<tr>
<td>$218$</td>
<td>$1092$</td>
</tr>
<tr>
<td>$2.68e0$</td>
<td>$2.68e0$</td>
</tr>
<tr>
<td>$0.4244$</td>
<td>$0.4244$</td>
</tr>
<tr>
<td>$0.4247$</td>
<td>$0.4242$</td>
</tr>
<tr>
<td>$0.4481$</td>
<td>$0.4221$</td>
</tr>
<tr>
<td>$0.4035$</td>
<td>$0.4035$</td>
</tr>
<tr>
<td>$187$</td>
<td>$5124$</td>
</tr>
<tr>
<td>$1339$</td>
<td>$3985$</td>
</tr>
<tr>
<td>$790$</td>
<td>$2914$</td>
</tr>
<tr>
<td>$363$</td>
<td>$1876$</td>
</tr>
<tr>
<td>$4.39e0$</td>
<td>$4.39e0$</td>
</tr>
<tr>
<td>$0.0948$</td>
<td>$0.0947$</td>
</tr>
<tr>
<td>$0.0951$</td>
<td>$0.0944$</td>
</tr>
<tr>
<td>$0.0982$</td>
<td>$0.0916$</td>
</tr>
<tr>
<td>$0.1296$</td>
<td>$0.0749$</td>
</tr>
<tr>
<td>$4124$</td>
<td>$10871$</td>
</tr>
<tr>
<td>$2896$</td>
<td>$8243$</td>
</tr>
<tr>
<td>$1648$</td>
<td>$5538$</td>
</tr>
<tr>
<td>$550$</td>
<td>$2797$</td>
</tr>
<tr>
<td>$7.20e0$</td>
<td>$7.20e0$</td>
</tr>
<tr>
<td>$0.0185$</td>
<td>$0.0185$</td>
</tr>
<tr>
<td>$0.0195$</td>
<td>$0.0195$</td>
</tr>
<tr>
<td>$0.0214$</td>
<td>$0.0145$</td>
</tr>
<tr>
<td>$0.0537$</td>
<td>$0.0972$</td>
</tr>
<tr>
<td>$8430$</td>
<td>$22272$</td>
</tr>
<tr>
<td>$5614$</td>
<td>$16112$</td>
</tr>
<tr>
<td>$2712$</td>
<td>$10046$</td>
</tr>
<tr>
<td>$598$</td>
<td>$3679$</td>
</tr>
<tr>
<td>$1.18e1$</td>
<td>$1.18e1$</td>
</tr>
<tr>
<td>$0.0020$</td>
<td>$0.0025$</td>
</tr>
<tr>
<td>$0.0030$</td>
<td>$0.0025$</td>
</tr>
<tr>
<td>$0.0050$</td>
<td>$0.0020$</td>
</tr>
<tr>
<td>$0.0794$</td>
<td>$0.7541$</td>
</tr>
<tr>
<td>$20920$</td>
<td>$54446$</td>
</tr>
<tr>
<td>$12452$</td>
<td>$36302$</td>
</tr>
<tr>
<td>$4207$</td>
<td>$14617$</td>
</tr>
<tr>
<td>$667$</td>
<td>$1598$</td>
</tr>
<tr>
<td>$1.93e1$</td>
<td>$1.93e1$</td>
</tr>
<tr>
<td>$0.0004$</td>
<td>$0.0003$</td>
</tr>
<tr>
<td>$0.0014$</td>
<td>$0.0008$</td>
</tr>
<tr>
<td>$0.0058$</td>
<td>$0.0635$</td>
</tr>
<tr>
<td>$0.0880$</td>
<td>$0.8092$</td>
</tr>
<tr>
<td>$47285$</td>
<td>$129593$</td>
</tr>
<tr>
<td>$24089$</td>
<td>$80981$</td>
</tr>
<tr>
<td>$2883$</td>
<td>$31417$</td>
</tr>
<tr>
<td>$721$</td>
<td>$726$</td>
</tr>
<tr>
<td>$3.16e1$</td>
<td>$3.16e1$</td>
</tr>
<tr>
<td>$0.0001$</td>
<td>$0.0001$</td>
</tr>
<tr>
<td>$0.0005$</td>
<td>$0.0002$</td>
</tr>
<tr>
<td>$0.0054$</td>
<td>$0.0202$</td>
</tr>
<tr>
<td>$0.0524$</td>
<td>$0.8218$</td>
</tr>
<tr>
<td>$10184$</td>
<td>$368343$</td>
</tr>
<tr>
<td>$39297$</td>
<td>$176201$</td>
</tr>
<tr>
<td>$1197$</td>
<td>$25052$</td>
</tr>
<tr>
<td>$769$</td>
<td>$736$</td>
</tr>
<tr>
<td>$5.18e1$</td>
<td>$5.18e1$</td>
</tr>
<tr>
<td>$0.0000$</td>
<td>$0.0000$</td>
</tr>
<tr>
<td>$0.0000$</td>
<td>$0.0205$</td>
</tr>
<tr>
<td>$0.0205$</td>
<td>$0.8261$</td>
</tr>
<tr>
<td>$266319$</td>
<td>$713139$</td>
</tr>
<tr>
<td>$102269$</td>
<td>$367429$</td>
</tr>
<tr>
<td>$1922$</td>
<td>$1041$</td>
</tr>
<tr>
<td>$1096$</td>
<td>$749$</td>
</tr>
<tr>
<td>$8.48e1$</td>
<td>$8.48e1$</td>
</tr>
<tr>
<td>$0.0029$</td>
<td>$0.0029$</td>
</tr>
<tr>
<td>$0.0092$</td>
<td>$0.2382$</td>
</tr>
<tr>
<td>$0.0555$</td>
<td>$0.8280$</td>
</tr>
<tr>
<td>$0.0509$</td>
<td>$0.8287$</td>
</tr>
<tr>
<td>$1048263$</td>
<td>$1454845$</td>
</tr>
<tr>
<td>$7037$</td>
<td>$419403$</td>
</tr>
<tr>
<td>$2132$</td>
<td>$1133$</td>
</tr>
<tr>
<td>$1016$</td>
<td>$818$</td>
</tr>
<tr>
<td>$1.36e2$</td>
<td>$1.36e2$</td>
</tr>
<tr>
<td>$0.0172$</td>
<td>$0.0195$</td>
</tr>
<tr>
<td>$0.0566$</td>
<td>$0.7442$</td>
</tr>
<tr>
<td>$0.0960$</td>
<td>$0.8296$</td>
</tr>
<tr>
<td>$0.0952$</td>
<td>$0.8300$</td>
</tr>
<tr>
<td>$171947$</td>
<td>$2940755$</td>
</tr>
<tr>
<td>$3427$</td>
<td>$284886$</td>
</tr>
<tr>
<td>$1751$</td>
<td>$1362$</td>
</tr>
<tr>
<td>$1189$</td>
<td>$959$</td>
</tr>
<tr>
<td>$2.28e2$</td>
<td>$2.28e2$</td>
</tr>
<tr>
<td>$0.0962$</td>
<td>$0.1276$</td>
</tr>
<tr>
<td>$0.0961$</td>
<td>$0.8317$</td>
</tr>
<tr>
<td>$0.0941$</td>
<td>$0.8317$</td>
</tr>
<tr>
<td>$0.0553$</td>
<td>$0.8317$</td>
</tr>
<tr>
<td>$5305$</td>
<td>$838135$</td>
</tr>
<tr>
<td>$4863$</td>
<td>$2261$</td>
</tr>
<tr>
<td>$2171$</td>
<td>$1746$</td>
</tr>
<tr>
<td>$1415$</td>
<td>$1167$</td>
</tr>
<tr>
<td>$3.73e2$</td>
<td>$3.73e2$</td>
</tr>
<tr>
<td>$0.0963$</td>
<td>$0.7943$</td>
</tr>
<tr>
<td>$0.0964$</td>
<td>$0.8289$</td>
</tr>
<tr>
<td>$0.0962$</td>
<td>$0.8289$</td>
</tr>
<tr>
<td>$0.0553$</td>
<td>$0.8306$</td>
</tr>
<tr>
<td>$7206$</td>
<td>$2742780$</td>
</tr>
<tr>
<td>$6353$</td>
<td>$2325$</td>
</tr>
<tr>
<td>$2495$</td>
<td>$1801$</td>
</tr>
<tr>
<td>$1466$</td>
<td>$1340$</td>
</tr>
<tr>
<td>$6.11e1$</td>
<td>$6.11e1$</td>
</tr>
<tr>
<td>$0.0964$</td>
<td>$0.9285$</td>
</tr>
<tr>
<td>$0.0964$</td>
<td>$0.8285$</td>
</tr>
<tr>
<td>$0.0962$</td>
<td>$0.8285$</td>
</tr>
<tr>
<td>$0.0553$</td>
<td>$0.8285$</td>
</tr>
<tr>
<td>$4112$</td>
<td>$3081$</td>
</tr>
<tr>
<td>$3196$</td>
<td>$2545$</td>
</tr>
<tr>
<td>$2117$</td>
<td>$2037$</td>
</tr>
<tr>
<td>$1342$</td>
<td>$1456$</td>
</tr>
<tr>
<td>$1.06e3$</td>
<td>$1.06e3$</td>
</tr>
<tr>
<td>$0.0964$</td>
<td>$0.8337$</td>
</tr>
<tr>
<td>$0.0964$</td>
<td>$0.8336$</td>
</tr>
<tr>
<td>$0.0963$</td>
<td>$0.8323$</td>
</tr>
<tr>
<td>$0.0553$</td>
<td>$0.8287$</td>
</tr>
<tr>
<td>$5157$</td>
<td>$3391$</td>
</tr>
<tr>
<td>$3979$</td>
<td>$2763$</td>
</tr>
<tr>
<td>$2420$</td>
<td>$2151$</td>
</tr>
<tr>
<td>$1480$</td>
<td>$1522$</td>
</tr>
</tbody>
</table>
### Appendix G. Results

#### Sensitivity-based termination tolerance

<table>
<thead>
<tr>
<th>$w$</th>
<th>$1e^{-4}$</th>
<th>$1e^{-6}$</th>
<th>$1e^{-8}$</th>
<th>$1e^{-10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00e0</td>
<td>3.2556</td>
<td>3.2556</td>
<td>3.2556</td>
<td>3.2556</td>
</tr>
<tr>
<td>1.64e0</td>
<td>1.3743</td>
<td>1.3743</td>
<td>1.3743</td>
<td>1.3743</td>
</tr>
<tr>
<td>2.68e0</td>
<td>0.4245</td>
<td>0.4245</td>
<td>0.4245</td>
<td>0.4245</td>
</tr>
<tr>
<td>4.35e0</td>
<td>0.0947</td>
<td>0.0947</td>
<td>0.0947</td>
<td>0.0947</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.0195</td>
<td>0.0195</td>
<td>0.0195</td>
<td>0.0195</td>
</tr>
<tr>
<td>1.18e1</td>
<td>0.0025</td>
<td>0.0025</td>
<td>0.0025</td>
<td>0.0025</td>
</tr>
<tr>
<td>1.93e1</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
</tr>
<tr>
<td>3.16e1</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
<tr>
<td>8.48e1</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
</tr>
<tr>
<td>1.39e2</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
</tr>
<tr>
<td>2.68e2</td>
<td>0.0005</td>
<td>0.0005</td>
<td>0.0005</td>
<td>0.0005</td>
</tr>
<tr>
<td>6.11e2</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
</tr>
<tr>
<td>1.00e3</td>
<td>0.0176</td>
<td>0.0176</td>
<td>0.0176</td>
<td>0.0176</td>
</tr>
</tbody>
</table>

#### Sensitivity-based with move limits termination tolerance

<table>
<thead>
<tr>
<th>$w$</th>
<th>$1e^{-4}$</th>
<th>$1e^{-6}$</th>
<th>$1e^{-8}$</th>
<th>$1e^{-10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00e0</td>
<td>3.2556</td>
<td>3.2556</td>
<td>3.2556</td>
<td>3.2556</td>
</tr>
<tr>
<td>1.64e0</td>
<td>1.3743</td>
<td>1.3743</td>
<td>1.3743</td>
<td>1.3743</td>
</tr>
<tr>
<td>2.68e0</td>
<td>0.4245</td>
<td>0.4245</td>
<td>0.4245</td>
<td>0.4245</td>
</tr>
<tr>
<td>4.35e0</td>
<td>0.0947</td>
<td>0.0947</td>
<td>0.0947</td>
<td>0.0947</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.0195</td>
<td>0.0195</td>
<td>0.0195</td>
<td>0.0195</td>
</tr>
<tr>
<td>1.18e1</td>
<td>0.0025</td>
<td>0.0025</td>
<td>0.0025</td>
<td>0.0025</td>
</tr>
<tr>
<td>1.93e1</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
</tr>
<tr>
<td>3.16e1</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
<tr>
<td>8.48e1</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
</tr>
<tr>
<td>1.39e2</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
</tr>
<tr>
<td>2.68e2</td>
<td>0.0005</td>
<td>0.0005</td>
<td>0.0005</td>
<td>0.0005</td>
</tr>
<tr>
<td>6.11e2</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
</tr>
<tr>
<td>1.00e3</td>
<td>0.0176</td>
<td>0.0176</td>
<td>0.0176</td>
<td>0.0176</td>
</tr>
</tbody>
</table>

#### Hybrid termination tolerance

<table>
<thead>
<tr>
<th>$w$</th>
<th>$1e^{-4}$</th>
<th>$1e^{-6}$</th>
<th>$1e^{-8}$</th>
<th>$1e^{-10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00e0</td>
<td>5.0768</td>
<td>5.0768</td>
<td>5.0768</td>
<td>5.0768</td>
</tr>
<tr>
<td>1.64e0</td>
<td>1.3743</td>
<td>1.3743</td>
<td>1.3743</td>
<td>1.3743</td>
</tr>
<tr>
<td>2.68e0</td>
<td>0.4245</td>
<td>0.4245</td>
<td>0.4245</td>
<td>0.4245</td>
</tr>
<tr>
<td>4.35e0</td>
<td>0.0947</td>
<td>0.0947</td>
<td>0.0947</td>
<td>0.0947</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.0195</td>
<td>0.0195</td>
<td>0.0195</td>
<td>0.0195</td>
</tr>
<tr>
<td>1.18e1</td>
<td>0.0025</td>
<td>0.0025</td>
<td>0.0025</td>
<td>0.0025</td>
</tr>
<tr>
<td>1.93e1</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
</tr>
<tr>
<td>3.16e1</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
<tr>
<td>8.48e1</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
</tr>
<tr>
<td>1.39e2</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
</tr>
<tr>
<td>2.68e2</td>
<td>0.0005</td>
<td>0.0005</td>
<td>0.0005</td>
<td>0.0005</td>
</tr>
<tr>
<td>6.11e2</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
<td>0.0003</td>
</tr>
<tr>
<td>1.00e3</td>
<td>0.0176</td>
<td>0.0176</td>
<td>0.0176</td>
<td>0.0176</td>
</tr>
</tbody>
</table>
G.2. Geometric optimization problem 2, decomposition 1

The results presented here are computed for geometric programming problem 1 of Section 5.3. The decomposed problem formulation is presented in Section A.14. Results are presented for all five evaluated coordination strategies: hierarchical scheme III, all-parallel, sensitivity-based, sensitivity-based with move limits, and the hybrid strategy.

### Scheme III

<table>
<thead>
<tr>
<th>w</th>
<th>1e-4</th>
<th>1e-6</th>
<th>1e-8</th>
<th>1e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.06e0</td>
<td>8.4590</td>
<td>8.4583</td>
<td>8.4586</td>
<td>8.4587</td>
</tr>
<tr>
<td></td>
<td>2461</td>
<td>1849</td>
<td>1441</td>
<td>1033</td>
</tr>
<tr>
<td>1.64e0</td>
<td>3.5662</td>
<td>3.5663</td>
<td>3.5684</td>
<td>3.5998</td>
</tr>
<tr>
<td></td>
<td>3090</td>
<td>2282</td>
<td>1676</td>
<td>1272</td>
</tr>
<tr>
<td>2.68e0</td>
<td>0.9901</td>
<td>0.9903</td>
<td>0.9925</td>
<td>1.0007</td>
</tr>
<tr>
<td></td>
<td>5291</td>
<td>4007</td>
<td>2723</td>
<td>1867</td>
</tr>
<tr>
<td>4.39e0</td>
<td>0.1958</td>
<td>0.1966</td>
<td>0.1978</td>
<td>0.2144</td>
</tr>
<tr>
<td></td>
<td>9330</td>
<td>6643</td>
<td>4226</td>
<td>2232</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.0318</td>
<td>0.0330</td>
<td>0.0333</td>
<td>0.0346</td>
</tr>
<tr>
<td></td>
<td>1783</td>
<td>11927</td>
<td>6434</td>
<td>2301</td>
</tr>
</tbody>
</table>

### All-parallel

<table>
<thead>
<tr>
<th>w</th>
<th>1e-4</th>
<th>1e-6</th>
<th>1e-8</th>
<th>1e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.06e0</td>
<td>8.4590</td>
<td>8.4583</td>
<td>8.4586</td>
<td>8.4587</td>
</tr>
<tr>
<td></td>
<td>2183</td>
<td>1807</td>
<td>1243</td>
<td>867</td>
</tr>
<tr>
<td>1.64e0</td>
<td>3.5662</td>
<td>3.5663</td>
<td>3.5684</td>
<td>3.5998</td>
</tr>
<tr>
<td></td>
<td>2491</td>
<td>1927</td>
<td>1511</td>
<td>1081</td>
</tr>
<tr>
<td>2.68e0</td>
<td>0.9901</td>
<td>0.9903</td>
<td>0.9925</td>
<td>1.0007</td>
</tr>
<tr>
<td></td>
<td>4824</td>
<td>3648</td>
<td>2472</td>
<td>1492</td>
</tr>
<tr>
<td>4.39e0</td>
<td>0.1958</td>
<td>0.1966</td>
<td>0.1978</td>
<td>0.2144</td>
</tr>
<tr>
<td></td>
<td>8378</td>
<td>6097</td>
<td>3912</td>
<td>1914</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.0318</td>
<td>0.0330</td>
<td>0.0333</td>
<td>0.0346</td>
</tr>
<tr>
<td></td>
<td>15813</td>
<td>10831</td>
<td>5840</td>
<td>1983</td>
</tr>
</tbody>
</table>

### Termination tolerance τ

<table>
<thead>
<tr>
<th>w</th>
<th>1e-4</th>
<th>1e-6</th>
<th>1e-8</th>
<th>1e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.06e0</td>
<td>8.4590</td>
<td>8.4583</td>
<td>8.4586</td>
<td>8.4587</td>
</tr>
<tr>
<td></td>
<td>2183</td>
<td>1807</td>
<td>1243</td>
<td>867</td>
</tr>
<tr>
<td>1.64e0</td>
<td>3.5662</td>
<td>3.5663</td>
<td>3.5684</td>
<td>3.5998</td>
</tr>
<tr>
<td></td>
<td>2491</td>
<td>1927</td>
<td>1511</td>
<td>1081</td>
</tr>
<tr>
<td>2.68e0</td>
<td>0.9901</td>
<td>0.9903</td>
<td>0.9925</td>
<td>1.0007</td>
</tr>
<tr>
<td></td>
<td>4824</td>
<td>3648</td>
<td>2472</td>
<td>1492</td>
</tr>
<tr>
<td>4.39e0</td>
<td>0.1958</td>
<td>0.1966</td>
<td>0.1978</td>
<td>0.2144</td>
</tr>
<tr>
<td></td>
<td>8378</td>
<td>6097</td>
<td>3912</td>
<td>1914</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.0318</td>
<td>0.0330</td>
<td>0.0333</td>
<td>0.0346</td>
</tr>
<tr>
<td></td>
<td>15813</td>
<td>10831</td>
<td>5840</td>
<td>1983</td>
</tr>
</tbody>
</table>

### Scheme III

<table>
<thead>
<tr>
<th>w</th>
<th>1e-4</th>
<th>1e-6</th>
<th>1e-8</th>
<th>1e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.06e0</td>
<td>8.4590</td>
<td>8.4583</td>
<td>8.4586</td>
<td>8.4587</td>
</tr>
<tr>
<td></td>
<td>2183</td>
<td>1807</td>
<td>1243</td>
<td>867</td>
</tr>
<tr>
<td>1.64e0</td>
<td>3.5662</td>
<td>3.5663</td>
<td>3.5684</td>
<td>3.5998</td>
</tr>
<tr>
<td></td>
<td>2491</td>
<td>1927</td>
<td>1511</td>
<td>1081</td>
</tr>
<tr>
<td>2.68e0</td>
<td>0.9901</td>
<td>0.9903</td>
<td>0.9925</td>
<td>1.0007</td>
</tr>
<tr>
<td></td>
<td>4824</td>
<td>3648</td>
<td>2472</td>
<td>1492</td>
</tr>
<tr>
<td>4.39e0</td>
<td>0.1958</td>
<td>0.1966</td>
<td>0.1978</td>
<td>0.2144</td>
</tr>
<tr>
<td></td>
<td>8378</td>
<td>6097</td>
<td>3912</td>
<td>1914</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.0318</td>
<td>0.0330</td>
<td>0.0333</td>
<td>0.0346</td>
</tr>
<tr>
<td></td>
<td>15813</td>
<td>10831</td>
<td>5840</td>
<td>1983</td>
</tr>
</tbody>
</table>

### All-parallel

<table>
<thead>
<tr>
<th>w</th>
<th>1e-4</th>
<th>1e-6</th>
<th>1e-8</th>
<th>1e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.06e0</td>
<td>8.4590</td>
<td>8.4583</td>
<td>8.4586</td>
<td>8.4587</td>
</tr>
<tr>
<td></td>
<td>2183</td>
<td>1807</td>
<td>1243</td>
<td>867</td>
</tr>
<tr>
<td>1.64e0</td>
<td>3.5662</td>
<td>3.5663</td>
<td>3.5684</td>
<td>3.5998</td>
</tr>
<tr>
<td></td>
<td>2491</td>
<td>1927</td>
<td>1511</td>
<td>1081</td>
</tr>
<tr>
<td>2.68e0</td>
<td>0.9901</td>
<td>0.9903</td>
<td>0.9925</td>
<td>1.0007</td>
</tr>
<tr>
<td></td>
<td>4824</td>
<td>3648</td>
<td>2472</td>
<td>1492</td>
</tr>
<tr>
<td>4.39e0</td>
<td>0.1958</td>
<td>0.1966</td>
<td>0.1978</td>
<td>0.2144</td>
</tr>
<tr>
<td></td>
<td>8378</td>
<td>6097</td>
<td>3912</td>
<td>1914</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.0318</td>
<td>0.0330</td>
<td>0.0333</td>
<td>0.0346</td>
</tr>
<tr>
<td></td>
<td>15813</td>
<td>10831</td>
<td>5840</td>
<td>1983</td>
</tr>
</tbody>
</table>
### Appendix G. Results

#### Sensitivity-based termination tolerance

<table>
<thead>
<tr>
<th>w</th>
<th>1e-4</th>
<th>1e-6</th>
<th>1e-8</th>
<th>1e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00e0</td>
<td>8.4890</td>
<td>8.4890</td>
<td>8.4890</td>
<td>8.4890</td>
</tr>
<tr>
<td>1.64e0</td>
<td>3.5882</td>
<td>3.5880</td>
<td>3.5878</td>
<td>3.5862</td>
</tr>
<tr>
<td>2.68e0</td>
<td>0.9901</td>
<td>0.9901</td>
<td>0.9901</td>
<td>0.9901</td>
</tr>
<tr>
<td>4.39e0</td>
<td>0.1958</td>
<td>0.1958</td>
<td>0.1958</td>
<td>0.1958</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.0318</td>
<td>0.0318</td>
<td>0.0318</td>
<td>0.0318</td>
</tr>
<tr>
<td>1.18e1</td>
<td>0.0070</td>
<td>0.0070</td>
<td>0.0070</td>
<td>0.0070</td>
</tr>
<tr>
<td>1.93e1</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>3.16e1</td>
<td>0.0004</td>
<td>0.0004</td>
<td>0.0004</td>
<td>0.0004</td>
</tr>
<tr>
<td>6.11e1</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>1.00e3</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

Elements indicated with a (*) did not terminate.

#### Sensitivity-based with move limits termination tolerance

<table>
<thead>
<tr>
<th>w</th>
<th>1e-4</th>
<th>1e-6</th>
<th>1e-8</th>
<th>1e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00e0</td>
<td>8.4890</td>
<td>8.4890</td>
<td>8.4890</td>
<td>8.4890</td>
</tr>
<tr>
<td>1.64e0</td>
<td>3.5882</td>
<td>3.5880</td>
<td>3.5878</td>
<td>3.5862</td>
</tr>
<tr>
<td>2.68e0</td>
<td>0.9901</td>
<td>0.9901</td>
<td>0.9901</td>
<td>0.9901</td>
</tr>
<tr>
<td>4.39e0</td>
<td>0.1958</td>
<td>0.1958</td>
<td>0.1958</td>
<td>0.1958</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.0318</td>
<td>0.0318</td>
<td>0.0318</td>
<td>0.0318</td>
</tr>
<tr>
<td>1.18e1</td>
<td>0.0070</td>
<td>0.0070</td>
<td>0.0070</td>
<td>0.0070</td>
</tr>
<tr>
<td>1.93e1</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>3.16e1</td>
<td>0.0004</td>
<td>0.0004</td>
<td>0.0004</td>
<td>0.0004</td>
</tr>
<tr>
<td>6.11e1</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>1.00e3</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

#### Hybrid termination tolerance

<table>
<thead>
<tr>
<th>w</th>
<th>1e-4</th>
<th>1e-6</th>
<th>1e-8</th>
<th>1e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00e0</td>
<td>8.4890</td>
<td>8.4890</td>
<td>8.4890</td>
<td>8.4890</td>
</tr>
<tr>
<td>1.64e0</td>
<td>3.5882</td>
<td>3.5880</td>
<td>3.5878</td>
<td>3.5862</td>
</tr>
<tr>
<td>2.68e0</td>
<td>0.9901</td>
<td>0.9901</td>
<td>0.9901</td>
<td>0.9901</td>
</tr>
<tr>
<td>4.39e0</td>
<td>0.1958</td>
<td>0.1958</td>
<td>0.1958</td>
<td>0.1958</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.0318</td>
<td>0.0318</td>
<td>0.0318</td>
<td>0.0318</td>
</tr>
<tr>
<td>1.18e1</td>
<td>0.0070</td>
<td>0.0070</td>
<td>0.0070</td>
<td>0.0070</td>
</tr>
<tr>
<td>1.93e1</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>3.16e1</td>
<td>0.0004</td>
<td>0.0004</td>
<td>0.0004</td>
<td>0.0004</td>
</tr>
<tr>
<td>6.11e1</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>1.00e3</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>
The results presented here are computed for geometric programming problem 2 of Section 5.4. The decomposed problem formulation is presented in Section A.14. Results are presented for all five evaluated coordination strategies: hierarchical scheme III, all-parallel, sensitivity-based, sensitivity-based with move limits, and the hybrid strategy.

### Scheme III

<table>
<thead>
<tr>
<th>termination tolerance $\tau$</th>
<th>$10^{-4}$</th>
<th>$10^{-6}$</th>
<th>$10^{-8}$</th>
<th>$10^{-10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00e0</td>
<td>11.9969</td>
<td>11.9907</td>
<td>11.8586</td>
<td>11.8742</td>
</tr>
<tr>
<td>1.64e0</td>
<td>7.1388</td>
<td>7.1968</td>
<td>7.1918</td>
<td>7.1863</td>
</tr>
<tr>
<td>2.68e0</td>
<td>3.8870</td>
<td>3.8586</td>
<td>3.6641</td>
<td>3.7807</td>
</tr>
<tr>
<td>4.39e0</td>
<td>1.3352</td>
<td>1.3375</td>
<td>1.3594</td>
<td>1.5924</td>
</tr>
<tr>
<td>7.26e0</td>
<td>0.3385</td>
<td>0.3462</td>
<td>0.3655</td>
<td>0.3677</td>
</tr>
</tbody>
</table>

Elements indicated with a (*) did not terminate

### All-parallel

<table>
<thead>
<tr>
<th>termination tolerance $\tau$</th>
<th>$10^{-4}$</th>
<th>$10^{-6}$</th>
<th>$10^{-8}$</th>
<th>$10^{-10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00e0</td>
<td>11.9968</td>
<td>11.9905</td>
<td>11.8666</td>
<td>11.8844</td>
</tr>
<tr>
<td>1.64e0</td>
<td>7.1388</td>
<td>7.1962</td>
<td>7.1861</td>
<td>7.2105</td>
</tr>
<tr>
<td>2.68e0</td>
<td>3.8870</td>
<td>3.8503</td>
<td>3.6108</td>
<td>3.7912</td>
</tr>
<tr>
<td>4.39e0</td>
<td>1.3383</td>
<td>1.3388</td>
<td>1.3729</td>
<td>1.7023</td>
</tr>
<tr>
<td>7.26e0</td>
<td>0.3380</td>
<td>0.3417</td>
<td>0.3810</td>
<td>0.4705</td>
</tr>
</tbody>
</table>

Elements indicated with a (*) did not terminate
## Appendix G. Results

### Sensitivity-based termination tolerance

<table>
<thead>
<tr>
<th>$w$</th>
<th>$1e-4$</th>
<th>$1e-6$</th>
<th>$1e-8$</th>
<th>$1e-10$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00e0</td>
<td>11.9009</td>
<td>11.9010</td>
<td>11.9018</td>
<td>11.9016</td>
</tr>
<tr>
<td>1.64e0</td>
<td>7.1689</td>
<td>7.1990</td>
<td>7.2083</td>
<td>7.2175</td>
</tr>
<tr>
<td>2.68e0</td>
<td>3.5877</td>
<td>3.5882</td>
<td>3.5943</td>
<td>3.6511</td>
</tr>
<tr>
<td>4.39e0</td>
<td>1.3351</td>
<td>1.3361</td>
<td>1.3484</td>
<td>1.4570</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.3794</td>
<td>0.3932</td>
<td>0.3544</td>
<td>0.5002</td>
</tr>
<tr>
<td>1.18e1</td>
<td>0.0034</td>
<td>0.0042</td>
<td>0.0038</td>
<td>0.0218</td>
</tr>
<tr>
<td>1.93e1</td>
<td>0.0008</td>
<td>0.0010</td>
<td>0.0047</td>
<td>0.0249</td>
</tr>
<tr>
<td>3.16e1</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0023</td>
<td>0.0250</td>
</tr>
<tr>
<td>5.18e1</td>
<td>0.0010</td>
<td>0.0013</td>
<td>0.0237</td>
<td>0.2634</td>
</tr>
<tr>
<td>8.48e1</td>
<td>0.0052</td>
<td>0.0129</td>
<td>0.2498</td>
<td>0.2667</td>
</tr>
<tr>
<td>1.35e2</td>
<td>0.0205</td>
<td>0.2466</td>
<td>0.2489</td>
<td>0.2635</td>
</tr>
<tr>
<td>2.28e2</td>
<td>0.0010</td>
<td>0.0041</td>
<td>0.0246</td>
<td>0.2595</td>
</tr>
<tr>
<td>3.48e2</td>
<td>0.0022</td>
<td>0.0040</td>
<td>0.0152</td>
<td>0.2273</td>
</tr>
<tr>
<td>6.11e2</td>
<td>0.0041</td>
<td>0.0222</td>
<td>0.2433</td>
<td>0.2568</td>
</tr>
<tr>
<td>1.00e3</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

Elements indicated with a * did not terminate

### Sensitivity-based with move limits termination tolerance

<table>
<thead>
<tr>
<th>$w$</th>
<th>$1e-4$</th>
<th>$1e-6$</th>
<th>$1e-8$</th>
<th>$1e-10$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00e0</td>
<td>11.9008</td>
<td>11.9006</td>
<td>11.9016</td>
<td>11.9016</td>
</tr>
<tr>
<td>1.64e0</td>
<td>7.1947</td>
<td>7.1941</td>
<td>7.1952</td>
<td>7.1931</td>
</tr>
<tr>
<td>2.68e0</td>
<td>3.5782</td>
<td>3.5844</td>
<td>3.5789</td>
<td>3.5800</td>
</tr>
<tr>
<td>4.39e0</td>
<td>1.3357</td>
<td>1.3399</td>
<td>1.3479</td>
<td>1.4137</td>
</tr>
<tr>
<td>7.20e0</td>
<td>0.3383</td>
<td>0.3422</td>
<td>0.3583</td>
<td>0.5583</td>
</tr>
<tr>
<td>1.18e1</td>
<td>0.0042</td>
<td>0.0066</td>
<td>0.0014</td>
<td>0.1985</td>
</tr>
<tr>
<td>1.93e1</td>
<td>0.0004</td>
<td>0.0021</td>
<td>0.0027</td>
<td>0.1814</td>
</tr>
<tr>
<td>3.16e1</td>
<td>0.0022</td>
<td>0.0065</td>
<td>0.0134</td>
<td>0.1661</td>
</tr>
<tr>
<td>5.18e1</td>
<td>0.0019</td>
<td>0.0061</td>
<td>0.0192</td>
<td>0.1709</td>
</tr>
<tr>
<td>8.48e1</td>
<td>0.0000</td>
<td>0.0049</td>
<td>0.0100</td>
<td>0.1719</td>
</tr>
<tr>
<td>1.35e2</td>
<td>0.0018</td>
<td>0.0057</td>
<td>0.0160</td>
<td>0.1741</td>
</tr>
<tr>
<td>2.28e2</td>
<td>0.0024</td>
<td>0.0060</td>
<td>0.0165</td>
<td>0.1741</td>
</tr>
<tr>
<td>3.48e2</td>
<td>0.0018</td>
<td>0.0061</td>
<td>0.0192</td>
<td>0.1709</td>
</tr>
<tr>
<td>6.11e2</td>
<td>0.0023</td>
<td>0.0065</td>
<td>0.0134</td>
<td>0.1661</td>
</tr>
<tr>
<td>1.00e3</td>
<td>0.0000</td>
<td>0.0049</td>
<td>0.0100</td>
<td>0.1719</td>
</tr>
</tbody>
</table>

### Hybrid termination tolerance

<table>
<thead>
<tr>
<th>$w$</th>
<th>$1e-4$</th>
<th>$1e-6$</th>
<th>$1e-8$</th>
<th>$1e-10$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00e0</td>
<td>11.9009</td>
<td>11.9000</td>
<td>11.8927</td>
<td>11.8561</td>
</tr>
<tr>
<td>1.64e0</td>
<td>7.1987</td>
<td>7.1972</td>
<td>7.1994</td>
<td>7.2374</td>
</tr>
<tr>
<td>2.68e0</td>
<td>3.5909</td>
<td>3.5916</td>
<td>3.5946</td>
<td>3.9946</td>
</tr>
<tr>
<td>4.39e0</td>
<td>1.3361</td>
<td>1.3200</td>
<td>1.3066</td>
<td>1.5075</td>
</tr>
<tr>
<td>1.18e1</td>
<td>0.0034</td>
<td>0.0042</td>
<td>0.0038</td>
<td>0.0218</td>
</tr>
<tr>
<td>1.93e1</td>
<td>0.0010</td>
<td>0.0210</td>
<td>0.0399</td>
<td>0.6407</td>
</tr>
<tr>
<td>3.16e1</td>
<td>0.0025</td>
<td>0.0412</td>
<td>0.0519</td>
<td>0.8176</td>
</tr>
<tr>
<td>5.18e1</td>
<td>0.0038</td>
<td>0.0411</td>
<td>0.0680</td>
<td>0.9904</td>
</tr>
<tr>
<td>8.48e1</td>
<td>0.0012</td>
<td>0.0436</td>
<td>0.0616</td>
<td>0.5790</td>
</tr>
<tr>
<td>1.35e2</td>
<td>0.0024</td>
<td>0.0420</td>
<td>0.0584</td>
<td>0.6134</td>
</tr>
<tr>
<td>2.28e2</td>
<td>0.0049</td>
<td>0.0625</td>
<td>0.0667</td>
<td>0.9270</td>
</tr>
<tr>
<td>3.48e2</td>
<td>0.0064</td>
<td>0.0876</td>
<td>0.0975</td>
<td>0.9920</td>
</tr>
<tr>
<td>6.11e2</td>
<td>0.0027</td>
<td>0.0636</td>
<td>0.0731</td>
<td>0.5884</td>
</tr>
<tr>
<td>1.00e3</td>
<td>0.0042</td>
<td>0.0443</td>
<td>0.0544</td>
<td>0.7552</td>
</tr>
<tr>
<td>2.28e3</td>
<td>0.0070</td>
<td>0.1694</td>
<td>0.1999</td>
<td>3.5574</td>
</tr>
</tbody>
</table>
Appendix H

MatLab files

This chapter holds all the MatLab files for the implementation of the three example problems used in Chapter 5.

H.1 Global functions

function cvrg=checkcvrg(q,cvrgl,tol,scheme);
[s,k]=size(q);
if norm(norm(cvrgl)^2-length(cvrgl))’^2<tol
   cvrg=0;
else
   if k<scheme
      cvrg=0;
   else
      if norm(q(:,k)-q(:,k-scheme),inf)^2<tol
         cvrg=1;
      else
         cvrg=0;
      end
   end
end

function [f,df]=fcoor(x,q,Q,J,A,w)
f=norm(w.*(A*Q+A*J*(x-q)-x))^2;
if nargout>1
   dfdu=[];
   u=(A*Q+A*J*(x-q)-x);
   dudx=A*J-eye(length(x));
   for i=1:length(x)
      dfdu=[dfdu 2*w(i)^2*u(i)];
   end
   df=dfdu*dudx;
else
end

function [Q,x,cvrg,J]=OptRun(q,nx,objfun,confun,lb,ub,w);
global Nfevals t
s=ones(nx,min(nx,1));
tol=1e-10;
options=optimset(’Display’,’off’,’MaxFunEval’,1000000,’MaxIter’,1000000,...
   ’tolFun’,tol,’tolX’,tol,’tolCon’,tol,...
   ’GradConstr’,’on’,’GradObj’,’on’,’Hessian’,’on’,’DerivativeCheck’,’off’);
xbar=[x;q];
[xbar,fval,exitflag,lambda]=fmincon(objfun,xbar,[],[],[],[],lb,ub,confun,options,q,w);

Nfevals=Nfevals+output.funcCount;
Q=xbar(nx+1:length(xbar));
if nx>0
  x=xbar(1:nx);
else
  end

cvrg=max(sign(exitflag),0);

if nargout==4
  [f,df,ddf,ddf2]=feval(objfun,xbar,q,w);
  [g,h,dg,dh,ddg,ddh]=feval(confun,xbar,q,w);
  Nfevals=Nfevals+2;
  N=[]; Z=0*eye(length(xbar));
  if length(lambda.lower)>0
    for i=1:length(lambda.lower)
      if abs(lambda.lower(i)) > t
        n=zeros(size(xbar)); n(i)=-1; N=[N n];
      else
        end
    end
  end
  if length(lambda.upper)>0
    for i=1:length(lambda.upper)
      if abs(lambda.upper(i)) > t
        n=zeros(size(xbar)); n(i)=1; N=[N n];
      else
        end
    end
  end
  if length(lambda.ineqlin)>0
    for i=1:length(lambda.ineqlin)
      if abs(lambda.ineqlin(i)) > t
        N=[N A(i,:)'];
      else
        end
    end
  end
  if length(lambda.eqlin)>0
    for i=1:length(lambda.eqlin)
      N=[N Aeq(i,:)'];
    end
  end
  if length(lambda.ineqnonlin)>0
    for i=1:length(lambda.ineqnonlin)
      if abs(lambda.ineqnonlin(i)) > t
        Z=Z+lambda.ineqnonlin(i)*ddg(1:length(xbar),((i-1)*length(xbar)+1):(i*length(xbar)));
      else
        end
    end
  end
  if length(lambda.eqnonlin)>0
    for i=1:length(lambda.eqnonlin)
      N=[N dh(:,i)';
    end
  end
  %[a,b]=size(N);
  %H=[ddf + Z h; zeros(b,b)];
  %S=[ddf2;zeros(b,length(q))];
  J=rref([RS]);
  J=J(nx+1:length(xbar),length(xbar)+b1:(length(xbar)+b+length(q)));
else
  end
H.2 Geometric optimization problem 1

Objective and constraint files:

```matlab
function [f,df,ddf,ddf2]=f0(x,q,w)
R01=x(1); Y01=x(2); R02=x(3); Y02=x(4);
r11=q(1); y11=q(2); r12=q(3); y12=q(4);
w11=x(1); w11=w(1); w12=x(3); w12=w(3); w1Y=w(4);
wY=[w1Y w12Y]; wR=[w12R w12R];
f=R01^2+R02^2+norm(wR.*([R01 R02]-[r11 r12]))^2+norm(wY.*([Y01 Y02]-[y11 y12]))^2;
if nargout >= 2
dfdR01= 2*w11R^2*(R01-r11)+2*R01; dfdY01= 2*w11Y^2*(Y01-y11);
dfdR02= 2*w12R^2*(R02-r12)+2*R02; dfdY02= 2*w12Y^2*(Y02-y12);
df=[dfdR01 dfdY01 dfdR02 dfdY02];
if nargout >= 3
ddf=diag([2*w11R^2+2 2*w11Y^2 2*w12R^2+2 2*w12Y^2]);
if nargout >= 4
ddf2=-diag([2*w11R^2 2*w11Y^2 2*w12R^2 2*w12Y^2]);
else
end
else
else
end
end

function [f,df,ddf,ddf2]=f11(x,q,w)
z3=x(1); z4=x(2); r11=x(3); y11=x(4);
R01=q(1); Y01=q(2);
wR=w(1); wY=w(2);
f= (wR*([R01]-[r11]))^2 + (wY*([Y01]-[y11]))^2 ;
if nargout >=2
dfdr11= -2*wR^2*(R01-r11); dfdy11= -2*wY^2*(Y01-y11);
df=[ 0 0 dfdr11 dfdy11];
if nargout >= 3
ddf=diag([0 0 2*wR^2 2*wY^2]);
if nargout >= 4
ddf2=-ddf(:,3:4);
else
end
else
else
end
end

function [f,df,ddf,ddf2]=f12(x,q,w)
z6=x(1); z7=x(2); r12=x(3); y12=x(4);
R02=q(1); Y02=q(2);
wR=w(1); wY=w(2);
f= (wR*([R02]-[r12]))^2 + (wY*([Y02]-[y12]))^2 ;
if nargout >=2
dfdr12= -2*wR^2*(R02-r12); dfdy12= -2*wY^2*(Y02-y12);
df=[ 0 0 dfdr12 dfdy12];
if nargout >= 3
ddf=diag([0 0 2*wR^2 2*wY^2]);
else
end
else
else
end
end
```
202

Appendix H. MatLab files

if nargout >= 4
ddf2=-ddf(:,3:4);
else
end
else
end
else
end

function [g,h,dg,dh,ddg,ddh]=c0(x,q,w)
R01=x(1); Y01=x(2); R02=x(3); Y02=x(4);

h=[Y01-Y02];
g=[];

if nargout >= 4
    dh=[0 1 0 -1]';
    dg=[];
    if nargout >= 6
        dddh=zeros(4,4);
        dddg=[];
    else
    end
end

else
end

function [g,h,dg,dh,ddg,ddh]=c11(x,q,w)

z3=x(1); z4=x(2); r11=x(3); y11=x(4);

h=[r11^2-z3^2-z4^-2-y11^2];
g=[z3^-2+z4^2-y11^2];

if nargout >= 4
    dh=[-2*z3 2*z4^-3 2*r11 -2*y11]';
    dg=[-2*z3^-3 -2*z4 0 -2*y11]';
    if nargout >= 6
        dddh=diag([-2 -6*z4^-4 2 -2]);
        dddg=diag([6*z3^-4 2 0 -2]);
    else
    end
end

else
end

function [g,h,dg,dh,ddg,ddh]=c12(x,q,w)

z6=x(1); z7=x(2); r12=x(3); y12=x(4);

h=[r12^2-y12^2-z6^2-z7^2];
g=[y12^2+z6^-2-z7^2];

if nargout >= 4
    dh=[-2*z6 -2*z7 2*r12 -2*y12]';
    dg=[-2*z6^-3 -2*z7 0 2*y12]';
    if nargout >= 6
        dddh=diag([-2 -2 2 -2]);
        dddg=diag([6*z6^-4 -2 0 2]);
    else
    end
end

else
end

ATC coordination strategy implementations

gcoi_atc.m

clear all
warning off
global Nfevals t
load x_opt
nxe=[0 2 2]; nqe=[4 2 2]; nq=sum(nqe);
lb0=zeros(nxe(1)+nqe(1),1); lb11=zeros(nqe(2),1); lb12=zeros(nq(3),1);
ub0=[]; ub11=[]; ub12=[]; lbq=zeros(nq,1); ubq=Inf*ones(nq,1);
A=[zeros(nqe(1),1) nxe(1)] eye(nqe(1))
I=eye(nq); B0=I(nxe(1),:); B11=I(nqe(1)+1:sum(nqe(1:2)),:); B12=I(sum(nqe(1:2))+1:sum(nqe(1:3)),:);
tol=...; % ATC termination tolerance
w=...; % Scaling factor
kmax=...; % Maximum number of ATC iterations
t=...; % Numerical tolerance factor. Defines within what tolerance numbers are considered zero
scheme=...; % Select scheme 1 = hierarchical III, 2 = all-parallel
sens=...; % Selection of sensitivity-based coordination? 0=No, 1=Yes
safe=...; % Selection of safeguard method? 0=None, 1=Move limits, 2=Hybrid
for i=1:length(w)
    for j=1:length(tol)
        cvrg=0; iter0=0; iter11=0; iter12=0; Nfevals=0; Nf=0;
        z=[1 1 1 1 1 1]; q=[z(1) z(5) z(1) z(5) z(2) z(5)]';
        e=norm(z-x_opt)^2; qhist=q;
        w0=w(i)*ones(nqe(1),1); w11=w(i)*ones(nqe(2),1); w12=w(i)*ones(nqe(3),1);
        while and(cvrg==0,k<kmax)==1
            if scheme==1
                [Q11,x11,cvrg11]=OptRun(B11*q(:,k),nxe(2),'f11','c11',lb11,ub11,w11); iter11=iter11+1;
                [Q12,x12,cvrg12]=OptRun(B12*q(:,k),nq(3),'f12','c12',lb12,ub12,w12); iter12=iter12+1;
                q(1:4,k)=Q11; Q11=x11;
            elseif scheme==2
                [Q0,x0,cvrg0,J0]=OptRun(B0*q(:,k),nxe(1),'f0','c0',lb0,ub0,w0); iter0=iter0+1;
                [Q11,x11,cvrg11,J11]=OptRun(B11*q(:,k),nqe(2),'f11','c11',lb11,ub11,w11); iter11=iter11+1;
                [Q12,x12,cvrg12,J12]=OptRun(B12*q(:,k),nq(3),'f12','c12',lb12,ub12,w12); iter12=iter12+1;
            else
                break
            end
            k=k+1; X(:,k)=x0;x11;x12; Q(:,k)=Q0; Q11; Q12;
            z=[z([Q(1,k); Q(3,k); X(nxe(1)+1:sum(nxe(1:2)),k-1); Q(2,k); X(sum(nxe(1:2))+1:sum(nxe(1:3)),k-1))];
            if and(scheme==2,sens==0)
                Qcheck=[B0*Q(:,k-2:k); B11*B12*Q(:,k-3:k-1)]
                cvrg=cvrgcvqg(Qcheck,[cvrg0 cvrg11 cvrg12],tol(j),2);
                cvrg0=0;
            end
            z=[z(:,k); Q(3,k); X(nxe(1)+1:sum(nxe(1:2)),k-1); Q(2,k); X(sum(nxe(1:2))+1:sum(nxe(1:3)),k-1)];
            else
                cvrg=cvrgcvqg([Q0(Q0(1);Q0(3);x11;Q0(2);x12)],tol(j),1); z=[z([Q0;Q1;Q11;Q12]);
            end
        end
    end
end

Appendix H. MatLab files

H.3 Geometric optimization problem 2, decomposition 1

Objective and constraint files:

function \([f,df,ddf,ddf2]=f0(x,q,w)\)

```matlab
R01=x(1); Y01=x(2); R02=x(3); Y02=x(4);
r11=x(1); y11=x(2); r12=x(3); y12=x(4);
w11=w(1); w11y=w(2); w12=w(3); w12y=w(4);
wR=[w11y w12y]; wY=[w11y w12y];
f=R01^2+R02^2+norm(wR.*([R01 R02]-[r11 r12]))^2+norm(wY.*([Y01 Y02]-[y11 y12]))^2;
if nargout >= 2
    dfdR01= 2*w11R^2*(R01-r11)+2*R01; dfdY01= 2*w11Y^2*(Y01-y11); dfdR02= 2*w12R^2*(R02-r12)+2*R02; dfdY02= 2*w12Y^2*(Y02-y12);
    df=[dfdR01 dfdY01 dfdR02 dfdY02];
    if nargout >= 3
ddf=diag([2*w11R^2+2 2*w11Y^2 2*w12R^2+2 2*w12Y^2]);
    elseif nargout >= 4
        ddf2=-diag([2*w11R^2 2*w11Y^2 2*w12R^2 2*w12Y^2]);
    end
else
    end
else
    end
end
```

function \([f,df,ddf,ddf2]=f11(x,q,w)\)

```matlab
z4=x(1); r11=x(2); y11=x(3); R11=x(4);
R01=q(1); Y01=q(2); r21=q(3);
```

Function files:
function \[ f, df, ddf, ddf2 \] = f12(x,q,w)

\[ z7 = x(1); r12 = x(2); y12 = x(3); R12 = x(4); \]
\[ R02 = q(1); Y02 = q(2); r22 = q(3); \]
\[ wR = [w(1) w(3)]; wY = w(2); \]
\[ f = \text{norm}(w.*([R02 R12] - [r12 r22]))^2 + (wY^2*(Y02 - y12))^2; \]

if nargout \geq 2
\[ dfdr12 = -2*wR(1)^2*(R02-r12); dfdy12 = -2*wY^2*(Y02-y12); dfdR12 = 2*wR(2)^2*(R12-r22); \]
\[ df = [0 dfdr12 dfdy12 dfdR12]; \]
if nargout \geq 3
\[ ddf = \text{diag}([0 2*wR^2 2*wY^2 2*wR(2)^2]); \]
if nargout \geq 4
\[ ddf2 = -ddf(:,2:4); \]
else
else
end

function \[ f, df, ddf, ddf2 \] = f21(x,q,w)

global p
\[ z8 = x(1); z9 = x(2); z10 = x(3); r21 = x(4); \]
\[ R11 = q(1); \]
\[ wR = w(1); \]
\[ f = wR*(R11-r21)^2; \]

if nargout \geq 2
\[ df = [0 0 0 -2*wR*(R11-r21)]; \]
if nargout \geq 3
\[ ddf = \text{diag}([0 0 0 2*wR^2]); \]
if nargout \geq 4
\[ ddf2 = -ddf(:,4); \]
else
else
end

function \[ f, df, ddf, ddf2 \] = f22(x,q,w)

global p
\[ z12 = x(1); z13 = x(2); z14 = x(3); r22 = x(4); \]
\[ R12 = q(1); \]
\[ wR = w(1); \]
\[ f = wR*(R12-r22)^2; \]

if nargout \geq 2
\[ df = [0 0 0 -2*wR*(R12-r22)]; \]
if nargout \geq 3
\[ ddf = \text{diag}([0 0 0 2*wR^2]); \]
if nargout \geq 4
\[ ddf2 = -ddf(:,4); \]
else
else
end

function [g, h, dg, dh, ddg, ddh] = c0(x, q, w)
R01=x(1); Y01=x(2); R02=x(3); Y02=x(4);
h=[Y01-Y02];
g=[];
if nargout >= 4
    dh=[0 1 0 -1]';
    dg=[];
    if nargout >= 6
        ddh=zeros(4,4);
        ddg=[];
    else
        end
    end
else
    end
end

function [g, h, dg, dh, ddg, ddh] = c11(x, q, w)
z4=x(1); r11=x(2); y11=x(3); R11=x(4);
h=[r11^2-R11^2-z4^2-y11^2];
g=[R11^-2+z4^2-y11^2];
if nargout >= 4
    dh=[2*z4^-3 2*r11 -2*y11 -2*R11]';
    dg=[2*z4 0 -2*y11 -2*R11^-3]';
    if nargout >= 6
        ddh=diag([-6*z4^-4 2 -2 -2]);
        ddg=diag([-2 0 2 6*R11^-4]);
    else
        end
    end
else
    end
end

function [g, h, dg, dh, ddg, ddh] = c12(x, q, w)
z7=x(1); r12=x(2); y12=x(3); R12=x(4);
h=[r12^2-y12^2-R12^2-z7^2];
g=[y12^2+R12^-2-z7^2];
if nargout >= 4
    dh=[-2*z7 2*r12 -2*y12 -2*R12]';
    dg=[-2*z7 0 2*y12 -2*R12^-3]';
    if nargout >= 6
        ddh=diag([-2 2 -2 -2]);
        ddg=diag([-2 0 2 6*R12^-4]);
    else
        end
    end
else
    end
end

function [g, h, dg, dh, ddg, ddh] = c21(x, q, w)
global p
z8=x(1); z9=x(2); z10=x(3); r21=x(4);
h=r21^2-z8^2-z9^-2-z10^-2-p^2;
g=[z8^2+z9^2-p^2; z9^-2-z10^-2-p^2];
if nargout >= 4
    dh=[-2*z8 2*z9^-3 2*z10^-3 2*r21]';
    dg=[2*z8 2*z9 0 0];
else
    end
end
H.3. Geometric optimization problem 2, decomposition 1

\[-2z_8-3 0 2z_{10} 0\];

if nargout > 6
    ddh=diag([-2 -6z_9^-4 -6z_{10}^-4 2]);
    ddg1=diag([2 0 0 0]);
    ddg2=diag([0 0 0 0 2 0 0 0 0 0 0 0]);
    ddg=[ddg1 ddg2];
else
    end
else
    end

function [g,h,dg,dh,ddg,ddh]=c22(x,q,w)
global p
z12=x(1); z13=x(2); z14=x(3); r22=x(4);
h=r22^2-z12^2-z13^2-z14^2-p^2;
g=[p^2-z12^-2-z13^2; p^2+z12^2-z14^2];
if nargout >= 4
    dh=[-2*z12 -2*z13 -2*z14 2*r22]';
    dg=[-2*z12^-3 -2*z13 0 0; 2*z12 0 -2*z14 0];
if nargout >= 6
    ddh=diag([-2 -2 -2 2]);
    ddg1=diag([6*z12^-4 -2 0 0]);
    ddg2=diag([2 0 -2 0]);
    ddg=[ddg1 ddg2];
else
    end
else
    end

ATC coordination strategy implementations

genco2deco1_atc.m
clear all
warning off
global Nfevals t p
load x_opt
p=x_opt(11);
x_opt=x_opt';
nxe=[0 1 3 3]; nqe=[4 3 3 1]; nq=sum(nqe);
lb0=zeros(nxe(1)+nqe(1),1); lb11=zeros(nxe(2)+nqe(2),1); lb12=zeros(nxe(3)+nqe(3),1); lb21=zeros(nxe(4)+nqe(4),1); lb22=zeros(nxe(5)+nqe(5),1); ub0=[]; ub11=[]; ub12=[]; ub21=[]; ub22=[];
lbq=zeros(nq,1); ubq=Inf*ones(nq,1);
A=zeros(nq,nq); A(1:2,5:6)=eye(2); A(3:4,8:9)=eye(2); A(5:6,1:2)=eye(2); A(7,11)=1; A(8:9,3:4)=eye(2); A(10,12)=1; A(11,7)=1; A(12,10)=1;
I=eye(nq); B0=I(1:nqe(1),:); B11=I(nqe(1)+1:sum(nqe(1:2)),:); B12=I(sum(nqe(1:2))+1:sum(nqe(1:3)),:);
B21=I(sum(nqe(1:3))+1:sum(nqe(1:4)),:); B22=I(sum(nqe(1:4))+1:sum(nqe),:);
tol=...; % ATC termination tolerance
w=...; % Scaling factor
s=...; % Move limits of coordination process
kmax=...; % Maximum number of ATC iterations
sens=...; % Selection of sensitivity-based coordination? 0=No, 1=Yes
safe=...; % Selection of safeguard? 0=None, 1=Move limits, 2=Hybrid
for i=1:length(w)
    for j=1:length(tol)
        cvrg=0; iter0=0; iter11=0; iter12=0; iter21=0; iter22=0; Nfevals=0; N=0;
        z=ones(14,1); q=[z(1) z(5) z(2) z(5) z(1) z(5) z(3) z(2) z(5) z(6) z(3) z(6) z(6)];
        X=[z(4) z(7) z(8:10)'' z(12:14)'']; Q=[];
        e=norm(z-x_opt)^2; qcoor=q; qatc=q;
        while and(cvrg==0,k<kmax)==1

    end
end
end
end
if scheme==1
    qbranch1=[zeros(nqe(2)+nqe(4),1)]; kbranch1=0; cvrgbranch1=0;
    while and(cvrgbranch1==0,kbranch1<kmax)==1
        [Q21,x21,cvrg21]=OptRun(B21*q(:,k),nxe(4),'f21','c21',lb21,ub21,w21);
        q(7,k)=Q21; iter21=iter21+1;
        [Q11,x11,cvrg11]=OptRun(B11*q(:,k),nxe(2),'f11','c11',lb11,ub11,w11);
        q(1:2,k)=Q11(1:2); q(11,k)=Q11(3); iter11=iter11+1;
        qbranch1=[qbranch1 [Q11;Q21]]; kbranch1=kbranch1+1;
        cvrgbranch1=checkcvrg(qbranch1,[cvrg11 cvrg21],tol(j),scheme);
    end
    qbranch2=[zeros(nqe(3)+nqe(5),1)]; kbranch2=0; cvrgbranch2=0;
    while and(cvrgbranch2==0,kbranch2<kmax)==1
        [Q22,x22,cvrg22]=OptRun(B22*q(:,k),nxe(5),'f22','c22',lb22,ub22,w22);
        q(10,k)=Q22; iter22=iter22+1;
        [Q12,x12,cvrg12]=OptRun(B12*q(:,k),nxe(3),'f12','c12',lb12,ub12,w12);
        q(3:4,k)=Q12(1:2); q(12,k)=Q12(3); iter12=iter12+1;
        qbranch2=[qbranch2 [Q12;Q22]]; kbranch2=kbranch2+1;
        cvrgbranch2=checkcvrg(qbranch2,[cvrg21 cvrg22],tol(j),scheme);
    end
    [Q0,x0,cvrg0]=OptRun(B0*q(:,k),nxe(1),'f0','c0',lb0,ub0,w0);
    q(5:6,k)=Q0(1:2); q(8:9,k)=Q0(3:4); iter0=iter0+1;
    elseif scheme==2
        [Q0,x0,cvrg0,J0]=OptRun(B0*q(:,k),nxe(1),'f0','c0',lb0,ub0,w0); iter0=iter0+1;
        [Q11,x11,cvrg11,J11]=OptRun(B11*q(:,k),nxe(2),'f11','c11',lb11,ub11,w11); iter11=iter11+1;
        [Q12,x12,cvrg12,J12]=OptRun(B12*q(:,k),nxe(3),'f12','c12',lb12,ub12,w12); iter12=iter12+1;
        [Q21,x21,cvrg21,J21]=OptRun(B21*q(:,k),nxe(4),'f21','c21',lb21,ub21,w21); iter21=iter21+1;
        [Q22,x22,cvrg22,J22]=OptRun(B22*q(:,k),nxe(5),'f22','c22',lb22,ub22,w22); iter22=iter22+1;
    else
        end
end
if scheme==2, sens==0)
    if k>4
        Qcheck=[B0*Q(:,k-2);[B11;B12]*Q(:,k-3);[B21;B22]*Q(:,k-4)]
        cvrg=checkcvrg(Qcheck,[cvrg0 cvrg11];tol(j),3);
    else
        cvrg=0;
    end
    z=[x [Q11(1:3);Q21(1:3);Q7(1:1);X(sum(nxe(1:2)),k-1);Q12(k-1);X(sum(nxe(1:3)),k-1);X(sum(nxe(1:3)))+1;sum(nxe(1:4)),k-1]; p; X(sum(nxe(1:4)))+1;sum(nxe(1:5))];
else
    cvrg=checkcvrg(Q,[cvrg0 cvrg11 cvrg21];tol(j),1);
    z=[x [Q0(1:3);Q11(3);x11;Q0(2);Q12(3);x12;x21;p;x22]];
end
=norm(x(k)-xopt)^2; Nf=[Nf Nfevals];
if (cvrg==0, sens==1)
    J=B0'*J0*B0 + B11'*J11*B11 + B12'*J12*B12 + B21'*J21*B21 + B22'*J22*B22;
    W=w(1:11,1:12);w12;w22)
    &eqcoor=[]; beqcoor=[];
    if safe==0
        S=Inf;ones(eq,1);
    elseif safe==1
        S=ones(eq,1);
    elseif safe==2
        S=Inf;ones(eq,1);
        if rank(J0)==null(1)
            &eqcoor=[eqcoor; B0*ones(1)];
            beqcoor=[beqcoor; B0*Q1(1));
        else
            end
        end
        if rank(J11)==null(2)
            &eqcoor=[eqcoor; B11*ones(A1)];
H.4. Geometric optimization problem 2, decomposition 2

beqcoor=[beqcoor; B11*Q(:,k)];
else
end
if rank(J12) == nQi(3)
Aeqcoor=[Aeqcoor; B12*inv(A)];
beqcoor=[beqcoor; B12*Q(:,k)];
else
end
if rank(J21) == nQi(4)
Aeqcoor=[Aeqcoor; B21*inv(A)];
beqcoor=[beqcoor; B21*Q(:,k)];
else
end
if rank(J22) == nQi(5)
Aeqcoor=[Aeqcoor; B22*inv(A)];
beqcoor=[beqcoor; B22*Q(:,k)];
else
else
end
end
if length(beqcoor) < nq
tolcoor=1e-10;
options=optimset('Display','off','TolX',tolcoor,'TolCon',tolcoor,'TolFun',tolcoor,...
'MaxIter',100000,'MaxFunEvals',100000,...
'GradObj','on','DerivativeCheck','off');
[qnew,f,exitflag,output]=fmincon('fcoor',A*Q(:,k),[],[],Aeqcoor,beqcoor,...
[max(lbq,q(:,k-1)-S)]',[min(ubq,q(:,k-1)+S)],[],options,q(:,k-1),Q(:,k),J,A,W);
Nfevals=Nfevals+output.funcCount;
qcoor=[qcoor qnew]; qatc=[qatc q(:,k)]; q(:,k)=qnew;
else
else
end
end
N(i,j)=max(Nf); K(i,j)=k-1; E(i,j)=e(length(e));
end
end

H.4 Geometric optimization problem 2, decomposition 2

Objective and constraint files:

function [f,df,ddf,ddf2]=f0(x,q,w)
R01=x(1); Y01=x(2:3); R02=x(4); Y02=x(5:6);
r11=q(1); y11=q(2:3); r12=q(4); y12=q(5:6);
w11wR=w(1); w11wY=w(2:3); w12wR=w(4); w12wY=w(5:6);
wY=[w11wY' w12wY' ]; wR=[w11wR w12wR ];
f=R01^2+R02^2+norm(wR.*([R01 R02]-[r11 r12]))^2+ norm(wY.*([Y01 Y02]'-[y11 y12]'))^2;
if nargout >= 2
dfdR01= 2*w11wR^2*(R01-r11)+2*R01; dfdY01= 2*w11wY.^2.*(Y01-y11); dfdR02= 2*w12wR^2*(R02-r12)+2*R02; dfdY02= 2*w12wY.^2.*(Y02-y12);
df=[dfdR01 dfdY01' dfdR02 dfdY02'];
if nargout >= 3
ddf=dig([2+1iR 2+1iY'.*2 2+1iR 2+1iY'.*2]);
if nargout >= 4
ddf2=diag([2+1iR 2+1iY'.*2 2+1iR 2+1iY'.*2]);
else
else
end
else
function [f,df,ddf,ddf2]=f11(x,q,w)
Appendix H. MatLab files

function \[f, df, ddf, ddf2\] = f12(x,q,w)
\[ f = \|wR \times (R02 - r22)\|^2 + \|wY \times (Y02 - y12)\|^2 \]
if nargout >= 2
\[ df = \begin{bmatrix} 0 & dfdr12 & dfdy12' & dfdR12' \end{bmatrix} \]
if nargout >= 3
\[ ddf = \begin{bmatrix} 0 & 2 \times wR(1)^2 & 2 \times wY'.^2 & 2 \times wR(2:3).'^2 \end{bmatrix} \]
if nargout >= 4
\[ ddf2 = -ddf(:,2:6) \]
else
end
else
end
end

function \[f, df, ddf, ddf2\] = f21(x,q,w)
\[ f = \|wR \times (R11 - r21)\|^2 \]
if nargout >= 2
\[ df = \begin{bmatrix} 0 & 0 & 0 & 0 & -2 \times wR'.*(R11 - r21) \end{bmatrix} \]
if nargout >= 3
\[ ddf = \begin{bmatrix} 0 & 0 & 0 & 2 \times wR'.^2 \end{bmatrix} \]
if nargout >= 4
\[ ddf2 = -ddf(:,4:5) \]
else
end
else
end
end

function \[f, df, ddf, ddf2\] = f22(x,q,w)
\[ f = \|wR \times (R11 - r21)\|^2 \]
if nargout >= 2
\[ df = \begin{bmatrix} 0 & 0 & 0 & 0 & -2 \times wR'.*(R11 - r21) \end{bmatrix} \]
if nargout >= 3
\[ ddf = \begin{bmatrix} 0 & 0 & 0 & 2 \times wR'.^2 \end{bmatrix} \]
if nargout >= 4
\[ ddf2 = -ddf(:,4:5) \]
else
end
else
end
end

function \[f, df, ddf, ddf2\] = f23(x,q,w)
\[ f = \|wR \times (R11 - r21)\|^2 \]
if nargout >= 2
\[ df = \begin{bmatrix} 0 & 0 & 0 & 0 & -2 \times wR'.*(R11 - r21) \end{bmatrix} \]
if nargout >= 3
\[ ddf = \begin{bmatrix} 0 & 0 & 0 & 2 \times wR'.^2 \end{bmatrix} \]
if nargout >= 4
\[ ddf2 = -ddf(:,4:5) \]
else
end
else
end
end

function \[f, df, ddf, ddf2\] = f24(x,q,w)
\[ f = \|wR \times (R11 - r21)\|^2 \]
if nargout >= 2
\[ df = \begin{bmatrix} 0 & 0 & 0 & 0 & -2 \times wR'.*(R11 - r21) \end{bmatrix} \]
if nargout >= 3
\[ ddf = \begin{bmatrix} 0 & 0 & 0 & 2 \times wR'.^2 \end{bmatrix} \]
if nargout >= 4
\[ ddf2 = -ddf(:,4:5) \]
else
end
else
end
end
H.4. Geometric optimization problem 2, decomposition 2

\[ R_{12} = q(1:2); \]
\[ w_{R} = w; \]
\[ f = \text{norm}(w_{R} \cdot (R_{12} - r_{22})); \]

if \ nargout >= 2
\[ df = [0 0 0 0 -2w_{R} \cdot (R_{12} - r_{22})]; \]
if \ nargout >= 3
\[ ddf = \text{diag}([0 0 0 0 2w_{R} \cdot -2]); \]
if \ nargout >= 4
\[ ddf2 = -ddf(:,4:5); \]
else
  end
else
  end
else
  end

function [g, h, dg, dh, ddg, ddh] = c0(x, q, w)
  R01 = x(1); Y01 = x(2:3); R02 = x(4); Y02 = x(5:6);
  h = [Y01 - Y02];
  g = [ ];

if nargout >= 4
  dh = [0 1 0 0 -1 0; 0 0 1 0 0 -1]';
  dg = [ ];
if nargout >= 6
  dddh1 = zeros(6, 6);
  dddh2 = zeros(6, 6);
  dddh = [dddh1 dddh2];
else
  end
else
  end

function [g, h, dg, dh, ddg, ddh] = c11(x, q, w)
  z4 = x(1); r11 = x(2); y11 = x(3:4); R11 = x(5:6);
  h = [r11^2 - R11(1)^2 - z4^-2 - y11(1)^2; R11(2) - y11(2)];
  g = [R11(1)^-2 + z4^2 - y11(1)^2];

if nargout >= 4
  dh = [2*z4^-3 2*r11 -2*y11(1) 0 -2*R11(1) 0; 0 0 0 -1 0 1]';
  dg = [2*z4 0 -2*y11(1) 0 -2*R11(1)^-3 0];
if nargout >= 6
  dddh1 = diag([-6*z4^-4 2 -2 0 -2 0]);
  dddh2 = zeros(6, 6);
  dddh = [dddh1 dddh2];
else
  end
else
  end

function [g, h, dg, dh, ddg, ddh] = c12(x, q, w)
  z7 = x(1); r12 = x(2); y12 = x(3:4); R12 = x(5:6);
  h = [r12^2 - y12(1)^2 - R12(1)^2 - z7^2; R12(2) - y12(2)];
  g = [y12(1)^2 + R12(1)^-2 - z7^2];

if nargout >= 4
  dh = [-2*z7 2*r12 -2*y12(1) 0 -2*R12(1) 0; 0 0 0 -1 0 1]';
  dg = [-2*z7 0 2*y12(1) 0 -2*R12(1)^-3 0];
if nargout >= 6
ddh1=diag([-2 2 -2 0 -2 @ 0]);
ddh2=zeros(6,6);
ddh=[ddh1 ddh2];
ddg=diag([-2 0 2 0 6*R12(1)^-4 0]);
else
end
else
end

function [g,h,dg,dh,ddg,ddh]=c21(x,q,w)
z8=x(1); z9=x(2); z10=x(3); z15=x(4); r21=x(5:6);
h=[r21(1)^2-z8^2-z9^2-z10^2-z15^2; r21(2)-z15];
g=[z8^2+z9^2+z15];
if nargout >= 4
dh=[-2*z8 2*z9 -3 2*z10 -3 -2*z15 2*r21(1) 0; 0 0 0 -1 0 1];
dg=[2*z8 2*z9 -2 2*z15 0 0; -2*z8^3 0 -2*z10 -2 z15 0 0];
if nargout >= 6
ddh1=diag([-2 -6*z9^-4 -6*z10^-4 -2 2 0]);
ddh2=zeros(6,6);
ddh=[ddh1 ddh2];
ddg1=diag([2 2 0 -2 0 0]);
ddg2=diag([6*z8^-4 0 2 -2 0 0]);
ddg=[ddg1 ddg2];
else
end
else
end

function [g,h,dg,dh,ddg,ddh]=c22(x,q,w)
z12=x(1); z13=x(2); z14=x(3); z16=x(4); r22=x(5:6);
h=[r22(1)^2-z12^2-z13^2-z14^2-z16^2; r22(2)-z16];
g=[z16^2+z12^-2-z13^2; z16^2+z12^2-z14^2];
if nargout >= 4
dh=[-2*z12 -2*z13 -2*z14 -2*z16 2*r22(1) 0; 0 0 0 -1 0 1];
dg=[-2*z12^-3 -2*z13 0 2*z16 0 0; 2*z12 0 -2*z14 2*z16 0 0];
if nargout >= 6
ddh1=diag([-2 -2 -2 0 0]);
ddh2=zeros(6,6);
ddh=[ddh1 ddh2];
ddg1=diag([6*z12^-4 -2 0 2 2 0]);
ddg2=diag([2 0 -2 0 2 0]);
ddg=[ddg1 ddg2];
else
end
else
end

ATC coordination strategy implementations
geo2dec0_atc.m

clear all
warning off
global Nfevals t
load x_opt
nxe=[0 1 1 4 4]; nqe=[6 5 5 2 2]; nqi=[4 4 4 2 2]; nq=nnz(nqe);
lb0=zeros(nxe(1)+nqe(1),1); lb11=zeros(nxe(2)+nqe(2),1); lb12=zeros(nxe(3)+nqe(3),1); lb21=zeros(nqe(4)+nqe(5),1);
lb22=zeros(nqe(5),1); ub0=lb0; ub11=lb11; ub12=lb12; ub21=lb21; ub22=lb22;
lbq=lb0(:); ubq=ub0(:);
A=zeros(nq,nq); A(1:3,4:6)=eye(3); A(4:6,7:9)=eye(3);
H.4. Geometric optimization problem 2, decomposition 2

\[
\begin{align*}
\text{tol} &= \ldots \quad \% \text{ATC termination tolerance} \\
\text{w} &= \ldots \quad \% \text{Scaling factor} \\
\text{nmax} &= \ldots \quad \% \text{Maximum number of ATC iterations} \\
\text{t} &= \ldots \quad \% \text{Numerical tolerance factor. Defines within what tolerance numbers are considered zero} \\
\text{sens} &= \ldots \quad \% \text{Selection of sensitivity-based coordination? 0=No, 1=Yes} \\
\text{safe} &= \ldots \quad \% \text{Selection of safeguard? 0=None, 1=Move limits, 2=Hybrid} \\
\end{align*}
\]

for \( i = 1 : \text{length}(w) \)
for \( j = 1 : \text{length}(\text{tol}) \)
\[
\begin{align*}
\text{cvrg} &= 0; \quad \text{iter0}=0; \quad \text{iter11}=0; \quad \text{iter21}=0; \quad \text{iter22}=0; \quad \text{Nfevals}=0; \quad \text{Nf}=0; \\
\text{z} &= \text{ones}(14,1); \\
\text{q} &= \text{[z}(1)\text{z}(5)\text{z}(11)\text{z}(2)\text{z}(5)\text{z}(11)\text{z}(3)\text{z}(6)\text{z}(11)\text{z}(5)\text{z}(6)\text{z}(11)]; \\
\text{q} &= \text{[z}(4)\text{z}(7)\text{z}(8:10)\text{z}(11)\text{z}(12:14)\text{z}(11)]; \\
\text{Q} &= \ldots; \\
\text{w0} &= \text{w}(i)\text{ones}(\text{nqe}(1),1); \\
\text{w11} &= \text{w}(i)\text{ones}(\text{nqe}(2),1); \\
\text{w12} &= \text{w}(i)\text{ones}(\text{nqe}(3),1); \\
\text{w21} &= \text{w}(i)\text{ones}(\text{nqe}(4),1); \\
\text{w22} &= \text{w}(i)\text{ones}(\text{nqe}(5),1); \\
\end{align*}
\]
\[
\begin{align*}
\text{while and} \left( \text{cvrg}==0, \text{iter}<\text{kmax} \right) = 1 \\
\text{if scheme=1} \\
\text{qbranch1} &= \text{zeros(\text{nqe}(2)+\text{nqe}(4),1); kbranch1=0; cvrgbranch1=0;} \\
\text{while and} \left( \text{cvrgbranch1}==0, \text{iter}<\text{kmax} \right) = 1 \\
\text{[Q21,x21,cvrg21]} &= \text{OptRun(B21*q(:,iter),\text{nxe}(4),'f21','c21',\text{lb21},\text{ub21},\text{w21});} \\
\text{q}(10:11,iter) &= \text{Q21}; \text{iter21} = \text{iter21}+1; \\
\text{[Q11,x11,cvrg11]} &= \text{OptRun(B11*q(:,iter),\text{nxe}(2),'f11','c11',\text{lb11},\text{ub11},\text{w11});} \\
\text{q}(1:3,iter) &= \text{Q11}(1:3); \text{q}(17:18,iter) &= \text{Q11}(4:5); \text{iter11} = \text{iter11}+1; \\
\text{qbranch1} &= \text{[qbranch1 [Q11;Q21]; kbranch1} = \text{kbranch1}+1; \\
\text{cvrgbranch1} &= \text{checkcvrg(qbranch1, [cvrg11 cvrg21], \text{tol}(j), scheme);} \\
\text{end} \\
\text{end} \\
\text{qbranch2} &= \text{zeros(\text{nqe}(3)+\text{nqe}(5),1); kbranch2=0; cvrgbranch2=0;} \\
\text{while and} \left( \text{cvrgbranch2}==0, \text{iter}<\text{kmax} \right) = 1 \\
\text{[Q22,x22,cvrg22]} &= \text{OptRun(B22*q(:,iter),\text{nxe}(5),'f22','c22',\text{lb22},\text{ub22},\text{w22});} \\
\text{q}(15:16,iter) &= \text{Q22}; \text{iter22} = \text{iter22}+1; \\
\text{[Q12,x12,cvrg12]} &= \text{OptRun(B12*q(:,iter),\text{nxe}(3),'f12','c12',\text{lb12},\text{ub12},\text{w12});} \\
\text{q}(4:6,iter) &= \text{Q12}(4:6); \text{q}(19:20,iter) &= \text{Q12}(4:5); \text{iter12} = \text{iter12}+1; \\
\text{qbranch2} &= \text{[qbranch2 [Q12;Q22]; kbranch2} = \text{kbranch2}+1; \\
\text{cvrgbranch2} &= \text{checkcvrg(qbranch2, [cvrg12 cvrg22], \text{tol}(j), scheme);} \\
\text{end} \\
\text{end} \\
\text{[Q0,x0,cvrg0]=OptRun(B0*q(:,iter),\text{nxe}(1),'f0','c0',\text{lb0},\text{ub0},\text{w0});} \\
\text{q}(7:9,iter) &= \text{Q0}(1:3); \text{q}(12:14,iter) &= \text{Q0}(4:6); \text{iter} = \text{iter}+1; \\
\text{elseif scheme=2} \\
\text{[Q0,x0,cvrg0,R]=OptRun(B0*q(:,iter),\text{nxe}(1),'f0','c0',\text{lb0},\text{ub0},\text{w0});} \\
\text{iter0} = \text{iter0}+1; \\
\text{[Q11,x11,cvrg11]=OptRun(B11*q(:,iter),\text{nxe}(2),'f11','c11',\text{lb11},\text{ub11},\text{w11});} \\
\text{iter11} = \text{iter11}+1; \\
\text{[Q12,x12,cvrg12,J12]=OptRun(B12*q(:,iter),\text{nxe}(3),'f12','c12',\text{lb12},\text{ub12},\text{w12});} \\
\text{iter12} = \text{iter12}+1; \\
\text{[Q21,x21,cvrg21,R]=OptRun(B21*q(:,iter),\text{nxe}(4),'f21','c21',\text{lb21},\text{ub21},\text{w21});} \\
\text{iter21} = \text{iter21}+1; \\
\text{[Q22,x22,cvrg22,J22]=OptRun(B22*q(:,iter),\text{nxe}(5),'f22','c22',\text{lb22},\text{ub22},\text{w22});} \\
\text{iter22} = \text{iter22}+1; \\
\text{end} \\
\text{else} \\
\text{end} \\
\text{k} += 1; \\
\text{X}(:,k) = \text{[x0;x11;x12;x21;x22]; Q}(:,k) = \text{[Q0;Q11;Q12;Q21;Q22]; Q}(:,k) += 4*Q(:,k); \\
\text{if and(scheme==2,sens==0} \\
\text{if k>3} \\
\text{Qcheck} &= \text{[B0;B11;B12;B21;B22]Q(:,3-k:1);[B21;B22]Q(:,k-2:k);} \\
\text{cvrg} &= \text{checkcvrg(Qcheck, [cvrg0 cvrg11 cvrg12 cvrg21 cvrg22], tol(j), 2);} \\
\text{else} \\
\text{cvrg} &= 0; \\
\text{end} \\
\text{R} &= \text{[Q1(1:k),Q(4:4),Q(10:k-1),X(1:k-1),Q(1:k), Q(15:k-1),X(2:k-1),X(3:5:k),Q(3:k),X(7:9:k)]}; \\
\text{else} \\
\text{cvrg} &= \text{checkcvrg(Q, [cvrg0 cvrg11 cvrg12 cvrg21 cvrg22], tol(j), 1);} \\
\text{R} &= \text{[Q0(1:4),Q(1:4),x1;Q(1:2),Q(2:4),x1;x21(1:3),Q(3:3),x22(1:3:3)];} \\
\text{end} \\
\end{align*}
\]
\[ e = \text{norm}(x(:,k) - x_{\text{opt}})^2 \] 
\[ N_f = \text{length}(\text{Nfevals}) \] 

\[
\begin{align*}
e & = \text{norm}(x(:,k) - x_{\text{opt}})^2; \\
N_f & = \text{length}(\text{Nfevals}); \\
\text{if and}(\text{cvrg}==0, \text{sens}==1) & \\
J & = B_0^*J_0B_0 + B_{11}^*J_{11}B_{11} + B_{12}^*J_{12}B_{12} + B_{21}^*J_{21}B_{21} + B_{22}^*J_{22}B_{22}; \\
W & = [w_0; w_{11}; w_{12}; w_{21}; w_{22}]; \\
\&qcoor & = []; \\
begqcoor & = []; \\
\text{if safe} & = 0 \\
S & = \text{Inf} \times \text{max}(nq,1); \\
\text{elseif safe} & = 1 \\
S & = \text{max}(\text{Nmax}(nq,1)); \\
\text{elseif safe} & = 2 \\
S & = \text{Inf} \times \text{max}(nq,1); \\
\text{if rank}(J_{01}) & = nQi(1) \\
\text{Aeqcoor} & = [\text{Aeqcoor}; B_{01} \times \text{inv}(A)]; \\
\text{begqcoor} & = [\text{begqcoor}; B_{01}Q(:,k)]; \\
\text{elseif} & \\
\text{if rank}(J_{11}) & = nQi(2) \\
\text{Aeqcoor} & = [\text{Aeqcoor}; B_{11} \times \text{inv}(A)]; \\
\text{begqcoor} & = [\text{begqcoor}; B_{11}Q(:,k)]; \\
\text{elseif} & \\
\text{if rank}(J_{12}) & = nQi(3) \\
\text{Aeqcoor} & = [\text{Aeqcoor}; B_{12} \times \text{inv}(A)]; \\
\text{begqcoor} & = [\text{begqcoor}; B_{12}Q(:,k)]; \\
\text{elseif} & \\
\text{if rank}(J_{21}) & = nQi(4) \\
\text{Aeqcoor} & = [\text{Aeqcoor}; B_{21} \times \text{inv}(A)]; \\
\text{begqcoor} & = [\text{begqcoor}; B_{21}Q(:,k)]; \\
\text{elseif} & \\
\text{if rank}(J_{22}) & = nQi(5) \\
\text{Aeqcoor} & = [\text{Aeqcoor}; B_{22} \times \text{inv}(A)]; \\
\text{begqcoor} & = [\text{begqcoor}; B_{22}Q(:,k)]; \\
\text{else} & \\
\text{end} \\
\text{end} \\
\text{end} \\
\text{else} \\
\text{end} \\
\text{if length(\text{begqcoor}) < nq} \\
tolcoor & = 10^{-10}; \\
\text{options} & = \text{optimset}(\{'\text{Display}','\text{off}', '\text{TolX}', \text{tolcoor}', '\text{TolCon}', \text{tolcoor}', '\text{TolFun}', \text{tolcoor}', \\
'\text{MaxIter}', 100000, '\text{MaxFunEvals}', 100000, \\
'GradObj', 'on', '\text{DerivativeCheck}', '\text{off}'\}); \\
[qcoor, f, \text{exitflag, output}] & = \text{fmincon}(\text{\text{fcoor}}, A*Q(:,k), [], [], \text{Aeqcoor}, \text{begqcoor}, [], [], [], [], \text{options}, q(:,k-1), Q(:,k), J, A, W); \\
\text{Nfevals} & = \text{Nfevals} + \text{output}.\text{funcCount}; \\
\text{Nfevals} & = \text{output}.\text{funcCount}; \\
\text{qcoor} & = [qcoor, qnew]; \\
\text{qatc} & = [\text{qatc}, q(:,k)]; \\
q(:,k) & = qnew; \\
\text{end} \\
\text{end} \\
\text{end} \\
\text{end} \\
\text{end} \\
\text{end} \\
\text{end} \\
\text{end} \\
\text{end} \\
\text{end} \\
\text{end}