Master of Science Thesis

Acceleration of partitioned fluid-structure interaction simulations by means of space mapping

An analysis of suitable approaches

Liesbeth Florentie

March 28, 2012

Aerodynamics

TU Delft

Faculty of Aerospace Engineering

Delft University of Technology

Acceleration of partitioned fluid-structure interaction simulations by means of space mapping

An analysis of suitable approaches

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

Liesbeth Florentie

March 28, 2012

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF AERODYNAMICS

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance the thesis entitled **"Acceleration of partitioned fluid-structure interaction simulations by means of space mapping"** by **Liesbeth Florentie** in fulfillment of the requirements for the degree of **Master of Science**.

Dated: March 28, 2012

Supervisors:

Prof. dr. ir. drs. H. Bijl

Dr. ir. A.H. van Zuijlen

Dr. D.J.P. Lahaye

Ir. T.P. Scholcz

Preface

Finishing this thesis marks the end of my time as a TU Delft student, and it was a great time! Thanks to my friends and teachers the past five and a half years have been both very joyful and instructive, making me look back at it with great pleasure.

Reaching this point would not have been possible without the help of some people. First of all I wish to thank my supervisors Hester Bijl, Sander van Zuijlen en Thomas Scholcz. I am very grateful for their guidance and enthusiasm throughout the past year. Furthermore I am glad they introduced me to the world of space mapping optimization, a technique I was completely unfamiliar with when I started this project. Therefore I would especially like to thank Thomas for supplying me with his code in order to get me started.

Also a huge thank you goes out to my parents for their endless support. I am very thankful they gave me the opportunity to come to Delft and to become an aerospace engineer.

And finally to my beloved boyfriend, Mirko, thanks a lot for all your love, support and distraction. Live would be boring without you!

Liesbeth Florentie,
Delft, March 2012

Summary

Modelling of fluid-structure interactions (FSI) plays a key role in many engineering applications. However, due to the interaction between fluid and structure the computational cost related to high fidelity models (especially for strongly coupled systems) limits the direct use of current FSI simulation techniques in industry. Since a thorough knowledge of FSI phenomena is very important in design processes, efficient simulation techniques that combine low cost with high accuracy need to be developed. To this end, the use of the space mapping optimization technique as coupling approach for partitioned FSI simulations is investigated.

The space mapping technique is a collection of efficient optimization approaches which use a low fidelity coarse model to accelerate the evaluation of expensive high fidelity fine models. A surrogate model, whose response resembles the fine model response, is build by defining a parameter mapping that is obtained through the iterative optimization and updating of the coarse model. In this way the computational effort is shifted from the high fidelity model to the much faster to evaluate coarse model without any losses in accuracy. As a result, only a few fine model evaluations (for calibration) are required to obtain high fidelity solutions. Several space mapping methods can be found in literature. Of these, four methods were identified to be potential candidates for the use as coupling method for partitioned FSI simulations. Those methods (aggressive space mapping (ASM), generalized space mapping (GSM), output space mapping (OSM) and implicit space mapping (ISM)) are implemented for the 1D piston problem on a fixed domain and the 2D panel problem.

ASM defines a surrogate that consists of the coarse model in conjunction with an input mapping. Every iteration starts with a fine model evaluation, after which the mapped fine model variable is obtained by matching the particular fine and coarse model responses. The subsequent estimate for the fine model solution is then obtained by applying an approximation for the inverse mapping (using a first order Taylor series) to the coarse model optimal solution. It is found that ASM is successful in performing strongly coupled FSI simulations for the considered test problems. Savings in computational cost up to 30% with respect to a conventional Quasi-Newton (QN) method are achieved.

GSM employs a predefined combination of input and output mappings. First the misalignment between both models is minimized over a region of interest by utilizing a linear input mapping. Afterward, an output mapping is used which locally matches the surrogate response with the fine model response and which minimizes the deficit between their first order derivatives. The mapping parameters are obtained by means of a parameter extraction (PE) process where a choice can be made between single point PE, multipoint PE, gradient PE and constrained PE. The next iterate is then found by optimizing the resulting surrogate model. A Taylor series expansion of the mapping parameters during the surrogate optimization is proposed to improve the convergence properties of GSM. This is found to be very effective. GSM is applied to both test problems. The method succeeded in obtaining strongly coupled solutions, but the efficiency gains are small or even non-existing with respect to QN. This is due to the parameter extraction process which appears to be very expensive for large systems. Moreover, GSM is complicated to implement due to the large choice in PE approaches and weighting factors, both of which have a large influence on the overall convergence rate.

Based on GSM, an output space mapping approach is developed in this work which predicts the deficit between the fine and the coarse model. This is done by adding a first order Taylor series approximation for the deficit to the coarse model, where the Jacobian is approximated using Broydens rank one scheme. In this way a locally perfect mapping is obtained which exhibits both excellent approximation and extrapolation properties. This method is successfully applied to both test problems. OSM is found to be up to 50% more efficient than QN. Moreover, the method is simple and robust.

Finally, the use of ISM was tested. ISM builds a surrogate by altering physical parameters of the coarse model like mass, stiffness or density. The optimal values for those preassigned parameters are obtained using a PE process, after which the resulting surrogate is optimized to find the next iterate. This method was not successful for the considered test problems. The influence of the preassigned parameters is too small to obtain a sufficient match between both models. Moreover, ISM does not allow for the use of black box solvers. Therefore it is found that ISM is not suitable for FSI simulations.

Of all coupling methods that were implemented, OSM is found to be the most efficient. It exhibits by far the largest convergence rate. This method is validated by applying it to other combinations of fine and coarse models. The piston problem on a deforming domain is considered, for which OSM yielded very good results as well. Moreover, coarse models that consist of the fine model on a coarse grid are used for the 1D and 2D test problems. Again OSM was able to obtain strongly coupled solutions, although the obtained efficiencies were lower than for coarse models that are based on simplified physics. The latter made clear that a successful implementation of OSM requires a coarse model that is both cheap ($\bar{W}_c/\bar{W}_f < 1\%$ is advised) and reasonably accurate. For the current test problems OSM became inefficient for cases where the average error of the coarse model is $> 25\%$. More research is required in order to be able to identify optimal coarse models.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction to the project

Modelling of fluid-structure interactions (FSI) plays a key role in many engineering applications. Examples of these include wind interaction with cable stayed bridges, bio-fluids in deformable vessels and flutter/buffeting of wings or buildings [11]. Being able to analyze those systems with high accuracy is very important, not only to predict the performance of fluid-structure systems, but also to investigate the instability of coupled (dynamical) systems. However, due to the interaction between fluid and structure the computational cost related to high fidelity models (especially for strongly coupled systems) limits their direct use in industry. So in order to be useful, efficient simulation techniques that combine low cost with high accuracy need to be developed.

Since a thorough knowledge of fluid-structure interaction phenomena is very important in the design of future aircraft, the FFAST project (future fast aeroelastic simulation technologies, [16]) aims at improving the efficiency of aeroelastic computations. The FFAST project will develop, implement and assess a range of candidate numerical simulation technologies to accelerate the aircraft design process. The research is conducted in three major areas, being

- fast identification of critical load cases,
- extraction of aerodynamic and aeroelastic reduced order models (ROMs),
- reduced order model acceleration of full order models.

To achieve these objectives, complementary experiences and expertise are gathered to form a trans-national consortium consisting of universities, research agencies and aerospace companies. Within this context, Delft University of Technology researches the acceleration of high fidelity aeroelastic simulations using aeroelastic reduced order models and will implement

these algorithms in a generic coupling shell. In addition to this goal the constraint of software modularity is identified. This allows CFD and CSM communities to develop their own codes using their own expertise. So a partitioned approach is favoured over a monolithic solution. Moreover, the final solution will have to be robust, minimal intrusive and generic.

As part of this FFAST project T.P. Scholcz [27] looked into different techniques that might lower the computational cost of fluid-structure interaction calculations by combining a fast but inaccurate coarse model with a highly accurate but expensive fine model. Expensive in this context means that the computational effort to solve the system is large with respect to both time and cost. He found that especially space mapping, a surrogate based optimization technique mainly used for electromagnetic design, has high potential in achieving this goal. The space mapping technique was first introduced by Bandler et al. in 1994 [4] and the idea behind it is to establish a connection between a coarse and a fine model through a parameter mapping. This mapping is then used to find an optimal set of parameters such that the response of the coarse model resembles the response of the fine model as good as possible. The optimization is then performed using the coarse model such that considerably less fine model iterations are needed, leading to savings in computational cost up to 90% [21; 20].

In this report a closer look is taken at the application of the space mapping method for strongly coupled partitioned fluid-structure interaction problems. After a description of the method in Chapter 3 the implementation and performance of several space mapping approaches are investigated in Chapters 4 to 8. This is done by application to two commonly used test problems, being the one-dimensional piston problem and the two-dimensional panel problem. Both are introduced in Chapter 2. However, this chapter continues with a general introduction to fluid-structure interaction simulations first.

1.2 Introduction to computational fluid-structure interaction

1.2.1 General problem formulation

When designing aircraft one has to take into account aeroelastic phenomena. Those arise when structural deformations induce changes on aerodynamic forces. These additional aerodynamic forces cause an increase in the structural deformations, which in turn influence the aerodynamic forces in a feedback process. So when investigating aeroelastic effects one has to solve a coupled fluid-structure system. A schematic representation of the FSI problem is shown in Fig. 1.1.

A FSI problem consists of the fluid domain Ω_f and the structure domain Ω_s. Along the boundary Γ_I between those two domains both displacements and pressure/stresses should match, leading to a kinematic

$$\mathbf{d}_f = \mathbf{d}_s \qquad \text{or} \qquad \mathbf{v}_f = \dot{\mathbf{d}}_s \tag{1.1}$$

and a dynamic

$$\mathbf{n}_f \cdot \tau_f = -\mathbf{n}_s \cdot \tau_s \tag{1.2}$$

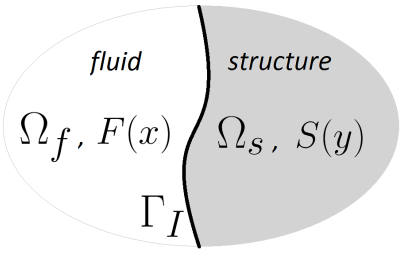


Figure 1.1: Schematic representation of the fluid-structure domain.

boundary condition. In (1.1) and (1.2) $\mathbf{d}$ represents the displacement, $\mathbf{v}$ the velocity and τ the stress tensor at the interface. The subscripts f and s indicate the fluid and the structure variables respectively, and $\mathbf{n}$ is the unit normal vector of the boundary pointing outward.

So in order to allow interactions between both systems information should be transferred across the interface. However, structural equations are usually formulated in material (Lagrangian) coordinates whereas the fluid dynamics are typically expressed using spatial (Eulerian) coordinates. To allow for the transfer of information both representations should coincide at the interface. This is done by rewriting the fluid equations in the Arbitrary Lagrangian Eulerian (ALE) formulation [11]. In this approach the observation scheme is not fixed in space nor moves with the material but has an arbitrary motion in order to follow the boundary conditions. The ALE discretization therefore can be locally Eulerian (mainly in the far field), purely Lagrangian (near the interface) or something in between. Modelling the interactions between the structure and the fluid can now be done according to two approaches: monolithic or partitioned.

Monolithic FSI approach When a monolithic approach is used the flow and structural dynamics are solved simultaneous using a single solver, such that their mutual influence can be taken into account directly. The aim is to put all the necessary components, including the physical models, discretizations and solution algorithms for the fluid as well as the structure, into a single computational model,

$$M(\mathbf{w_s}, \mathbf{w_f}) = 0, \tag{1.3}$$

with $\mathbf{w_s}$ and $\mathbf{w_f}$ the discrete states of the structure and fluid. This simulation approach is favourable from a stability and accuracy point of view, but impractical since a certain monolithic solver is only applicable for one specific type of problems. A new, complicated solver has to be build for every new FSI problem to be analyzed. Moreover, it is very hard to keep the solver up to date with the latest developments in each research field.

Partitioned FSI approach Contrary to a monolithic approach, when applying a partitioned approach to perform FSI calculations the governing equations for the flow and structure are solved separately using distinct solvers. This eliminates the problems encountered for monolithic simulations, since software modularity is preserved and therefore the most suitable

solvers for both the fluid and the structure can be chosen independent of each other. However, a partitioning error is introduced since the implicit system of equations is now split up and thus not solved simultaneously. A coupling algorithm, which allows for the interaction between both solvers, therefore is required to assure the matching conditions at the fluid-structure interface. By iterating over the interface the partitioning error can be minimized. This coupling method determines to a large extent the overall accuracy and efficiency of the simulation.

Comparing both approaches to the requirements set for the FFAST project, it is clear that a partitioned approach should be used. This is why issues regarding monolithic FSI approaches are not considered here.

In a partitioned approach, the fluid and structure solvers can be considered to be arbitrary units generating a desired output from a given input and are often referred to as "black boxes". Denoting the black box fluid solver by F, generating a pressure $\mathbf{y}$ from a certain displacement $\mathbf{x}$, and the black box structure solver by S, generating a displacement from a given pressure distribution,

$$\mathbf{y} = F(\mathbf{x}) \qquad \text{and} \qquad \mathbf{x} = S(\mathbf{y}), \tag{1.4}$$

the FSI problem can be formulated as a nonlinear problem in the discrete interface position with all other variables internal to the black box solvers. So the goal of the coupling algorithm is to enforce

$$\mathbf{R}(\mathbf{x}) = S \circ F(\mathbf{x}) - \mathbf{x} = \mathbf{0} \tag{1.5}$$

along the interface. If only one iteration is performed, the problem is referred to as being loosely coupled. This can be the case if e.g. the structure is very stiff. However, for flexible structures loose coupling will result in a large partitioning error. As this is undesirable, multiple iterations are required until (1.5) is satisfied. Those cases are referred to as being strongly coupled. In the current work, the aim is to obtain strongly coupled solutions using as few iterations as possible.

1.2.2 Coupling approaches

Two kinds of coupling at the interface have to be ensured when performing FSI calculations: spatial coupling and temporal coupling, both of which allow for the transfer of information between the fluid and the structural domain.

Spatial coupling

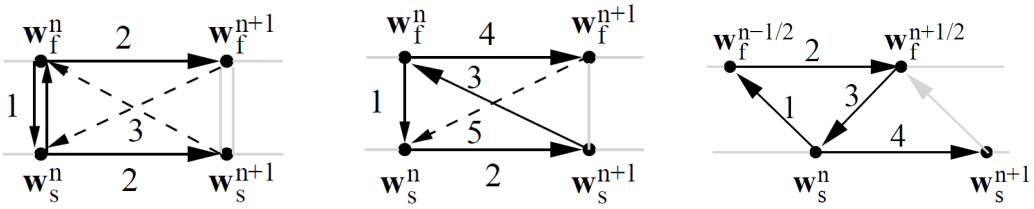
The goal of spatial coupling is to deal with non-matching meshes at the interface, such that data can be transferred. Usually the fluid and structure meshes do not match at the interface due to the different mesh requirements posed on the flow and the structure (e.g. the flow generally requires a finer grid than the structure) and the use of independent solvers. Several approaches which can be applied to ensure spatial coupling can be found in the work of de Boer [12]. Most widely applied are

- Nearest neighbour interpolation,
- Weighted residual method, and
- Radial basis function (RBF) interpolation.

For the test problems used in this work the fluid and structure meshes match at the interface hence no spatial interpolation is required.

Temporal coupling

On the other hand, temporal coupling aims at ensuring (1.5) in time, so updating the solution from time level n to $n+1$. Three schemes which are often used to couple both black box solvers are the additive Schwartz (or block-Jacobi) method, multiplicative Schwartz (or block-Gauss-Seidel) method and the midpoint staggered scheme. Those approaches are visualized in Fig. 1.2. Note that the block-Jacobi and block-Gauss-Seidel methods allow for subiterating whereas the midpoint staggered scheme is only applicable for loose coupling. For strongly



(a) Block-Jacobi (b) Block-Gauss-Seidel (c) Midpoint staggered

Figure 1.2: Temporal coupling methods [11], the dotted arrows indicate the subiterations for strong coupling.

coupled problems either a block-Jacobi or a block-Gauss-Seidel method with sub-iterations to ensure a perfect match between the displacements and stresses (pressure) along the boundary can be considered. This sub-iteration process can be accelerated by making use of a suitable iterative method. Commonly used approaches include

- fixed point iterations,
- Aitken relaxation,
- the Newton-Raphson method,
- Richardsons method,
- Incremental quasi-Newton (QN),
- Newton-Krylov methods.

For the FSI simulations performed in this work a Gauss-Seidel temporal coupling approach is used, which first evaluates the fluid and then uses the pressure output of the fluid solver as input for the structure solver. The Gauss-Seidel partitioning scheme is preferred over the block-Jacobi method since it generally needs only half the number of subiterations. A drawback over block-Jacobi is that the fluid and structure solvers cannot be evaluated in parallel. However, since the required computing time for the structure is in general much less than for the fluid solver, this does not increase the total computing time. A fully coupled solution will be obtained by performing space mapping subiterations, where a QN method is used as reference.

1.2.3 Efficiency improvements for strongly coupled FSI problems by means of reduced order models (ROM)

Although the solution of strong coupling procedures converges to the monolithic solution if sufficient subiterations are applied, thus decreasing the partitioning error and thereby increasing the accuracy, subiterating increases the computational cost to a large extend. Even when an appropriate method is applied to perform the subiterations, the flow and structure have to be solved multiple times per timestep, leading to a large computational cost. Therefore, much research is conducted regarding methods that can improve the efficiency of partitioned FSI simulations lately. A large amount of this research focuses on the construction and application of reduced order models (ROM's). In those methods, the original high-order system of governing equations (called the fine model) is transformed to a system of much lower order, whereby only the most important or dominant part of the system dynamics is preserved. By combining the information obtained by solving both the fine and the coarse models the number of required subiterations is reduced. Several methods to construct those ROM's and to use them in order to accelerate the coupling procedure are available, an overview of which can be found in [23; 25; 26; 31]. Although promising results are obtained by these new methods, most of them do not satisfy all constraints as identified by the FFAST project [16], being software modularity, minimal intrusiveness, robustness and genericity. Moreover, a lot of the existing reduced order model methods require access to the mathematical or conceptual abstraction level of the fine model and are therefore not suitable for use as generic coupling algorithm. The space mapping method however is believed to satisfy all those constraints [28].

1.3 Research plan

The current research deals with the application of the space mapping technique to FSI problems. The goal of it can be summarized as:

> *"To reduce the computational cost required to obtain a fully coupled solution for partitioned fluid-structure interaction problems by means of a space mapping method."*

The desired output of this research is a space mapping algorithm which only needs a fraction of the number of subiterations required by conventional coupling algorithms like QN to eliminate the partitioning error, without increasing the computational effort per subiteration. Moreover, the resulting algorithm should be relatively simple to implement and allow for the use of black box fluid and structure solvers. The following approach is applied in order to reach this goal:

- Performing a literature survey in order to identify different space mapping methods currently employed in other research areas and investigating which methods are most likely to be useful for FSI problems. This part of the project can be found in [15].
- Implementation of the most promising space mapping approaches for two academic test problems, a one- and a two-dimensional. The details regarding the applied test problems can be found in Chapter 2 and a description and the results of the implemented algorithms are gathered in Chapters 4 to 7.
- Comparison of the implemented space mapping methods with QN as reference, for different fluid-to-structure mass ratios and different timesteps (see Chapter 8).
- Validation of the space mapping approach which yields the highest efficiency by application to some other fine-coarse model combinations (see Chapter 9).

1.4 Efficiency measures

For all methods a fully coupled solution will be computed such that the partitioning error will be zero for all methods. Hence, the algorithms will be judged purely on the computational effort required to obtain this solution. The most efficient approach therefore is the method which proves to be the cheapest to obtain the required solution. The computational cost of each method is normalized with respect to the average CPU time required to perform one fine model evaluation ($\bar{W}_f$) and is defined as

$$W^{SM}(t_n) = N_f^{SM}(t_n) + \frac{\bar{W}_c}{\bar{W}_f} N_c^{SM}(t_n) \qquad \text{and} \qquad W^{QN}(t_n) = N_f^{QN}(t_n). \tag{1.6}$$

In the above $N_f(t_n)$ and $N_c(t_n)$ are the number of fine and coarse model iterations evaluated at time t_n respectively, $\bar{W}_c$ is the average cost of one coarse model evaluation and the superscripts SM and QN indicate that use has been made of a space mapping method or the Quasi-Newton method to perform the subiterations. Making use of (1.6), the total normalized computational cost thus becomes

$$W_{tot}^{SM} = \sum_{n=1}^{t_{final}/\Delta t} \left(N_{f_n}^{SM} + N_{c_n}^{SM} \frac{\bar{W}_c}{\bar{W}_f} \right) \qquad \text{and} \qquad W_{tot}^{QN} = \sum_{n=1}^{t_{final}/\Delta t} N_{f_n}^{QN} \tag{1.7}$$

for the space mapping and Quasi-Newton method respectively. Here the subscript n denotes the time levels. As long as the coarse computational cost is small compared to the fine computational cost, the number of fine model evaluations required per timestep is a good indication for the overall efficiency of the method.

Apart from the total computational cost, also the relative efficiency with respect to the Quasi-Newton method can be calculated for every space mapping method under consideration. The time dependent and total efficiency are defined as

$$\eta(t_n) = 1 - \frac{W^{SM}(t_n)}{W^{QN}(t_n)} \qquad \text{and} \qquad \eta_{tot} = 1 - \frac{W^{SM}_{tot}}{W^{QN}_{tot}}. \tag{1.8}$$

If $\eta > 0$ the space mapping method appears to be more efficient than the Quasi-Newton method.

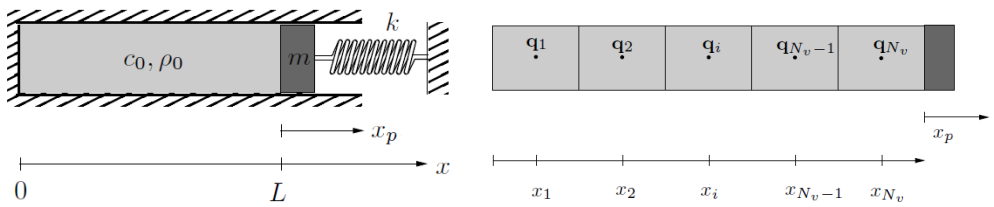
Chapter 2

Test Problems

This chapter contains a description of the fluid-structure interaction problems which will be used to test several space mapping methods and to compare their numerical performance to conventional coupling approaches. A simple one dimensional problem, the frequently used piston problem, as well as a more complicated two dimensional problem are presented below. These test problems are the same as the ones used by Scholcz in [29].

2.1 The 1D piston problem

The one dimensional piston problem describes a compressible inviscid fluid (with equilibrium density ρ_0 and speed of sound c_0) in a closed section which is in interaction with a piston having a certain mass m and stiffness k, see Fig. 2.1.



(a) Conceptual domain.

(b) Computational domain.

Figure 2.1: Visualization of the piston problem, [29].

2.1.1 Governing equations

The conceptual domain is discretized using a central finite volume scheme with a transpiration boundary condition on the fluid-structure interface, the piston. This results in the computational domain as visualized in Fig. 2.1. The structural model is given by a simple linear mass-spring system, which in nondimensional form looks like

$$\bar{m}\ddot{\bar{x}}_p + \bar{k}\bar{x}_p = \Delta\bar{p}, \tag{2.1}$$

with x_p the piston displacement. Since the fluid is considered to be one-dimensional, compressible, inviscid and isentropic it is governed by the Euler equations

$$\frac{\partial}{\partial t}\mathbf{q} + \frac{\partial}{\partial x}\mathbf{F}(\mathbf{q}) = \mathbf{0} \tag{2.2}$$

with the state and flux vectors give by

$$\mathbf{q} = \begin{pmatrix} \rho \\ \rho u \end{pmatrix} \qquad \text{and} \qquad \mathbf{F}(\mathbf{q}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \end{pmatrix}. \tag{2.3}$$

In (2.3) ρ represents the fluid density and u the horizontal fluid velocity. Applying the finite volume method to the integral form of (2.2), both the structure and fluid dynamics equations can be transformed into a semi-discrete nonlinear system of equations

$$\frac{\partial}{\partial t}\begin{pmatrix} \mathbf{w_f} \\ \mathbf{w_s} \end{pmatrix} + \begin{pmatrix} \mathbf{A_f} & \mathbf{A_{fs}} \\ \mathbf{A_{sf}} & \mathbf{A_s} \end{pmatrix}\begin{pmatrix} \mathbf{w_f} \\ \mathbf{w_s} \end{pmatrix} = \mathbf{0} \tag{2.4}$$

where $\mathbf{w_f} = (\ \mathbf{q}_1^T \quad \mathbf{q}_2^T \quad \dots \quad \mathbf{q}_{N_v}^T\ )^T$ and $\mathbf{w_s} = (\ \bar{x}_p \quad \dot{\bar{x}}_p\ )^T$ are the state vectors of the fluid and structure respectively, $\mathbf{A_f}$ and $\mathbf{A_s}$ the fluid and structure discretization matrices and $\mathbf{A_{fs}}$ and $\mathbf{A_{sf}}$ are the fluid-structure coupling matrices. The fluid dynamics are solved iteratively, using Aitken relaxation with a convergence tolerance $\epsilon^{fluid} = 1 \cdot 10^{-15}$.

The semi-discrete structure and fluid equations of motion (2.4) are integrated using the BDF-2 time-integration scheme. The tolerance in the coupling algorithm is set to a very small value $\epsilon = 1 \cdot 10^{-12}$, resulting in a strongly coupled solution. When the BDF-2 scheme is used, the piston velocity is derived from the piston displacement as

$$\dot{\bar{x}}_p^{n+1} = \frac{\frac{3}{2}\bar{x}_p^{n+1} - 2\bar{x}_p^n + \frac{1}{2}\bar{x}_p^{n-1}}{\Delta t} \tag{2.5}$$

such that the piston displacement can be assumed to be the only unknown during the coupling iterations.

2.1.2 Coarse model

When applying space mapping in order to solve the fluid-structure interaction problem, a coarse model governing the same physics but which is much faster to solve (hence less accurate) should be available. Here a coarse model corresponding to the piston problem is found

from linearizing the fluid flux vector $\mathbf{F}(\mathbf{q})$ with respect to the equilibrium state of the fluid ($\mathbf{q_0} = (\ \rho_0 \quad 0\ )^T$). The fluid is governed by

$$\frac{\partial}{\partial t}\mathbf{q}' + \frac{\partial}{\partial \mathbf{q}}\,\mathbf{F}|_{\mathbf{q}=\mathbf{q_0}}\,\frac{\partial}{\partial x}\mathbf{q}' = \mathbf{0} \tag{2.6}$$

with $\frac{\partial}{\partial \mathbf{q}}\mathbf{F}$ the Jacobian of the nonlinear flux vector and $\mathbf{q}'$ the perturbation with respect to $\mathbf{q_0}$. The structural model remains the same as above since the computational effort required so solve it is only a fraction of the cost of the fine fluid model. So the coarse fluid dynamics are now governed by the linear semi-discrete system

$$\frac{\partial}{\partial t}\mathbf{w_f}' + \mathbf{A_f}\mathbf{w_f}' + \mathbf{A_{fs}}\tilde{\mathbf{w}}_\mathbf{s} = \mathbf{0} \tag{2.7}$$

with the coarse state vector $\tilde{\mathbf{w}}_\mathbf{s} = (\ \bar{x}_p \quad \dot{\bar{x}}_p\ )^T_c$. For this linear piston problem an exact solution exists which can be found in [24; 32].

2.1.3 1D Testcases

The numerical parameters and initial conditions for both the fine and coarse model are gathered in table 2.1, where P represents the (non-dimensional) coupled period of the fluid-structure model.

		Fine model	Coarse model
Initial piston displacement	$\bar{x}_p^0$	0.5	0.5
Initial fluid density	$\bar{\rho}^0$	2/3	-1/3
Number of fine volume cells	N_v	64	64
Number of timesteps	N_t	$10 \cup 20 \cup 30$	$10 \cup 20 \cup 30$
Timestep	Δt	$P/10 \cup P/20 \cup P/30$	$P/10 \cup P/20 \cup P/30$

Table 2.1: Nondimensional initial conditions and numerical parameters for the 1D piston problem.

In order to study the performance of the space mapping methods, various levels of coupling strength are considered. This is done by varying the fluid-to-structure mass ratio

$$\zeta = \frac{\rho_0 L}{m}, \tag{2.8}$$

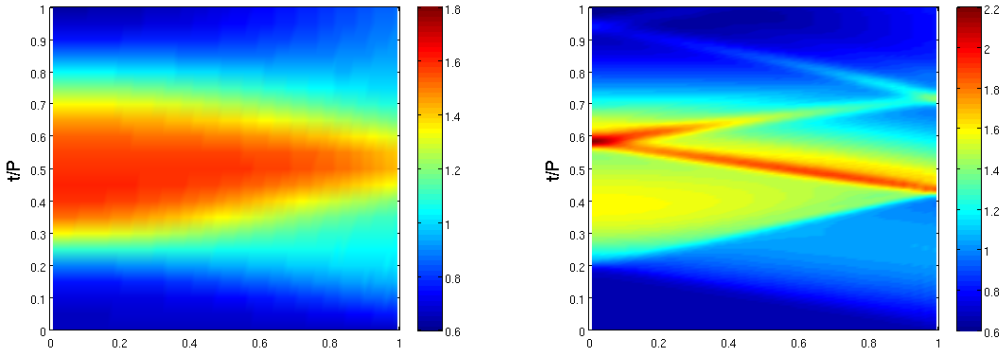
while keeping the ratio of characteristic time scales

$$\lambda = \frac{L\omega}{c_0} \tag{2.9}$$

constant. In (2.9) $\omega = \sqrt{\frac{k}{m}}$ is the natural frequency of the mass-spring system. Three testcases with increasing fluid-to-structure mass ratios will be used, which are given in table 2.2.

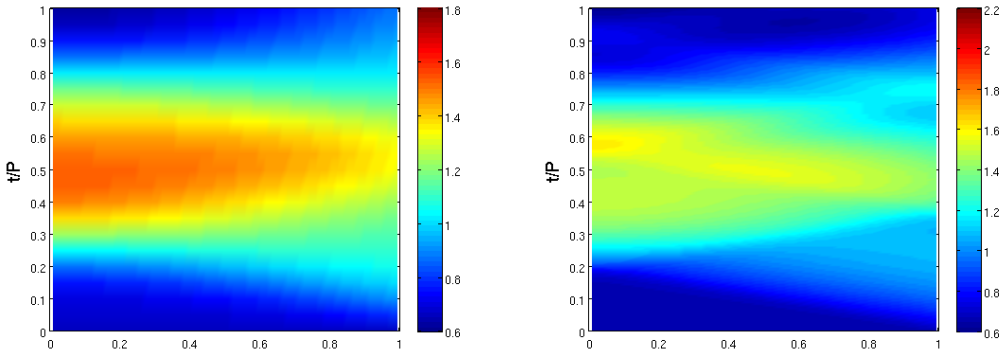
	Similarity parameters		Structural parameters		Fluid parameters			P
	$\zeta[-]$	$\lambda[-]$	$m[kg]$	$k[\frac{kg}{ms^2}]$	$L[m]$	$\rho_0[\frac{kg}{m^3}]$	$c_0[\frac{m}{s}]$	
FSI-1	1/2	0.85	4	64300	2	1	300	6.19
FSI-2	2/3	0.85	3	48225	2	1	300	5.96
FSI-3	2	0.85	1	16075	2	1	300	5.04

Table 2.2: Physical and similarity parameters of the testcases for the 1D piston problem.



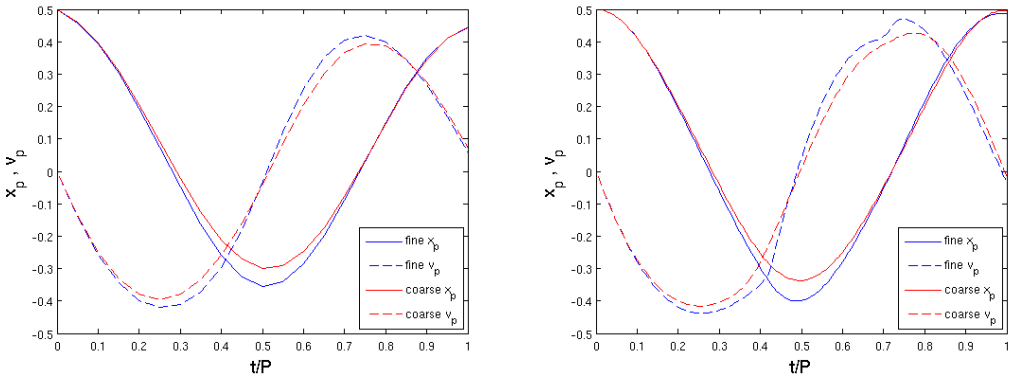
(a) $\Delta t = P/20$ (b) $\Delta t = P/200$

Figure 2.2: Nondimensional density field for the fine model, testcase FSI-2.



(a) $\Delta t = P/20$ (b) $\Delta t = P/200$

Figure 2.3: Nondimensional density field for the coarse model, testcase FSI-2.



(a) $\Delta t = P/20$ (b) $\Delta t = P/200$

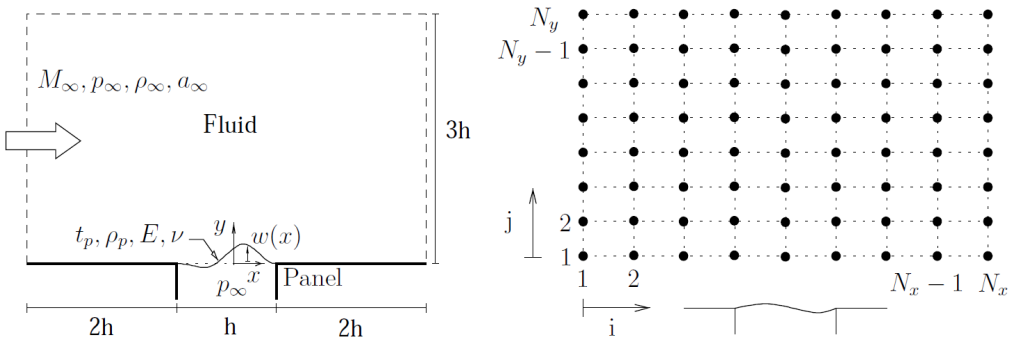
Figure 2.4: Piston displacement and velocity for both the fine and coarse model, testcase FSI-2.

In Figs. 2.2, 2.3 and 2.4 the density field, piston displacement and piston velocity are shown for both the fine and the coarse model. Because the pressure wave departing from the piston requires a small timestep size in order to be resolved completely, the results are given for both $\Delta t = P/20$ and $\Delta t = P/200$. Large differences in the resulting fluid density are observed between both the fine and the coarse model and the small and the large timesteps. However, when looking at the piston response it can be seen that the coarse model approximates the fine model well, the observed deficits for both the amplitude and frequency are rather small. The piston responses are also alike for both timestep sizes, the only differences occur at the moments the pressure wave departs from the piston and when it hits the piston again. For the other testcases the same behaviour is observed.

Because decreasing the applied timestep seems to have little influence on the deficit between the fine and coarse piston response, in the following chapters rather large timesteps are used to analyze the performance of different space mapping methods. Only in Chapter 8 time integration with $\Delta t = P/200$ is used again. This is done in order to save computing time when looking at the space mapping performance over multiple periods. Finally, it is found that the computational cost to perform one coarse model evaluation (so including both the structure and fluid dynamics) is approximately $\frac{\bar{W}_c}{\bar{W}_f} = 0.3\%$ of the cost of one fine model evaluation.

2.2 The 2D panel problem

The two-dimensional panel problem [29; 30] considers a flexible panel clamped at both ends and in interaction with an inviscid fluid at supersonic speed, see Fig. 2.2. Here M_∞, a_∞, p_∞ and ρ_∞ are the free stream Mach number, speed of sound, pressure and density respectively. t_p, ρ_p, E and ν are the panel thickness, density, Young modulus and Poisson ratio. The vertical displacement of the flexible panel is given by $w(x)$. On the fluid-structure interface a slip boundary condition holds, which results from the impermeability of the interface.



(a) Conceptual fluid domain.

(b) Computational fluid domain.

Figure 2.5: Visualization of the panel problem, [29].

2.2.1 Governing equations

The structural dynamics of the flexible panel are described by the Euler Bernoulli beam equation

$$\rho_p t_p \frac{\partial^2 w}{\partial t^2} + \frac{\partial^2}{\partial x^2}\left[\frac{EI}{1-\nu^2}\frac{\partial^2 w}{\partial x^2}\right] = -\Delta p(x) \tag{2.10}$$

with moment of inertia $I = \frac{1}{12}t_p^3$ and forcing term $\Delta p = p(x) - p_\infty$. Using the finite element method with Hermite polynomials as basis functions [30] the beam equation is transformed to a linear system of equations

$$\mathbf{M}^{pp}\ddot{\mathbf{a}}^p + \mathbf{K}^{pp}\mathbf{a}^p = \mathbf{f}^p. \tag{2.11}$$

Note that the superscript p refers to the panel, whereas the superscript f will be used to indicate the fluid. In (2.11) $\mathbf{M}^{pp}$ and $\mathbf{K}^{pp}$ are the panel's mass and stiffness matrices, $\mathbf{f}^p$ is the external force acting on the panel and the vector $\mathbf{a}^p$ contains the discrete state of the panel. The fluid is governed by the two-dimensional unsteady linearized potential equation

$$(1 - M_\infty^2)\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} - \frac{1}{a_\infty}\left(\frac{\partial^2 \phi}{\partial t^2} + 2u_\infty \frac{\partial^2 \phi}{\partial x \partial t}\right) = 0 \tag{2.12}$$

where ϕ denotes the perturbed fluid potential and

$$u = u_\infty + \frac{\partial \phi}{\partial x} \qquad \text{and} \qquad v = \frac{\partial \phi}{\partial y}. \tag{2.13}$$

The potential equation (2.12) is discretized using the finite difference method on an equidistant grid (visualized in Fig. 2.2), resulting in the semi-discrete system

$$\mathbf{M}^{ff}\ddot{\mathbf{a}}^f + \mathbf{C}^{ff}\ddot{\mathbf{a}}^f + \mathbf{K}^{ff}\mathbf{a}^f = \mathbf{f}^f \tag{2.14}$$

where $\mathbf{a}^f$ is the discrete state of the fluid, containing the potential values at the grid points. The external fluid force depends on the state of the panel via the coupling matrices $\mathbf{C}^{pf}$ and $\mathbf{K}^{fp}$ and is given by

$$\mathbf{f}^f = -\mathbf{C}^{fp}\dot{\mathbf{a}}^p - \mathbf{K}^{fp}\dot{\mathbf{a}}^p. \tag{2.15}$$

The pressure difference along the panel is given by

$$\Delta p(x,t) = -\rho_\infty \left(\frac{\partial \phi}{\partial t} + u_\infty \frac{\partial \phi}{\partial x} \right) \tag{2.16}$$

such that the external force in (2.11) can be expressed as

$$\mathbf{f}^p = -\mathbf{C}^{pf}\dot{\mathbf{a}}^f - \mathbf{K}^{pf}\mathbf{a}^f. \tag{2.17}$$

The coupled fluid-structure problem is now described by (2.11), (2.14), (2.15) and (2.17).

The fluid and structure equations for the coupled problem are integrated using a Newmark-β time-integration scheme, which is second order accurate and unconditionally stable for $\gamma = 1/2$ and $\beta = 1/4$. See [29] for a pseudocode. For the 1D test problem, only the interface displacement was assumed to be unknown at each timestep in the time integration procedure. However, here the computation of the fluid force $\mathbf{f}^f$ requires both the panel displacement and velocity. Therefore every coupling iteration does also require the derivation of the interface velocity from the interface displacement, which is also done using the Newmark-β scheme.

2.2.2 Coarse model

In order to accelerate the evaluation of the fine 2D FSI problem described above, first order piston theory will be used as the coarse model. In this case the pressure difference over the panel is given by

$$\Delta p(x) = \rho_\infty a_\infty \frac{\partial w}{\partial t} + \rho_\infty M_\infty a_\infty^2 \frac{\partial w}{\partial x} \tag{2.18}$$

such that the forcing term in the beam equation can be expressed in terms of the discrete state of the panel

$$\mathbf{f}^p = -\mathbf{C}^{fop}\dot{\mathbf{a}}^p - \mathbf{K}^{fop}\mathbf{a}^p. \tag{2.19}$$

This leads to a compact dynamical system for the panel state,

$$\mathbf{M}^{coarse}\ddot{\mathbf{a}}^p + \mathbf{C}^{coarse}\dot{\mathbf{a}}^p + \mathbf{K}^{coarse}\mathbf{a}^p = \mathbf{0} \tag{2.20}$$

which is cheaper to solve than the fine model due to the smaller dimensions of the coupling matrices. In (2.20) $\mathbf{M}^{coarse} = \mathbf{M}^{pp}$, $\mathbf{C}^{coarse} = \mathbf{C}^{fop}$ and $\mathbf{K}^{coarse} = \mathbf{K}^{pp} + \mathbf{K}^{ff}$.

2.2.3 2D testcases

The numerical parameters and initial conditions for both the fine and coarse model are gathered in table 2.3. The initial panel displacement is expressed as a function of the flutter mode ξ and the timestep depends on the non-dimensional critical coupled period

$$P_{cr} = \frac{2\pi a_\infty}{\omega_{cr} L}. \tag{2.21}$$

		Fine model	Coarse model
Initial panel displacement	w^0	$0.1\xi/\|\xi\|$	$0.1\xi/\|\xi\|$
Fluid grid size	$N_x \times N_y$	321×193	-
Number of timesteps	N_t	$10 \cup 30 \cup 50$	$10 \cup 30 \cup 50$
Timestep	Δt	$P_{cr}/10 \cup P_{cr}/30 \cup P_{cr}/50$	$P_{cr}/10 \cup P_{cr}/30 \cup P_{cr}/50$

Table 2.3: Nondimensional initial conditions and numerical parameters for the 2D panel problem.

For this two-dimensional problem the fluid-to-structure mass ratio and the ratio of characteristic time-scales are defined as

$$\zeta = \frac{\rho_\infty L}{\rho_p t_p} \qquad \text{and} \qquad \lambda = \frac{1}{a_\infty L}\sqrt{\frac{EI}{\rho_p t_p}}. \tag{2.22}$$

Testcases 4 to 6 (table 2.4) are again chosen such that ζ varies while λ remains constant. For each testcase the panel is excited in the flutter mode at the critical Mach number, such that the panel would vibrate exactly at the critical frequency ω_{cr} in the absence of physical damping.

	Similarity parameters		Structural parameters				Fluid parameters	
	$\zeta[-]$	$\lambda[-]$	$\rho_s[\frac{kg}{m^3}]$	$E[\frac{kg}{ms^2}]$	$L[m]$	$t_p[m]$	$\rho_\infty[\frac{kg}{m^3}]$	$a_\infty[\frac{m}{s}]$
FSI-4	5.47 E-2	1.47 E-2	2.71 E3	8.67 E10	0.5	1.35 E-3	0.4	300
FSI-5	7.41 E-2	1.47 E-2	2.00 E3	6.40 E10	0.5	1.35 E-3	0.4	300
FSI-6	3.00 E-1	1.47 E-2	5.00 E2	1.60 E10	0.5	1.35 E-3	0.4	300

Table 2.4: Physical and similarity parameters of the test cases with fixed time-scales for the 2D panel problem.

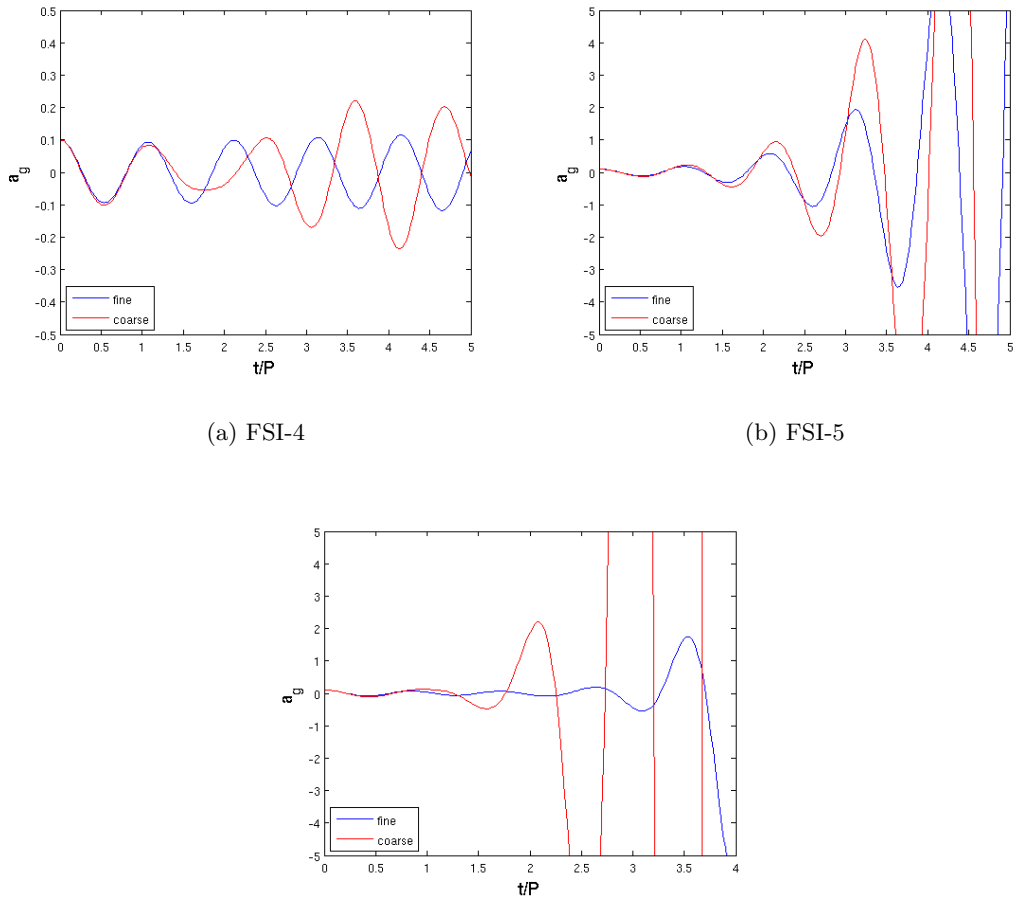
	M_{cr}	ω_{cr}	P_{cr}
FSI-4	2.27	460	8.2
FSI-5	2.28	536	7.0
FSI-6	2.33	1075	3.5

Table 2.5: Critical parameters for the two-dimensional testcases.

To visualize the behaviour of the panel in time an additional generalized coordinate

$$a_g(t) = \xi^T \mathbf{a}^p(t) \tag{2.23}$$

is introduced, which is the projection of the panel displacement on the flutter mode. This results in the graphs shown in Fig. 2.6, which shows a_g for both the fine and coarse model. It can be seen that the coarse model is less damped than the fine model, resulting in larger oscillations. For testcases 5 and 6 flutter occurs for both models, however, the panels amplitude increases faster for the coarse model. For testcase FSI-4 on the other hand the panel vibrates nicely at the critical frequency for the fine model. Furthermore it is observed that the coarse model gets out of phase for testcase FSI-4, whereas for the other testcases the largest deficits are observed in the amplitude. The cost of one coarse model evaluation is found to be 0.03% of the cost of one fine model evaluation.



(a) FSI-4

(b) FSI-5

(c) FSI-6

Figure 2.6: Evolution of the fine and coarse model generalized coordinate for different testcases with $\Delta t = P/50$.

Chapter 3

Space mapping theory

The space mapping technique is a collection of efficient optimization approaches which use a low fidelity (coarse) model to solve otherwise expensive optimization problems. The iterative optimization and updating of these coarse models, which are cheap to evaluate but not very accurate, replaces the direct optimization of the accurate but computationally expensive high fidelity (fine) model of interest. In this way the computational effort is replaced from the fine model to the much faster coarse model, such that only a few fine model evaluations are required to obtain satisfactory results, thereby decreasing the overall simulation time to a large extent.

The technique was first described by Bandler et al. in 1994 [4] after which it became widely applied in the field of electromagnetics and microwave circuit design in particular. The current work aims at investigating the application of the space mapping optimization technique as coupling algorithm for fluid-structure interaction problems. To this end a literature survey was performed [15] which resulted in the identification of some promising approaches, which are applied in the following chapters. First, this chapter starts with a general introduction to space mapping.

3.1 General principle of the space mapping technique

Space mapping aims at solving optimization problems like

$$\mathbf{x}^* = arg \min_{\mathbf{x}} U(\mathbf{R}(\mathbf{x})) \tag{3.1}$$

while minimizing the need for expensive evaluations of $\mathbf{R}(\mathbf{x})$, the fine model. In (3.1) $\mathbf{R}$ is the vector with responses of the model, $\mathbf{x}$ a vector containing the n parameters to be optimized and U is the objective function for the optimization. For FSI applications the input vector $\mathbf{x}$ contains the interface displacements, $\mathbf{R}$ is the residual as defined in (1.5) and the objective function U equals the norm of the response vector. Space mapping solves (3.1) by creating a

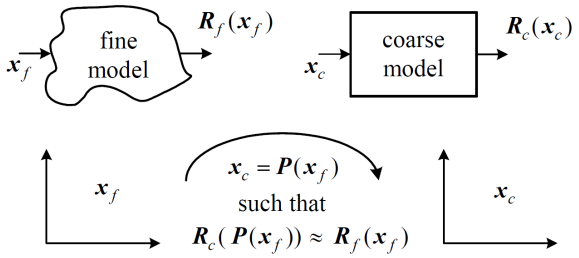


Figure 3.1: Illustration of the input space mapping procedure [13].

suitable surrogate model that combines the low computational cost of a certain coarse model (with response $\mathbf{R_c}$ and input variable $\mathbf{x_c}$) with the accuracy of the fine model (with response $\mathbf{R_f}$ and variable $\mathbf{x_f}$). Building this surrogate is done by enhancing the coarse model with a parameter transformation (mapping), denoted as P,

$$\mathbf{x_c} = P(\mathbf{x_f}) \qquad \text{such that} \qquad \mathbf{R_c}(P(\mathbf{x_f})) \approx \mathbf{R_f}(\mathbf{x_f}), \tag{3.2}$$

that establishes a link between the fine model and coarse model parameter spaces (X_f and X_c respectively), see Fig. 3.1. The term surrogate model is used to denote the combination of coarse model and mapping, such that at a certain subiteration j

$$\mathbf{R_s}^{(j)}(\mathbf{x_f}) = \mathbf{R_c}(P_j(\mathbf{x_f})). \tag{3.3}$$

The mapping P in (3.2) is determined using an iterative process and is chosen such that the response of the surrogate model resembles the output of the fine model as much as possible within some local modeling region around the optimal fine model solution $\mathbf{x_f}^*$. When the latter holds an estimate $\mathbf{x_f}^{(j)}$ for the fine model optimal solution can be obtained by simply applying the inverse mapping to the coarse model optimal solution, $\mathbf{x_c}^* = arg\min_{\mathbf{x_c}} \|\mathbf{R_c}(\mathbf{x_c})\|$, such that

$$\mathbf{x_f}^{(j+1)} = P_j^{-1}(\mathbf{x_c}^*). \tag{3.4}$$

When the mapping function P_j maps $\mathbf{x_f}^*$ exactly onto $\mathbf{x_c}^*$, so if $\mathbf{x_c}^* = P_j(\mathbf{x_f}^*)$, the mapping is said to be perfect. However, having only an input mapping as in (3.2) often does not suffice to achieve an exact match between the fine and coarse model responses. Therefore the surrogate can comprise, apart from the coarse model and an input mapping, also an output mapping $O(\mathbf{R_c})$. In the latter case the entire mapping is often denoted as Q which depends on the coarse model response, the fine model input vector and a set of preassigned parameters $\mathbf{x_p}$,

$$Q_j(\mathbf{R_c}, \mathbf{x_p}^{(j)}, \mathbf{x_f}) = \mathbf{R_s}^{(j)}(\mathbf{x_f}) \approx \mathbf{R_f}(\mathbf{x_f}). \tag{3.5}$$

The preassigned parameters contained in $\mathbf{x_p}^{(j)}$ are the parameters that define the mapping at iteration j. They can be artificial like the coefficients defining a linear mapping as well as physical model variables used to adapt the coarse model itself like density, mass and stiffness. By making use of an appropriate combination of input and output mappings the surrogate is always able to match the fine model response at least locally. The amount of parameters

in $\mathbf{x_p}$ determines the flexibility of the surrogate model. Surrogates with a large number of preassigned parameters are very flexible and thus provide very good local matching. However, very good approximation properties often lead to lower extrapolation properties and thus to poor matching elsewhere. The flexibility of the surrogate should therefore be chosen carefully. Once the nature of the mapping and the preassigned parameters are identified, selection of the adequate value for the mapping parameters in $\mathbf{x_p}$ is done during a parameter extraction process.

3.2 Parameter extraction approaches

Parameter extraction (PE) is one of the most important steps in the space mapping optimization process, where the parameters $\mathbf{x_p}$ that define the mapping or surrogate to match the fine model are extracted. Special care should be given to the uniqueness of the solution, since non-unique solutions may cause the algorithm to converge slowly, oscillate or fail to converge at all. Moreover, a balance has to be found between the approximation and extrapolation properties of the resulting surrogate. Therefore several PE approaches are available, an overview of which can be found in, among others, [3; 13].

3.2.1 Methods to perform the PE

Single point extraction (SPE) The most basic way of PE calibrates the mapping locally by matching the responses of both models at only one fine model point $\mathbf{x_f}^{(j)}$. Assuming the mapping is determined by the parameters $\mathbf{x_p}$, this approach looks like

$$\mathbf{x_p}^{(j)} = arg \min_{\mathbf{x_p}} \|\mathbf{R_f}(\mathbf{x_f}^{(j)}) - \mathbf{R_s}(\mathbf{x_p}, \mathbf{x_f}^{(j)})\|. \tag{3.6}$$

In general, SPE has excellent approximation properties since a perfect match at the considered fine model point can usually be obtained. However, the extrapolation quality of the obtained surrogate is expected to be rather low since only one point is taken into account to determine to mapping parameters. This formulation yields a unique solution if the number of elements in $\mathbf{x_p}$ is smaller or equal to the number of elements in the response vectors $\mathbf{R_f}$ and $\mathbf{R_s}$, so if $n_p \leq n_f$. If $n_p > n_f$ however, the system is underdetermined and regularization is necessary to obtain a unique solution. This can be done by using one of the following methods.

Multipoint extraction (MPE) MPE aims at improving the uniqueness of the PE solution by overdetermining the problem. This is done by simultaneously matching the responses at multiple fine model points, taking into account appropriate weighting factors, as shown in (3.7).

$$\mathbf{x_p}^{(j)} = arg \min_{\mathbf{x_p}} \|[\ \omega_0 \mathbf{e}_0^T \quad \omega_1 \mathbf{e}_1^T \quad \dots \quad \omega_N \mathbf{e}_N^T\ ]^T\| \tag{3.7}$$

with

$$\mathbf{e}_i = \mathbf{R_f}(\mathbf{x_f}^{(i)}) - \mathbf{R_s}(\mathbf{x_p}, \mathbf{x_f}^{(j)}) \qquad , \qquad i = 1, \dots, N. \tag{3.8}$$

The number of fine model points taken into account is given by N. The choice of the weighting coefficients ω_i determines which points are taken into account and to what extent. A possible choice can be

$$\omega_i = \alpha^{N-i} \tag{3.9}$$

where α is a constant smaller than one [19]. Apart from enhancing the uniqueness of the solution, this MPE formulation does also increase the region of applicability of the mapping (extrapolation quality). The latter is especially beneficial in the early stages of the optimization when the solution estimate may not be close to the optimal solution yet. However, in order to apply MPE several fine model evaluations should have been performed, which is generally not the case at the start of the algorithm. Since it is undesirable to determine a set of base points at the start of each new timestep, one can start with using SPE for the first iteration and increasing the number of points taken into account every subsequent iteration. When applying MPE the amount of coarse model evaluations increases by a factor equal to the number of fine model points taken into account. Therefore it might be desirable to limit the number of points used for MPE. Furthermore, since the mapping parameters are based on multiple points it is not possible to extract a perfect mapping. Consequently, the unique extraction process and improved extrapolation quality comes with the cost of decreased approximation properties.

Gradient parameter extraction (GPE) GPE [9] enhances the uniqueness of the parameter extraction process by taking into account derivative information. For this approach only one fine model point is required, so it can be applied from the first iteration on. This also implies that GPE results in a local mapping. However, due to the derivative information the region of applicability is still larger than for SPE. Moreover, it can be extended by combining GPE with MPE. The gradient parameter extraction procedure looks like

$$\mathbf{x_p}^{(j)} = arg\min_{\mathbf{x_p}} \| [\ \mathbf{e}_0^T \quad \omega\mathbf{e}_1^T \quad \dots \quad \omega\mathbf{e}_n^T\ ]^T \| \tag{3.10}$$

with

$$\mathbf{e}_0 = \mathbf{R_f}(\mathbf{x_f}^{(j)}) - \mathbf{R_s}(\mathbf{x_p}, \mathbf{x_f}^{(j)}) \qquad \text{and} \qquad \mathbf{E} = \mathbf{J_f}(\mathbf{x_f}^{(j)}) - \mathbf{J_s}(\mathbf{x_p}, \mathbf{x_f}^{(j)}), \tag{3.11}$$

where ω is a weighting factor, $\mathbf{E} = [\ \mathbf{e}_1 \quad \mathbf{e}_2 \quad \dots \quad \mathbf{e}_n\ ]$, n the number of fine model design variables, $\mathbf{J_f}$ the fine model Jacobian and $\mathbf{J_s}$ the surrogate model Jacobian. For FSI simulations usually no Jacobian information is available such that both $\mathbf{J_f}$ and $\mathbf{J_s}$ have to be approximated. This evidently decreases the accuracy of the GPE approach considerably. Moreover, approximating the fine and surrogate model derivatives requires at least two fine model responses, such that in this case GPE can only be used from the second iteration on.

Constrained parameter extraction An easy and computationally inexpensive technique which can be used to assure unique PE is the addition of constraints for the mapping parameters,

$$\mathbf{x_p}^{(j)} = arg \min_{\mathbf{x_p}} \left(\|\mathbf{R_f}(\mathbf{x_f}^{(j)}) - \mathbf{R_c}(\mathbf{x_p})\| + \omega \|\mathbf{x_p} - \mathbf{x_{p_0}}\| \right) \tag{3.12}$$

where $\mathbf{x_{p_0}}$ is the objective. For example, one can enforce the mapping to be as close as possible to the unit mapping or the previous mapping. The latter approach is expected to be most suitable for use when time increases and therefore the deviation between fine and coarse model increases. Apart from SPE, as shown here, constraints can be added to all previously mentioned parameter extraction techniques. However, care should be taken when selecting the weighting factor. Too large values for ω cause poor agreement between fine and coarse model, whereas a too small ω may be ineffective in alleviating PE non-uniqueness. According to Bandler et al. [6] values between 0.05 and 0.2 are most effective.

Statistical parameter extraction Another way to perform parameter extraction is statistical PE [7]. This method is designed to be used in cases where non-uniqueness is a problem. It extracts the optimal solution by utilizing several starting points to initiate the PE process. Those starting points are statistically generated within a certain exploration region,

$$\mathbf{x_p} \in [\mathbf{x_{p_0}} - 2|\delta^{(j)}|, \mathbf{x_{p_0}} + 2|\delta^{(j)}|] \qquad , \qquad \delta^{(j)} = \mathbf{x_p}^{(j-1)} - \mathbf{x_{p_0}}. \tag{3.13}$$

For all starting points SPE is performed. If the same values for the extracted parameters $\mathbf{x_p}^{(j)}$ are found the solution is unique. If not, the resulting solutions are ranked according to the value achieved for the objective function and the best solution is taken as the final result. A large drawback of this PE approach consists of the computational cost, which increases by a factor equal to the number of starting points used. This makes the cost of statistical PE comparable to the cost required for MPE, while the resulting surrogate has the quality of SPE (good approximation but bad extrapolation properties).

3.2.2 Implementation

Since the function to be minimized in the previous expressions is generally nonlinear, usually a nonlinear least squares method is used to extract the desired parameters. To this end the build in Matlab® function *lsqnonlin* is used, which is designed to provide a numerical solution to the problem of minimizing a function over a space of function parameters. This function employs a trust-region-reflective algorithm but switches to the Levenberg-Marquardt algorithm when the system of equations is underdetermined. The tolerance on the parameter extraction process is kept at the standard value of $1 \cdot 10^{-6}$. This is done for two reasons. First of all this slightly lowers the approximation quality in favour of the extrapolation properties, which is believed to be beneficial for the convergence of the space mapping algorithm. Secondly, this way the coarse computational cost is kept low. This is especially important for 2D problems where the number of extractable parameters and thus the required coarse model evaluations will be high. Apart from *lsqnonlin* also other minimization methods were tested, like *fconmin* and *fminimax*, but it was found that *lsqnonlin* yields the best results.

3.3 Updating the solution estimate

The goal of space mapping is to find the solution of the fine model using as few as possible fine model evaluations. So when a good mapping is found the second step consists of using it to establish a good estimate for $\mathbf{x_f}^*$. In [14] two approaches are identified to do this. Primal space mapping on the one hand defines the next estimate $\mathbf{x_f}^{(j+1)}$ by minimizing the residual vector $\mathbf{f}_j$,

$$\mathbf{f}_j(\mathbf{x_f}) = P_j(\mathbf{x_f}) - \mathbf{x_c}^*, \tag{3.14}$$

so by performing an optimization

$$\mathbf{x_f}^* \approx \mathbf{x_f}^{(j+1)} = arg\min_{\mathbf{x_f}} \|P_j(\mathbf{x_f}) - \mathbf{x_c}^*\| \tag{3.15}$$

in the coarse input space X_c. The solution of (3.15) consists of applying the inverse mapping to $\mathbf{x_c}^*$, as already shown earlier in (3.4). This means that when the inverse of the mapping is available (or a reasonably accurate approximation of it) and $\mathbf{x_c}^* \in P_j(X_f)$ no optimization is required to determine the next estimate $\mathbf{x_f}^{(j+1)}$. It is clear however that this primal approach can only be utilized for cases where only an input mapping is applied. If also an output mapping is applied or physical parameters are adapted to align both models, $\mathbf{x_c}^*$ is no longer constant but changes every iteration such that (3.15) is not appropriate for use anymore. An alternative approach therefore consists of defining the next estimate as the optimal solution of the surrogate model,

$$\mathbf{x_f}^{(j+1)} = arg\min_{\mathbf{x}} \|\mathbf{R_s}^{(j)}(\mathbf{x})\|, \tag{3.16}$$

since $\mathbf{R_s}$ is constructed such that its response resembles the fine model response. This dual space mapping approach is very general and can be used in all cases. A drawback is that it requires two optimization steps (parameter extraction and surrogate optimization) during every iteration whereas the primal approach only needs one in most cases. So especially for the dual space mapping method it is very important that the computational cost of the coarse model is low.

From the foregoing it is clear that the space mapping technique comprises three major steps, which are repeated until the optimization requirements are satisfied. Those main steps are:

1. Fine model evaluation (verification)
2. Updating the surrogate / mapping by means of parameter extraction (calibration)
3. Determination of the next estimate

As the above steps are rather vague, it is obvious that there exist several approaches that can be used to accomplish them. Step 1, the verification, is straight forward. For step 3 one distinguishes between the primal and dual approach as explained before. The largest variation in space mapping methods follow from the calibration step, which is also the crucial part. By varying the kind of mapping and the way in which the corresponding mapping parameters are extracted a large amount of different algorithms can be identified. Different space mapping variations are discussed in the following chapters.

3.4 Convergence improvements

Although the space mapping technique has shown to be a reliable and efficient method, convergence is not guaranteed in general [20]. The algorithm may exhibit fast initial progress but can stagnate as the number of iterations increases due to a lack of convergence and oscillations in the matching error. In this section, two measures which may improve the convergence properties of the algorithm are presented, the first one already being proposed in literature.

3.4.1 Adaptively constrained parameter extraction

The flexibility of the surrogate model is an important parameter which determines the convergence properties of the space mapping method to a large extend. Often the suitable amount of flexibility is selected by trial and error, which obviously is both inefficient and not user friendly. Therefore Koziel et al. [22] introduced an adaptively constrained parameter extraction process that adapts the parameter space of the surrogate model, and thus its flexibility. The method is initiated by selecting an over-flexible surrogate model which, without any constraints on the bounds for parameter extraction, is able to approximate the fine model with sufficient accuracy. As a rule of thumb it is stated that the number of parameters in $\mathbf{x_p}$ should be larger than the number of design variables in $\mathbf{x_f}$. During the first iteration, parameter extraction is performed with loose lower and upper bounds that allow for a large flexibility of $\mathbf{x_p}$, such that $\mathbf{l}^{(j)} \leq \mathbf{x_p} \leq \mathbf{u}^{(j)}$. The bounds are defined as

$$\mathbf{l}^{(j)} = {\mathbf{x_p}}^{(j-1)} - \delta^{(j)} \tag{3.17}$$

$$\mathbf{u}^{(j)} = {\mathbf{x_p}}^{(j-1)} + \delta^{(j)}, \tag{3.18}$$

were $\delta^{(1)}$ is a user-defined vector representing the initial parameter space size. From the second iteration on the parameter extraction bounds are tightened if the approximation error ϵ is sufficiently small and the bounds are loosened otherwise. This is done according to

$$\epsilon^{(j)} = \|\mathbf{R_f}({\mathbf{x_f}}^{(j)}) - {\mathbf{R_s}}^{(j)}({\mathbf{x_f}}^{(j)})\|, \tag{3.19}$$

- $\epsilon^{(j)} \leq \alpha_{decr} \cdot \epsilon_{max}$ then $\delta^{(j+1)} = \delta^{(j)}/\beta_{decr}$ (3.20)
- $\epsilon^{(j)} > \alpha_{incr} \cdot \epsilon_{max}$ then $\delta^{(j+1)} = \delta^{(j)} \cdot \beta_{incr}$. (3.21)

In the above ϵ_{max} represents the maximum allowed deficit between the fine and surrogate model (which is user defined) and α_{decr}, α_{incr}, β_{decr} and β_{incr} define the adaptation of the bounds. The values used by Koziel et al. [22] are: $\alpha_{decr} = 1$, $\alpha_{incr} = 2$, $\beta_{decr} = 5$ and $\beta_{incr} = 2$. Under normal circumstances δ is reduced from iteration to iteration, such that closeness of ${\mathbf{x_p}}^{(j)}$ and ${\mathbf{x_p}}^{(j+1)}$ is enforced. Although this method does not guarantee convergence, it has been shown that it is able to substantially improve the space mapping performance.

3.4.2 Taylor expansion of preassigned parameters

Here a new convergence improvement is proposed. When applying methods like GSM and ISM (Chapters 5 and 7) preassigned parameters are extracted that ensure a match between the fine and coarse model at the currenct evaluation point. Those extracted parameters define the new surrogate model which is subsequently optimized in order to define the next fine model iterate $\mathbf{x_f}^{(j+1)}$. However, the surrogate model is defined such that its response mainly resembles the response of the fine model for $\mathbf{x_f}^{(j)}$. Therefore it is not ensured to be accurate at the new estimate. By defining the set of preassigned parameters such that they depend on the quantity that is to be optimized, in this case $\mathbf{x_f}$, it is expected that this inaccuracy is minimized. This is accomplished by using a first order Taylor series expansion around $\mathbf{x_p}^{(j)}$ instead of $\mathbf{x_p}^{(j)}$ itself for the determination of the next iterate. So

$$\mathbf{x_p}^{(j+1)} \approx \tilde{\mathbf{x}}_\mathbf{p}^{(j)} = \mathbf{x_p}^{(j)} + \nabla_{\mathbf{x_f}}\mathbf{x_p}^{(j)} \cdot \Delta\mathbf{x_f} \tag{3.22}$$

where $\Delta\mathbf{x_f} = \mathbf{x_f} - \mathbf{x_f}^{(j)}$, such that the next iterate is defined as

$$\mathbf{x_f}^{(j+1)} = arg\min_{\mathbf{x_f}} \|\mathbf{R_s}(\mathbf{x_f}, \tilde{\mathbf{x}}_\mathbf{p}^{(j)}(\mathbf{x_f}))\|. \tag{3.23}$$

The gradient of the preassigned parameters with respect to the fine model design variable $\mathbf{x_f}$ is updated every iteration by means of Broyden's method,

$$\nabla_{\mathbf{x_f}}\mathbf{x_p}^{(j)} = \nabla_{\mathbf{x_f}}\mathbf{x_p}^{(j-1)} + \frac{\mathbf{x_p}^{(j)} - \mathbf{x_p}^{(j-1)} - \nabla_{\mathbf{x_f}}\mathbf{x_p}^{(j-1)} \cdot \Delta\mathbf{x_f}}{\Delta\mathbf{x_f}^T\Delta\mathbf{x_f}}\Delta\mathbf{x_f}^T. \tag{3.24}$$

In (3.24) $\Delta\mathbf{x_f} = \mathbf{x_f}^{(j)} - \mathbf{x_f}^{(j-1)}$ which implies that this Taylor expansion improvement can only be applied from the second iteration on. For the 1D test problem, where $\mathbf{x_f}$ is a scalar, the gradient becomes a vector with length equal to the number of preassigned parameters m. For the 2D test problem on the other hand $\mathbf{x_f}$ is a vector of length n, such that the gradient in (3.22) becomes a matrix with size $m \times n$. However, in practice the method is implemented for each preassigned parameter separate. In the case of GSM (see Chapter 5) one gets

$$\mathbf{x_f}^{(j+1)} = arg\min_{\mathbf{x_f}} \left\| \mathbf{R_s}\left(\mathbf{x_f}, \tilde{\mathbf{A}}^{(j)}(\mathbf{x_f}), \tilde{\mathbf{B}}^{(j)}(\mathbf{x_f}), \tilde{\mathbf{c}}^{(j)}(\mathbf{x_f})\right)\right\|. \tag{3.25}$$

Since this method predicts the next values of the preassigned parameters by taking into account an approximation to their derivatives, it is expected that its influence is largest for algorithms that do not take into account any derivative information yet, like algorithms applying SPE or MPE. For those cases where surrogate derivative information is already utilized, e.g. when using GPE, the gains are expected to be much smaller.

3.5 Space mapping as coupling technique for partitioned fluid-structure interaction simulations

In section 1.2 it was explained that for strongly coupled problems that are solved in a partitioned fashion subiterating is necessary to obtain accurate solutions. However, this increases the computational cost drastically, thereby limiting the direct use of such algorithms in industry. Although several acceleration techniques can be found in literature, none of them fully satisfies the requirements for a general and efficient FSI simulation technology as identified in the FFAST project [16; 27]. The space mapping technique, however, has shown to be very efficient in expensive electromagnetic optimization while being minimal intrusive, robust, generic and allowing for software modularity. Therefore in the remainder of this work the application of the space mapping technique to FSI problems is investigated.

3.5.1 FSI problem formulation for space mapping

When solving a fluid-structure interaction problem, it is required that the interface displacements in both the fluid and solid domain match. So

$$\mathbf{R}(\mathbf{x}) = S \circ F(\mathbf{x}) - \mathbf{x} = 0 \tag{1.5}$$

should be satisfied for every timestep (where S and F represent respectively the black box solid and fluid solvers and $\mathbf{x}$ is the interface displacement). This requirement can be reformulated as an optimization problem which aims at finding the optimal interface displacement $\mathbf{x}^*$,

$$\mathbf{x}^* = arg \min_{\mathbf{x}} \|\mathbf{R}(\mathbf{x})\|. \tag{3.26}$$

Which corresponds to equation (3.1) with the objective function U equal to the norm of the response vector. In order to be able to apply space mapping to solve this optimization problem, an appropriate coarse model is required which is based on the same physics but is much cheaper to evaluate. This is done by using separate approximate black box fluid and structure solvers and keeping the interface residual formulation unchanged. In general the fine and coarse models look like

$$\mathbf{R_f}(\mathbf{x_f}) = S_f \circ F_f(\mathbf{x_f}) - \mathbf{x_f} \qquad \text{and} \qquad \mathbf{R_c}(\mathbf{x_c}) = S_c \circ F_c(\mathbf{x_c}) - \mathbf{x_c} \tag{3.27}$$

where $\mathbf{x_f}$ and $\mathbf{x_c}$ denote the vectors of fine and coarse design variables, which are the interface displacements in this case. Note that the vectors $\mathbf{x_f}$ and $\mathbf{x_c}$ can have different dimensions. Due to the definition of the residual, this means that in the latter case the response vectors $\mathbf{R_f}$ and $\mathbf{R_c}$ will have different dimensions too. Therefore some kind of interpolation method may have to be added to the coarse model to allow for the transfer of information between both models. The coarse models which will be used in this work are presented in Sections 2.1.2 and 2.2.2. By Koziel et al. [21] it is advised that for space mapping to be successful the cost of the coarse model should be less than 1%, and preferable even less than 0.1%, of the cost of the fine model. This is satisfied for both the 1D and 2D test problems.

When applied to fluid-structure interaction simulations, space mapping replaces the conventional subiterating process to ensure temporal coupling. The implementation of space mapping within this dynamical process is illustrated in Fig. 3.2, where the background colour intensity represents the computational cost of that step.

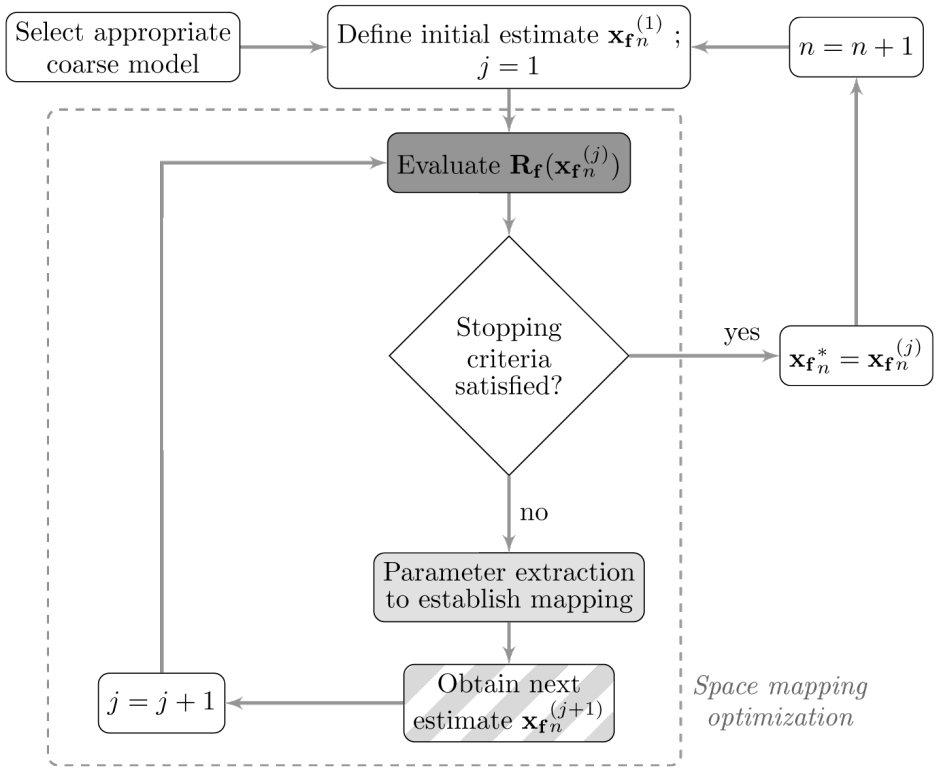


Figure 3.2: Illustration of the implementation of the space mapping method in the temporal coupling process.

Compared to the conventional statical applications of space mapping, the fact that FSI simulations are dynamical has some consequences for the application of the method. First of all the required amount of fine model evaluations is expected to be rather low. The reason for this is that due to the time stepping an initial estimate to start the optimization which is already within the neighbourhood of the final solution is likely to be found, especially when the applied timestep is rather small. As a consequence, approaches that require several fine model evaluations in order to build the initial surrogate are not likely to be suitable for use in this case. Secondly, the time stepping will cause the initial misalignment between the fine and coarse model to grow in time. The expected consequence of this is that when the misalignment becomes large, an input mapping alone might not be able to match both models and an output correction (and therefore dual space mapping) is advised. So in general the required flexibility will increase with time. Algorithms which might be suitable for use are presented in Chapters 4 to 7, but first several possibilities to define the initial estimate are identified.

3.5.2 Approaches to define the initial estimate

The determination of the initial estimate $\mathbf{x_f}_n^{(1)}$ is an important step in the FSI simulation since an initial estimate which is far off the final solution will increase the number of required iterations. When space mapping is applied the number of possibilities to determine $\mathbf{x_f}_n^{(1)}$ increases due to the presence of the surrogate and coarse models. Several approaches are presented below.

Approach 1 The initial estimate equals the converged result of the previous timestep:

$$\mathbf{x_f}_n^{(1)} = \mathbf{x_f}_{n-1}^* \tag{3.28}$$

This is the easiest way to initialize the next timestep and is often used in conventional FSI simulations, which yields good results as long as the timesteps are small.

Approach 2 Instead of using only the solution of the previous timestep, one can also base $\mathbf{x_f}_n^{(1)}$ on the history of fine model solutions. This approach is often referred to as state extrapolation. So

$$\mathbf{x_f}_n^{(1)} = f(\mathbf{x_f}_{n-1}^*, \mathbf{x_f}_{n-2}^*, \ldots) = \mathbf{x_f}_{n-1}^* + \delta_n \tag{3.29}$$

with the penalty term δ defined by

$$\delta_n \approx \mathbf{x_f}_n^* - \mathbf{x_f}_{n-1}^* \tag{3.30}$$

$$= \delta_{n-1} + \nabla_t \delta \cdot \Delta t. \tag{3.31}$$

A correction term is added to the previous solution, which consists of a prediction of the deficit between the solution at the current timestep and the solution at the previous timestep. The gradient of this correction δ is calculated by using upwind finite differences which looks like (3.32) or (3.33).

$$1^{st} \text{ order:} \qquad \nabla_t \delta = \frac{\delta_{n-1} - \delta_{n-2}}{\Delta t} \tag{3.32}$$

$$2^{nd} \text{ order:} \qquad \nabla_t \delta = \frac{3\delta_{n-1} - 4\delta_{n-2} + \delta_{n-3}}{2\Delta t} \tag{3.33}$$

Approach 3 Taking advantage of the available coarse model, one can define the first estimate for the fine model solution of the next timestep to be the optimal coarse model solution.

$$\mathbf{x_f}_n^{(1)} = \mathbf{x_c}_n^* \tag{3.34}$$

Although accurate for the first few timesteps, this approach is likely to be inefficient when time evolves and the misalignment between the fine and coarse models increases, resulting in a bad initial estimate. Although space mapping is supposed to be able to deal with this, strong misalignment will nevertheless decrease the efficiency of the algorithm since more iterations will be required to satisfy the termination condition.

Approach 4 The misalignment problem can be minimized by utilizing the surrogate model solution as initial estimate,

$$\mathbf{x_f}_n^{(1)} = arg\min_{\mathbf{x}} \left\| \mathbf{R}_{\mathbf{s}_{n-1}^*}(\mathbf{x}) \right\|, \tag{3.35}$$

so combining the coarse model with the optimal mapping one timestep prior. By using the optimal surrogate model found during the previous timestep, which at that particular time instant yielded a match with the fine model, the misalignment is likely to be minimized throughout time. A drawback of this method is that it requires an additional coarse optimization.

Approach 5 Similar to approach 1, approach 4 can be improved by taking into account a correction term which aims at predicting the deficit between the estimate and the optimal solution. So

$$\mathbf{x_f}_n^{(1)} = arg\min_{\mathbf{x}} \left\| \mathbf{R}_{\mathbf{s}_{n-1}^*}(\mathbf{x}) \right\| + \delta_n \tag{3.36}$$

where

$$\delta_k \approx \mathbf{x_f}_k^* - arg\min_{\mathbf{x}} \left\| \mathbf{R}_{\mathbf{s}_{n-1}^*}(\mathbf{x}) \right\| \tag{3.37}$$

$$= \delta_{k-1} + \nabla_t \delta \cdot \Delta t. \tag{3.38}$$

Since it is known that $\delta_{n-1} = 0$ the gradient of the correction term is now given by (3.39) or (3.40).

$$1^{st} \text{ order:} \qquad \nabla_t \delta = -\frac{\delta_{n-2}}{\Delta t} \tag{3.39}$$

$$2^{nd} \text{ order:} \qquad \nabla_t \delta = \frac{-4\delta_{n-3} + \delta_{n-4}}{2\Delta t} \tag{3.40}$$

Note that when using this approach, the correction term is determined by looking at how well the surrogate obtained at the previous solution represents the fine model at nearby moments in time whereas for approach 2 the deficit is followed in time. This implies that multiple additional surrogate optimizations are required, increasing the coarse computational cost.

Approach 6 Something similar can be done for approach 3. The correction term added to (3.34) now accounts for the difference between the fine and coarse model optimal solution. So δ_n is given by

$$\delta_n \approx \mathbf{x_f}_n^* - \mathbf{x_c}_n^* \tag{3.41}$$

$$= \delta_{n-1} + \nabla_t \delta \cdot \Delta t \tag{3.42}$$

with $\nabla_t \delta$ as in approach 2, given by (3.32) or (3.33).

Chapter 4

Aggressive space mapping (ASM)

4.1 The aggressive space mapping algorithm

The aggressive space mapping approach was proposed by Bandler et al. [5] in 1995 and incorporates a quasi-Newton iteration in conjunction with first-order derivative approximations. High convergence rates are obtained and no initial set of fine model results is required, thereby making it a very efficient and popular method. ASM applies only an input mapping and therefore utilizes a primal approach to update the fine model solution estimate.

Every ASM iteration starts with determining the value of the mapped fine model point $P_j(\mathbf{x_f}^{(j)})$. This is done by matching the particular fine and coarse model responses in the single point parameter extraction process

$$P_j(\mathbf{x_f}^{(j)}) = \mathbf{x_c}^{(j)} = arg \min_{\mathbf{x_c}} \|\mathbf{R_f}(\mathbf{x_f}^{(j)}) - \mathbf{R_c}(\mathbf{x_c})\|, \tag{4.1}$$

which yields a unique solution as long as $n_c \leq n_f$. It can safely be assumed that this is always the case, since coarse models with $n_c > n_f$ would be computationally too expensive. This mapping is perfect at the point $\mathbf{x_f}^{(j)}$, where the responses are matched, but unknown at other points. Therefore a general mapping is approximated by means of a first order Taylor series

$$P_j(\mathbf{x_f}) \approx P_j(\mathbf{x_f}^{(j)}) + \nabla_{\mathbf{x_f}} P_j \cdot (\mathbf{x_f} - \mathbf{x_f}^{(j)}). \tag{4.2}$$

The approximation of the inverse mapping, necessary to apply the primal space mapping update, thus becomes

$$P_j^{-1}(\mathbf{x_c}) \approx \mathbf{x_f}^{(j)} + (\nabla_{\mathbf{x_f}} P_j)^{-1} \cdot (\mathbf{x_c} - \mathbf{x_c}^{(j)}). \tag{4.3}$$

The next estimate for the fine model solution is now obtained by applying this inverse mapping to the coarse model optimal solution $\mathbf{x_c}^* = arg \min_{\mathbf{x_c}} \|\mathbf{R_c}(\mathbf{x_c})\|$. This can be written as a quasi-Newton iteration like

$$\mathbf{x_f}^{(j+1)} = \mathbf{x_f}^{(j)} + \mathbf{J_p}^{(j)^{-1}} \mathbf{f}^{(j)} \tag{4.4}$$

where the (approximated) mapping Jacobian $\mathbf{J_p}$ and the mapping residual $\mathbf{f}$ are given by

$$\mathbf{J_p}^{(j)} \approx \nabla_{\mathbf{x_f}} P_j \qquad \text{and} \qquad \mathbf{f}^{(j)} = \mathbf{x_c}^* - P_j(\mathbf{x_f}^{(j)}). \tag{4.5}$$

The key to a successful implementation of this algorithm is the determination of the latter Jacobian matrix. This is usually done by means of the Broyden rank one scheme,

$$\mathbf{J_p}^{(j)} = \mathbf{J_p}^{(j-1)} + \frac{P_j(\mathbf{x_f}^{(j)}) - P_{j-1}(\mathbf{x_f}^{(j-1)}) - \mathbf{J_p}^{(j-1)}\Delta\mathbf{x_f}}{\Delta\mathbf{x_f}^T\Delta\mathbf{x_f}}\Delta\mathbf{x_f} \tag{4.6}$$

which ensures that the accuracy of the approximation increases with every iteration. In (4.6) $\Delta\mathbf{x_f} = \mathbf{x_f}^{(j)} - \mathbf{x_f}^{(j-1)}$. In the rare cases where both the Jacobians of the fine and coarse models are available, one can also use a Jacobian based update of $\mathbf{J_p}$,

$$\mathbf{J_p} = (\mathbf{J_c}^T\mathbf{J_c})^{-1}\mathbf{J_c}^T\mathbf{J_f}. \tag{4.7}$$

Note that the parameter extraction approach used in (4.1) is SPE, which can be evaluated using fixed-point iterations in combination with Aitken underrelaxation to enhance the stability. Multipoint extraction cannot be used to improve the region of applicability of the mapping because in that case the mapping would no longer be locally perfect, such that expressions (4.2) and (4.3) loose their validity. GPE on the other hand is possible in theory, however, no exact Jacobians are available for FSI simulations such that the expected gains are low. Moreover, for 1D problems the Broyden update reduces to an ordinary first order finite difference approximation. Hence for methods that obtain locally perfect mappings (as ASM) the distinction between SPE and GPE vanishes when approximate derivatives are used.

The foregoing space mapping optimization method is repeated until $\mathbf{R_f}(\mathbf{x_f}^{(j)}) \approx 0$, as visualized in figure 3.2. If the latter is satisfied the current timestep is resolved and one has to switch to the next. For this specific case approach 4 to determine the next initial estimate becomes

$$\mathbf{x_f}_n^{(1)} = arg\min_{\mathbf{x_f}} \|\mathbf{R_c}(P_{*_{n-1}}(\mathbf{x_f}))\|. \tag{4.8}$$

Because ASM applies only an input mapping, expression (4.8) can be evaluated without the need to optimize the surrogate model. The initial estimate (when applying approach 4) is determined as

$$\mathbf{x_f}_n^{(1)} = P_{*_{n-1}}^{-1}(\mathbf{x_c}_n^*) = \mathbf{x_f}_{n-1}^* + \mathbf{J_p}_{n-1}^{*^{-1}}\left(\mathbf{x_c}_n^* - P_{*n-1}(\mathbf{x_f}_{n-1}^*)\right) \tag{4.9}$$

which does not increase the coarse computational cost since ASM requires the coarse model to be optimized anyway. For approach 6 a correction term is added which can also be calculated without a coarse optimization.

4.2 1D piston problem results

The ASM algorithm described above has been successfully applied to a typical one dimensional fluid-structure interaction problem, being the piston problem as described in section 2.1, by T.P. Scholcz [29]. The numerical performance of ASM for this simple single interface point problem is investigated by using three testcases (FSI-1, FSI-2 and FSI-3), at three different timestep sizes ($\Delta t = P/10$, $\Delta t = P/20$ and $\Delta t = P/30$). In this way the influence of the fluid-to-structure mass ratio ζ and the size of the applied timestep (thus the amount of numerical damping) is analyzed. The total relative cost (with respect to the application of the Quasi-Newton method) and the average final residual values for testcase FSI-2 (for which the nondimensional period $P = 5.96$) are gathered in Table 4.1. The results for the other testcases appear to be similar and can be found in Appendix B.

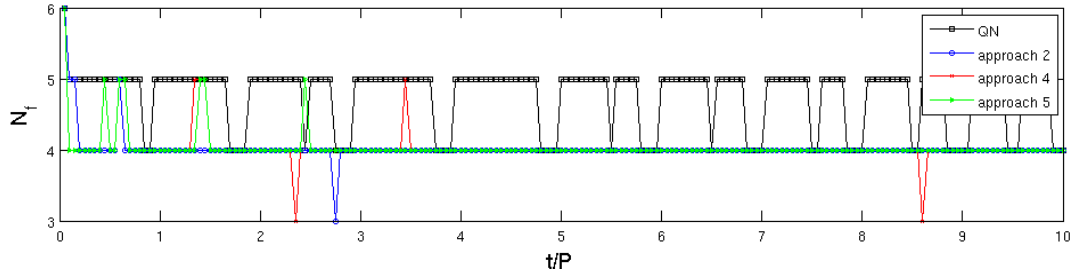
The table below contains six different variations of the ASM method which differ in the approach that was used to define the initial fine model point at the start of every timestep. Those approaches can be found in Section 3.5.2 and are based on previous solutions of the fine model (1 and 2), the coarse model (3 and 6) or the mapped coarse model (the surrogate, 4 and 5). It is expected that especially for small timesteps the approaches based on previous fine model solutions will show good behaviour, since the new solution will be in the neighbourhood of the previous one. For large timesteps however the accuracy of these methods decreases and an initial estimate based on the (mapped) coarse model may prove to be more efficient.

	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	6.5 E-16	100	2.2 E-16	100	2.3 E-15
1: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1}$	98.1	8.5 E-16	103.3	3.3 E-16	111.3	9.5 E-16
2: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1} + \delta_n$	89.0	2.4 E-15	85.5	2.8 E-15	94.8	2.2 E-16
3: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n$	96.3	4.4 E-16	100.7	4.0 E-16	107.8	1.5 E-15
4: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1}$	94.8	3.2 E-16	95.3	1.2 E-15	100.5	7.5 E-15
5: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1} + \delta_n$	84.4	1.1 E-14	85.9	2.1 E-15	96.5	4.1 E-16
6: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n + \delta_n$	83.8	1.1 E-14	85.9	1.6 E-15	95.8	4.7 E-16

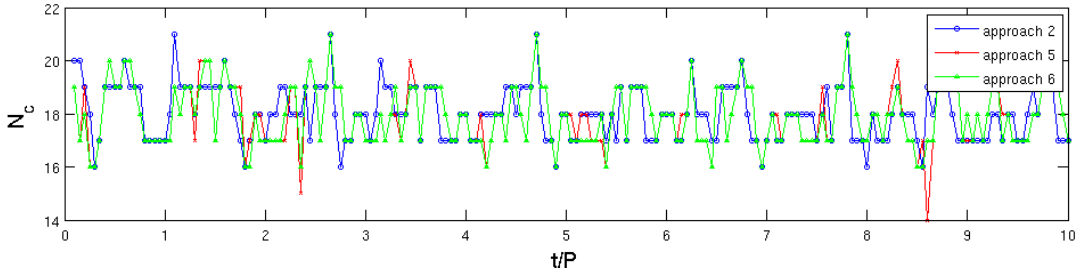
Table 4.1: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, ASM testcase FSI-2.

From the results in Table 4.1 it can be seen that for most cases the computational cost of ASM is well below the cost required when performing the same simulations with a Quasi-Newton method. Note that for the Quasi-Newton method applied in this work state extrapolation (approach 2) is used to define every first fine iterate. Especially for large timesteps ASM proves to be more efficient than QN, leading to savings in computational cost up to 17%. Apart from the timestep size, the relative efficiency of ASM with respect to QN also increases with increasing fluid-to-structure ratio. The higher efficiency of ASM is visualized in Fig. 4.1, which shows the number of fine and coarse model evaluations per timestep and the relative efficiency η as function of time for testcase FSI-2 with $\Delta t = P/20$. It is seen that ASM decreases the amount of required fine model iterations with one almost every timestep. Moreover, the amount of coarse model evaluations is very low, indicating that the parameter

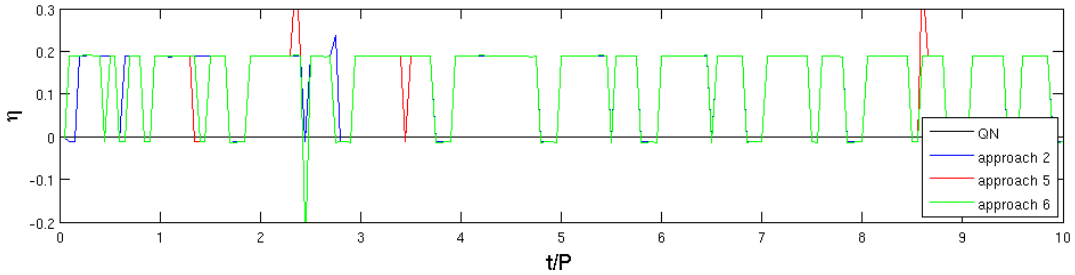
extraction procedure is very efficient. The latter also implies that, because the cost of one coarse model evaluation is approximately 0.3% of one fine evaluation, by far the largest part of the computational effort is due to the fine model. A saving of one fine model iteration per timestep therefore indicates a large increment in efficiency for the overall coupling method.



(a) Fine model evaluations



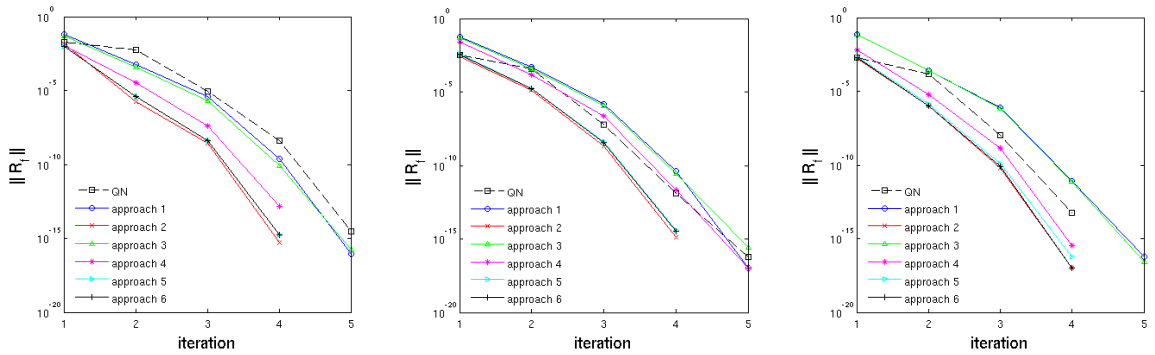
(b) Coarse model evaluations



(c) Relative efficiency w.r.t. Quasi-Newton

Figure 4.1: Comparison of different approaches to define the initial estimate for the 1D piston problem (testcase FSI-2), applying ASM with $\Delta t = P/20$.

Looking at the convergence within one timestep for the different methods in Fig. 4.2, it is seen that the overall convergence rate is higher for ASM than for QN. The largest difference in convergence rate occurs at the first iterate. This first drop in $\|\mathbf{R_f}\|$ is much larger for ASM than for QN. Afterward the convergence rates of both methods are more alike. From this it can be concluded that the use of ASM might also be beneficial for loosely coupled simulations, since for one iteration the resulting partitioning error is much smaller.



(a) $\Delta t = P/10$ (b) $\Delta t = P/20$ (c) $\Delta t = P/30$

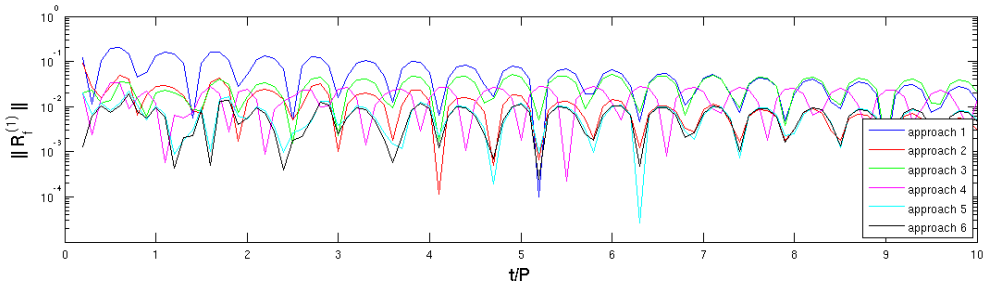
Figure 4.2: Convergence at $t = 6P$ for different timesteps for testcase FSI-2.

For all methods the norm of the first residual value is shown in Fig. 4.3. Here it can be seen that all approaches show an oscillatory behaviour in their initial residual value. For approaches 1 and 2 this is due to the velocity of the piston (see Fig. 2.4). When the velocity is high, the previous solution is further off. For approach 4, where the previous mapping is used, the opposite occurs. In the regions where the piston has a high velocity the behaviour of the piston is almost linear such that the linear mapping is very accurate, resulting in a good initial estimate. For the other approaches the oscillations are proportional to the deficit between the fine and coarse model piston displacement, which is shown in Fig. 4.4. In all cases the correction term δ_n is very efficient and damps the amplitude of the oscillations.

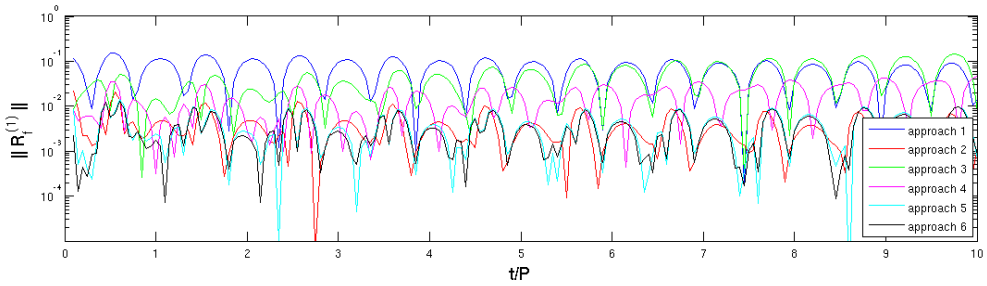
From Table 4.1 as well as Figs. 4.1, 4.2 and 4.3 it is clear that the ASM variations which yield the best results are

- Approach 2: the first estimate is a function of previous results (state extrapolation),
- Approach 5: the optimal mapping of the previous timestep is applied to the coarse model solution at the current timestep in combination with a penalty term,
- Approach 6: the first estimate equals the coarse model solution at the current timestep in combination with a penalty term.

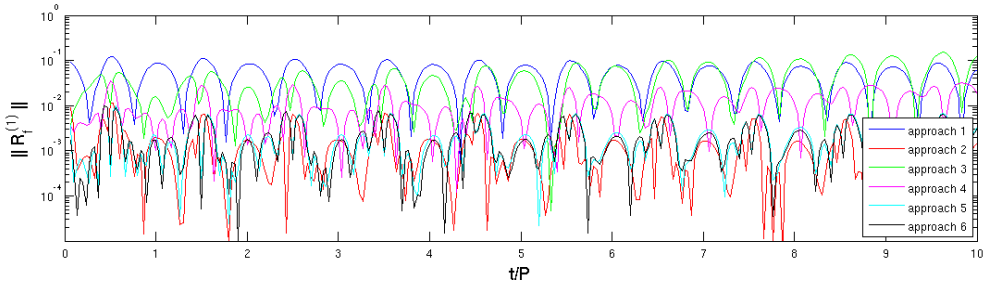
Approaches 5 and 6 are the most efficient when the applied timestep is large, like $\Delta t = P/10$. For slightly smaller timesteps the efficiency of approach 2 increases and tends to outperform all other approaches. This is as expected and is also observed for testcases FSI-1 and FSI-3. For this combination of fine and coarse models the coarse model response resembles the fine model response relatively well, especially early in the simulation. Therefore also the approaches based on the coarse and surrogate model without any correction term perform reasonably well. However, both when time (and thus the deficit) increases and for coarse models which resemble the fine model less, those approaches are likely to become less efficient.



(a) $\Delta t = P/10$



(b) $\Delta t = P/20$



(c) $\Delta t = P/30$

Figure 4.3: Initial residual value for several approaches to define the initial estimate $\mathbf{x_f}_n^{(1)}$ (test case FSI-2).

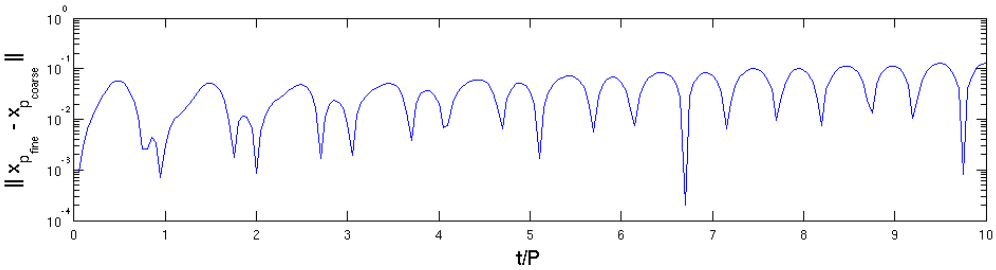


Figure 4.4: Difference between the fine and coarse model piston displacement.

Apart from the relative computational cost, also the average fine residual value at the end of every timestep, $\|\mathbf{R_f}\|_{av}$, is included in Table 4.1. This value is a measure of the average partitioning error, which in these simulations is required to be less than $\epsilon = 1 \cdot 10^{-12}$ in order to have a fully coupled solution. This quantity is included since it might alter the conclusions when only looking at the computational cost. For example, if the cost of two methods is similar one method might still be considered more efficient over the other if its average partitioning error is smaller and thus its accuracy higher. If the bound on the partitioning error would be loosened, this method would require less subiterations and thus have a lower cost than the less accurate method. For the current simulation results, all partitioning errors are well below the required value of $1 \cdot 10^{-12}$. Looking at the applied ASM approaches it is observed that the final partitioning error is in general slightly larger than for QN. Therefore the actual savings in computational cost might by a little lower for other applications.

4.3 2D panel problem results

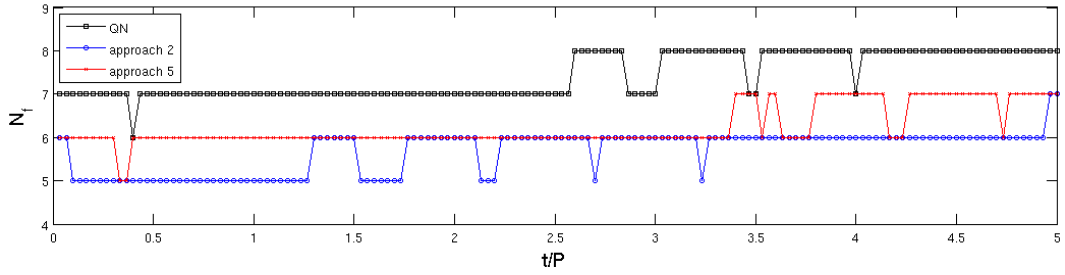
The aggressive space mapping method is also applied to the two-dimensional panel problem for testcases FSI-4, FSI-5 and FSI-6. Detailed results for testcase FSI-5 are presented in the current section, the other cases can be found in Appendix B. This 2D problem is different from the 1D testcases in that it has multiple interface points. The parameter extraction process therefore is expected to be more expensive. Moreover, from Section 2.2 it is seen that the deficit between the fine and coarse model is much larger, especially after a few periods. This testcase therefore is useful to investigate the robustness of the aggressive space mapping method.

	$\Delta t = P/10$		$\Delta t = P/30$		$\Delta t = P/50$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	1.2 E-13	100	1.3 E-13	100	1.3 E-13
1: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1}$	77.9	2.8 E-13	89.3	2.7 E-13	103.6	2.3 E-13
2: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1} + \delta_n$	74.5	2.3 E-13	77.1	2.6 E-13	88.6	2.0 E-13
3: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n$	76.6	2.6 E-13	91.3	3.2 E-13	110.7	1.7 E-13
4: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1}$	76.0	2.7 E-13	87.9	2.7 E-13	103.1	2.3 E-13
5: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1} + \delta_n$	73.8	3.0 E-13	84.5	2.4 E-13	94.9	1.8 E-13
6: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n + \delta_n$	76.6	2.6 E-13	91.3	3.2 E-13	110.7	1.7 E-13

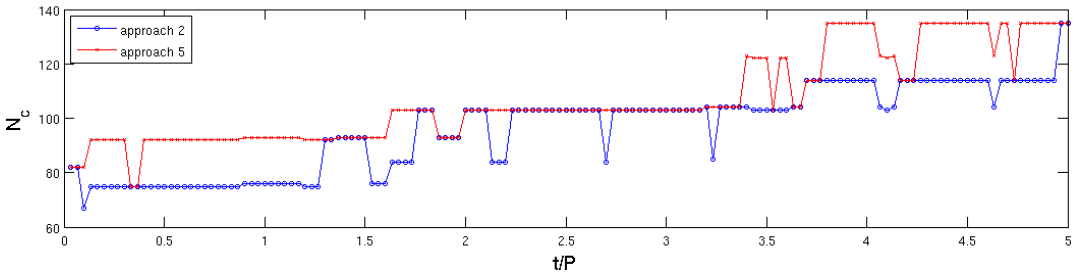
Table 4.2: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, ASM test case FSI-5.

First of all it is observed that ASM is also very effective for this more expensive 2D problem, as the obtained efficiencies are even higher than for the 1D test problem. For the piston problem the savings in computational cost were maximum 17%, whereas in this case savings up to 33% are obtained. This is caused by the presence of the Broyden approximation, which reduces to a less accurate finite difference approximation for one-dimensional problems. Again the efficiency is highest for large timesteps and for strongly coupled problems. Moreover, the partitioning errors for ASM are all of the same order of magnitude as for QN, thus similar efficiency results are expected for other values of the coupling strength ϵ.

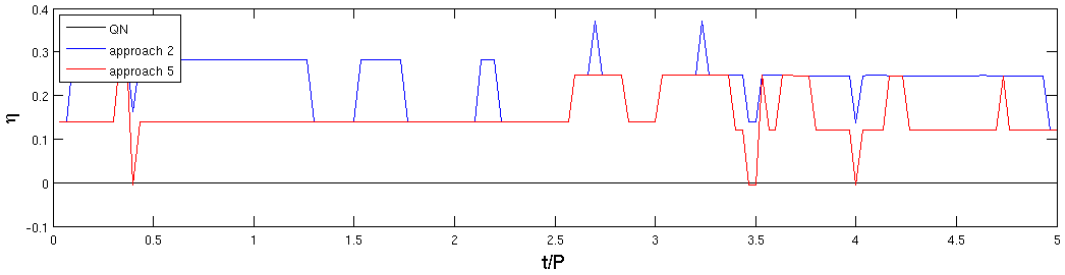
The number of fine and coarse model evaluations per timestep as well as the relative frequency as a function of time are shown in Fig. 4.5. Every timestep at least one and often two fine model evaluations are saved by using ASM. This comes at the cost of a large amount of coarse model evaluations, about five times as much as were required for the 1D test problem. This is due to the increased number of interface points which causes the parameter extraction process to be more expensive. However, the coarse model is also cheaper (about 0.03% of a fine evaluation) such that the coarse cost is still only a fraction of the fine cost. Furthermore it can be seen that the algorithm is able to perform the coupling between fluid and structure several periods after the first occurrence of flutter, where the deficit between the fine and coarse model is very large, without a decrease in efficiency. This proves the robustness of ASM.



(a) Fine model evaluations

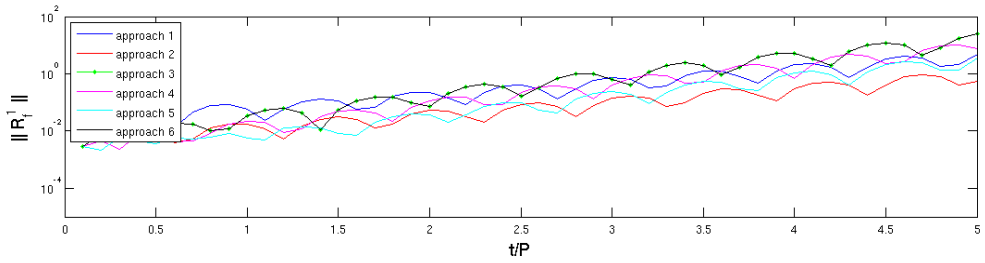


(b) Coarse model evaluations

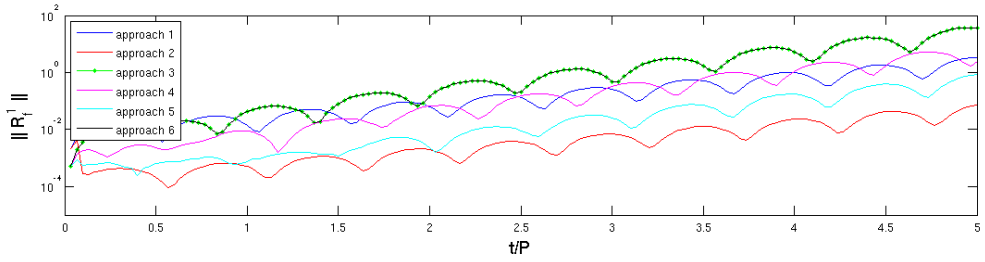


(c) Relative efficiency w.r.t. Quasi-Newton

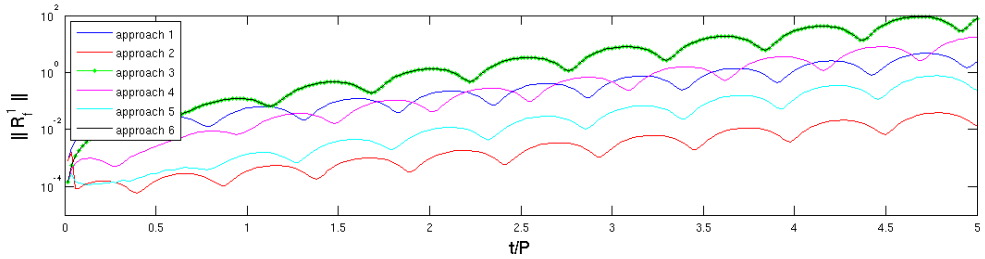
Figure 4.5: Comparison of different approaches to define the initial estimate for the 2D panel problem (test case FSI-5), applying ASM with $\Delta t = P/30$.



(a) $\Delta t = P/10$



(b) $\Delta t = P/30$



(c) $\Delta t = P/50$

Figure 4.6: Initial residual value for several approaches to define the initial estimate $\mathbf{x_f}_n^{(1)}$ (test case FSI-5).

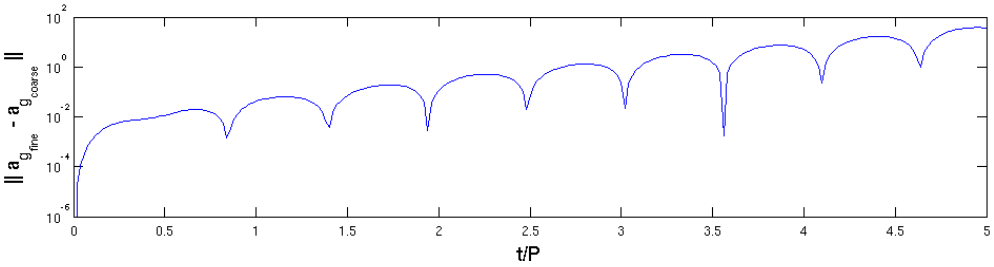
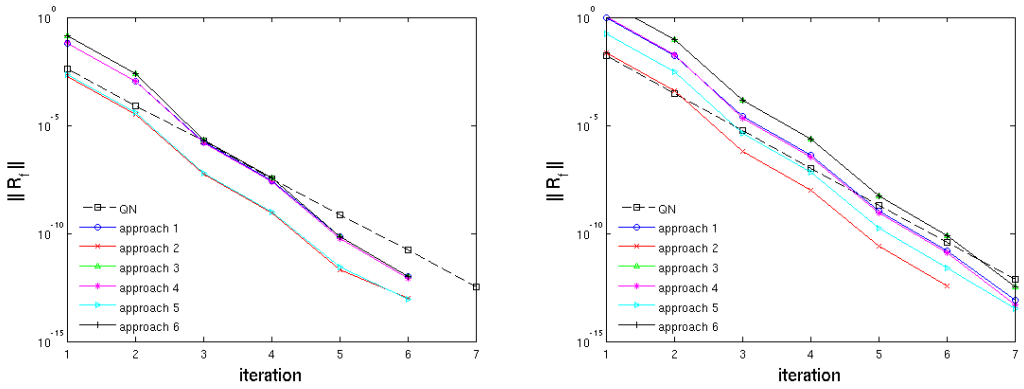


Figure 4.7: Difference between the fine and coarse model panel displacement.



(a) $t = 2P$ (b) $t = 4P$

Figure 4.8: Convergence during one timestep at different moments for test case FSI-5 with $\Delta t = P/30$.

Finally, in Fig. 4.6 the initial residual values are shown for the different approaches that are used to define the first fine iterate. State extrapolation (approach 2) clearly is the most efficient, even for large timesteps. Approaches based on the (mapped) coarse model are less effective here due to the large deficit between both models.

4.4 ASM conclusions

Overall it can be concluded that the use of aggressive space mapping to perform strongly coupled fluid-structure interaction simulations works very well, for the 1D as well as the 2D test problem. The method can handle a wide range of fluid-to-structure mass ratios with relatively large timesteps. Even when the fine and coarse model differ considerably the method still shows good performances. The largest savings in computational cost are obtained for strongly coupled problems for which large timesteps are used. The method is also more efficient for 2D than for 1D problems.

From the previous results it is also concluded that the most efficient way to define the first estimate of every new timestep is to use a function of solutions of previous moments in time, here corresponding to approach 2. Although for large timesteps (mapped) coarse model based approaches might yield better results, in general approach 2 is considered to be most applicable for use due to its simplicity and reliability.

Chapter 5

The generalized space mapping framework (GSM)

In this chapter the use of the generalized space mapping (GSM) framework for fluid-structure interaction problems is investigated. GSM is fundamentally different form ASM in that it is a dual method which uses both input and output mappings.

5.1 The GSM algorithm

5.1.1 The surrogate model

A first example of dual space mapping is the generalized space mapping framework [8; 17]. It consists of a family of surrogate models which utilize an input mapping to reduce the misalignment between the coarse and fine models over a region of interest. Furthermore, an output mapping is used to ensure matching of responses and first-order derivatives between the mapped coarse model and the fine model at the given iteration point. The general form of the surrogate is given in (5.1), with mapping parameters $\mathbf{x_p} = (\mathbf{A}, \mathbf{B}, \mathbf{c}, \mathbf{d}, \mathbf{E})$.

$$\mathbf{R_s}^{(j)}(\mathbf{x}) = \mathbf{A}^{(j)} \cdot \mathbf{R_c}(\mathbf{B}^{(j)} \cdot \mathbf{x} + \mathbf{c}^{(j)}) + \mathbf{d}^{(j)} + \mathbf{E}^{(j)} \cdot (\mathbf{x} - \mathbf{x_f}^{(j)}) \tag{5.1}$$

The mapping parameters are obtained in three stages. First of all the input mapping parameters $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{c}$ at iterate j are calculated by performing the parameter extraction process

$$(\mathbf{A}^{(j)}, \mathbf{B}^{(j)}, \mathbf{c}^{(j)}) = arg \min_{(\mathbf{A},\mathbf{B},\mathbf{c})} \varepsilon^{(j)}(\mathbf{A}, \mathbf{B}, \mathbf{c}) \tag{5.2}$$

which aims at matching the coarse with the fine model residual. This is done by minimizing a certain matching condition $\varepsilon^{(j)}$, which is given in (5.7). Afterward, a perfect match between

the responses of the surrogate and the fine model is ensured by the addition of the output mapping

$$\mathbf{d}^{(j)} = \mathbf{R_f}(\mathbf{x_f}^{(j)}) - \mathbf{A}^{(j)} \cdot \mathbf{R_c}(\mathbf{B}^{(j)} \cdot \mathbf{x_f}^{(j)} + \mathbf{c}^{(j)}). \tag{5.3}$$

Finally, GSM aims at matching the first order derivatives of the surrogate and the fine model through the last term in (5.1). The mapping parameter $\mathbf{E}^{(j)}$ is defined as

$$\mathbf{E}^{(j)} = \mathbf{J_f}(\mathbf{x_f}^{(j)}) - \mathbf{A}^{(j)} \cdot \mathbf{J_c}(\mathbf{B}^{(j)} \cdot \mathbf{x_f}^{(j)} + \mathbf{c}^{(j)}) \cdot \mathbf{B}^{(j)}. \tag{5.4}$$

As mentioned before, the fine and coarse model derivatives are not available for FSI simulations. Therefore the output term $\mathbf{E}^{(j)}$ in (5.4) cannot be determined exact. However, a fairly accurate approximation can be obtained by applying Broyden's method. When applying Broyden's method to approximate both $\mathbf{J_f}$ and $\mathbf{J_c}$, the subsequent calculation of $\mathbf{E}^{(j)}$ is based on two first order approximations. This will diminish the accuracy of the output term in the GSM framework aimed at matching the first order derivatives, causing the term to be of little use. A better approach consists of approximating $\mathbf{E}^{(j)}$ directly. With $\mathbf{A}^{(j)}$, $\mathbf{B}^{(j)}$ and $\mathbf{c}^{(j)}$ constants and noting that $\mathbf{E}^{(j)}$ is actually the derivative of $\mathbf{d}^{(j)}$,

$$\begin{aligned}
\mathbf{E}^{(j)} &= \frac{\partial \mathbf{R_f}}{\partial \mathbf{x_f}}(\mathbf{x_f}^{(j)}) - \mathbf{A}^{(j)} \frac{\partial \mathbf{R_c}}{\partial \mathbf{x_f}}(\mathbf{B}^{(j)} \cdot \mathbf{x_f}^{(j)} + \mathbf{c}^{(j)}) \\
&= \frac{\partial}{\partial \mathbf{x_f}} \left(\mathbf{R_f}(\mathbf{x_f}^{(j)}) - \mathbf{A}^{(j)} \cdot \mathbf{R_c}(\mathbf{B}^{(j)} \cdot \mathbf{x_f}^{(j)} + \mathbf{c}^{(j)}) \right) \\
&= \frac{\partial}{\partial \mathbf{x_f}} \mathbf{d}^{(j)},
\end{aligned} \tag{5.5}$$

the approximation of $\mathbf{E}^{(j)}$ becomes

$$\mathbf{E}^{(j)} \approx \mathbf{E}^{(j-1)} + \frac{\mathbf{d}^{(j)} - \mathbf{d}^{(j-1)} - \mathbf{E}^{(j-1)} \Delta \mathbf{x_f}}{\Delta \mathbf{x_f}^T \Delta \mathbf{x_f}} \Delta \mathbf{x_f}^T. \tag{5.6}$$

Note that this Broyden method is not reliable if $\Delta \mathbf{x_f} \to 0$. Therefore $\mathbf{E} = 0$ is used if this happens, such that no derivative information is taken into account anymore.

5.1.2 Parameter extraction

The matrices $\mathbf{A}^{(j)}$ and $\mathbf{B}^{(j)}$ and the vector $\mathbf{c}^{(j)}$ have to be obtained by means of parameter extraction applied to a suitable matching condition $\varepsilon^{(j)}$. In literature [8; 17] it is found that $\mathbf{A}^{(j)}$ should be diagonal. However, $\mathbf{B}^{(j)}$ is usually a full matrix. This means that a total of $n_f^2 + 2n_f$ parameters have to be extracted during the parameter extraction process. When the number n_f of elements in the fine model response vector is large, as is the case for FSI simulations, the computational cost of the parameter extraction process becomes excessively high, thereby causing the entire space mapping optimization to be inefficient. Moreover, such a large amount of extractable variables causes the system to be severely underdetermined. To prevent those problems, $\mathbf{B}^{(j)}$ is assumed to be diagonal as well in the remainder of this work. The total number of extractable parameters thus becomes $3n_f$. The system therefore is still underdetermined. However, the use of a suitable parameter extraction approach can

alleviate this problem. In general the applied matching condition $\varepsilon^{(j)}$ looks like (5.7), which accounts for MPE, GPE and constrained PE.

$$\begin{aligned}\varepsilon^{(j)}(\mathbf{A},\mathbf{B},\mathbf{c}) = &\sum_{k=1}^{j} w_k \|\mathbf{R_f}(\mathbf{x_f}^{(k)}) - \mathbf{A}\cdot\mathbf{R_c}(\mathbf{B}\cdot\mathbf{x_f}^{(k)}+\mathbf{c})\| \\ &+ \sum_{k=1}^{j} v_k \|\mathbf{E}^{(k)}\| \\ &+ \omega\|\mathbf{A}-\mathbf{A}_0\| + \omega\|\mathbf{B}-\mathbf{B}_0\| + \omega\|\mathbf{c}-\mathbf{c}_0\|\end{aligned} \tag{5.7}$$

In (5.7) the coefficients w_k and v_k are weighting factors which allow to choose the points where the responses and derivatives are matched. The value of ω determines whether or not and to which extend the mapping parameters are constrained to be close to their objectives. Those objectives are chosen equal to the initial value of $\mathbf{x_p}$, such that the mapping closest to the unit mapping is favoured. The penalty terms are effective to ensure uniqueness of the parameter extraction process as long as not enough fine model evaluations have been performed to determine the system or to use GPE. However, the coefficient ω should be small such that its influence is small compared to the primary matching criteria, being matching responses and/or derivatives.

5.1.3 Implementation

The general surrogate model defined by (5.1) allows for a broad range of different surrogate models. By preconditioning some of the mapping parameters different surrogates with different flexibility (the ability to match the fine model, which depends on the number of model parameters) can be created. For example, when no derivative information is available one can set $\mathbf{E} = 0$ and $v_k = 0$. Choosing the proper amount of flexibility is an important issue, since it affects the uniqueness of the parameter extraction process as well as the extrapolation properties of the model. Models with a too high flexibility may provide excellent matching at the current iteration point, but poor matching elsewhere (so poor extrapolation) and therefore slow down or even prevent convergence of the algorithm. A possible solution to ensure convergence consists of the adaptive bounds approach, see section 3.4. By adapting the bounds for the extractable parameters relative to the obtained match, the flexibility of the model is decreased every iteration, thereby ensuring convergence to unique and suitable mapping parameters.

Note that this dual space mapping approach requires twice as many coarse/surrogate optimizations per timestep as fine model evaluations. This means that the coarse computational cost will be at least twice as high compared to the aggressive space mapping method. This method will therefore only be beneficial when less iterations are required to resolve a timestep and the cost per coarse model evaluation is small. However, due to the combination of both input and output mappings the current surrogate model can be very flexible. Hence, it is expected that GSM might still be applicable at times where ASM fails.

5.2 1D piston problem results

The generalized space mapping framework is tested by application to the one-dimensional piston problem. Several variations with respect to parameter extraction, convergence improvements and the definition of the first estimate $\mathbf{x_f}_n^{(1)}$ are compared such that the best approach can be identified. Only results for testcase FSI-2 are presented here, testcases FSI-1 and FSI-3 can be found in Appendix C.

5.2.1 Parameter extraction

First several approaches to perform the parameter extraction are compared with each other. Distinction is made between single point extraction (SPE), multipoint extraction (MPE) and gradient parameter extraction (GPE). For those methods also the influence of the addition of a penalty term, thereby constraining the extracted parameters to be close to a certain objective, is investigated. The results for testcase FSI-2 (the cost and average final residual values) are presented in table 5.1 for different PE approaches with different timesteps.

For SPE only the fine and coarse residual values at the most recent iteration point are matched. The system to be minimized during parameter extraction is always underdetermined since three mapping parameters have to be extracted in order to define the surrogate. In this case a Levenberg-Marquardt algorithm is used to perform the parameter extraction since this method can deal with underdetermined systems. However, as can be seen from Table 5.1, using standard SPE as parameter extraction approach is not very efficient. Therefore constraints are introduced which make sure that the extracted parameters do not only match the residual values, but also that the obtained surrogate model is as close to the standard coarse model as possible. Thus $\|\mathbf{x_p}_n^{(j)} - \mathbf{x_p}_0\|$ is minimized as well, thereby making the system determined again such that a nonlinear least squares method can be used to extract the mapping parameters. The weight which is used for this penalty term is $\omega = 0.01$, since it is found that this value is large enough to improve the convergence of the system but also small enough to keep the surrogate flexible. It is found that this form of constrained SPE improves the standard SPE approach a lot. Other possibilities for the implementation of constrained SPE would be to use $\mathbf{x_p}_{n-1}^*$ instead of the unit mapping $\mathbf{x_p}_0$ as objective for $\mathbf{x_p}_n^{(j)}$, or the use of a variable weighting factor which depends on the difference between the fine and coarse residual. Both were tested but did not yield additional convergence improvements.

The second parameter extraction approach which is tested for the piston problem is multipoint extraction (MPE). Since three mapping parameters have to be extracted, also three test points are required in order to make the system determined. This is only possible from the third iteration on, so during the first and second iteration the Levenberg-Marquardt method is used. From the third iteration on *lsqnonlin* is used where the three most recent iteration points are taken into account. Since the most recent iteration result should have the most influence on the matching condition, the weighting factor w_k is defined similar to (3.9), so

$$w_k = \alpha^{j-k} \qquad \text{with} \qquad k = j-2, j-1, j. \tag{5.8}$$

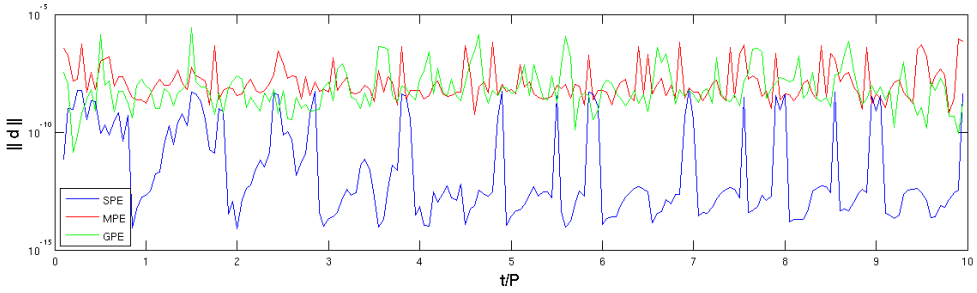
This implies that the weight of the last iteration point is largest (and equal to 1) and decreases for earlier iteration points. It was found that the influence of the value of α is small. Here $\alpha = 0.3$ is used since this is in the range proposed by Koziel [19] and good results are obtained for this value. As can be seen in Table 5.2, this form of MPE performs well and is more efficient than QN for large timesteps. Since this parameter extraction method is underdetermined for the first two iterations, it is tested whether the use of constraints, similar as for SPE, improves the performance of GSM with MPE. However, it is found that it does not: the addition of constraints during the first two iterations has an adverse effect on the rate of convergence.

Finally, GPE is considered as the parameter extraction approach. In this case the system becomes determined from the second iteration on, since then two points can be used to match the residuals and enough information is available to approximate the Jacobian $\mathbf{E}$. In this work, $v_j = 1$ and $v_k = 0$ for $k = 1, \cdots, j-1$ such that the Jacobians are matched at the last iteration point only. This is similar as done in [8]. Applying this parameter extraction approach to GSM does not lead to good performances. The computational cost is much higher than when MPE is used. It is believed that this is due to the absence of exact derivative information. The Jacobian $\mathbf{E}^{(j)} = \frac{\partial \mathbf{d}^{(j)}}{\partial \mathbf{x_f}}$ is approximated using Broydens method, which causes an approximation error which is present in both the parameter extraction process and the output mapping term. Moreover, the function $\mathbf{d}(\mathbf{x_f})$ changes every iteration which lowers the accuracy of the approximation of $\mathbf{E}^{(j)}$. This way the accuracy of the surrogate model and therefore the rate of convergence decreases. Again, also the effect of applying constraints for the parameter extraction at the first iteration is investigated. This does yield some improvements for small timesteps, but GPE still performs worse than QN.

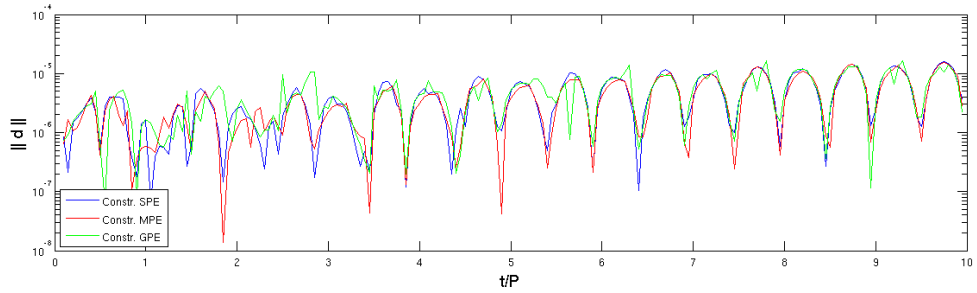
	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	6.4 E-16	100	2.2 E-16	100	2.3 E-15
SPE	176.5	7.8 E-14	242.4	3.6 E-14	284.4	3.1 E-14
Constr. SPE	104.9	5.1 E-15	104.7	3.1 E-15	107.6	9.7 E-15
Constr. SPE + TE	97.1	1.4 E-15	97.3	3.2 E-15	105.2	2.9 E-15
Constr. SPE + AB	102.2	1.5 E-15	104.4	6.9 E-16	107.9	4.7 E-16
MPE 3pt.	95.9	3.1 E-15	92.5	6.6 E-15	102.4	3.7 E-16
Constr. MPE 3pt.	112.0	2.9 E-15	111.1	6.9 E-15	110.7	1.1 E-14
MPE 4pt.	96.0	3.0 E-15	92.7	6.5 E-15	102.4	3.6 E-15
MPE 3pt. + TE	98.3	1.8 E-15	95.7	1.0 E-14	103.6	4.6 E-16
MPE 3pt. + AB	95.9	2.5 E-15	93.1	6.5 E-15	102.0	4.0 E-15
GPE 2pt.	135.2	5.3 E-16	186.1	2.6 E-16	200.4	2.1 E-16
Constr. GPE 2pt.	155.9	4.5 E-16	166.7	1.1 E-15	177.6	9.1 E-16
GPE 3pt.	122.0	4.7 E-16	127.1	9.3 E-16	151.4	8.2 E-16
GPE 2pt. + TE	98.7	6.9 E-16	99.7	2.4 E-15	105.0	5.5 E-16
GPE 2pt. + AB	135.2	2.4 E-15	185.6	1.5 E-15	200.4	1.8 E-15

Table 5.1: Relative computational cost and average final accuracy for different PE approaches for GSM, testcase FSI-2 (TE = Taylor expansion and AB = adaptive bounds).

The approximation quality of the different parameter extraction approaches is visualized in Fig. 5.1, which shows the matching error between the fine and coarse model residuals at the most recent iteration point after PE, both without and with constraints. As expected, standard SPE yields the best match whereas $\mathbf{d}$ is much larger for MPE and GPE. This is due to the fact that other factors are implemented in the matching condition. When constraints are added the matching error is of the same order of magnitude for all methods. Note that the difference is largest for SPE, which is also the PE approach for which the addition of constraints is the most beneficial. Fig. 5.1 and Table 5.1 clearly show for SPE that good approximation properties yield worse extrapolation properties, thus slow convergence.



(a)



(b)

Figure 5.1: Quality of the parameter extraction, testcase FSI-2 with $\Delta t = P/20$.

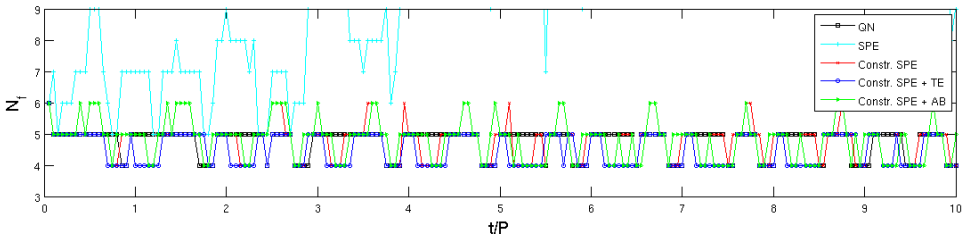
5.2.2 Convergence improvements

From the foregoing results it is concluded that MPE performs best, whereas the performance of GPE is rather disappointing. However, in Section 3.4 some measures which are likely to improve the convergence are introduced. Those measures consist of predicting the values of the mapping parameters in $\mathbf{x_p}$ for the next iteration by means of a first order Taylor series expansion, and introducing upper and lower limits for $\mathbf{x_p}$ which are adjusted every iteration with respect to the quality of the mapping. Those convergence improvements are applied to GSM with constrained SPE, MPE and GPE. The results are also included in table 5.1 and the number of required iterations for every PE approach is visualized in Fig. 5.2.

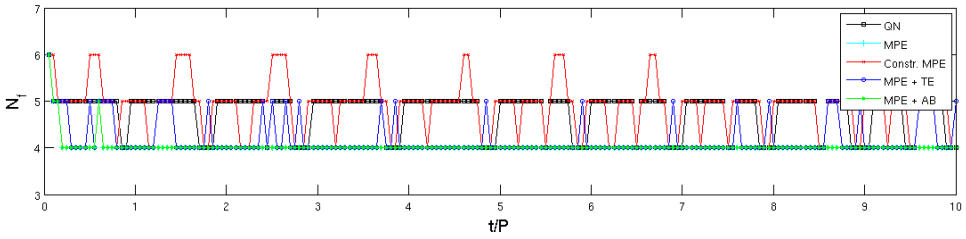
It is observed that the use of a Taylor series expansion for $\mathbf{x_p}$ has a positive effect on both SPE and GPE, where the improvement is especially large for GPE and rather small for SPE.

For both methods the computational cost is now comparable to the cost of MPE and thus more efficient than QN. No positive effect is observed for MPE. On the contrary, MPE even performs slightly worse in combination with a Taylor expansion. Also here, $\nabla_{\mathbf{x_f}}\mathbf{x_p}^{(j)} = 0$ is used if $\Delta\mathbf{x_f} \to 0$.

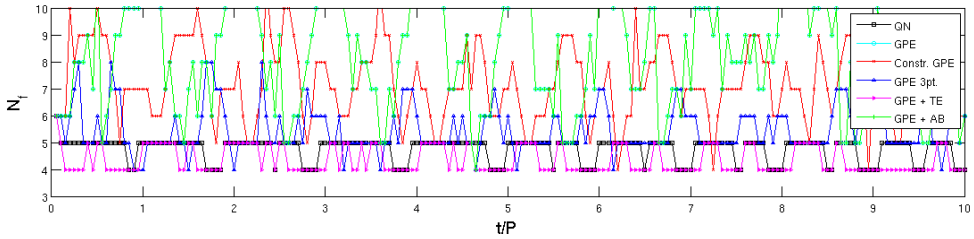
The addition of adaptive bounds to the parameter extraction process on the other hand does not yield any noticeable effects for constrained SPE, MPE and GPE. In [22] two conditions for the application of adaptively constrained bounds are identified: the surrogate should be over-flexible and be able to approximate the fine model with sufficient accuracy. The first condition is satisfied. The second condition however, is only satisfied when standard SPE is used (see Fig. 5.1), as done in [22]. For all other PE approaches some kind of measure to reduce the approximation quality in favour of the extrapolation quality of the surrogate is applied, thereby eliminating the effect of adaptive bounds. However, the algorithm applied for PE does not handle bound constraints when the system is underdetermined such that no adaptive bounds can be applied for standard SPE.



(a) Fine model evaluations for SPE



(b) Fine model evaluations for MPE



(c) Fine model evaluations for GPE

Figure 5.2: Number of fine model evaluations per timestep for the 1D piston problem (test case FSI-2), applying GSM with different PE approaches and $\Delta t = P/20$.

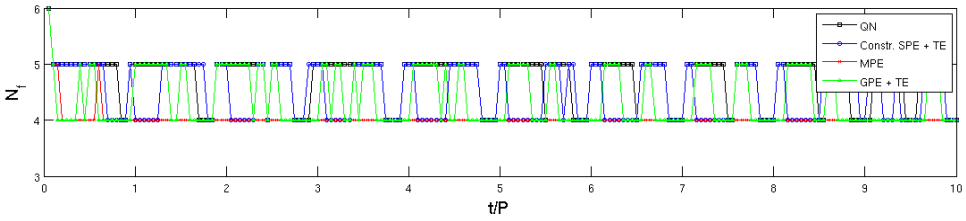
Apart from a Taylor series expansion for the mapping parameters and adaptive bounds during parameter extraction, it is also tested whether taking into account more points than strictly necessary improves the convergence properties of MPE and GPE. This way the region of applicability of the surrogate would be increased. For MPE this does not effect the computational cost, as any savings in fine evaluations are counteracted by the increased number of coarse evaluations. This could have been expected, since the next iterate will be closer to the current point than the previous one. Increasing the region over which the surrogate can be used to approximate the fine model from three to four points therefore has little use. For GPE on the other hand, the computational cost is lowered significantly. Here originally only two points are used, and thus the third point that is added is still close enough to the current iterate.

Although the convergence properties of SPE and GPE can be improved significantly by using constraints, a Taylor series expansion and an additional point during the parameter extraction, it is found that combining those convergence improvements does not always lead to additional savings in computational cost. For constrained SPE for example, the addition of a Taylor series expansion for $\mathbf{x_p}$ still leads to minor improvements, but for GPE this is not the case.

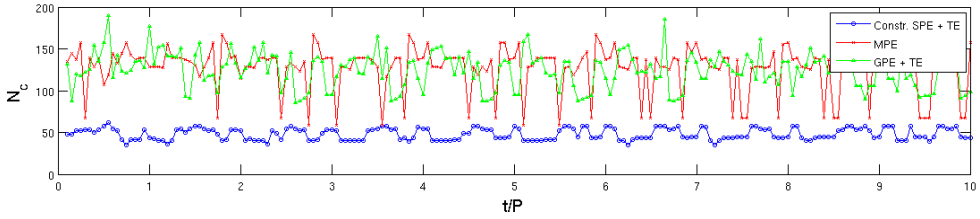
5.2.3 General performance of GSM

Comparing all three parameter extraction approaches and their variations, one sees from Table 5.1 that standard MPE appears to be the most efficient method for the parameter extraction process, exhibiting both the lowest computational cost and highest accuracy. Standard GPE performs surprisingly bad whereas the results of constrained SPE are much better than expected, even close to the performance of MPE. Those results are also visualized in Fig. 5.3, which shows the number of fine and coarse iterations required to resolve each timestep and the resulting efficiency. It can be seen that for constrained SPE with a Taylor series expansion for the mapping parameters the computational cost is comparable to the cost of standard MPE, while it requires more fine iterations per timestep than MPE. This is however balanced by the coarse computational cost, which is lowest for SPE. Looking at MPE, approximately 150 surrogate evaluations per timestep are required to perform the parameter extraction and subsequent surrogate optimization. Since the cost of one surrogate evaluation is about 0.3% of a fine model evaluation, this implies that the surrogate cost per timestep equals almost half the cost of one fine evaluation (45% for 150 surrogate evaluations). So for GSM the number of fine iterations does not represent the total cost of the algorithm. However, it is a good measure to compare different types of the same parameter extraction approach.

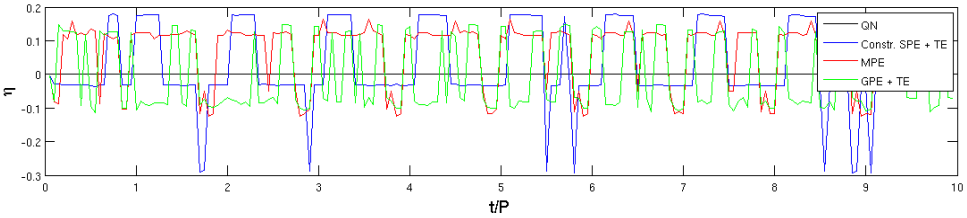
For completeness, in Fig. 5.5 the evolution of all mapping parameters (so $\mathbf{A}$, $\mathbf{B}$, $\mathbf{c}$, $\mathbf{d}$ and $\mathbf{E}$ of equation (5.1)) for the different GSM approaches is shown. A clear distinction is observed when constraints are applied, even for GPE where they are used during the first iteration only. In those cases $\mathbf{A}$ and $\mathbf{B}$ change little from their nominal values and the match is mainly enforced by $\mathbf{c}$, which is rather large. The addition of one extra point during the parameter extraction for MPE does not change the obtained surrogate model. As mentioned before however, the influence of one additional point is larger for GPE, which can be seen in Fig. 5.5. Note that gaps are present in Fig. 5.5(o). At those points $\Delta\mathbf{x_f}$ became too small at the end of the iterative process such that $\mathbf{E} = 0$ was used.



(a) Fine model evaluations

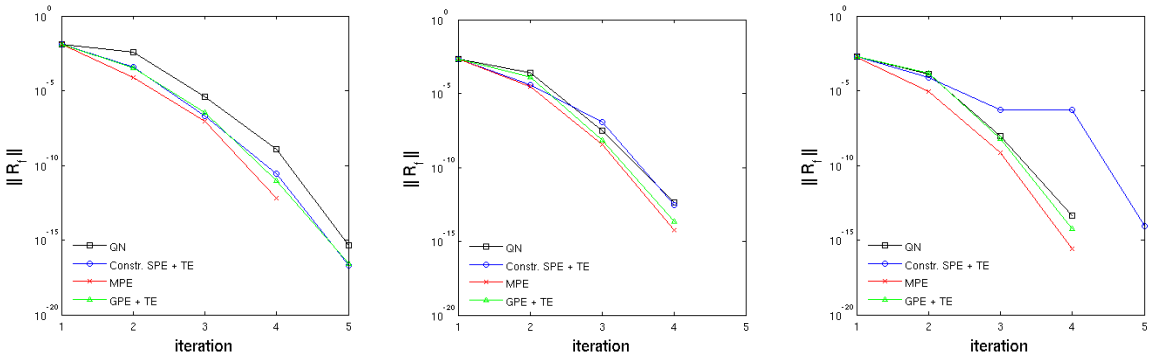


(b) Coarse model evaluations



(c) Relative efficiency w.r.t. Quasi-Newton

Figure 5.3: Computational cost of the GSM method for the 1D piston problem (FSI-2, $\Delta t = P/20$), applying different parameter extraction techniques.



(a) $\Delta t = P/10$ (b) $\Delta t = P/20$ (c) $\Delta t = P/30$

Figure 5.4: Convergence at $t = 6P$ for different timesteps for test case FSI-2.

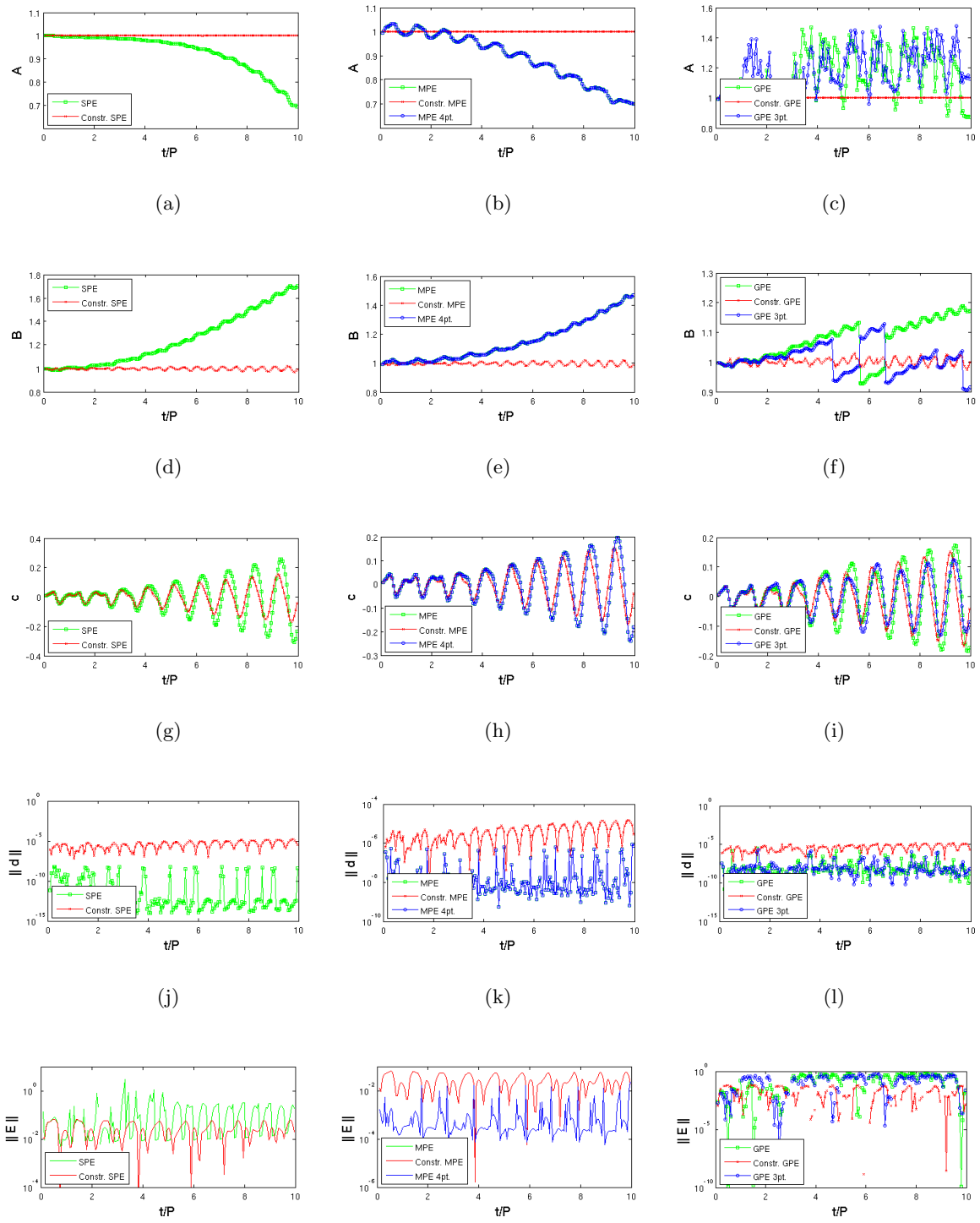


Figure 5.5: Evolution of the mapping parameters for GSM applying different PE approaches and convergence improvements, testcase FSI-2 with $\Delta t = P/20$.

5.2.4 Initial estimate

The previous simulations were performed using state extrapolation (approach 2, (3.29)) to define the initial estimate for every timestep. For completeness, the results for 3-point MPE (which appears to be the most efficient and reliable parameter extraction approach for GSM) for the other initial estimate approaches are presented in Table 5.2. In Fig. 5.6 also the first residual values are shown, which indicate the effectiveness of the initial estimate approaches. As for ASM, approach 2 together with approach 6, where the coarse model solution in combination with a correction term is used to predict the fine model solution, appear to be the most efficient. For small t, where the difference between the coarse and the fine model is still small, approach 6 is the most efficient, whereas approach 2 tends to perform better than approach 6 when time increases. From Table 5.2 it can also be seen that when the size of the timestep decreases the efficiency of QN increases and all GSM methods tend to be less efficient than QN.

	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	6.4 E-16	100	2.2 E-16	100	2.3 E-15
1: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1}$	107.9	2.8 E-15	114.1	2.4 E-15	124.6	6.2 E-16
2: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1} + \delta_n$	95.9	3.1 E-15	92.5	6.6 E-15	102.3	3.5 E-15
3: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n$	105.2	4.6 E-16	110.6	1.4 E-15	121.6	1.6 E-15
4: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1}$	100.4	8.0 E-16	103.1	1.2 E-15	110.7	4.9 E-15
5: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1} + \delta_n$	94.9	6.8 E-15	95.4	7.3 E-15	104.2	5.2 E-15
6: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n + \delta_n$	91.0	1.4 E-14	92.6	8.1 E-15	103.1	3.1 E-15

Table 5.2: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, GSM with 3-point MPE, testcase FSI-2.

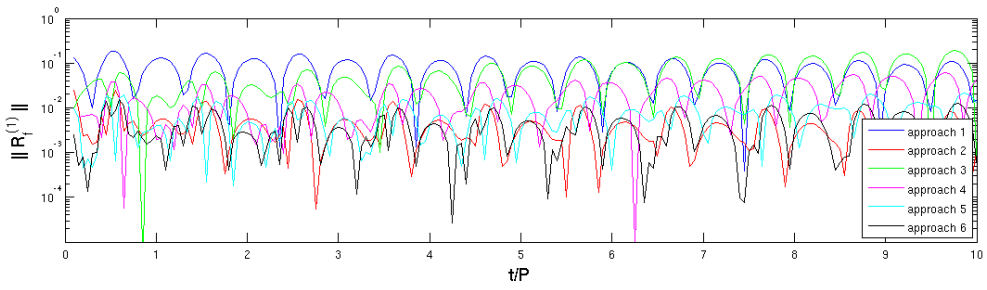


Figure 5.6: Initial residual value for several approaches to define the initial estimate $\mathbf{x_f}_n^{(1)}$ for the 1D piston problem using GSM (test case FSI-2 with $\Delta t = P/20$).

5.3 2D panel problem results

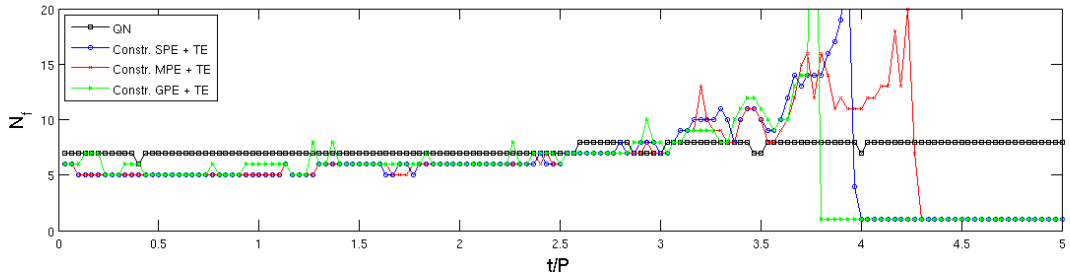
The generalized space mapping framework is also applied to the more complex two-dimensional panel problem. The choice of the parameter extraction approach is very important in this case, since the number of parameters to be extracted is much larger and it is observed that the method does not converge for every parameter extraction approach. Furthermore, it is found that solving an underdetermined system by means of the Levenberg-Marquardt method does not work well for this amount of mapping parameters. Therefore constraints are applied for SPE as well as for MPE and GPE as long as the system is underdetermined. The computational cost and average accuracy for testcase FSI-5 can be found in Table 5.3 for those PE approaches as well as the combination with a Taylor series expansion for the mapping parameters. Similar to the one-dimensional test problem, the addition of a Taylor series expansion improves the efficiency of the method. Note that these results are computed over a time span of two periods. After this moment flutter occurs and the performance of GSM decreases drastically.

	$\Delta t = P/10$		$\Delta t = P/30$		$\Delta t = P/50$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	7.4 E-14	100	1.3 E-13	100	1.4 E-13
Constr. SPE	94.2	5.0 E-13	97.7	1.9 E-13	104.2	6.1 E-14
Constr. SPE + TE	86.6	2.9 E-13	87.9	2.1 E-13	103.2	6.2 E-14
Constr. MPE	∞	∞	122.2	2.2 E-13	126.7	5.9 E-14
Constr. MPE + TE	136.5	3.3 E-13	107.8	2.4 E-13	124.5	6.3 E-14
Constr. GPE	121.7	3.5 E-13	126.6	1.4 E-13	148.5	7.0 E-14
Constr. GPE + TE	107.3	3.3 E-13	114.4	1.6 E-13	125.9	6.8 E-14

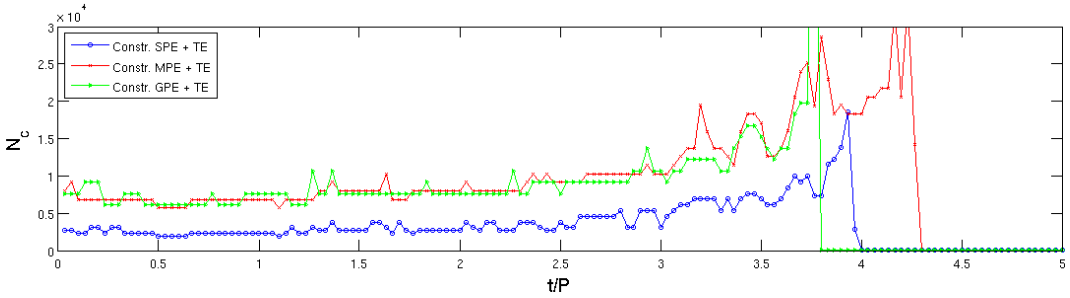
Table 5.3: Computational cost and average final accuracy for the 2D panel problem applying GSM (testcase FSI-5, taking into account 2 periods).

The number of fine and coarse model evaluations required per timestep is shown in Fig. 5.9. All GSM methods require less iterations than the Quasi-Newton method for $t < 3P$. After three periods both models deviate too much to build an accurate surrogate and the rate of convergence decreases rapidly. After approximately four periods the deficit becomes too large to handle and the algorithm fails to converge. It is seen that GPE crashes first, followed by SPE and MPE respectively.

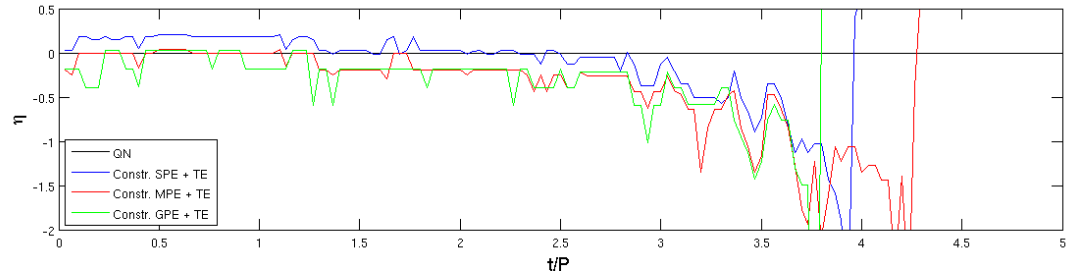
Although GSM is able to reduce the required number of iterations for the first few periods, the efficiency gains are small. This is due to the excessive number of coarse evaluations that are required to obtain these savings in fine computational cost. $3n_f$ parameters have to be extracted and n_f (the number of interface points) is large, causing the parameter extraction process to be very expensive. For SPE almost 3000 coarse evaluations are required per timestep, leading to an additional cost of almost one fine model evaluation. For MPE and GPE this is even twice as much. This results in the low efficiencies as shown in Fig. 5.7(c), even before flutter occurs.



(a) Fine model evaluations



(b) Coarse model evaluations



(c) Relative efficiency w.r.t. Quasi-Newton

Figure 5.7: Computational cost of the GSM method for the 2D panel problem (FSI-5, $\Delta t = P/30$), applying different parameter extraction techniques.

In Fig. 5.8 the evolution of the mapping parameters is displayed. As time increases and both models start to deviate more, the mapping parameters deviate further from their initial values. This proceeds until it is no longer possible to obtain a good match and the coarse model input becomes unphysical. Note that the different parameter extraction approaches result in different values for $\mathbf{A}$ whereas $\mathbf{B}$ and $\mathbf{c}$ are the same for all cases. Also the resulting output parameter $\mathbf{d}$ differs, where it is clearly seen that SPE and GPE experience difficulties with the parameter extraction first.

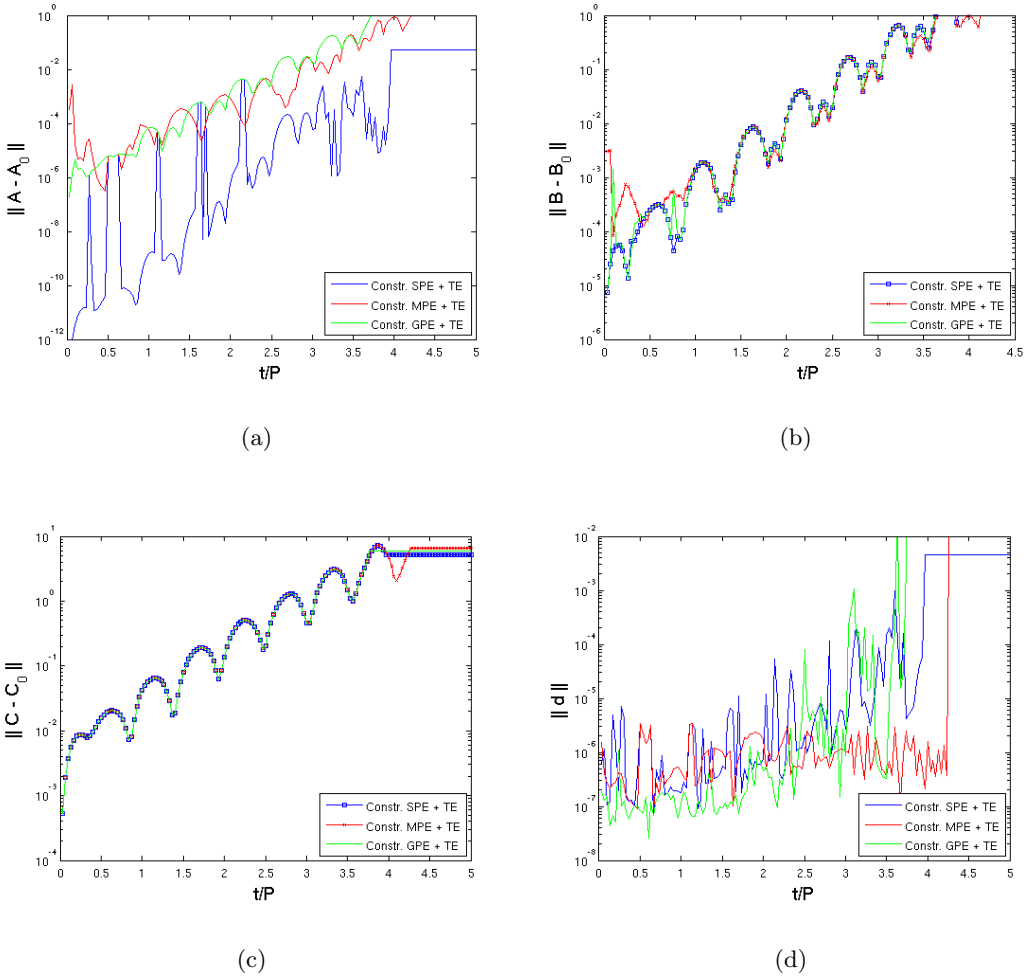


Figure 5.8: Variation of the mapping parameters for GSM for the 2D panel problem, FSI-5 with $\Delta t = P/30$.

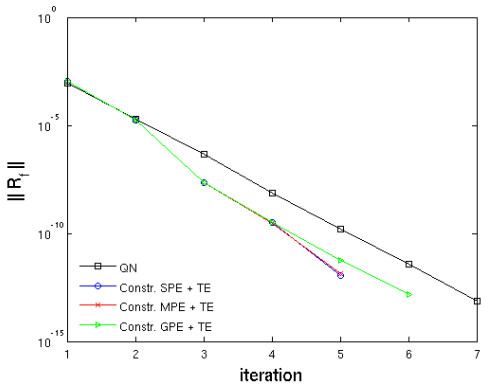


Figure 5.9: Convergence of the GSM algorithm at $t = 1.5P$ for testcase FSI-5.

5.4 Conclusions regarding GSM for FSI applications

Looking at the results of the GSM algorithm discussed in the previous sections it is seen that it is a space mapping method which is able to yield an improved efficiency over the Quasi-Newton method for large timesteps. When the temporal resolution is high, QN proves to be more efficient. However, even for large Δt the efficiency gains are small (less than 5% and 14% for 1D and 2D respectively) and the correct parameter extraction approach has to be chosen. The latter is very important since the amount of parameters that have to be extracted in order to build the surrogate is large, resulting in large coarse computational costs.

For the 1D piston problem it is found that MPE for which the last three iteration points are taking into account is the optimal approach for parameter extraction. However, also constrained SPE with a Taylor series expansion to improve the convergence properties yields good results. For the 2D panel problem on the other hand, MPE performs much worse. For very large timesteps it does not converge without the addition of a Taylor series expansion for the mapping parameters. For this test problem constrained SPE + TE is found to perform the best. The application of adaptive bounds during PE did not yield any improvements in efficiency.

However, no general statements concerning the efficiency of GSM and the optimal approach to implement it can be made. Further investigation to the influence of all coefficients has to be performed and more test problems should be considered before one can draw conclusions regarding the applicability of GSM for FSI problems.

Chapter 6

Output space mapping (OSM)

6.1 The OSM algorithm

Comparing aggressive space mapping to the generalized space mapping framework presented in the previous chapter, one notices that ASM resembles a special case of GSM. Defining

$$\mathbf{B}^{(j)} = \mathbf{J_p}^{(j)} \tag{6.1}$$

and

$$\mathbf{c}^{(j)} = P_j(\mathbf{x_f}^{(j)}) - \mathbf{J_p}^{(j)} \cdot \mathbf{x_f}^{(j)}, \tag{6.2}$$

ASM reduces to that particular case of GSM where only an input mapping is applied (so $\mathbf{R_s}(\mathbf{x_f}) = \mathbf{R_c}(\mathbf{B}\cdot\mathbf{x_f}+\mathbf{c})$). However, due to the distinction in the determination of the mapping ($\mathbf{B}$ and $\mathbf{c}$ are calculated instead of extracted) and the next estimate for the fine model solution (primal instead of dual) both methods are generally considered to be completely different.

If the subset of GSM surrogates containing only an input mapping changes the entire algorithm in such a drastic way, utilizing only an output mapping might do the same. Therefore in this chapter a new surrogate is considered which only consists of the coarse model and an output penalty term depending on the design variable,

$$\mathbf{R_s}(\mathbf{x}) = \mathbf{R_c}(\mathbf{x}) + \mathbf{O}(\mathbf{x}). \tag{6.3}$$

This space mapping method is called output space mapping (OSM). Note that this surrogate requires $\mathbf{x_c}$ to be in the same parameter space as $\mathbf{x_f}$, so the coarse model should contain some kind of interpolation if this is not the case. Therefore in the remainder of this chapter no distinction is made and the variable $\mathbf{x}$ is used for both the fine and the coarse model. Furthermore, the presence of the output term requires the use of a dual approach to determine the next estimate for the fine model solution.

The output term $\mathbf{O}^{(j)}(\mathbf{x})$ in (6.3) is defined such that the surrogate model response as well as its first derivative matches with the fine model at point $\mathbf{x}^{(j)}$. Similar to the generalized space mapping framework an output term $\mathbf{O}^{(j)}$ is defined as

$$\mathbf{O}^{(j)}(\mathbf{x}) = \mathbf{d}^{(j)} + \mathbf{E}^{(j)} \cdot (\mathbf{x} - \mathbf{x}^{(j)}) \tag{6.4}$$

with $\mathbf{d}^{(j)}$ and $\mathbf{E}^{(j)}$ given by (6.6) and (6.7) respectively.

$$\mathbf{d}^{(j)} = \mathbf{d}(\mathbf{x}^{(j)}) = \mathbf{R_f}(\mathbf{x}^{(j)}) - \mathbf{R_c}(\mathbf{x}^{(j)}) \tag{6.5}$$

$$\tag{6.6}$$

$$\mathbf{E}^{(j)} = \frac{\partial}{\partial \mathbf{x}}(\mathbf{R_f}(\mathbf{x}) - \mathbf{R_c}(\mathbf{x})) = \frac{\partial \mathbf{d}}{\partial \mathbf{x}} \tag{6.7}$$

Where $\mathbf{E}^{(j)}$ is again approximated by Broydens rank one scheme (5.6). Note that, as mentioned before, the Broyden approximation for $\mathbf{E}^{(j)}$ looses its accuracy when $\Delta\mathbf{x} \to 0$. When this happens $\mathbf{E}^{(j)} = 0$ is used, which does not have a negative influence on the convergence of the algorithm since for those cases $\mathbf{d}^{(j+1)} \approx \mathbf{d}^{(j)}$.

Comparing this OSM method with the surrogate used for GSM (equation (5.1)) it is clear that this method corresponds to the case of GSM where $\mathbf{A} = \mathbf{I}$, $\mathbf{B} = \mathbf{I}$ and $\mathbf{c} = \mathbf{0}$. These restrictions simplify the method to a great amount. Due to the absence of an input mapping, a parameter extraction process (which was the most expensive part of GSM) is no longer required. Building the surrogate now only demands one coarse model evaluation, after which the mapping parameters $\mathbf{d}$ and $\mathbf{E}$ can be calculated directly. In this way, a locally perfect mapping at $\mathbf{x}^{(j)}$ is obtained in a cheap way. Moreover, due to the Broyden update used to determine the first derivative term $\mathbf{E}$, information of previous iterations is still taken into account thus improving the extrapolation properties of the surrogate. Furthermore, the absence of a parameter extraction process means that the function $\mathbf{d}(\mathbf{x})$ does not change for different iterations or timesteps, making the output term $\mathbf{E}$ more effective than for GSM.

This space mapping method thus requires only one optimization step per iteration, optimizing the surrogate (6.3) in order to determine the next estimate. As a result, the coarse computational cost per iteration of this OSM method consists of only a fraction of the cost required for GSM, comparable to the coarse cost of ASM. An additional advantage of OSM is its simplicity: the lack of parameter extraction makes this method easy to implement and no tuning of coefficients is required.

Other than a special case of GSM, OSM can also be regarded as a method which enhances the coarse model with a prediction term. The output term $\mathbf{O}(\mathbf{x})$ actually predicts the deficit between the fine and coarse model at a certain point $\mathbf{x}$. Looking at (6.4) one recognizes a first order Taylor series expansion of $\mathbf{d}(\mathbf{x})$ around the point $\mathbf{x}^{(j)}$, such that $\mathbf{O}^{(j)}(\mathbf{x}^{(j+1)})$ is a first order approximation of $\mathbf{d}^{(j+1)}$.

6.2 1D piston problem results

As for the ASM and GSM algorithms, the output space mapping method is applied to a one-dimensional (the piston problem, section 2.1) and a two-dimensional testcase (the panel problem, section 2.2). The numerical performance of OSM is investigated by application to testcases with different fluid-to-structure mass ratios and timesteps. The results of testcase FSI-2 (1D) are presented below. The results of the other testcases can be found in Appendix D.

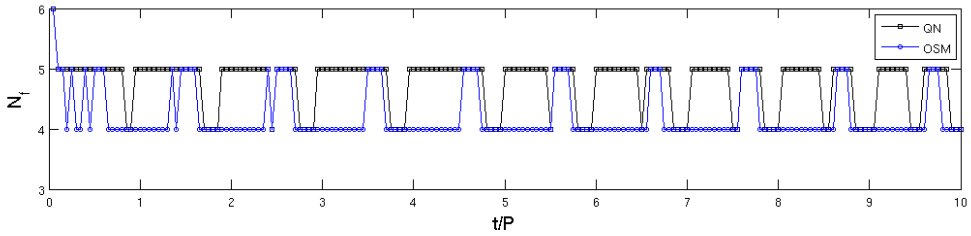
In the previous chapters it was found that the best approach to define the initial estimate at the start of every new timestep is using state extrapolation. Therefore this is the approach which is used in Figs. 6.1, 6.3 and 6.2 to compare the performances of OSM with QN. For completeness also the cost and average partitioning error for the other approaches is included in Table 6.1.

	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	6.4 E-16	100	2.2 E-16	100	2.3 E-15
1: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1}$	101.7	5.2 E-16	105.7	8.3 E-16	117.4	3.8 E-16
2: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1} + \delta_n$	92.4	2.6 E-15	89.3	2.5 E-15	97.6	6.7 E-16
3: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n$	99.3	3.8 E-16	103.6	7.7 E-16	116.3	5.1 E-16
4: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1}$	88.7	2.2 E-15	92.5	2.3 E-15	104.3	1.5 E-15
5: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1} + \delta_n$	87.1	2.7 E-15	90.4	2.3 E-15	99.5	1.8 E-15
6: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n + \delta_n$	89.9	2.7 E-15	90.0	1.9 E-15	98.8	1.4 E-15

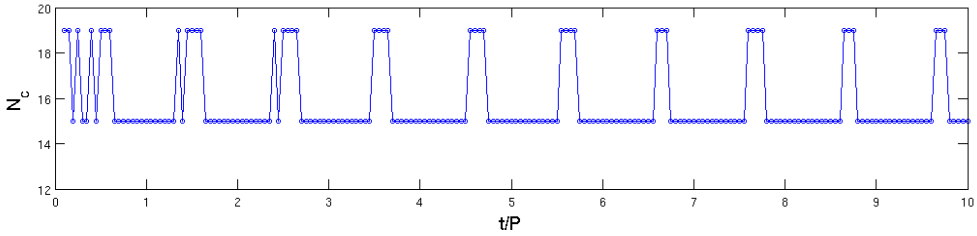
Table 6.1: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, OSM test case FSI-2.

First of all it is observed that OSM performs better than QN for this test problem. Savings in computational cost up to 16% are achieved, where the best results are obtained for problems with a high fluid-to-structure mass ratio and for large timesteps. The final partitioning error is much smaller than the benchmark value of $1 \cdot 10^{-12}$, but still larger than the error for QN. Therefore the actual savings in computational cost to obtain solutions with the same accuracy might be slightly lower than the values indicated here. This can also be seen from Fig. 6.2: for timesteps where both methods require the same amount of fine evaluations the accuracy of OSM is higher, but when OSM saves an iteration its accuracy is lower than for QN.

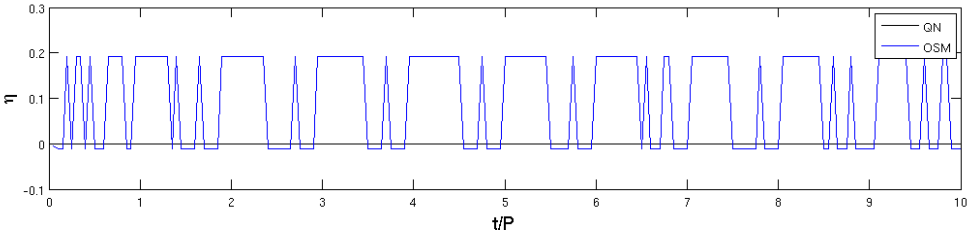
The number of required fine and coarse model evaluations to obtain the fully coupled solution is visualized in Fig. 6.1 for 10 periods. It can be seen that for approximately half of the timesteps OSM needs one fine evaluation less whereas for the other half of the timesteps both methods need the same number of iterations to obtain the required accuracy. The moments at which no savings in computational cost are achieved correspond to the times at which a pressure wave hits the structure. Due to the additional coarse cost OSM is slightly less efficient than QN at those moments. At all other moments however, OSM yields savings over QN up to 20% such that overall OSM is much more efficient than QN. Finally, also the values of the mapping parameters $\mathbf{d}$ and $\mathbf{E}$ are shown in Fig. 6.3.



(a) Fine model evaluations

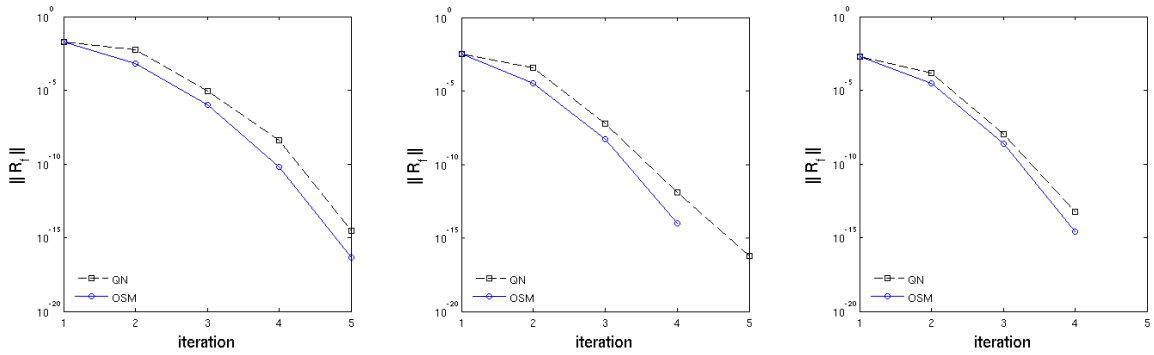


(b) Coarse model evaluations



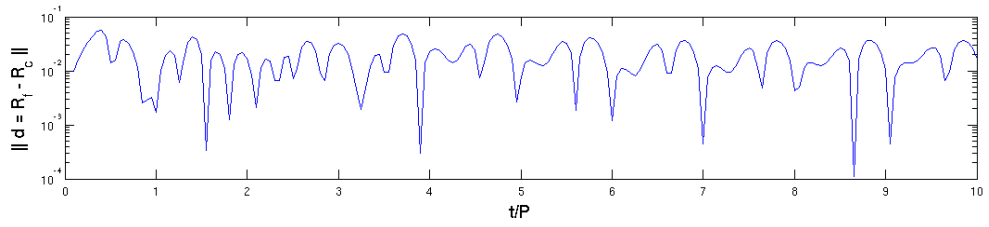
(c) Relative efficiency w.r.t. Quasi-Newton

Figure 6.1: Computational cost for the 1D piston problem (testcase FSI-2) applying OSM with $\Delta t = P/20$.

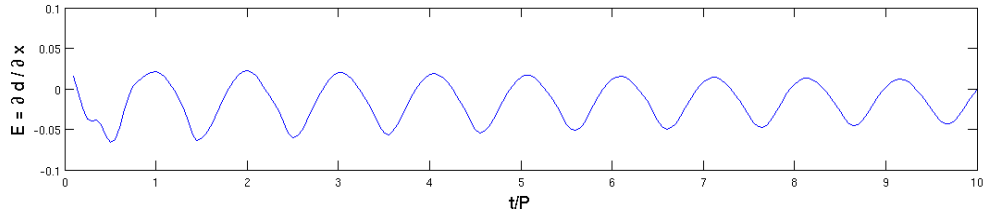


(a) $\Delta t = P/10$ (b) $\Delta t = P/20$ (c) $\Delta t = P/30$

Figure 6.2: Convergence of the OSM method at $t = 6P$ for testcase FSI-2.



(a)



(b)

Figure 6.3: Evolution of the parameters defining the output mapping for OSM, testcase FSI-2 with $\Delta t = P/20$.

6.3 2D panel problem results

OSM is also applied to the two-dimensional panel problem, where it performs very good. Savings in computational cost up to 55% compared to the Quasi-Newton method are obtained while the partitioning error is of the same order of magnitude for both methods. Again, the largest savings are obtained for problems with high values for ζ where the applied timestep is rather large. Fig. 6.4 shows the convergence rate for both OSM and QN, where the curve for OSM is clearly steeper than for QN, proving the large difference in convergence for both methods. The high efficiency of OSM is also shown in Fig. 6.5, from which it is clear that the required amount of fine model evaluations to obtain the fully coupled solution is almost halved. Furthermore, since the cost of a coarse evaluation is approximately 0.03% of the cost of a fine model evaluation, the additional coarse computational cost is very small. Note that where the deficit between both models increases with time, which is visualized by the value of $\mathbf{d}$ in Fig. 6.6, the efficiency of OSM improves. This is a very beneficial property of OSM, since it indicates that the method does not put stringent requirements on the coarse model in order to obtain high efficiencies. Even when flutter is observed the computational cost to resolve the problem does not increase, proving the robustness of the method.

Note that the good performance of OSM for this 2D problem compared to the piston problem is not a coincidence. The output term $\mathbf{E}^{(j)}$, which should ensure matching derivatives and good extrapolation properties, reduces to a simple finite difference approximation for one-

dimensional problems. This means that the term $\mathbf{E}^{(j-1)}$ drops from (5.6) and the derivative information of previous iterations is not taken into account anymore. So OSM bases its surrogate on the last two iterations only for 1D problems, whereas more information can be taken into account for 2D problems. This results in a higher accuracy and wider range of application for the two-dimensional surrogate, thus a higher efficiency for OSM in 2D. The difference in convergence rate between 1D and 2D OSM is visible when comparing Figs. 6.2 and 6.4.

Similar as for the one-dimensional problem, the different approaches to define $\mathbf{x}_{\mathbf{f}}{}_n^{(1)}$ are applied for the panel problem, the results of which can be found in Table 6.2. Again it is found that, especially when the size of Δt decreases, state extrapolation is the most efficient.

	$\Delta t = P/10$		$\Delta t = P/30$		$\Delta t = P/50$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	1.2 E-13	100	1.3 E-13	100	1.3 E-13
1: $\mathbf{x}_{\mathbf{f}}{}_n^{(1)} = \mathbf{x}_{\mathbf{f}}{}_{n-1}^{*}$	49.1	3.9 E-13	63.3	1.5 E-13	74.8	1.2 E-13
2: $\mathbf{x}_{\mathbf{f}}{}_n^{(1)} = \mathbf{x}_{\mathbf{f}}{}_{n-1}^{*} + \delta_n$	48.0	3.5 E-13	56.0	2.5 E-13	69.9	1.3 E-13
3: $\mathbf{x}_{\mathbf{f}}{}_n^{(1)} = \mathbf{x}_{\mathbf{c}}{}_n^{*}$	50.6	3.2 E-13	68.0	1.9 E-13	82.1	1.2 E-13
4: $\mathbf{x}_{\mathbf{f}}{}_n^{(1)} = \mathbf{x}_{\mathbf{s}}{}_{n-1}^{*}$	48.9	3.4 E-13	67.9	1.9 E-13	81.2	8.7 E-14
5: $\mathbf{x}_{\mathbf{f}}{}_n^{(1)} = \mathbf{x}_{\mathbf{s}}{}_{n-1}^{*} + \delta_n$	48.4	3.5 E-13	67.7	1.7 E-13	78.9	9.7 E-14
6: $\mathbf{x}_{\mathbf{f}}{}_n^{(1)} = \mathbf{x}_{\mathbf{c}}{}_n^{*} + \delta_n$	47.8	3.6 E-13	62.9	1.1 E-13	71.1	1.7 E-13

Table 6.2: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, OSM test case FSI-5.

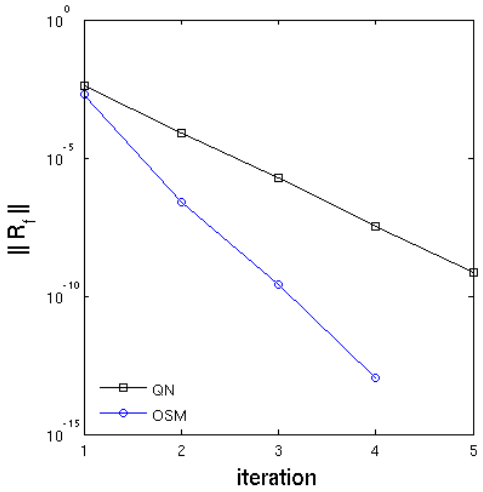
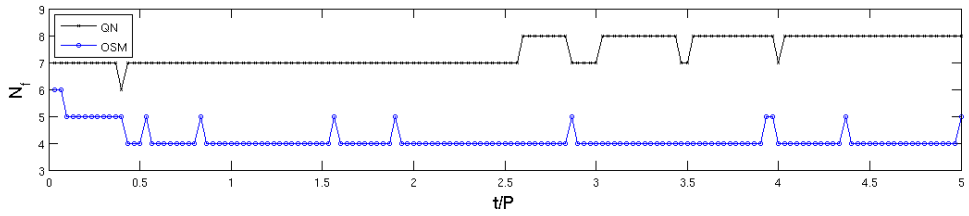
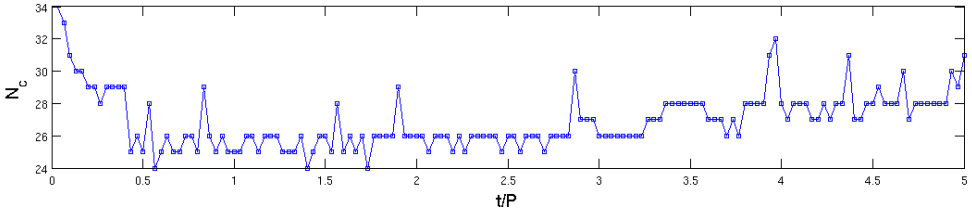


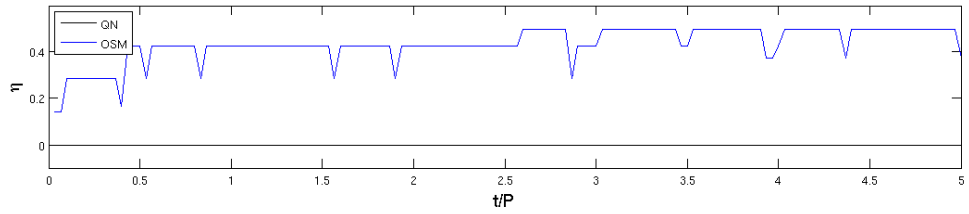
Figure 6.4: Convergence of the OSM method at $t = 2P$ for testcase FSI-5.



(a) Fine model evaluations



(b) Coarse model evaluations



(c) Relative efficiency w.r.t. Quasi-Newton

Figure 6.5: Computational cost for the 2D panel problem (test case FSI-5) applying OSM with $\Delta t = P/30$.

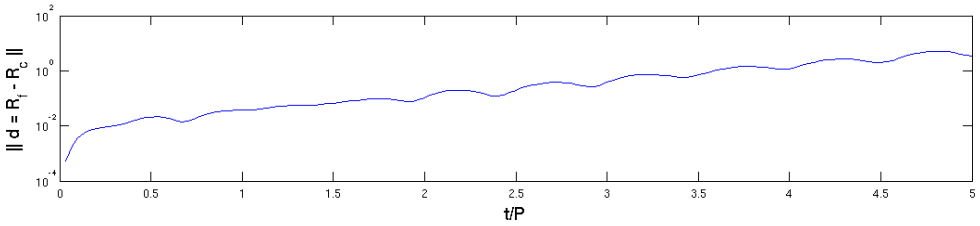


Figure 6.6: Evolution of the output parameter $\mathbf{d}$ for testcase FSI-5, $\Delta t = P/30$.

6.4 Conclusion

The output space mapping method is successfully applied as a coupling method to both a one-dimensional and a two-dimensional fluid-structure interaction problem. The method is able to obtain a fully coupled solution with the same accuracy as the Quasi-Newton method while requiring less fine model evaluations. The additional coarse cost is negligible, less than 1% of the cost of a fine evaluation per timestep for the panel problem and approximately 5% for the piston problem.

Whereas for the 1D piston problem OSM is less efficient than QN at the moments the pressure wave hits the structure, for the 2D panel problem large savings in computational cost are obtained at all time. Moreover, the efficiency of OSM even increases for the 2D problem when flutter occurs and the deficit between both models increases. The reason why OSM is so much more efficient for the 2D problem is because the output term $\mathbf{E}$ reduces to a finite difference approximation for one-dimensional problems, which has a large effect on the quality of the surrogate.

Overall it is found that OSM is able to solve the partitioned FSI problem, where the total efficiency increases with fluid-to-structure mass ratio and the size of the timestep. Moreover, OSM appears to perform best for multi-dimensional problems.

Chapter 7

Implicit space mapping (ISM)

7.1 The ISM algorithm

Another commonly used dual space mapping approach is implicit space mapping, introduced by Cheng et al. in 2004 [10; 13]. The key to ISM is the concept of preassigned parameters ($\mathbf{x_p}$). These are model parameters, different from the design parameters (the interface displacement), which are normally fixed during the optimization process but do have a significant influence on the optimal solution. By varying those physical parameters for the coarse model the flexibility of the surrogate model increases thereby allowing a better match with the fine model. For this case of dual space mapping the surrogate model looks like

$$\mathbf{R_s}^{(j)}(\mathbf{x}) = \mathbf{R_c}(\mathbf{x}, \mathbf{x_p}^{(j)}), \tag{7.1}$$

where the preassigned parameters $\mathbf{x_p}^{(j)}$ used to calibrate the coarse model are extracted from

$$\begin{aligned} \mathbf{x_p}^{(j)} = & arg \min_{\mathbf{x_p}} \left(\sum_{k=1}^{j} \omega_k \left\| \mathbf{R_f}(\mathbf{x_f}^{(j)}) - \mathbf{R_c}(\mathbf{x_f}^{(j)}, \mathbf{x_p}) \right\| \right. \\ & + \sum_{k=1}^{j} v_k \left\| \frac{\partial}{\partial \mathbf{x_f}} \left(\mathbf{R_f}(\mathbf{x_f}^{(j)}) - \mathbf{R_c}(\mathbf{x_f}^{(j)}, \mathbf{x_p}) \right) \right\| \\ & \left. + \omega \left\| \mathbf{x_p} - \mathbf{x_{p_0}} \right\| \right). \end{aligned} \tag{7.2}$$

Similar as for GSM one can choose between MPE, GPE, constrained parameter extraction or a suitable combination. In this case especially constrained PE might be useful since it is advised to keep the physical preassigned parameters as close as possible to their original values. This in order to prevent unphysical situations for which the coarse model fails.

In (7.2) the second term on the right hand side should be approximated. This is done using Broydens method, which is similar to (5.6) when denoting the deficit between the fine and surrogate model as $\mathbf{d}$.

The next estimate for the fine model optimal solution is now found in a dual fashion, so by optimizing the coarse model with the adapted preassigned parameters

$$\mathbf{x_f}^{(j+1)} = arg\min_{\mathbf{x}} \|\mathbf{R_s}^{(j)}(\mathbf{x})\| = arg\min_{\mathbf{x}} \|\mathbf{R_c}(\mathbf{x}, \mathbf{x_p}^{(j)})\|. \tag{7.3}$$

Similar to GSM, ISM requires two coarse model optimizations per iteration, thereby leading to a coarse computational cost which is at least twice as large as for ASM and OSM. Therefore the number of required fine model evaluations will have to be smaller than for the case of ASM/OSM in order to be efficient.

Implicit space mapping is a very simple optimization approach where the next iterate is found by updating (physical) coarse model parameters. It has been applied for several electromagnetic applications with great success. There it was found that it is especially useful in cases with severe misalignment between the fine and coarse models.

7.2 Selection of the preassigned parameters

Many successful examples of the application of ISM are found in literature. Therefore it seems worthwhile to investigate the use of it for fluid-structure interaction simulations. To do so, first a closer look is taken at the influence of the physical parameters on the coarse model solution in order to be able to identify appropriate preassigned parameters. For simplicity and to prevent additional computational work, parameters defining the domain and the spatial and temporal discretizations are not allowed to be changed. It is found that for the 1D piston problem only the (nondimensional) spring stiffness and piston mass

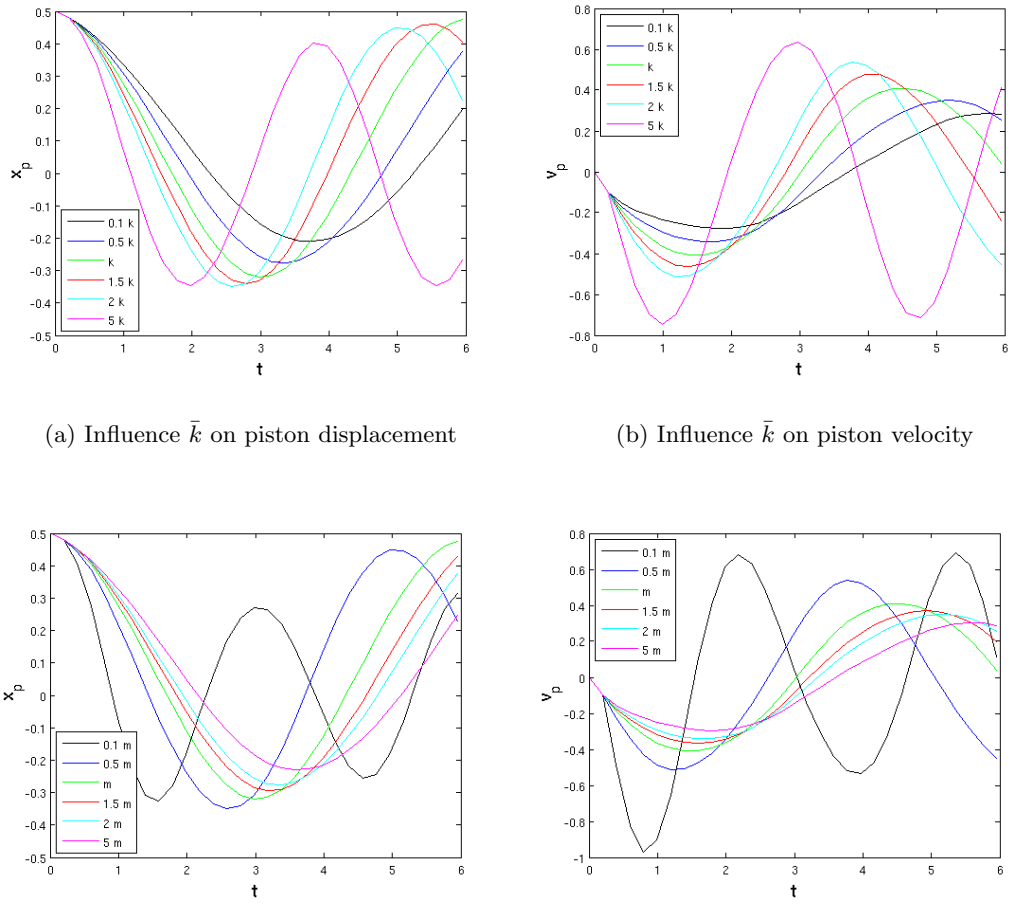
$$\bar{m} = \frac{m}{\rho_0 L} \quad , \quad \bar{k} = \frac{kL}{\rho_0 c_0^2} \tag{7.4}$$

are suited as preassigned parameters. Adapting those preassigned parameters corresponds to locally changing the fluid-to-structure mass ratio ζ, the natural frequency of the mass-spring system ω and the characteristic time-scale λ,

$$\zeta = \frac{\rho_0 L}{m} \quad , \quad \omega = \sqrt{\frac{k}{m}} \quad , \quad \lambda = \frac{L\omega}{c_0}. \tag{7.5}$$

The influence of these parameters on the coarse model solution is shown in Fig. 7.1. For every parameter the piston displacement and velocity are shown for different values, with reference to the standard (fine model) value of the particular parameter.

Similar as for the 1D piston problem, for the 2D panel test problem the mass and stiffness matrices are chosen as preassigned parameters. In order to avoid inconsistencies, those matrices are adapted by multiplying them with a scalar coefficient which equals one for the standard case. The influence of those coefficients (c_m for the mass matrix and c_k for the stiffness matrix) is shown in Fig. 7.2.



(a) Influence $\bar{k}$ on piston displacement

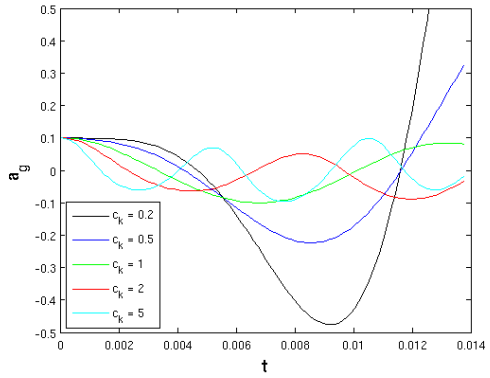
(b) Influence $\bar{k}$ on piston velocity

(c) Influence $\bar{m}$ on piston displacement

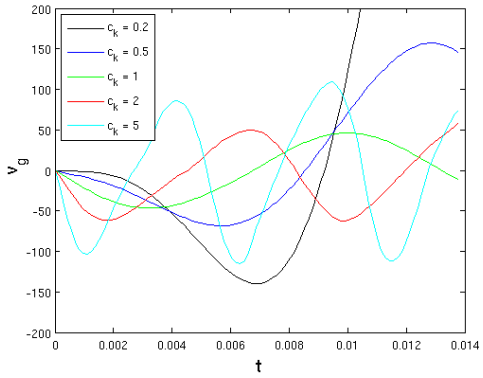
(d) Influence $\bar{m}$ on piston velocity

Figure 7.1: Influence of the nondimensional spring stiffness and piston mass for the coarse 1D piston problem, testcase FSI-2.

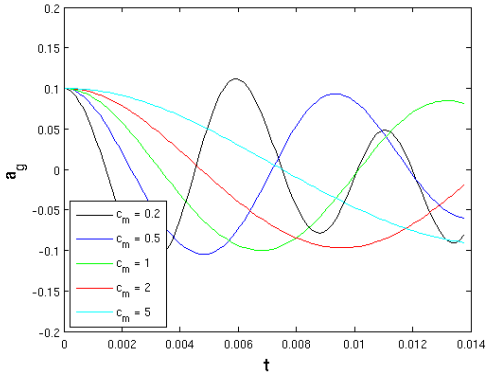
As expected, changing $\bar{m}$ and $\bar{k}$ (or equivalently for 2D c_k and c_m) has a large influence on both the frequency and amplitude of the final solution. However, because the preassigned parameters are physical, it is not possible to assign them every arbitrary value. To start with, they all should be positive. Moreover, they cannot be excessively small or large, as is clearly visible in Figs. 7.2(a) and 7.2(b) where flutter occurs much sooner for small values of c_k (so low panel stiffness). Which boundaries should be applied depends on the problem.



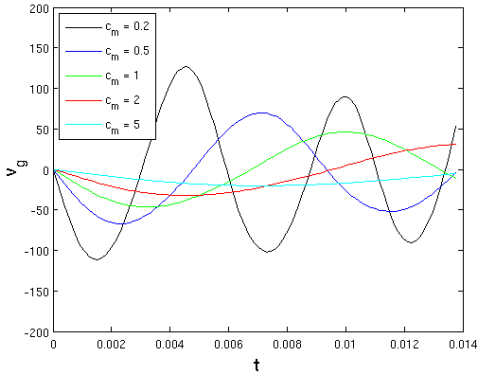
(a) Influence c_k on interface displacement



(b) Influence c_k on interface velocity



(c) Influence c_m on interface displacement



(d) Influence c_m on interface velocity

Figure 7.2: Influence of scaling the stiffness and mass matrices for the 2D panel flutter problem, testcase FSI-4.

7.3 ISM results

In this section the implicit space mapping method is tested by applying it to the one-dimensional piston problem with preassigned parameters $\bar{k}$ and $\bar{m}$ that alter the stiffness and mass of the structure. Just as for GSM, the space mapping method is tested with different parameter extraction approaches. Since the number of unknowns to be extracted during the parameter extraction process is two for this case, MPE is applied using the last two points and GPE is used in combination with SPE. The computational cost and the average partitioning error are gathered in Table 7.1 for testcase FSI-2.

When looking at the different parameter extraction approaches the observations are in line with the results for GSM (see Table 5.1). SPE performs best when constraints are added

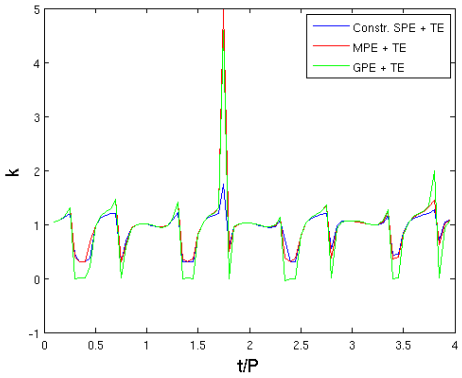
	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	6.4 E-16	100	2.2 E-16	100	2.3 E-15
Constr. SPE	160.2	1.1 E-13	151.5	1.5 E-13	159.1	1.1 E-13
Constr. SPE + TE	153.9	1.2 E-13	147.1	1.1 E-13	154.7	1.4 E-13
MPE	153.5	1.1 E-13	152.4	1.2 E-13	159.8	1.1 E-13
MPE + TE	161.3	9.1 E-14	152.1	1.2 E-13	159.6	1.0 E-13
GPE	164.8	1.0 E-13	154.8	1.5 E-13	163.2	1.1 E-13
GPE + TE	160.5	1.2 E-13	156.4	1.2 E-13	157.4	1.2 E-13

Table 7.1: Relative computational cost and corresponding accuracy for different approaches to perform the parameter extraction, ISM testcase FSI-2.

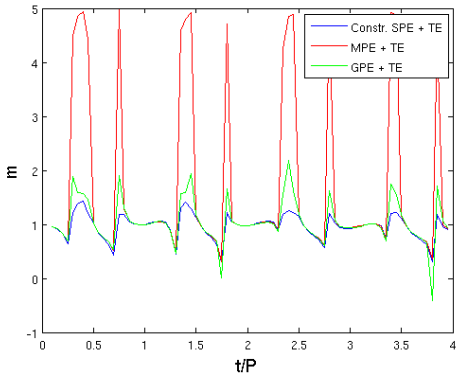
to keep the preassigned parameters as close as possible to their nominal values and in order to prevent an underdetermined system, whereas for MPE this yields an adverse effect. For GPE the influence of constraints during the first iteration is negligible. Furthermore, is was found for GSM that the addition of a Taylor series expansion for the preassigned parameters during the surrogate optimization has a positive effect on the convergence of the algorithm. However, for ISM the benefits are much smaller. Only for SPE clear benefits are observed when applying TE. It is believed that this is due to the influence of the constraints on the result of the parameter extraction. This causes the changes in preassigned parameters to be smoother than for MPE for example, such that the Broyden approximation of their Jacobians is more accurate. The obtained values for $\bar{k}$ and $\bar{m}$ at the end of every timestep are shown in Fig. 7.3 together with the final matching error. Especially the values for the nondimensional mass appear to be smoother for SPE than for MPE. Furthermore, the final matching error is large, comparable to the error when no ISM is applied (see Fig. 6.3). This indicates that although the nondimensional mass and stiffness have a large influence on the behaviour of the piston, they do not alter the solution considerably from one timestep/iteration to the next. This causes the differences in performance between the applied parameter extraction approaches to be small.

Since the preassigned parameters do not succeed in aligning both models, the performance of implicit space mapping is rather bad. From the results shown in Table 7.1 and Fig. 7.4 it is seen that Quasi-Newton is much more efficient than ISM. ISM requires at some points almost twice the number of iterations required by QN. Also the coarse cost necessary to perform the parameter extraction is high, adding approximately 30% of the cost of one fine model evaluation every timestep. This therefore results in an average computational cost which is about 50% higher than for QN, while the accuracy of the final solution is up to three orders of magnitude lower.

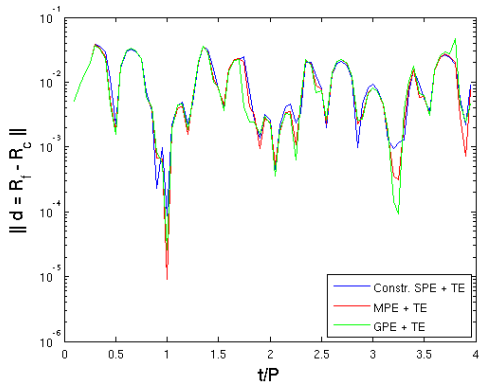
It is concluded that, although ISM is able to obtain a fully coupled solution for the piston problem, its performance is not good enough to be an alternative for conventional coupling methods. In this case ISM is not able to obtain a surrogate model which sufficiently resembles the fine model. Moreover, the attempt to do so is rather expensive, resulting in a computational cost which is much higher than for the Quasi-Newton method used as reference.



(a) Variation of nondimensional stiffness $\bar{k}$.

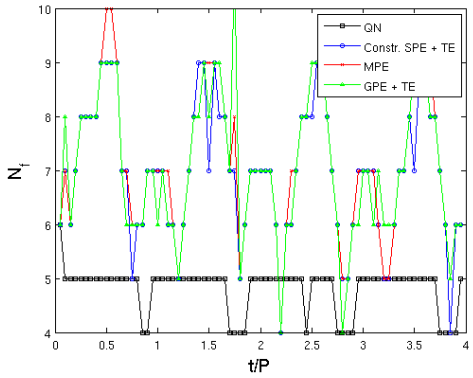


(b) Variation of nondimensional mass $\bar{m}$.

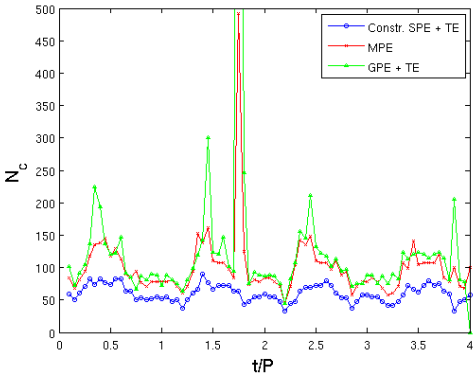


(c) Final matching error $\mathbf{d}$

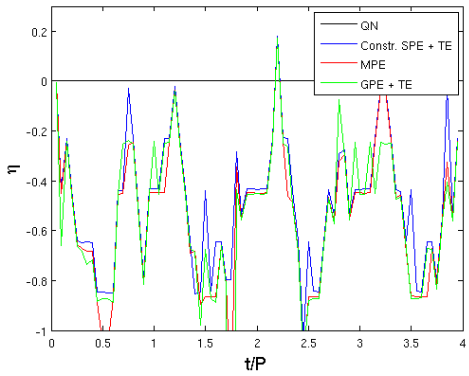
Figure 7.3: Variation of the preassigned parameters and the resulting matching error between the fine and coarse model for ISM, FSI-2 with $\Delta t = P/20$.



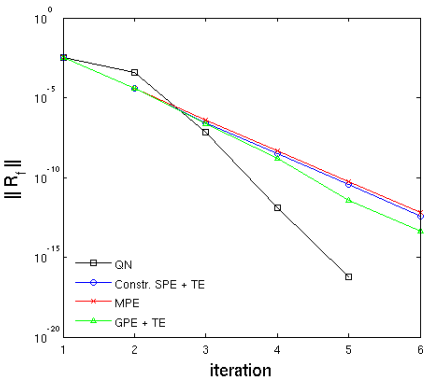
(a) Number of fine model evaluations



(b) Number of coarse model evaluations



(c) Relative efficiency w.r.t. Quasi-Newton



(d) Convergence at $t = 4P$

Figure 7.4: Results for the 1D piston problem (testcase FSI-2 with $\Delta t = P/20$), applying ISM with different parameter extraction approaches.

7.4 Combination of ISM with other mappings

Implicit space mapping can be used by itself, but often it is combined with another form of space mapping, thereby allowing for a surrogate model with a larger flexibility. For example, one can combine ISM with an output mapping as identified in section 6.1. The resulting surrogate then looks like

$$\mathbf{R_s}^{(j)}(\mathbf{x}) = \mathbf{R_c}(\mathbf{x}, \mathbf{x_p}^{(j)}) + \mathbf{O}^{(j)}(\mathbf{x}). \tag{7.6}$$

Due to the output mapping a locally perfect match with the fine model can be achieved, while the computational cost stays approximately the same as for standard ISM. The output term $\mathbf{O}(\mathbf{x})$ does also improve the extrapolation properties of the surrogate, as seen in Chapter 6.

Another example consists of combining ISM with GSM, leading to the so-called Generalized Implicit Space mapping framework GISM [18]

$$\mathbf{R_s}^{(j)}(\mathbf{x}) = \mathbf{A}^{(j)} \cdot \mathbf{R_c}(\mathbf{B}^{(j)} \cdot \mathbf{x} + \mathbf{c}^{(j)}, \mathbf{x_p}^{(j)}) + \mathbf{d}^{(j)} + \mathbf{E}^{(j)} \cdot (\mathbf{x} - \mathbf{x_f}^{(j)}). \tag{7.7}$$

This approach leads to an even higher flexibility of the surrogate model than when only an output mapping is added, and can increase the region of applicability considerably. However, the number of extractable parameters increases (from 2 to 5 for the piston problem) which causes a large increment in the coarse computational cost per iteration.

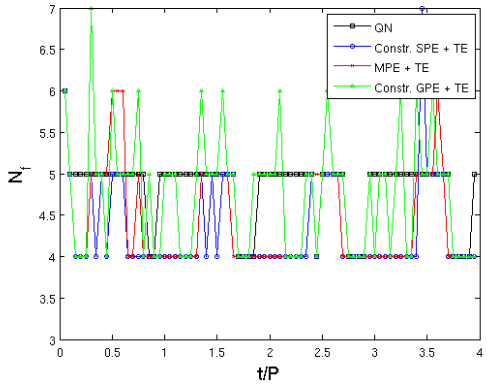
7.4.1 ISM with OSM results

Since implicit space mapping in itself does not work well for the 1D problem due to the limited influence of $\bar{k}$ and $\bar{m}$ on the solution, it is investigated whether or not the combination of ISM with an output mapping (so using a surrogate model as defined by (7.6)) improves the results. Note that for this case constrained GPE is used instead of normal GPE since that approach is found to be more efficient here, although the difference is small.

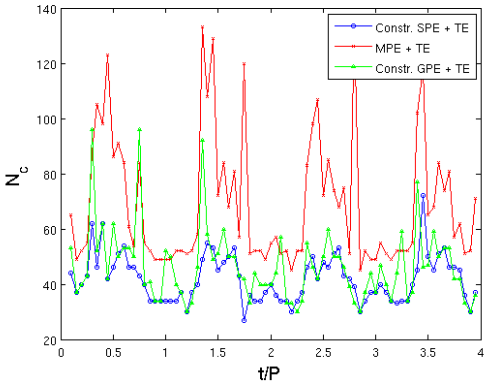
From Fig. 7.5 and Table 7.2 it is seen that the application of the output mapping improves the performance of ISM a lot. The number of required fine model evaluations has decreased significantly, and therefore also has the coarse cost. This causes the combination of implicit and output space mapping to be more efficient than QN at some moments. However, the overall gains in computational cost are small or even non-existing, while the accuracy is lower than for QN. Moreover, looking back to the performance of OSM only in Section 6.2 it is observed that although OSM has a positive influence on ISM, the addition of ISM to standard OSM lowers the efficiency.

	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\|\mathbf{R_f}\|\|_{av}$	Cost [%]	$\|\|\mathbf{R_f}\|\|_{av}$	Cost [%]	$\|\|\mathbf{R_f}\|\|_{av}$
QN	100	6.4 E-16	100	2.2 E-16	100	2.3 E-15
Constr. SPE	104.8	4.5 E-15	99.2	2.4 E-15	103.6	3.4 E-15
Constr. SPE + TE	98.6	1.2 E-15	91.9	4.0 E-15	101.8	8.1 E-16
MPE	106.9	2.1 E-15	104.2	2.1 E-15	106.6	4.3 E-15
MPE + TE	104.8	7.3 E-16	96.6	4.8 E-15	104.8	8.1 E-16
Constr. GPE	111.3	2.5 E-15	103.1	2.1 E-15	106.1	2.7 E-15
Constr. GPE + TE	109.8	8.2 E-16	101.5	1.8 E-15	106.4	6.4 E-16

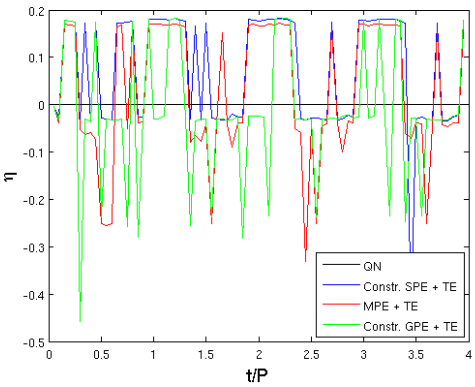
Table 7.2: Relative computational cost and corresponding accuracy for ISM with OSM, FSI-2.



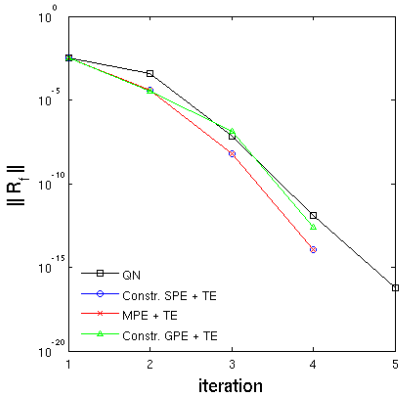
(a) Fine model evaluations



(b) Coarse model evaluations

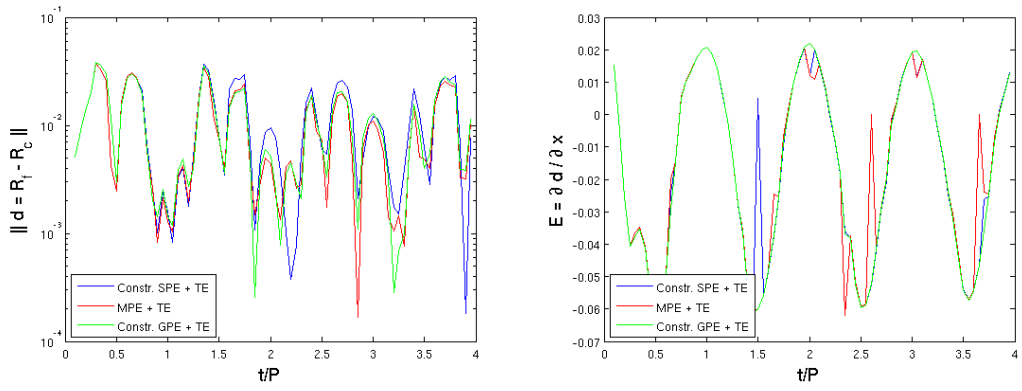


(c) Relative efficiency w.r.t. Quasi-Newton



(d) Convergence at $t = 4P$

Figure 7.5: Results for the 1D piston problem (testcase FSI-2 with $\Delta t = P/20$), applying ISM in combination with OSM.



(a) (b)

Figure 7.6: Variation of the output output mapping parameters for the combination of ISM and OSM, testcase FSI-2 with $\Delta t = P/20$

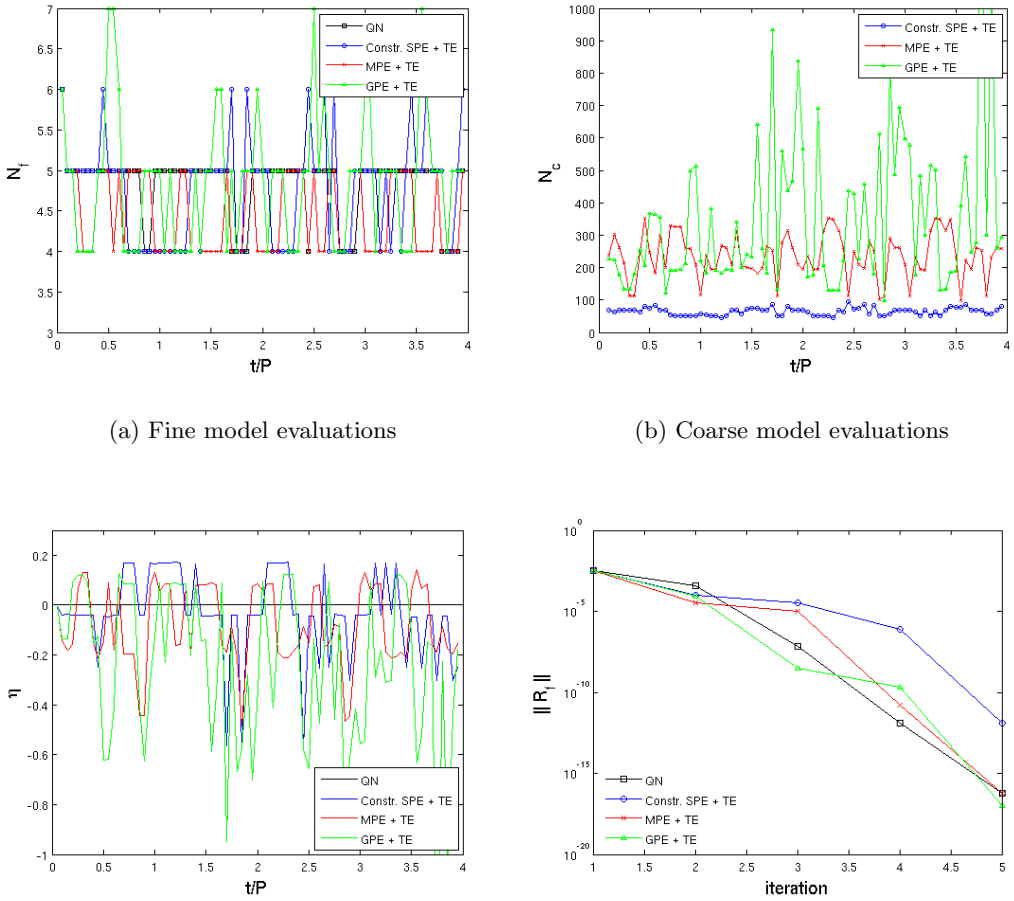
7.4.2 ISM with GSM results

In this section the results for the combination of ISM and GSM applied to the piston problem are presented. Note that due to the presence of GSM the amount of extractable mapping parameters is increased to 5. Therefore 5 points are taken into account for MPE and GPE is combined with 4-point MPE.

Just as for the combination with OSM, GISM succeeds in lowering the required number of fine model evaluations per timestep. However, N_f is not lowered enough to be below the level for QN. Especially when taking into account the coarse computational cost required to build the surrogate, GISM is less efficient. The coarse cost is very large in this case since the amount of extractable parameters is high, making the PE process expensive.

	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	6.4 E-16	100	2.2 E-16	100	2.3 E-15
Constr. SPE	115.0	5.4 E-15	107.1	3.9 E-15	109.9	6.9 E-15
Constr. SPE + TE	103.1	8.5 E-16	102.7	2.2 E-15	109.7	1.4 E-15
MPE	126.2	1.1 E-16	121.5	9.6 E-16	143.9	2.9 E-16
MPE + TE	110.7	3.3 E-16	107.4	1.4 E-15	109.6	3.7 E-16
GPE	145.2	2.3 E-14	143.3	7.1 E-14	169.4	4.1 E-14
GPE + TE	135.8	6.6 E-14	122.5	1.0 E-13	142.4	6.5 E-14

Table 7.3: Relative computational cost and corresponding accuracy for different approaches to perform the parameter extraction, ISM with GSM testcase FSI-2.



(a) Fine model evaluations

(b) Coarse model evaluations

(c) Relative efficiency w.r.t. Quasi-Newton

(d) Convergence at $t = 4P$

Figure 7.7: Results for the 1D piston problem applying GISM for FSI-2, $\Delta t = P/20$.

When looking at the parameter values which determine the surrogate model in the case of GISM, see Fig. 7.8, it is seen that $\bar{k}$ and $\bar{m}$ barely change. The local alignment of the surrogate and fine model is accomplished mainly by the GSM parameters **A**, **B** and **c**. The presence of the nondimensional mass and stiffness as preassigned parameters only causes an increase in coarse computational cost while their effect on the final surrogate is minimal. It is found that the addition of GSM to ISM is beneficial compared to standard ISM, but GSM without ISM performs better (see Table 5.1).

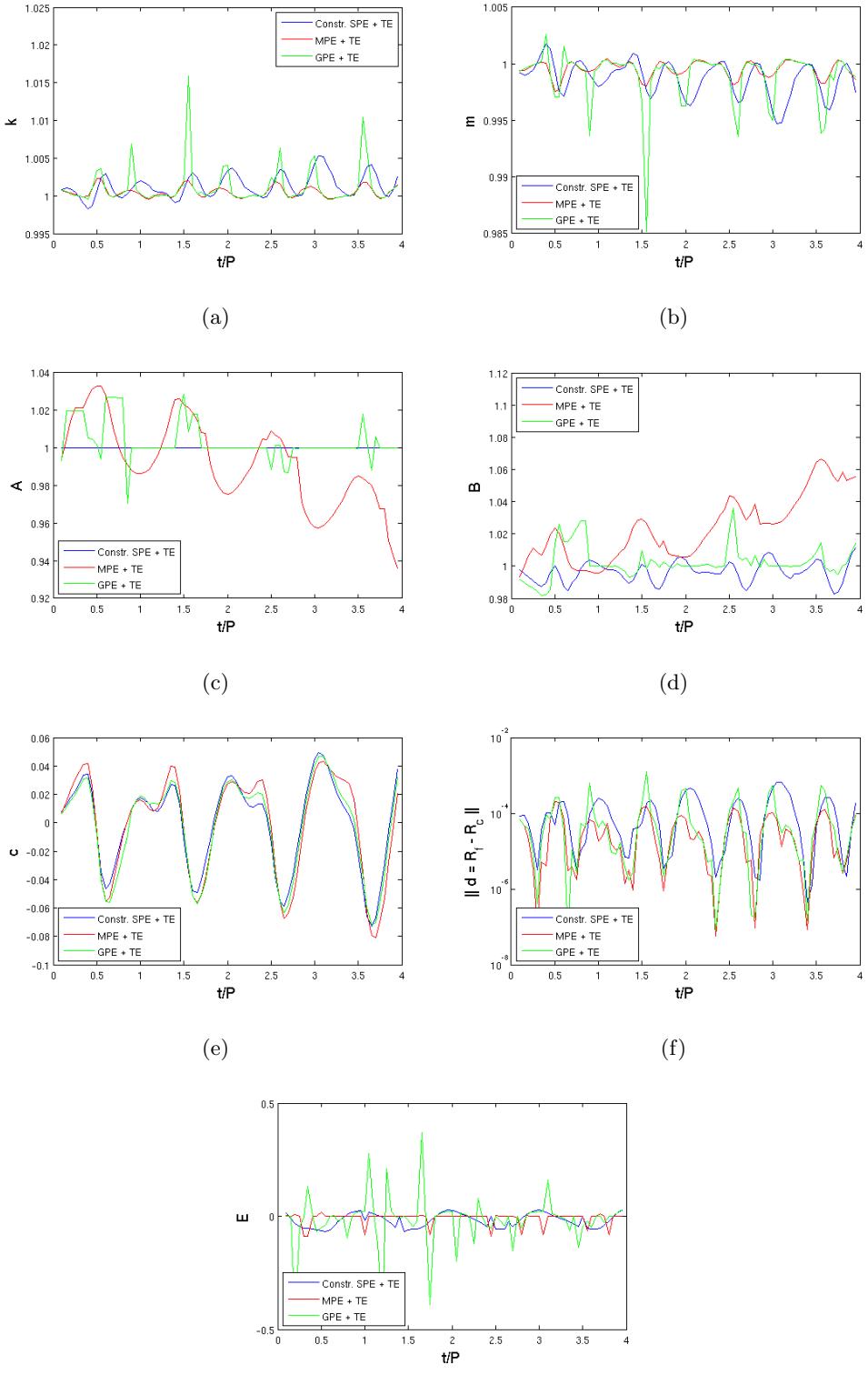


Figure 7.8: Variation of the mapping parameters for GISM, testcase FSI-2 with $\Delta t = P/20$.

7.5 ISM for the 2D panel problem

For the 2D panel problem a stiffness-coefficient c_k and a mass-coefficient c_m are used as preassigned parameters for the application of ISM. However, it is found that their influence is not large enough to obtain a sufficient match between both models. Whereas for the one-dimensional problem ISM did still converge, this is not the case for the panel problem and no solution applying only ISM can be obtained.

The only way in which ISM works for the panel problem is when it is combined with OSM or GSM. However, as found for the piston problem, the reason for this is just the good performance of the latter two space mapping methods. The combination with ISM does not yield any benefits those methods. Therefore no 2D results are presented for ISM.

7.6 Conclusion

ISM does work but is not efficient. Combining ISM with OSM or GSM improves the efficiency a lot. However, when comparing to OSM and GSM without ISM one notices that the addition of those extra model parameters does not yield an improved match. The number of fine model iterations does not reduce and since two extra parameters have to be extracted the coarse cost increases. So ISM on itself is rather inefficient and it does not improve the efficiency when adding it to other methods.

The way ISM was programmed in this work required access to the structure solver, thereby violating the black box approach. In theory it would be possible to apply ISM to black box solvers, however, that would require rebuilding the system matrices every coarse iteration thus increasing the computational cost dramatically.

Overall it is concluded that ISM does not work particularly well for FSI problems. Moreover, since it is hard to implement for black box solvers it does not satisfy the FFAST requirements. Therefore this method will not be considered anymore in the remainder of this work.

Chapter 8

Comparison of the space mapping methods

In this section the previously discussed space mapping algorithms which yield the best results are compared to each other and to the conventional Quasi-Newton method. It is investigated how well every method performs in obtaining strongly coupled partitioned fluid-structure interaction solutions. The methods which are considered here are:

- ASM: Aggressive space mapping
- GSM: Generalized space mapping (using 3-point MPE for the piston problem and constrained SPE with a Taylor series expansion for the 2D panel problem)
- OSM: Output space mapping
- QN: Quasi Newton

Approach 2 (see equation (3.29)) is used to determine the initial estimate for each timestep since this approach was found to yield the best results for longer periods of time. For both the 1D piston problem and the 2D panel problem three testcases with variable fluid-to-structure mass ratios are considered (see Chapter 2) for both small and large timesteps.

8.1 1D piston problem

Table 8.1 contains the computational cost (as fraction of the cost required for the Quasi-Newton method) and the average partitioning error at the end of each timestep for the three space mapping methods that were successfully implemented. Testcases FSI-1, FSI-2 and FSI-3 are considered for $\Delta t = P/20$ and $\Delta t = P/200$, where the latter timestep size is interesting since the amount of numerical diffusion is small and thus the effect of the pressure wave is visible (see Figs. 2.2, 2.3 and 2.4).

Testcase	Method	$\Delta t = P/20$		$\Delta t = P/200$	
		Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
FSI-1	QN	100	1.3 E-15	100	6.5 E-16
	ASM	89.6	1.5 E-15	98.4	1.2 E-16
	GSM	96.9	3.7 E-15	132.1	4.8 E-17
	OSM	92.7	2.1 E-15	99.6	3.6 E-16
FSI-2	QN	100	2.2 E-16	100	1.1 E-15
	ASM	85.5	2.8 E-15	98.4	1.8 E-16
	GSM	92.5	6.6 E-15	131.3	5.5 E-17
	OSM	89.3	2.5 E-15	99.3	5.6 E-16
FSI-3	QN	100	2.4 E-16	100	5.5 E-15
	ASM	84.4	6.5 E-15	97.5	8.9 E-16
	GSM	92.1	1.1 E-14	128.2	1.1 E-16
	OSM	88.7	4.9 E-15	99.2	3.1 E-15

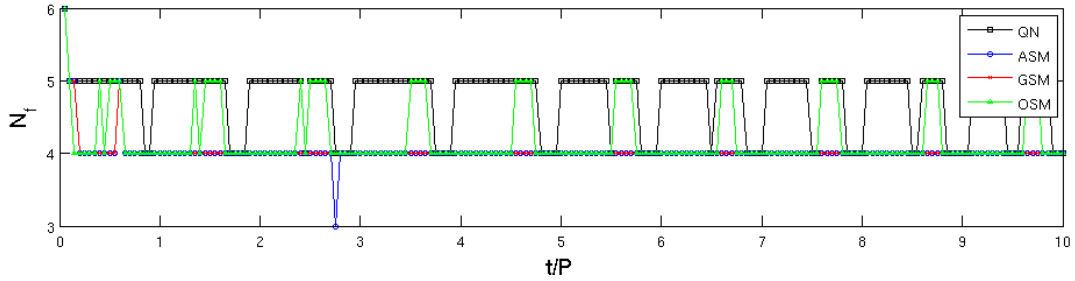
Table 8.1: Relative computational cost and average final accuracy for different space mapping methods for the 1D piston problem.

First of all it is observed that ASM proves to be the most efficient coupling method for all cases, closely followed by OSM. The efficiency gains with respect to the Quasi-Newton method are largest for strongly coupled problems which are solved using large timesteps, making space mapping the method of choice for those FSI problems. However, the space mapping efficiency decreases with the size of the applied timestep. This is especially the case for GSM, where the method becomes much slower than QN for $\Delta t = P/200$. So it seems like for one-dimensional problems which require a high temporal resolution space mapping does not yield noticeable improvements and in particular the use of GSM is not advised.

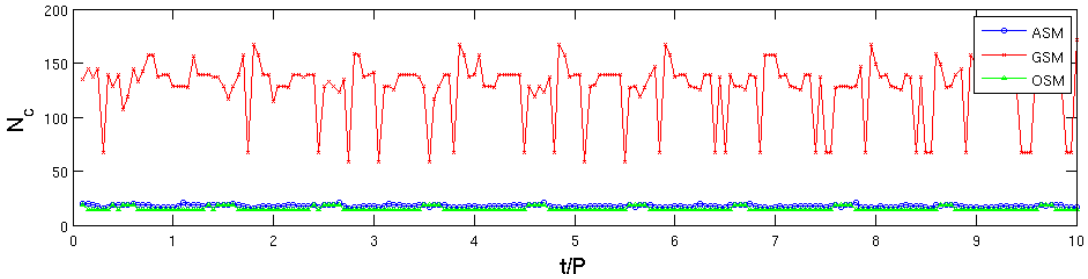
Looking at the composition of the computational cost for $\Delta t = P/20$ if Fig. 8.1, it is observed that QN requires the most fine model evaluations each timestep to obtain the fully coupled solution whereas ASM and GSM require the least. The difference in performance between the latter two space mapping methods is caused by the coarse cost necessary to lower the number of fine model evaluations. This cost is much larger for GSM (approximately 10% of the total cost) than for both ASM and OSM (less than 1.5%). This makes that overall ASM is most efficient and OSM performs better than GSM although more fine model evaluations are required. However, if a coarse model evaluation would be made cheaper, it is likely that GSM would be more efficient than OSM for this case. This is also confirmed by the rate of convergence shown in Fig. 8.3 where the curve for GSM lies just below the curve for OSM.

For the piston problem with a high temporal resolution the graphs in Fig. 8.2 look completely different. Now no space mapping method is able to reduce the required number of fine model evaluations significantly. GSM even needs more iterations than the other methods, while its coarse cost is still high. The coarse cost of ASM and OSM also causes those methods to be less efficient than QN at most moments. Only when a pressure wave hits the piston a peak in efficiency is observed for space mapping, making the overall performance of ASM and OSM still a little better than QN.

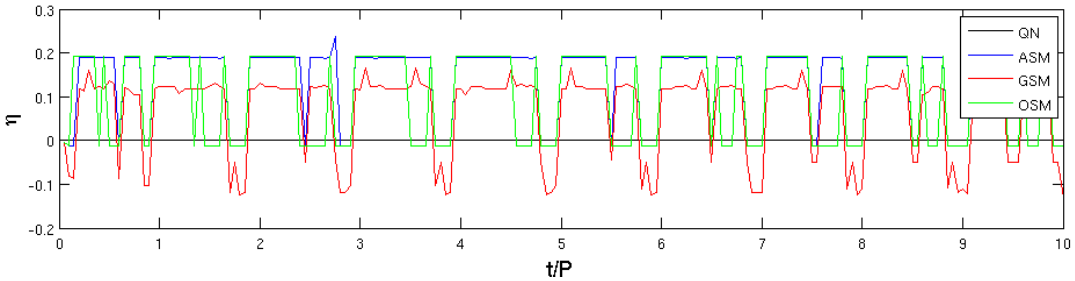
The disappointing performance of space mapping for problems that are solved using very small timesteps is not a surprise. ASM, GSM and OSM all make use of available input-output information of both the fine and coarse model to build a surrogate that approximates the fine model sufficient (at least locally) to predict its fully coupled interface displacement. So when the timestep is that small such that already the first estimate is close to the final solution and thus few iterations are required, little can be improved. In those cases the surrogate is not accurate enough yet to yield huge improvements. For GSM this even slows down the algorithm since the next iterate is determined using a surrogate that is not accurate enough yet. This also explains why space mapping is more efficient for strongly coupled FSI problems. For those cases more iterations are required to solve the problem, thus enough information is available to build an accurate surrogate model and still save some iterations.



(a) Fine model evaluations

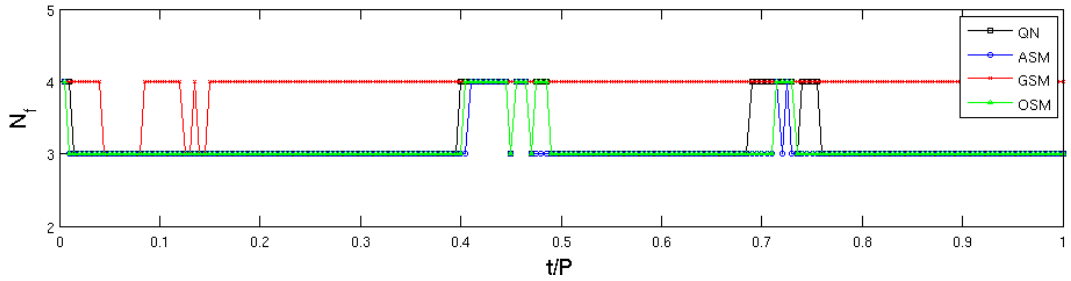


(b) Coarse model evaluations

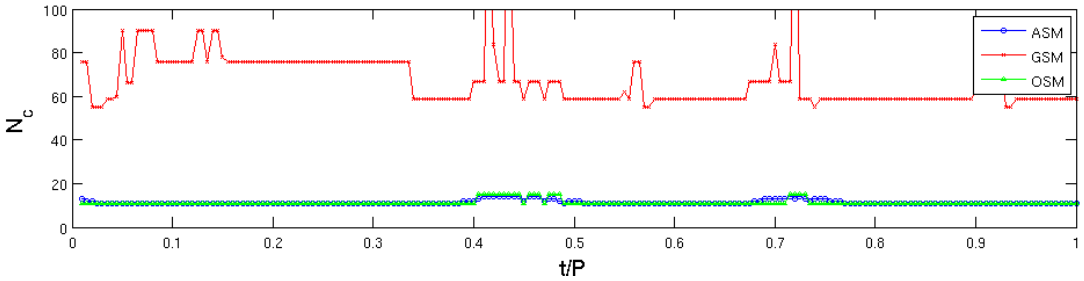


(c) Relative efficiency w.r.t. Quasi-Newton

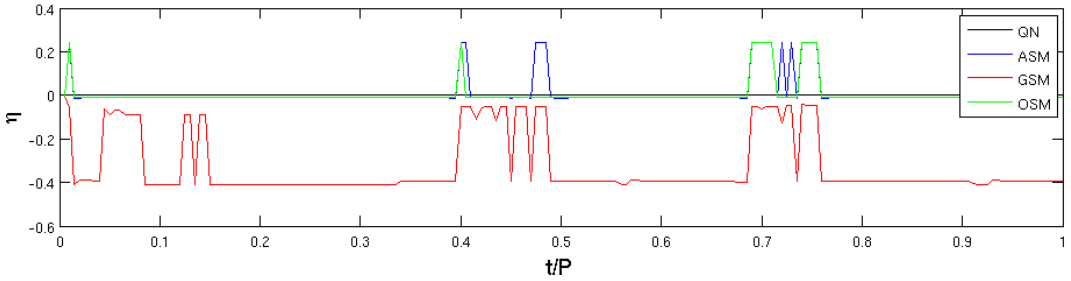
Figure 8.1: Computational cost of the space mapping methods for the 1D piston problem, test-case FSI-2 with $\Delta t = P/20$.



(a) Fine model evaluations

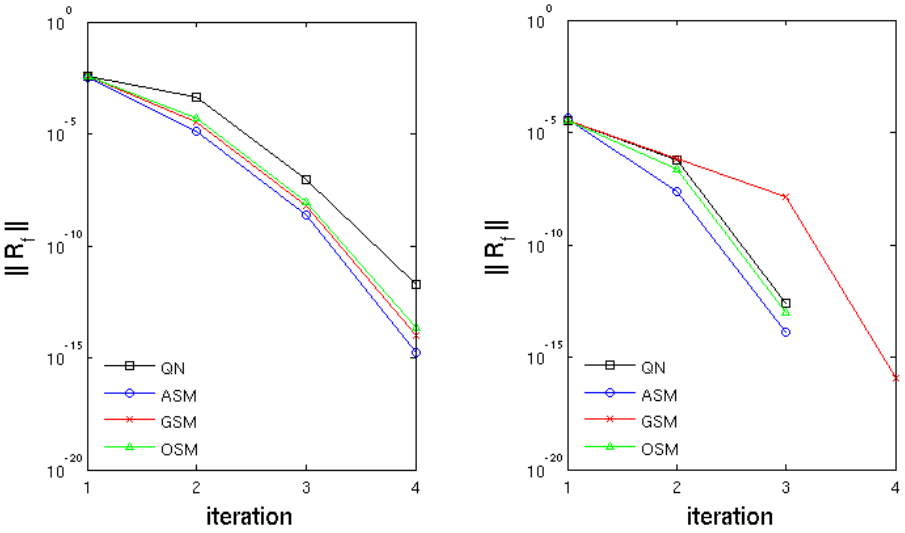


(b) Coarse model evaluations



(c) Relative efficiency w.r.t. Quasi-Newton

Figure 8.2: Computational cost of the space mapping methods for the 1D piston problem, test-case FSI-2 with $\Delta t = P/200$.



(a) $t = 5P$, $\Delta t = P/20$ (b) $t = 0.5P$, $\Delta t = P/200$

Figure 8.3: Convergence of the space mapping methods for the 1D piston problem, testcase FSI-2.

8.2 2D panel problem

The computational cost and average partitioning error of the different space mapping methods for the two-dimensional panel problem are shown in Table 8.2. Results for different coupling strengths and different temporal resolutions are obtained. It is observed that all methods perform well, especially for strongly coupled problems with large timesteps. OSM is by far the most efficient method, its computational cost is less than half of the cost required for QN for large timesteps. When the timestep decreases its efficiency decreases as well, but still savings of more than 30% are observed. As seen from Fig. 8.4 ASM and GSM have approximately the same rate of convergence, in between OSM and QN. The difference in efficiency for those methods is caused by the coarse cost. The parameter extraction process is very expensive for GSM and rather cheap for ASM. Moreover, ASM does not require an additional surrogate optimization but obtains the next estimate for the fine model solution by approximating the inverse of the mapping.

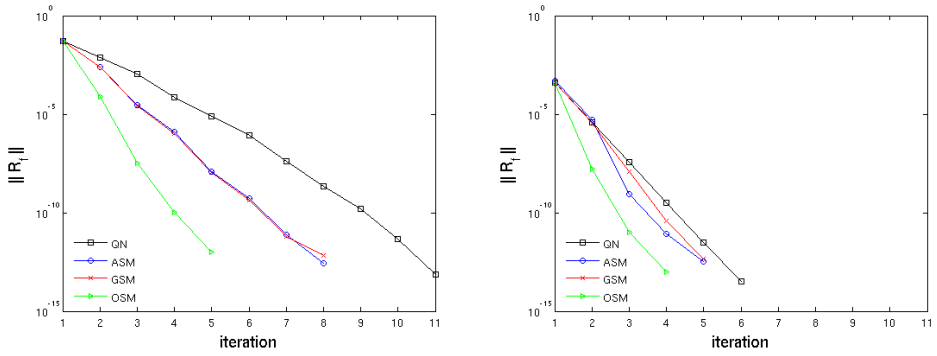
The number of fine and coarse model evaluations together with the relative efficiency with respect to QN is shown in Fig. 8.5 for $\Delta t = P/10$ and Fig. 8.6 for $\Delta t = P/50$. The most salient detail is the performance of GSM. For small deviations between the fine and the coarse model its required number of iterations is almost the same as for ASM. Its coarse cost however is excessive, approximately 4000 evaluations which is equivalent to 1.2 fine

evaluations, resulting in a lower efficiency. However, after a few periods GSM is not able o perform the coupling anymore and its cost drastically increases until the algorithm fails. The other coupling methods (QN, ASM and OSM) on the other hand are not significantly influenced by the occurrence of flutter, keeping the efficiency relatively constant.

Note that space mapping appears to be more efficient for two-dimensional than for one-dimensional problems. Two reasons are identified. First of all, the Broyden Jacobian approximation which is used several times in the different space mapping methods reduces to a first order finite difference approximation in 1D. Only information of the last two iterations is taken into account, which obviously lowers the efficiency considerably. This especially influences OSM, since there the surrogate is determined to a large extend by a Broyden approximation. Also the effectiveness of the addition of a Taylor series expansion for the mapping parameters lowers significantly for 1D. So space mapping, and especially OSM, cannot be applied in its optimal form for one-dimensional problems. A second reason for the higher space mapping efficiencies in 2D consists of the required number of iterations. Two-dimensional problems are generally more complex, hence they usually require more subiterations in order to obtain a fully coupled solution than 1D problems. Since space mapping requires some iterations for initialization before it is able to perform optimal, this means that more room for improvements is left for 2D problems with low temporal resolutions. The latter also explains the higher efficiencies for problems with large fluid-to-structure mass ratios.

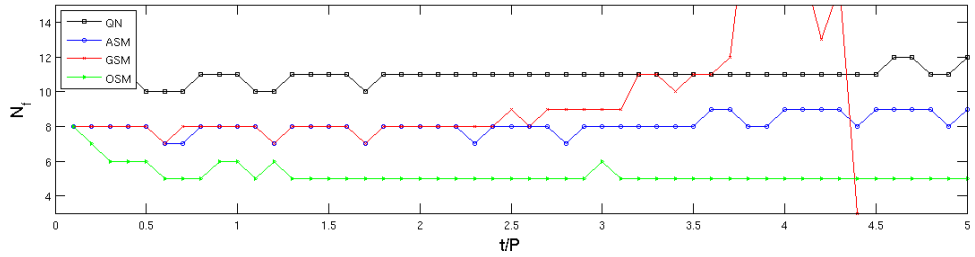
Testcase	Method	$\Delta t = P/10$		$\Delta t = P/50$	
		Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
FSI-4	QN	100	1.8 E-13	100	2.9 E-13
	ASM	71.7	4.5 E-13	96.9	8.6 E-14
	GSM	84.1	4.8 E-13	108.5	3.5 E-14
	OSM	51.6	2.7 E-13	70.5	9.3 E-14
FSI-5	QN	100	9.7 E-14	100	1.1 E-13
	ASM	74.5	2.3 E-13	88.6	2.0 E-13
	GSM	86.6	2.9 E-13	103.2	6.2 E-14
	OSM	48.0	3.5 E-13	69.9	1.3 E-13
FSI-6	QN	100	2.3 E-13	100	8.0 E-14
	ASM	66.9	3.4 E-13	84.5	2.7 E-13
	GSM	84.6	5.4 E-13	100.4	2.0 E-13
	OSM	44.7	3.3 E-13	68.9	3.5 E-13

Table 8.2: Relative computational cost and average final accuracy for different space mapping methods for the 1D piston problem.

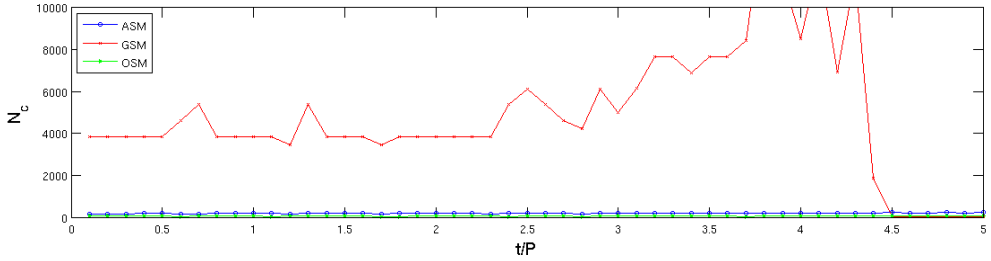


(a) $t = 2P$, $\Delta t = P/10$ (b) $t = 2P$, $\Delta t = P/50$

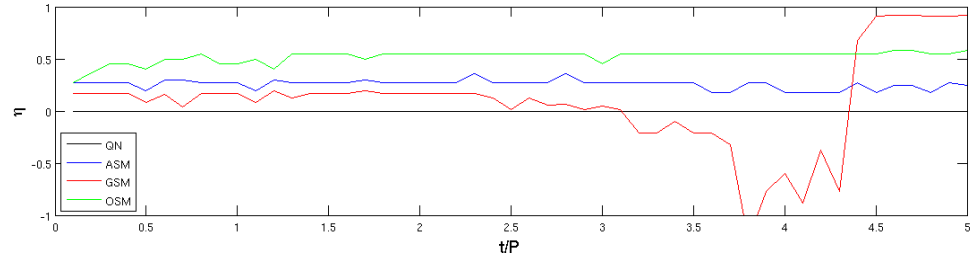
Figure 8.4: Convergence of the space mapping methods for the 2D panel problem, testcase FSI-5.



(a) Fine model evaluations

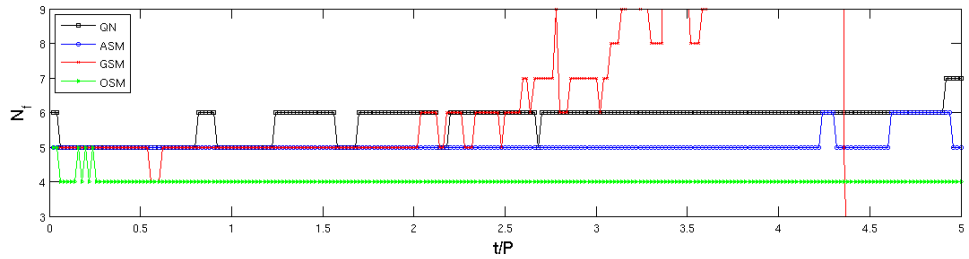


(b) Coarse model evaluations

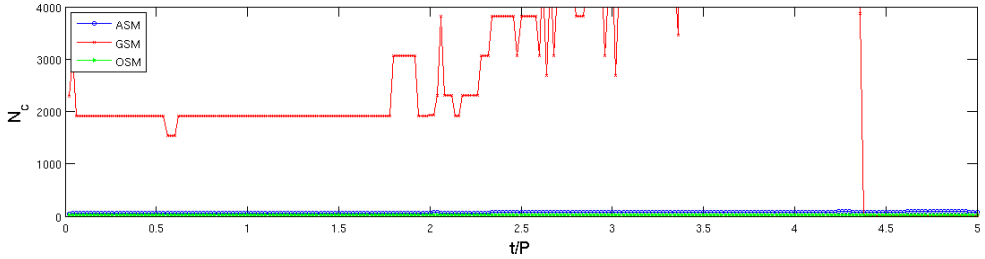


(c) Relative efficiency w.r.t. Quasi-Newton

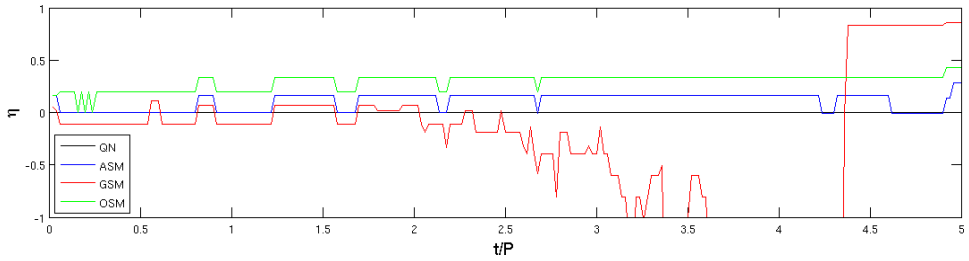
Figure 8.5: Computational cost of the space mapping methods for the 2D panel problem, testcase FSI-5 with $\Delta t = P/10$.



(a) Fine model evaluations



(b) Coarse model evaluations



(c) Relative efficiency w.r.t. Quasi-Newton

Figure 8.6: Computational cost of the space mapping methods for the 2D panel problem, testcase FSI-5 with $\Delta t = P/50$.

8.3 Conclusion

In this chapter the different space mapping algorithms that were discussed previously are compared to each other and the reference method, Quasi-Newton. It is found that space mapping is effective in lowering the computational cost in order to obtain a fully coupled solution for the 1D piston problem and especially for the 2D panel problem. Overall, output space mapping is found to show the best performance, closely followed by aggressive space mapping. Especially for 2D problems, where space mapping can be applied in its optimal form, the gains are large: OSM proves to be twice as efficient as QN. Generalized space mapping on the other hand has a convergence rate which is comparable to ASM, but fails to be efficient due to its huge parameter extraction cost.

Chapter 9

Validation

In the previous chapters several space mapping methods are discussed and applied to both a one- and a two-dimensional test problem. From the comparison of those different approaches in Chapter 8 it is found that for the 1D piston problem both aggressive space mapping and output space mapping yield the best results. For the 2D panel problem output space mapping performs the best. This analysis is performed for the fine-coarse model combinations as presented in Chapter 2. However, the question arises whether this good performance is a general property of ASM and OSM or whether it depends on the choice of this specific fine-coarse model combination. Therefore in the current chapter the general applicability of output space mapping is investigated by applying this method for different fine-coarse model combinations. This validation is performed for OSM because this method yielded by far the best performance for the 2D problem while the difference with ASM was small for the piston problem. Moreover, OSM is a new method which is very simple to implement and is therefore believed to be a very promising alternative for the standard coupling methods.

9.1 Fine-coarse model combinations for the piston problem

As seen from Section 2.1, the structural dynamics are much cheaper to solve than the governing equations for the fluid. So the overall cost of the FSI problem mainly depends on the fluid model. Therefore only the governing fluid equations will be varied when defining different fine and coarse models, and the structural dynamics remain the same. For the one-dimensional piston problem, OSM will be applied to the following combinations of fine and coarse models:

1A. **Fine:** Nonlinear Euler equations on a fixed grid,
Coarse: Linearized equations on the same grid.

1B. **Fine:** Nonlinear Euler equations on a fixed grid,
Coarse: Nonlinear Euler equations on a fixed but coarser grid.

2A. **Fine:** Nonlinear Euler equations on a deforming domain,
Coarse: Linearized equations on a fixed grid.

2B. **Fine:** Nonlinear Euler equations on a deforming domain,
Coarse: Nonlinear Euler equations on a fixed but coarser grid.

The first fine-coarse model combination is the same as used before and which is described in detail in section 2.1. The other combinations are discussed below.

9.1.1 Coarse grid coarse model for the nonlinear piston problem on a fixed domain

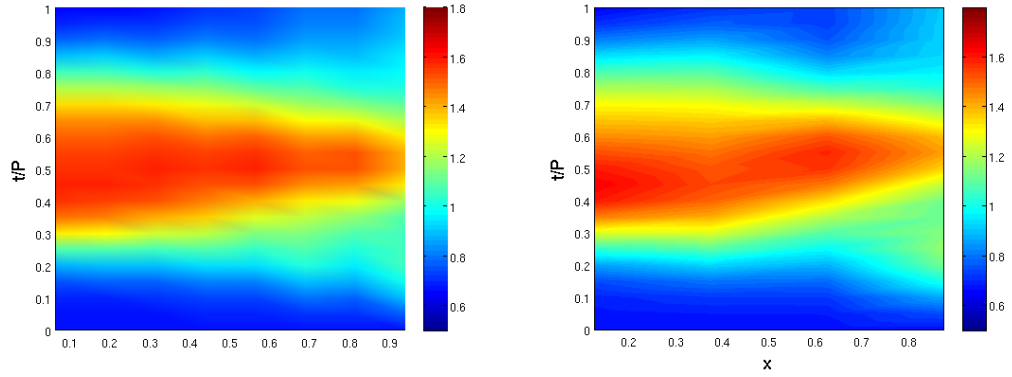
This is the original fine model. In Chapter 8 it is seen that OSM performs well for the case where the coarse model is a linearized version of the fine model. However, since the fine model is nonlinear it might be beneficial to take into account nonlinear effects for the coarse model as well. This is done by using the same physical model for the coarse as the fine model but with a lower spatial accuracy, so using less finite volume cells. This implies that the coarse model takes into account the large scale nonlinear phenomena, but neglects the smaller scales of the solution. Due to the one-dimensionality the FSI problem has only one interface point and hence no spatial coupling is required when using different meshes for the fine and coarse models. Since for one-dimensional problems the computational cost scales linearly with the number of cells, coarsening the mesh only will not lead to coarse models with a computational cost of less than 1% compared to the fine model, as is advised by Koziel et al. [21]. Therefore also coarsening of the convergence tolerance for the fluid dynamics is considered. The coarse grid coarse models which are investigated here are gathered in Table 9.1.

	Model	N_v	ϵ_c^{fluid}	$\epsilon_c^{\mathbf{R}}$	$\bar{W}^c/\bar{W}^f$	$\Delta t = P/20$		$\Delta t = P/200$	
						$\lvert E\rvert_{abs}$	$\lvert E\rvert_{rel}$	$\lvert E\rvert_{abs}$	$\lvert E\rvert_{rel}$
fine	nonlinear	64	1E-15	1E-12	-	-	-	-	-
1A	linear	64	1E-15	1E-12	0.3 %	0.022	4.4 %	0.025	5 %
1B-1	nonlinear	8	1E-15	1E-12	11.9 %	0.014	2.8 %	0.016	3.2 %
1B-2	nonlinear	8	1E-8	1E-12	6.8 %	0.014	2.8 %	0.016	3.2 %
1B-3	nonlinear	8	1E-8	1E-8	6.8 %	0.014	2.8 %	0.016	3.2 %
1B-4	nonlinear	4	1E-8	1E-12	4.3 %	0.030	6.0 %	0.041	8.2 %

Table 9.1: Coarse models which are used for the nonlinear piston problem on a fixed fluid domain.

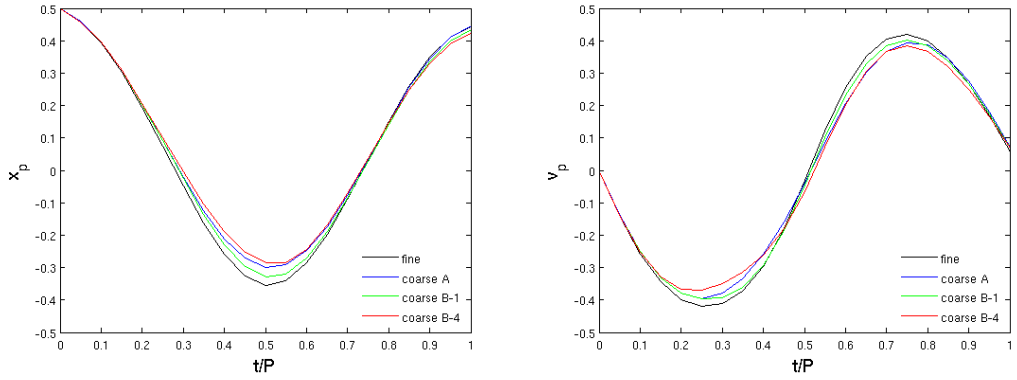
For those coarse models the fluid density field and the piston response are shown for one period in Figs. 9.1 and 9.2 for $\Delta t = P/20$. The results for $\Delta t = P/200$ are shown in Appendix E. Note that in Fig. 9.2 the piston response of coarse models 1B-2 and 1B-3 is not included. The reason for this is that the difference with model 1B-1 is only caused by the different convergence criteria. However, this results in a difference in piston response which is less than ϵ_c^{fluid}, so less than $1 \cdot 10^{-8}$, which is negligible and would not be visible in the response graphs. The accuracy of the applied coarse models is also included in Table 9.1, which contains both the average absolute and relative errors $|E|_{abs}$ and $|E|_{rel}$. Those errors are calculated with respect to the fine solution on a dense grid with $N_v = 64$ considering a

time span of 1 period. As time increases so will the errors between the models. Note that the average errors are slightly larger for $\Delta t = P/200$. This is caused by the influence of the pressure wave on the piston, which the coarse models cannot resolve accurately.



(a) Coarse model 1B-1, $N_v = 8$ (b) Coarse model 1B-4, $N_v = 4$

Figure 9.1: Nondimensional fluid density on a coarse grid, FSI-2 with $\Delta t = P/20$.



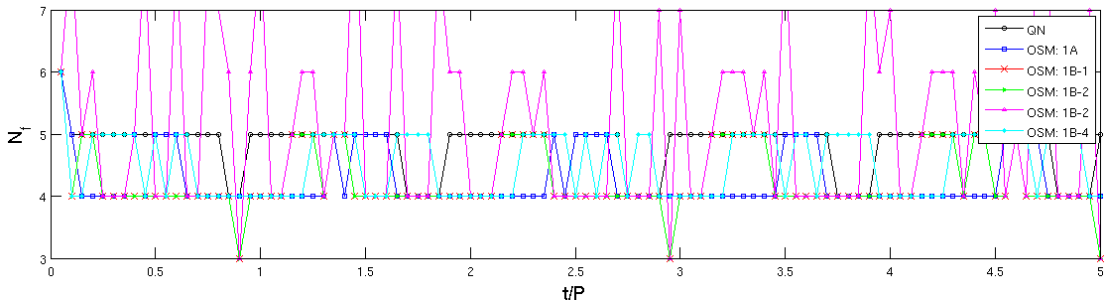
(a) Piston displacement (b) Piston velocity

Figure 9.2: Nonlinear piston response for different models, FSI-2 with $\Delta t = P/20$.

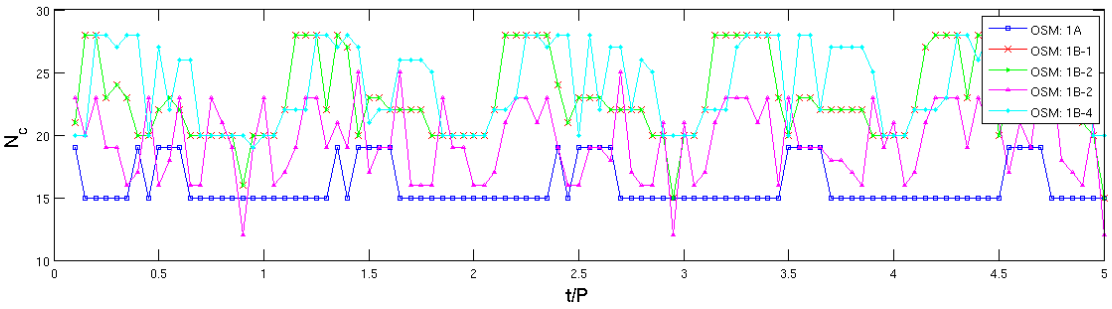
Comparing models 1B-1 and 1B-4 in Fig. 9.3, it is seen that, as expected, coarsening the grid increases the required fine computational cost to obtain a fully coupled solution. This makes sense, since it is harder to build an accurate surrogate for less accurate coarse models. However, since model 1B-4 is cheaper to evaluate its coarse cost is lower, resulting in a higher efficiency than for model 1B-1. So for this case the additional fine cost when using a coarser grid is compensated by the reduced coarse cost.

However, the coarse computational cost of model 1B-1 can be reduced by loosening the convergence criteria. When allowing a larger error during the time integration of the fluid, the coarse cost is effectively lowered by almost 50% without affecting the fine cost. On the other hand, increasing also the allowable coarse partitioning error (resulting in less coarse evaluations per iteration) has a strong adverse effect on the fine work. The reason for this

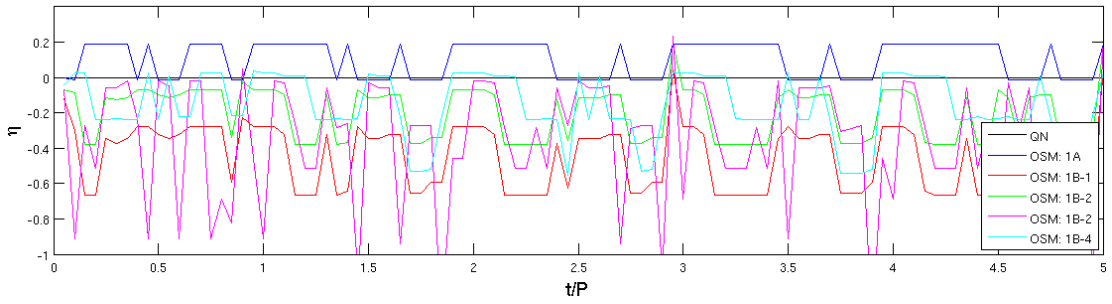
is that the output space mapping method makes use of a surrogate model (which resembles the output of the fine model) for which the interface optimization is performed. Hence, if the coarse model is not fully coupled more work has to be performed by the fine model to obtain a fully coupled solution. Due to the increased number of required fine iterations in total more coarse evaluations have to be performed.



(a) Number of fine model evaluations.



(b) Additional coarse model work.



(c) Efficiency w.r.t. Quasi Newton

Figure 9.3: Computational cost for the nonlinear piston problem on a fixed domain, OSM with different coarse models, FSI-2, $\Delta t = P/20$.

Overall it is observed that the use of a coarse model consisting of a coarser grid version of the fine model does work for output space mapping and yields good convergence results. For this one-dimensional problem, no savings in computational cost are obtained compared to a Quasi-Newton coupling method. However, it is shown that a fully coupled solution can be obtained using a limited number of fine and coarse model evaluations. Therefore it is expected that for higher dimensional problems, where the computational cost scales quadratic or even cubic with the number of finite volume cells, the influence of the coarse evaluations on the total cost will be limited and coarse grid OSM might perform better than Quasi-Newton methods or even better than OSM with a linearized coarse model.

	$\Delta t = P/20$		$\Delta t = P/200$	
	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	2.2 E-16	100	1.1 E-15
OSM: 1A	89.2	3.9 E-15	99.3	5.6 E-16
OSM: 1B-1	144.0	2.5 E-15	188.8	1.0 E-16
OSM: 1B-2	119.9	2.6 E-15	161.5	9.7 E-17
OSM: 1B-3	125.1	1.4 E-13	178.3	8.9 E-14
OSM: 1B-4	115.3	1.4 E-15	150.8	8.5 E-16

Table 9.2: Relative computational cost and corresponding accuracy for different coarse models, OSM testcase FSI-2.

9.1.2 The nonlinear piston problem on a deforming domain

Whereas in the previous section an alternative coarse model is presented, here a substitute for the fine model is discussed. The structural model remains the same, but the simplification that the fluid domain does not deform is dropped. So the length of the fluid domain and hence the size of the finite volume cells changes with the movement of the piston. This implies that the governing equations for the fluid are given in the Arbitrary Lagrangian Eulerian (ALE) formulation in integral form by

$$\frac{\partial}{\partial t}\int_{\Omega(t)} \mathbf{q}\ \mathrm{dV} + \oint_{\partial\Omega(t)} \mathbf{F}(\mathbf{q},\kappa)\cdot\mathbf{n}\ \mathrm{dS} = 0, \tag{9.1}$$

with $\Omega(t)$ the deforming control volume, $\mathbf{n}$ the unit normal of the boundary $\partial\Omega(t)$ pointing outward and $\mathbf{q}$ and $\mathbf{F}$ the state and flux vectors. The latter are defined as

$$\mathbf{q} = \begin{pmatrix} \rho \\ \rho u \end{pmatrix} \qquad \text{and} \qquad \mathbf{F}(\mathbf{q},\kappa) = \begin{pmatrix} \rho(u-\kappa) \\ \rho u(u-\kappa)+p \end{pmatrix}. \tag{9.2}$$

Here κ is the time-dependent velocity of the boundary. Since (9.1) should hold for every arbitrary domain, the governing fluid equations can be written as the semi-discrete system

$$\frac{\partial}{\partial t}(\mathbf{w_f}\Delta x) + \mathbf{A_f}\mathbf{w_f} + \mathbf{A_{fs}}\mathbf{w_s} = 0, \tag{9.3}$$

with again $\mathbf{w_f} = (\ \mathbf{q}_1^T\ \ \mathbf{q}_2^T\ \ \dots\ \ \mathbf{q}_{N_v}^T\ )^T$ and $\mathbf{w_s} = (\ \bar{x}_p\ \ \dot{\bar{x}}_p\ )^T$, $\mathbf{A_f}$ containing the fluid discretization and $\mathbf{A_{fs}}$ the influence of the structure on the fluid dynamics. This is similar

to the semi-discrete fluid system obtained for the fixed domain problem (2.4). However, now the grid size varies with time and should therefore be taken into account into the BDF2 time integration scheme:

$$\frac{\partial}{\partial t}(\mathbf{w_f}\Delta x) = \frac{3(\mathbf{w_f}\Delta x)^{n+1} - 4(\mathbf{w_f}\Delta x)^{n} + (\mathbf{w_f}\Delta x)^{n-1}}{2\Delta t}. \tag{9.4}$$

The velocity κ at the boundary of each finite volume cell is chosen such that the geometric conservation law (GCL) is satisfied. This law requires that a uniform flow is preserved on a moving mesh and is given by

$$\frac{\partial}{\partial t}\int_{\Omega(t)} \mathrm{dV} - \oint_{\partial\Omega(t)} \kappa \cdot \mathbf{n}\ \mathrm{dS} = 0. \tag{9.5}$$

For this case where a BDF2 scheme is used to perform the time integration, the discrete GCL results in a BDF2 scheme for κ as well

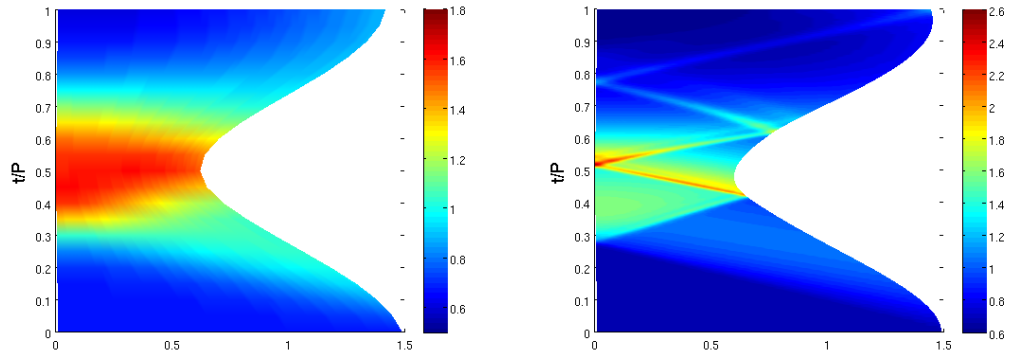
$$\kappa_{i+1/2} = \frac{3x_{i+1/2}^{n+1} - 4x_{i+1/2}^{n} + x_{i+1/2}^{n-1}}{2\Delta t}. \tag{9.6}$$

Since for the first iteration of every new timestep the domain did not change, $\kappa^{n,(1)} = \kappa^{n-1}$ to ensure continuous mesh velocities. The boundary conditions at the interface are

$$u_I = \kappa_I = \dot{x}_p \qquad \text{and} \qquad \frac{\partial \rho_I}{\partial x} = 0. \tag{9.7}$$

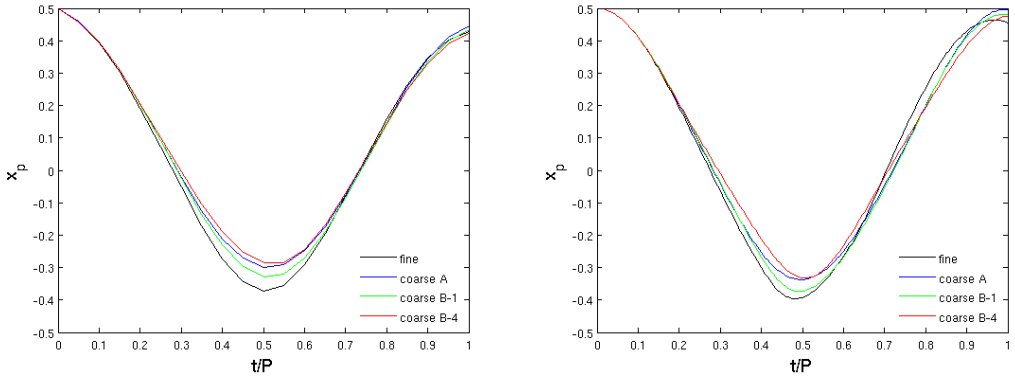
However, for a partitioned integration scheme in general $\kappa_I \neq \dot{x}_p$. Both velocities are only equal when a fully coupled solution is obtained. If using $u_I = \kappa_I$ as boundary condition, the matrix $\mathbf{A_{fs}}$ drops from (9.3) and results in a cheaper fluid solve than when using $u_I = \dot{x}_p$. However, it is found that this has an adverse effect on the convergence speed and thus more subiterations are required to obtain a fully coupled solution. Therefore the boundary condition $u_I = \dot{x}_p$ is used in the remainder. Note that from the boundary conditions it follows that at the interface $\kappa = u$ and thus a locally purely Lagrangian formulation is obtained.

The density field, piston displacement and piston velocity for this nonlinear piston problem on a deforming domain are shown in Figs. 9.4, 9.5 and 9.6 respectively for both a high temporal resolution, thus resolving the smallest nonlinearities, and a lower temporal resolution. The piston displacement is large and therefore the obtained density field differs considerably from the density field obtained on a fixed mesh (which is shown in Fig. 2.2). Note that due to the deformation of the fluid domain the period of the motion is decreased slightly compared to the static fluid domain case. So when using space mapping for this nonlinear fine model in combination with a coarse model on a fixed mesh, the difference in frequency between both models will be larger compared to the fixed domain case used in the previous chapters of this report.



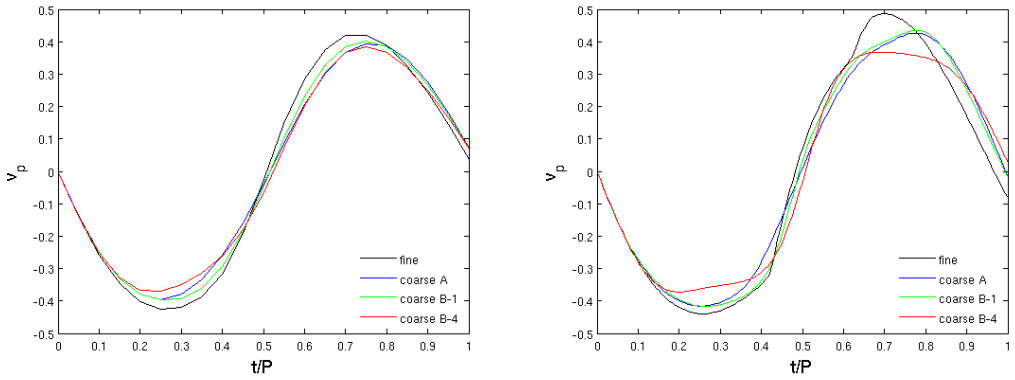
(a) $\Delta t = P/20$ (b) $\Delta t = P/200$

Figure 9.4: Density field of the nonlinear piston problem on a deforming domain, FSI-2.



(a) $\Delta t = P/20$ (b) $\Delta t = P/200$

Figure 9.5: Piston displacement of the nonlinear piston problem on a deforming domain, FSI-2.



(a) $\Delta t = P/20$ (b) $\Delta t = P/200$

Figure 9.6: Piston velocity of the nonlinear piston problem on a deforming domain, FSI-2.

Output space mapping is applied to improve the convergence speed of this nonlinear problem. The coarse models which are used are the same as for the nonlinear piston problem on the non-deforming domain. They include the linearized piston problem on a fixed fluid domain and the nonlinear problem on a fixed but coarse domain. The details of the coarse models are gathered in Table 9.3. Note that, since the computational cost of the piston problem on a deforming domain is slightly larger, the ratios of the coarse and fine costs are smaller for this fine model. Also the errors differ, they are larger now due to the different computational domains.

	Model	N_v	ϵ_c^{fluid}	$\bar{W}^c/\bar{W}^f$	$\Delta t = P/20$		$\Delta t = P/200$	
					$\|E\|_{av}$	$\|E\|_{rel}$	$\|E\|_{av}$	$\|E\|_{rel}$
fine	nonlin., moving	64	1E-15	-	-	-	-	-
2A	lin., fixed	64	1E-15	0.2 %	0.029	5.8 %	0.037	7.4 %
2B-2	nonlin., fixed	8	1E-8	4.2 %	0.020	4.0 %	0.029	5.8 %
2B-4	nonlin., fixed	4	1E-8	2.6 %	0.034	6.8 %	0.047	9.4 %

Table 9.3: Coarse models which are used for the nonlinear piston problem on a deforming fluid domain.

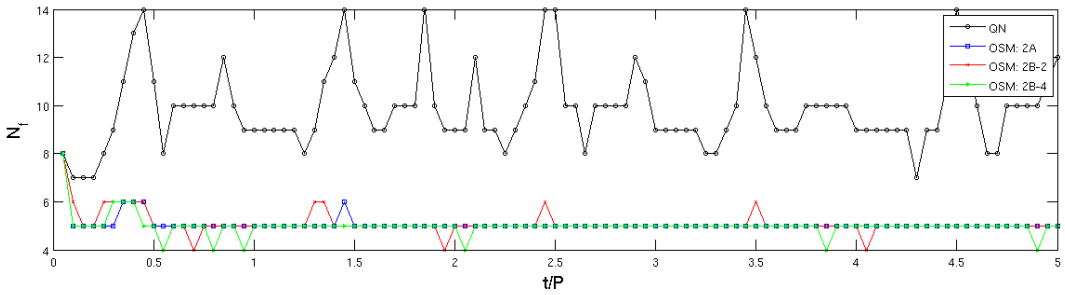
The space mapping method is applied to the test problem with settings FSI-2 for two timestep sizes, $\Delta t = P/20$ and $\Delta t = P/200$. For both cases OSM is able to obtain a fully coupled solution, see Table 9.4. Moreover, for all coarse models the efficiency is found to be higher than for the Quasi-Newton method, while yielding a partitioning error which is approximately two orders of magnitude smaller. The achieved savings in computational cost are especially large for large timesteps, since then the Quasi-Newton method converges slowly. For small timesteps QN does not experience convergence problems, but still OSM proves to be the most efficient method. The computational cost of both the OSM and QN methods are visualized in Fig. 9.7 for $\Delta t = P/20$ and in Fig. E.4 for $\Delta t = P/200$. This test problem shows again that OSM is especially very efficient for those cases that would normally require a relative large amount of subiterations.

With regard to the optimal coarse model it is again found that the linearized piston problem on a fixed grid yields the best performance. This is due to its low coarse model evaluation cost in combination with a reasonable error. However, now also the coarse grid nonlinear models yield higher efficiencies than QN. When the number of QN subiterations is high the output space mapping algorithm has enough time to build an accurate surrogate model and still save computing time. For those cases OSM is able to save computing time even when the quality and cost of the coarse model are not optimal.

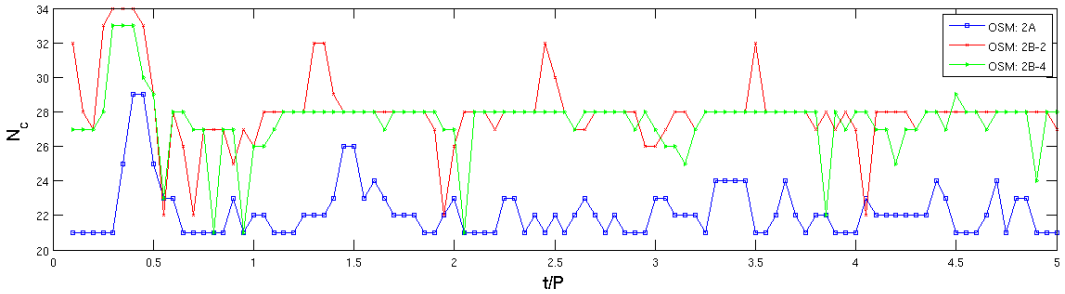
From the previous results it is found that, for 1D problems, OSM does also work well for different fine-coarse model combinations. Furthermore, it is shown that coarse grid coarse models can be suited for use with OSM if the required number of iterations is high. When higher dimensional problems are considered coarse grid coarse models might even outperform linearized coarse models since their evaluation cost decreases drastically when an extra dimension is considered. The choice of grid size in those cases is important to obtain optimal results. As observed large cells are preferred since they have a low computational cost while the additional fine work is minimal (model B-4).

	$\Delta t = P/20$		$\Delta t = P/200$	
	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	1.9 E-13	100	9.1 E-14
OSM: 2A	51.8	4.1 E-15	75.2	2.0 E-16
OSM: 2B-2	63.3	4.0 E-15	88.3	1.2 E-15
OSM: 2B-4	57.8	1.4 E-15	82.8	1.5 E-15

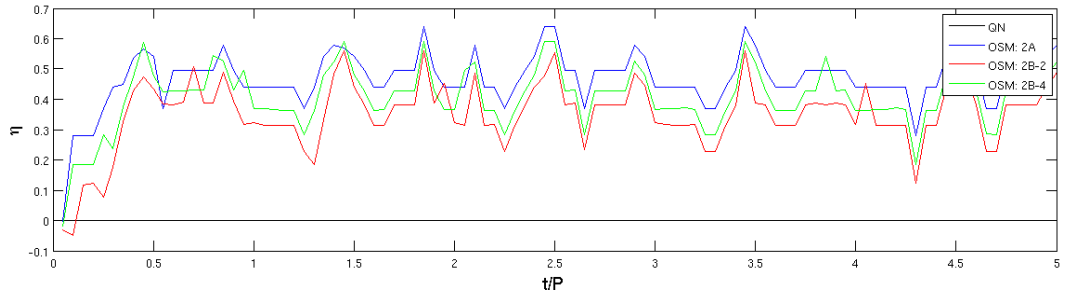
Table 9.4: Relative computational cost and corresponding accuracy for different coarse models, OSM, deforming domain testcase FSI-2.



(a) Number of fine model evaluations.



(b) Number of coarse model evaluations.



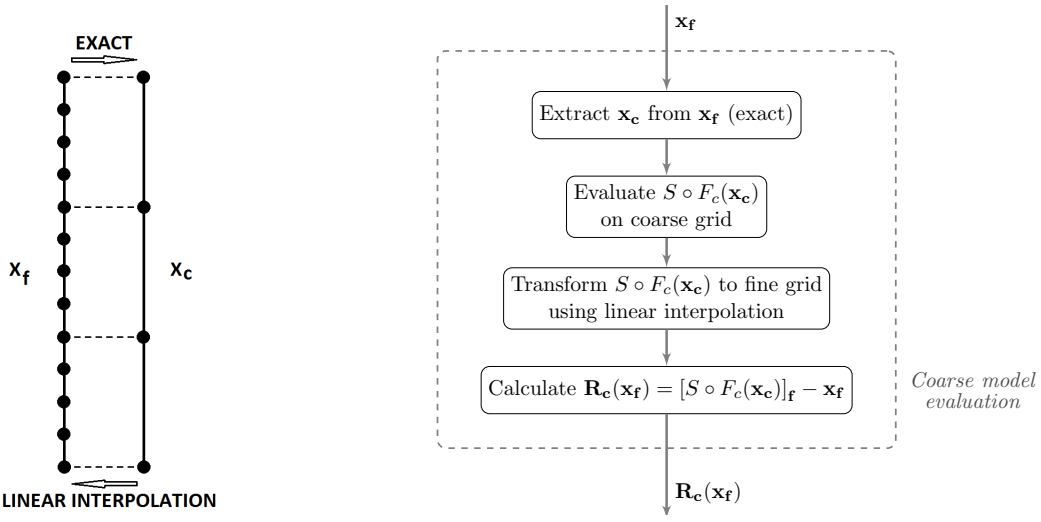
(c) Efficiency w.r.t. Quasi Newton

Figure 9.7: Computational cost for the nonlinear piston problem on a deforming domain, OSM with different coarse models, FSI-2, $\Delta t = P/20$.

9.2 Coarse grid models for the two-dimensional panel problem

In the previous section it is found that for the one-dimensional piston problem OSM with a coarse model consisting of a coarse grid version of the same fine model is able to reduce the required number of subiterations. Since for one-dimensional problems coarsening the grid does not reduce the evaluation time to a very large extend, no savings in overall computational cost are achieved for the piston problem on a fixed domain. However, for two-dimensional problems the evaluation time scales quadratically with the mesh size and thus the efficiency of OSM with a coarse grid coarse model is expected to be higher.

For the piston theory coarse model (which is used in the previous parts of this report) the same grid is used as for the fine model. Hence, the interface points coincide and thus the coarse model solution can be transferred directly to the fine model variable space and vice versa. When a coarse grid is used for the coarse model however, different meshes are obtained for both models and thus interpolation is required when transferring data between both parameter spaces along the panel. Therefore it is advised to choose the coarse mesh such that the number of finite volume cells along the interface for the fine model is a multiple of the number of coarse interface cells. In this case the transformation of variables from the fine to the coarse variable space is fast and exact, see Fig. 9.8(a). To transfer coarse data to the fine grid linear interpolation is used. This means that both the fluid and the structural dynamics are evaluated for coarse input data, after which the output is transferred to the fine parameter space. The coarse interface residual is then calculated in the fine space (see Fig. 9.8(b)), just as the mapping parameters $\mathbf{d}$ and $\mathbf{E} = \frac{\partial \mathbf{d}}{\partial \mathbf{x}}$.



(a) Fine and coarse interface mesh.

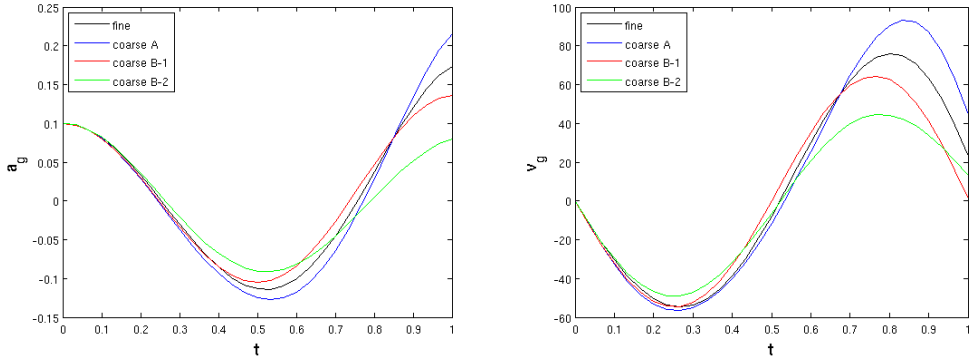
(b) Evaluation of $\mathbf{R_c}(\mathbf{x})$.

Figure 9.8: Illustration of the spatial coupling between the fine and the coarse interface mesh.

Two coarse grid models are considered to accelerate the fluid-structure coupling by means of OSM, models 3B-1 and 3B-2 where model 3B-2 has the coarsest mesh, see Table 9.5. The average coarse computational cost used to calculate the ratio $\bar{W}^c/\bar{W}^f$ includes all steps shown in Fig. 9.8(b), so also the linear interpolation. Note that the low coarse cost comes with a large error in panel response, especially when the temporal resolution is high. This is also seen in Fig. 9.9 which shows the deviations of the different models for $\Delta t = P/30$. For $\Delta t = P/10$ the errors will be a bit smaller due to the low temporal accuracy whereas for $\Delta t = P/50$ the errors increase.

	Model	N_v	$\bar{W}^c/\bar{W}^f$	$\Delta t = P/10$		$\Delta t = P/50$	
				$\lvert E\rvert_{av}$	$\lvert E\rvert_{rel}$	$\lvert E\rvert_{av}$	$\lvert E\rvert_{rel}$
fine	lin. pot. eq.	64	-	-	-	-	-
3A	piston theory	64	0.03 %	0.010	6.7 %	0.011	7.3 %
3B-1	lin. pot. eq.	16	2.08 %	0.031	20.7 %	0.043	28.7 %
3B-2	lin. pot. eq.	8	0.36 %	0.056	37.3 %	0.093	62.0 %

Table 9.5: Coarse models which are used in combination with OSM for the 2D panel problem.



(a) The generalized coordinate a_g

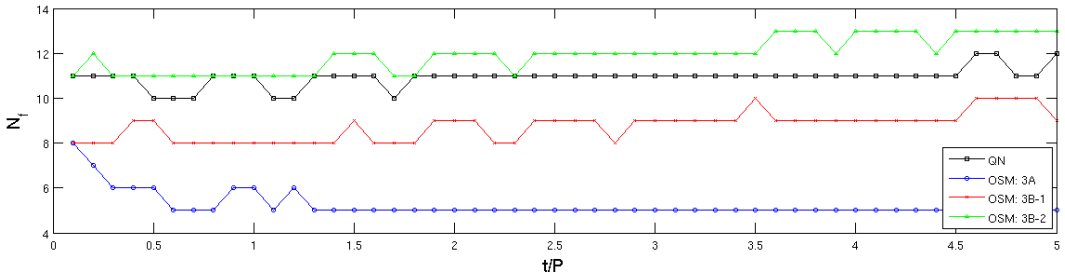
(b) Time derivative of the generalized coordinate, $\dot{a}_g$

Figure 9.9: Evolution of the generalized coordinate for $\Delta t = P/30$ for different models.

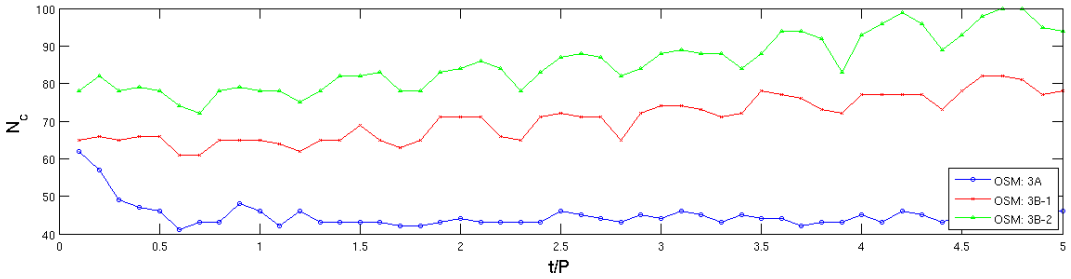
The output space mapping results for these combinations of fine and coarse models are gathered in Table 9.6 and Figs. 9.10 and E.5. The efficiency of OSM in combination with the coarse grid models is rather disappointing. Due to the large errors for those models multiple iterations are necessary to build an accurate surrogate. Therefore combination 3B-1 does only save two subiterations for $\Delta t = P/10$, whereas combination 3A succeeded in saving 6 subiterations per timestep. For model 3B-2 the error is even that large that OSM becomes less efficient than QN. Moreover, due to the large error the grid cannot be coarsened a lot, resulting in coarse costs that are still rather large. This cancels the savings obtained by 3B-1 to a large extend such that eventually the overall efficiency improvement is small for this OSM variant. This example thus shows that the gains which can be achieved by the application of OSM depend on both the coarse computational cost and the accuracy of the coarse model. It is desired that the coarse evaluation cost is $< 1\%$ of the fine model evaluation cost. For this example it appears that the error should be $< 25\%$, but it is expected that the allowable error increases if the number of subiterations required to solve the problem increases. However, the latter statement requires further investigation.

	$\Delta t = P/10$		$\Delta t = P/50$	
	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	1.1 E-13	100	1.1 E-13
OSM: 3A	48.0	2.3 E-13	69.9	1.3 E-13
OSM: 3B-1	93.3	4.1 E-13	102.5	1.6 E-13
OSM: 3B-2	112.1	3.9 E-13	103.0	1.3 E-13

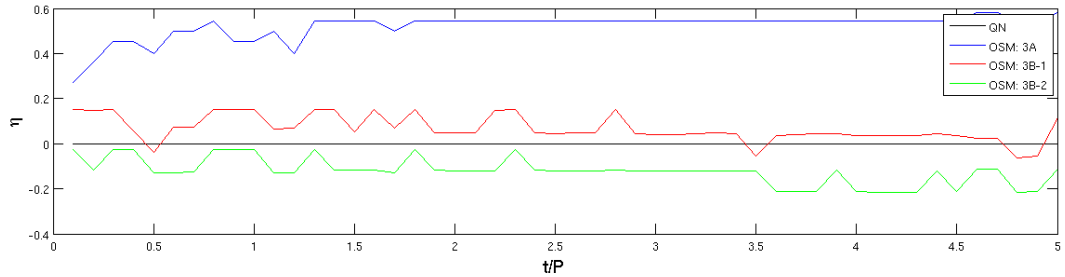
Table 9.6: Relative computational cost and corresponding accuracy for different coarse models for different methods to solve the 2D panel problem.



(a) Fine model evaluations



(b) Coarse model evaluations



(c) Efficiency w.r.t. Quasi-Newton

Figure 9.10: Performance of the OSM method for the 2D panel problem when using coarse grid models, FSI-5 with $\Delta t = P/10$.

Chapter 10

Conclusions and Recommendations

This thesis contains an investigation to the use of the space mapping technique as coupling algorithm for fluid-structure interaction simulations. The aim was to identify a generic and robust method that is able to obtain a fully coupled solution for partitioned problems while lowering the required computational cost with respect to conventional methods. To this end four promising space mapping approaches were identified, implemented and compared to a Quasi-Newton coupling algorithm for two academic test problems, being the one-dimensional piston problem and the two-dimensional flexible panel problem.

10.1 Conclusions

Space mapping is a technique which combines the accuracy of an expensive fine model with the computing speed of a less accurate coarse model. By building a surrogate based on the coarse model whose output resembles the response of the fine model, most of the optimizing computational effort is effectively shifted from the expensive fine model to the cheap coarse model without any loss of accuracy. The space mapping approaches which were believed to yield high efficiencies for FSI simulations are aggressive space mapping (ASM), generalized space mapping (GSM), implicit space mapping (ISM) and output space mapping (OSM).

ASM Aggressive space mapping makes use of an input mapping for the coarse model where an approximation to the inverse mapping is used to obtain an estimate for the fine model solution. It is found that

- ASM performs well in obtaining strongly coupled FSI solutions. It can handle a wide range of fluid-to-structure mass ratios for both low and high temporal resolutions and its efficiency does not decrease when the deviation between the fine and coarse model increases.

- ASM lowers the computing time with respect to QN, where the gains are largest for the 2D problem with a large fluid-structure coupling that is evaluated using a low temporal resolution.

GSM GSM builds a surrogate by employing a combination of both input and output mappings which are predefined. The optimal values for the mapping parameters are obtained in a parameter extraction (PE) process. GSM was successfully implemented for both problems where the following is observed:

- Several parameter extraction approaches are possible. However, no general conclusion regarding the optimal approach could be drawn: for the piston problem multipoint extraction was most efficient whereas for the panel problem constrained single point extraction yielded the best results.

- The convergence of GSM can be improved by applying a first order Taylor series expansion for the mapping parameters during the surrogate optimization.

- GSM is able to reduce the required number of subiterations and thus the fine cost with respect to QN. However, the PE process is very expensive (due to the large amount of parameters to be extracted) which reduces the overall efficiency gains: the improvement over QN is small for low temporal resolutions and GSM is less efficient than QN when the applied timestep is small.

- GSM is hard to implement due to the large number of choices that can be made for the PE approach and weighting factors. Convergence strongly depends on the choice you make and is not guaranteed. Moreover, GSM is not robust: its performance diminishes with increasing deviation between both models and it fails when the deficit becomes too large.

OSM The output space mapping method developed in this work employs only an output mapping. Therefore no parameter extraction process is required, making it simple to implement.

- OSM is found to be very successful for both test problems. The largest reduction in computational cost ($> 50\%$ with respect to QN) is achieved for the 2D problem for high fluid-to-structure mass ratios and large timesteps. For single interface point problems OSM performs not optimal but is still very efficient.

- OSM is simple, stable and robust. The efficiency remains constant, even when the deficit between both models increases.

ISM This space mapping method tries to align both models by extracting the optimal values for some physical parameters of the coarse model. The method has not been found successful.

- The influence of the preassigned parameters does not suffice to obtain a (local) match between both models. For the piston problem ISM does converge but is very inefficient. For the 2D panel problem ISM does not converge and thus no solution can be obtained.
- The performance of ISM can be improved by combining it with GSM or OSM. However, in both cases those methods are more efficient without ISM.
- Due to the adaptation of physical parameters of the coarse model it is very hard to implement ISM for black box solvers.

Furthermore it is found that, although accurate surrogate models are present at the end of every timestep, state extrapolation is the most efficient approach to determine the initial estimate for the following timestep.

Overall it is concluded that both ASM and OSM are very suited to be used for FSI simulations, where output space mapping is found to be the most promising. Therefore, a validation is performed for OSM in order to investigate its performance for different combinations of fine and coarse models: coarse grid versions of the fine model are considered as coarse model and OSM is also tested for the piston problem on a deforming domain. This resulted in the following observations:

- The good performance of OSM seems to be model independent; it performed well for all testcases.
- The largest gains in efficiency are obtained for problems which would normally require a large amount of subiterations.
- Coarse grid coarse models can be used for OSM. However, it is hard to make those models both cheap and sufficiently accurate. When the model is cheap ($\bar{W}_c/\bar{W}_f < 1\%$ is advised) but very inaccurate few iterations are saved but the coarse cost is low. When on the other hand the model is reasonably accurate but rather expensive many iterations can be saved but the coarse cost is relatively high. Therefore, coarse grid models can be used for problems that require much subiterations with conventional methods, but are not advised for FSI problems that are already cheap to solve using conventional methods.

10.2 Recommendations and future work

Since the new FSI coupling algorithms presented in this work are only tested for some simple academic problems, further research is required in order to be able to make general statements regarding their performances:

- Both OSM and ASM should be tested for more complex problems. Apart from one- and two-dimensional problems, also three-dimensional testcases have to be investigated.
- A coarse model analysis has to be performed in order to identify requirements that should be met for a successful implementation. Limits for the coarse-to-fine cost ratio and the coarse model error have to be obtained. This has to be done for both OSM and ASM. The desired output of this would be a framework that helps to identify the most suitable coarse model for every FSI problem under consideration.
- If from the above recommendations OSM and/or ASM are found to be very effective, it would be beneficial to develop a general coupling shell for black box solvers which uses space mapping to perform the subiterations.
- Although in this work GSM is not found to be efficient, its performance may be improved by using a more efficient algorithm to perform the parameter extraction (here the Matlab® function *lsqnonlin* is used). Moreover, GSM has to be applied to more test problems in order to draw general conclusions regarding the optimal PE approach and coefficient values. Only when more research is performed general statements concerning the performance of GSM for FSI problems can be made.

Bibliography

[1] M.H. Bakr, J.W. Bandler, R.M. Biernacki, S.H. Chen and K. Madsen, *A trust region aggressive space mapping algorithm for EM optimization*, IEEE Trans. Microwave Theory Tech., vol. 46, no. 12, pp. 2412-2425, 1998

[2] M.H. Bakr, J.W. Bandler, N.Georgieva and K. Madsen, *A hybrid aggressive space mapping algorithm for EM optimization*, IEEE Trans. Microwave Theory Tech., vol. 47, no. 12, pp. 2440-2449, 1999

[3] M.H. Bakr, J.W. Bandler, K. Madsen and J. Sondergaard, *Review of the space mapping approach to engineering optimization and modeling*, Optimization and engineering, vol. 1, pp. 241-276, 2000

[4] J.W. Bandler, R.M. Biernacki, S.H. Chen, P.A. Grobelny and R.H. Hemmers, *Space mapping technique for electromagnetic optimization*, IEEE Trans. Microwave Theory Tech., vol. 42, pp. 2536-2544, 1994

[5] J.W. Bandler, R.M. Biernacki, S.H. Chen, R.H. Hemmers and K. Madsen, *Electromagnetic optimization exploiting aggressive space mapping*, IEEE Trans. Microwave Theory Tech., vol. 42, no. 12, pp. 2874-2882, 1995

[6] J.W. Bandler, R.M. Biernacki, S.H. Chen and Y.F. Huang, *Design optimization of interdigital filters using aggressive space mapping and decomposition*, IEEE Trans. Microwave Theory Tech., vol. 45, no.5, pp. 761-769, 1997

[7] J.W. Bandler, R.M. Biernacki, S.H. Chen and D. Omeragic, *Space mapping optimization of waveguide filters using finite element and mode-matching electromagnetic simulators*, IEEE MTT-S Digest, pp. 635-638, 1997

[8] J.W. Bandler, N. Georgieva, M.A. Ismail, J.E. Rayas-Sanchez and Q.J. Zhang, *A generalized space mapping tableau approach to device modeling*, IEEE Trans. Microwave Theory Tech., vol. 49, no. 1, pp. 67-79, 2001

[9] J.W. Bandler, A.S. Mohammed, J.H. Bakr, K. Madsen and J. Sondergaard, *EM-based optimization exploiting partial space mapping and exact sensitivities*, IEEE Trans. Microwave Theory Tech., vol. 50, no. 12, pp. 2741-2751, 2002

[10] J.W. Bandler, Q.S. Cheng, N.K. Nikolova and M.A. Ismail, *Implicit space mapping optimization exploiting preassigned parameters*, IEEE Trans. Microwave Theory Tech., vol. 52, no. 1, pp. 378-384, 2004

[11] H. Bijl, A.H. van Zuijlen, A. de Boer and D.J. Rixen, *Fluid-structure interaction (wb1417) - An introduction to numerical coupled simulation*, Lecture notes, Delft University of Technology, 2008

[12] A. de Boer, *Computational fluid-structure interaction, spatial coupling, coupling shell and mesh deformation*, Ph.D. Thesis, Delft University of Technology, 2008

[13] Q. Cheng, *Advances in space mapping technology exploiting implicit space mapping and output space mapping*, Ph.D. Thesis, McMaster University, 2004

[14] D. Echeverria and P.W. Hemkers, *Space mapping and defect correction*, Comp. Meth. in Applied Math., vol. 5, no. 2, pp. 107-136, 2005

[15] L. Florentie, *Acceleration of partitioned fluid-structure interaction simulations by means of space mapping - Literature Survey*, Delft University of Technology, 2011

[16] Future Fast Aeroelastic Simulation Technologies (FFAST) project, *Annex I - Description of work*, Project no. 233665, 2009

[17] S. Koziel, J.W. Bandler and K. Madsen, *Space mapping optimization algorithms for engineering design*, IEEE MTT-S Int. Microw. Symp. Dig., 2006

[18] S. Koziel, J.W. Bandler and K. Madsen, *A space mapping framework for engineering optimization - Theory and implementation*, IEEE Trans. Microwave Theory Tech., vol. 54, pp. 3721-3730, 2006

[19] S. Koziel and J.W. Bandler, *Controlling convergence of space-mapping algorithms for engineering optimization*, Int. Symp. Signals, Systems and Electronics, URSI ISSSE 2007, Montreal, Canada, pp. 21-23, 2007

[20] S. Koziel, J.W. Bandler and K. Madsen, *Quality assessment of coarse models and surrogates for space mapping optimization*, Optim. Eng., vol. 9, pp. 375-391, 2008

[21] S. Koziel and J.W. Bandler, *Coarse models for efficient space mapping optimization of microwave structures*, IET Microw. Antennas Propag., vol. 4, iss. 4, pp. 453-465, 2010

[22] S. Koziel, J.W. Bandler and Q.S. Cheng, *Adaptively constrained parameter extraction for robust space mapping optimization of microwave circuits*, IEEE MTT-S Int. Microw. Symp. Dig., pp. 205-208, 2010

[23] R. Markovinovic and J.D. Jansen, *Accelerating iterative methods using reduced order models as solution predictors*, Int. J. Numer. Meth. Eng., vol. 68, pp. 525-541, 2006

[24] S. Piperno, C. Farhat and B. Larrouturou, *B. Partitioned procedures for the transient solution of coupled aeroelastic problems - Part I: Model problem, theory and two-dimensional application*, Comp. Methods in Appl. Mech. Eng., vol. 124, pp. 79-112, 1995

[25] D. Ryckelynck and F. Chinesta, *On the a priori model reduction: Overview and recent developments*, Arch. Comput. Meth. Eng., vol. 13, nr. 1, pp. 91-128, 2006

[26] D. Ryckelynck and D. Missoum Benziane, *Multi-level a priori hyper reduction of mechanical models involving internal variables*, Comput. Methods Appl. Mech. Eng, vol. 199, pp. 1134-1142, 2010

[27] T.P. Scholcz, *Multi-model coupling for fluid-structure interaction, a literature survey*, Delft University of Technology, 2010

[28] T.P. Scholcz, A.H. van Zuijlen and H. Bijl, *A multi-model incremental adaptive strategy to accelerate partitioned fluid-structure algorithms using space mapping*, IV International conference on computational methods for coupled problems in science and engineering, 2011

[29] T.P. Scholcz, *Aggressive space mapping applied to fluid-structure interaction*, D3.4: Report on implementation and validation of initial hybrid scheme, FFAST, 2011

[30] C.V. Verhoosel, T.P. Scholcz, S.J. Hulshoff and M.A. Gutiérrez, *Uncertainty and reliability analysis of fluid-structure stability boundaries*, AIAA Journal, vol. 47, no. 1, pp. 91-104, 2009

[31] J. Vierendeels, L. Lanoye, J. Degroote and P. Verdonk, *Implicit coupling of partitioned fluid-structure interaction problems with reduced order models*, Computers and structures, vol. 85, pp. 970-976, 2007

[32] A.H. van Zuijlen, *Fluid-structure interaction simulations - Efficient higher order time integration of partitioned systems*, Ph.D, Thesis, Delft University of Technology, 2006

Appendix A

Pseudocode of the space mapping algorithms

A.1 Pseudocode of the ASM algorithm

Aggressive space mapping	
1:	**for** $t = 0 : \Delta t : t_{final}$ **do**
2:	$j = 1$
3:	$\nabla_{\mathbf{x_f}} P_1 = \mathbf{I}$
4:	$\mathbf{x_c}_n^* = arg\min_{\mathbf{x}} \|\mathbf{R_c}(\mathbf{x})\|$
5:	$\mathbf{x_f}_n^1 = \mathbf{x_f}_{n-1}^* + \delta_{n-1} + \nabla_t \delta \cdot \Delta t$
6:	$\mathbf{R_f}_n^1 = S \circ F(\mathbf{x_f}_n^1) - \mathbf{x_f}_n^1$
7:	**while** $\|\mathbf{R_f}_n^j\| > \epsilon$
8:	$P(\mathbf{x_f}_n^j) = arg\min_{\mathbf{x}} \|\mathbf{R_c}(\mathbf{x})\|$
9:	**if** $j > 1$
10:	$\Delta \mathbf{x} = \mathbf{x_f}_n^j - \mathbf{x_f}_n^{j-1}$
11:	$\nabla_{\mathbf{x_f}} P_j = \nabla_{\mathbf{x_f}} P_{j-1} + \frac{P(\mathbf{x_f}_n^j) - P(\mathbf{x_f}_n^{j-1}) - \nabla_{\mathbf{x_f}} P_{j-1} \cdot \Delta \mathbf{x}}{\Delta \mathbf{x}^T \Delta \mathbf{x}} \Delta \mathbf{x}^T$
12:	**end if**
13:	$\mathbf{x_f}_n^{j+1} = \mathbf{x_f}_n^j + (\nabla_{\mathbf{x_f}} P_j)^{-1} \cdot (\mathbf{x_c}_n^* - P(\mathbf{x_f}_n^j))$
14:	$\mathbf{R_f}_n^{j+1} = S \circ F(\mathbf{x_f}_n^{j+1}) - \mathbf{x_f}_n^{j+1}$
15:	$j = j + 1$
16:	**end while**
17:	$\mathbf{x_f}_n^* = \mathbf{x_f}_n^j$
18:	$\delta_n = \mathbf{x_f}_m^* - \mathbf{x_f}_{n-1}^*$
19:	$\nabla_t \delta = \frac{3\delta_n - 4\delta_{n-1} + \delta_{n-1}}{2\Delta t}$
20:	$n = n + 1$
21:	**end for**

Table A.1: Pseudocode of the ASM algorithm

A.2 Pseudocode for the GSM framework

For the generalized space mappig framework the number of mapping parameters is five, three of which have to be obtained by means of a parameter extraction process. In the pseudocodes presented below the vector $\mathbf{x_p}^j$ is used to represent the collection of extractable parameters, so $\mathbf{x_p}^j = (\mathbf{A}^j, \mathbf{B}^j, \mathbf{c}^j)$.

Matching condition $\varepsilon(\mathbf{A}, \mathbf{B}, \mathbf{c})$ for GSM	
1:	**if** $j < N$ **then** $N = j$
2:	**if** $N \neq 1$
3:	**if** $(j \geq 2$ **and** $v_k \neq 0)$
4:	$\omega = 0$
5:	**elseif** $j \geq 3$
6:	$\omega = 0$
7:	**end**
8:	**end**
9:	$\varepsilon = [\quad]$
10:	**for** $k = j - N + 1 : 1 : j$
11:	$\varepsilon = [\ \varepsilon \ ; \ \alpha^{j-k} \cdot \|\mathbf{R_f}^k - \mathbf{A} \cdot \mathbf{R_c}(\mathbf{B} \cdot \mathbf{x_f}_n^k + \mathbf{c})\| \ ]$
12:	**end**
13:	**if** $v_k \neq 0$
14:	$\Delta\mathbf{x} = \mathbf{x_f}_n^j - \mathbf{x_f}_n^{j-1}$
15:	$\varepsilon = [\ \varepsilon \ ; \ v_k \cdot \left\| \mathbf{E}^{j-1} + \frac{\mathbf{A}\cdot\mathbf{R_c}(\mathbf{B}\cdot\mathbf{x_f}^j+\mathbf{c})-\mathbf{d}^{j-1}-\mathbf{E}^{j-1}\Delta\mathbf{x}}{\Delta\mathbf{x}^T\Delta\mathbf{x}}\Delta\mathbf{x}^T \right\| \ ]$
16:	**end**
17:	**if** $\omega \neq 0$
18:	$\varepsilon = [\ \varepsilon \ ; \ \omega \cdot \|\mathbf{x_p} - \mathbf{x_{p_0}}\| \ ]$
19:	**end**

Table A.2: Pseudocode for the matching condition as used for GSM.

Generalized space mapping framework

Initialize: $\mathbf{A} = \mathbf{I}$, $\mathbf{B} = \mathbf{I}$, $\mathbf{c} = \mathbf{0}$, $\mathbf{d} = \mathbf{0}$, $\mathbf{E} = \mathbf{0}$, $n = 1$
for $t = 0 : \Delta t : t_{final}$ **do**
  $j = 1$
  $\mathbf{x_c}_n^* = arg\min_{\mathbf{x}} \|\mathbf{R_c}(\mathbf{x})\|$
  $\mathbf{x_f}_n^1 = \mathbf{x_f}_{n-1}^* + \delta_{n-1} + \nabla_t\delta \cdot \Delta t$
  $\mathbf{R_f}_n^1 = S \circ F(\mathbf{x_f}_n^1) - \mathbf{x_f}_n^1$
  while $\|\mathbf{R_f}_n^j\| > \epsilon$
    $\mathbf{A}^j, \mathbf{B}^j, \mathbf{c}^j = arg\min_{\mathbf{l}^j < \mathbf{A},\mathbf{B},\mathbf{c} < \mathbf{u}^j} \|\varepsilon^j(\mathbf{A}, \mathbf{B}, \mathbf{c})\|$
    $\mathbf{d}^j = \mathbf{R_f}_n^j - \mathbf{A}^j \cdot \mathbf{R_c}(\mathbf{B}^j \cdot \mathbf{x_f}_n^j + \mathbf{c}^j)$
    if $(AB = 1$ **and** $\mathbf{d}^j < \alpha_{decr} \cdot \mathbf{d}_{max})$
      $\delta^{j+1} = \delta^j / \beta_{decr}$
    elseif $(AB = 1$ **and** $\mathbf{d}^j > \alpha_{incr} \cdot \mathbf{d}_{max})$
      $\delta^{j+1} = \delta^j \cdot \beta_{incr}$
    end if
    $\mathbf{l}^{j+1} = \mathbf{x_p}^j - \delta^{j+1}$ **and** $\mathbf{u}^{j+1} = \mathbf{x_p}^j + \delta^{j+1}$
    if $j > 1$
      $\Delta\mathbf{x} = \mathbf{x_f}_n^j - \mathbf{x_f}_n^{j-1}$
      $\mathbf{E}^j = \mathbf{E}^{j-1} + \frac{\mathbf{d}^j - \mathbf{d}^{j-1} - \mathbf{E}^{j-1}\Delta\mathbf{x}}{\Delta\mathbf{x}^T\Delta\mathbf{x}}\Delta\mathbf{x}^T$
      if $TE = 1$
        $\nabla^j_{\mathbf{x_f}}\mathbf{x_p} = \nabla^{j-1}_{\mathbf{x_f}}\mathbf{x_p} + \frac{\mathbf{x_p}^j - \mathbf{x_p}^{j-1} - \nabla^{j-1}_{\mathbf{x_f}}\mathbf{x_p}\cdot\Delta\mathbf{x}}{\Delta\mathbf{x}^T\Delta\mathbf{x}}\Delta\mathbf{x}^T$
        $\tilde{\mathbf{x}}^j_{\mathbf{p}}(\mathbf{x_f}) = \mathbf{x_p}^j + \nabla^j_{\mathbf{x_f}}\mathbf{x_p} \cdot (\mathbf{x_f} - \mathbf{x_f}_n^j)$
      else
        $\tilde{\mathbf{x}}^j_{\mathbf{p}} = \mathbf{x_p}^j$
      end if
    end if
    $\mathbf{R_s}^j(\mathbf{x}) = \tilde{\mathbf{A}}^j \cdot \mathbf{R_c}(\tilde{\mathbf{B}}^j \cdot \mathbf{x} + \tilde{\mathbf{c}}^j) + \tilde{\mathbf{d}}^j + \tilde{\mathbf{E}}^j \cdot (\mathbf{x} - \mathbf{x_f}_n^j)$
    $\mathbf{x_f}_n^{j+1} = arg\min_{\mathbf{x}} \|\mathbf{R_s}^j(\mathbf{x})\|$
    $\mathbf{R_f}_n^{j+1} = S \circ F(\mathbf{x_f}_n^{j+1}) - \mathbf{x_f}_n^{j+1}$
    $j = j + 1$
  end while
  $\mathbf{x_f}_n^* = \mathbf{x_f}_n^j$
  $\delta_n = \mathbf{x_f}_m^* - \mathbf{x_f}_{n-1}^*$
  $\nabla_t\delta = \frac{3\delta_n - 4\delta_{n-1} + \delta_{n-1}}{2\Delta t}$
  $n = n + 1$
end for

Table A.3: Pseudocode of the GSM algorithm

A.3 ISM pseudocode

Implicit space mapping is a dual space mapping approach, just like GSM. Therefore the pseudocode of the ISM algorithm is very similar to the pseudocode of GSM. The only differences are the extractable parameters, which are now $\mathbf{x_p}^j = (\bar{k}^j, \bar{m}^j)$ with $\mathbf{x_{p_0}} = (1, 1)$, and the surrogate model, now $\mathbf{R_s}^j(\mathbf{x}) = \mathbf{R_c}(\mathbf{x}, \bar{k}^j, \bar{m}^j)$. When ISM is combined with OSM also the output mapping parameters $\mathbf{d}^j$ and $\mathbf{E}^j$ are taken into account, similar to as done for the GSM algorithm. When ISM is combined with GSM the extractable parameters are $\mathbf{x_p}^j = (\bar{k}^j, \bar{m}^j, \mathbf{A}^j, \mathbf{B}^j, \mathbf{c}^j)$ and the surrogate becomes $\mathbf{R_s}(\mathbf{x}) = \mathbf{A}^j \cdot \mathbf{R_c}(\mathbf{B}^j \cdot \mathbf{x} + \mathbf{c}^j, \bar{k}^j, \bar{m}^j) + \mathbf{d}^j + \mathbf{E}^j \cdot (\mathbf{x} - \mathbf{x_f}_n^j)$.

A.4 Pseudocode for the output mapping algorithm

Output space mapping	
1:	Initialize: $\mathbf{d} = \mathbf{0}$, $\mathbf{E} = \mathbf{0}$, $n = 1$
2:	**for** $t = 0 : \Delta t : t_{final}$ **do**
3:	$j = 1$
4:	$\mathbf{x_c}_n^* = arg\min_{\mathbf{x}} \|\mathbf{R_c}(\mathbf{x})\|$
5:	$\mathbf{x_f}_n^1 = \mathbf{x_f}_{n-1}^* + \delta_{n-1} + \nabla_t\delta \cdot \Delta t$
6:	$\mathbf{R_f}_n^1 = S \circ F(\mathbf{x_f}_n^1) - \mathbf{x_f}_n^1$
7:	**while** $\|\mathbf{R_f}_n^j\| > \epsilon$
8:	$\mathbf{d}^j = \mathbf{R_f}_n^j - \mathbf{R_c}(\mathbf{x_f}_n^j)$
9:	**if** $j > 1$
10:	$\Delta\mathbf{x} = \mathbf{x_f}_n^j - \mathbf{x_f}_n^{j-1}$
11:	$\mathbf{E}^j = \mathbf{E}^{j-1} + \frac{\mathbf{d}^j - \mathbf{d}^{j-1} - \mathbf{E}^{j-1}\Delta\mathbf{x}}{\Delta\mathbf{x}^T\Delta\mathbf{x}}\Delta\mathbf{x}^T$
12:	**end if**
13:	$\mathbf{R_s}^j(\mathbf{x}) = \mathbf{R_c}(\mathbf{x}) + \mathbf{d}^j + \mathbf{E}^j \cdot (\mathbf{x} - \mathbf{x_f}_n^j)$
14:	$\mathbf{x_f}_n^{j+1} = arg\min_{\mathbf{x}} \|\mathbf{R_s}^j(\mathbf{x})\|$
15:	$\mathbf{R_f}_n^{j+1} = S \circ F(\mathbf{x_f}_n^{j+1}) - \mathbf{x_f}_n^{j+1}$
16:	$j = j + 1$
17:	**end while**
18:	$\mathbf{x_f}_n^* = \mathbf{x_f}_n^j$
19:	$\delta_n = \mathbf{x_f}_m^* - \mathbf{x_f}_{n-1}^*$
20:	$\nabla_t\delta = \frac{3\delta_n - 4\delta_{n-1} + \delta_{n-1}}{2\Delta t}$
21:	$n = n + 1$
22:	**end for**

Table A.4: Pseudocode of the OSM algorithm

Appendix B

Additional results for ASM

B.1 1D piston problem

	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	3.7 E-16	100	1.3 E-15	100	2.5 E-15
1: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}_{n-1}^*$	97.4	8.3 E-16	107.7	2.6 E-16	109.7	2.5 E-15
2: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}_{n-1}^* + \delta_n$	88.5	2.7 E-15	89.6	1.5 E-15	96.9	1.2 E-16
3: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}_n^*$	95.8	4.1 E-16	105.5	3.9 E-16	109.7	2.1 E-15
4: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}_{n-1}^*$	93.3	6.7 E-16	99.1	1.5 E-15	102.8	4.8 E-15
5: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}_{n-1}^* + \delta_n$	84.0	9.4 E-15	89.4	1.7 E-15	97.6	2.9 E-16
6: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}_n^* + \delta_n$	83.6	9.0 E-15	89.4	1.4 E-15	97.1	3.1 E-16

Table B.1: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, ASM test case FSI-1.

	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	3.4 E-15	100	2.4 E-16	100	8.2 E-16
1: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}_{n-1}^*$	95.8	1.6 E-15	101.3	7.5 E-16	104.4	1.1 E-15
2: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}_{n-1}^* + \delta_n$	87.1	3.1 E-15	84.4	6.5 E-15	88.3	8.2 E-16
3: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}_n^*$	92.2	6.5 E-16	99.7	2.5 E-15	113.4	1.4 E-15
4: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}_{n-1}^*$	89.9	8.8 E-16	93.7	1.2 E-15	99.7	1.4 E-15
5: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}_{n-1}^* + \delta_n$	82.5	6.3 E-15	86.4	3.1 E-15	90.6	2.8 E-15
6: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}_n^* + \delta_n$	83.1	4.3 E-15	86.8	2.7 E-15	90.8	2.7 E-15

Table B.2: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, ASM test case FSI-3.

B.2 2D panel problem

	$\Delta t = P/10$		$\Delta t = P/30$		$\Delta t = P/50$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	1.8 E-13	100	1.8 E-13	100	3.0 E-13
1: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1}$	78.8	1.5 E-13	94.7	1.5 E-13	101.1	2.8 E-13
2: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1} + \delta_n$	71.7	4.5 E-13	80.1	1.8 E-13	96.9	8.6 E-14
3: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n$	77.7	2.3 E-13	101.8	2.2 E-13	114.6	1.0 E-13
4: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1}$	77.3	1.7 E-13	95.2	3.3 E-13	102.8	2.9 E-13
5: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1} + \delta_n$	75.7	1.7 E-13	93.4	1.4 E-13	100.2	1.3 E-13
6: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n + \delta_n$	77.7	2.3 E-13	101.9	2.1 E-13	114.9	1.1 E-13

Table B.3: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, ASM test case FSI-4.

	$\Delta t = P/10$		$\Delta t = P/30$		$\Delta t = P/50$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	2.3 E-13	100	1.9 E-13	100	7.8 E-14
1: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1}$	69.8	3.0 E-13	96.2	2.4 E-13	99.6	1.9 E-13
2: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1} + \delta_n$	66.9	3.4 E-13	85.0	2.5 E-13	84.5	2.7 E-13
3: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n$	76.9	3.2 E-13	102.5	9.2 E-13	110.1	3.7 E-13
4: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1}$	75.7	3.1 E-13	101.2	2.9 E-13	107.2	2.3 E-13
5: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1} + \delta_n$	74.6	3.0 E-13	96.6	3.4 E-13	104.8	1.5 E-13
6: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n + \delta_n$	76.9	3.2 E-13	102.5	9.1 E-13	108.3	5.7 E-13

Table B.4: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, ASM test case FSI-6.

Appendix C

Additional results for GSM

C.1 1D piston problem

	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	3.7 E-16	100	1.3 E-15	100	2.5 E-15
SPE	180.8	7.4 E-14	252.4	4.5 E-14	298.3	5.1 E-14
Constr. SPE	105.1	4.0 E-15	107.3	5.1 E-15	108.6	1.0 E-14
Constr. SPE + TE	98.6	1.9 E-15	108.4	1.3 E-15	110.5	2.0 E-15
Constr. SPE + AB	103.9	3.6 E-16	110.3	3.7 E-16	110.3	5.0 E-16
MPE 3pt.	96.4	2.7 E-15	97.4	3.7 E-15	104.3	1.8 E-15
Constr. MPE 3pt.	112.3	2.8 E-15	112.1	7.0 E-15	112.1	9.8 E-15
MPE 4pt.	96.5	2.7 E-15	97.4	3.7 E-15	104.3	1.7 E-15
MPE 3pt. + TE	99.6	2.0 E-15	102.2	1.9 E-15	105.1	3.3 E-16
MPE 3pt. + AB	96.4	2.5 E-15	97.3	4.4 E-15	104.2	2.4 E-15
GPE 2pt.	133.7	9.1 E-15	193.0	3.7 E-14	214.1	3.2 E-14
Constr. GPE 2pt.	154.3	6.7 E-14	172.6	1.1 E-13	178.9	1.3 E-13
GPE 3pt.	122.5	2.3 E-15	135.6	1.4 E-15	157.4	2.3 E-15
GPE 2pt. + TE	98.9	2.1 E-15	102.8	3.1 E-15	107.8	1.9 E-15
GPE 2pt. + AB	133.6	9.1 E-15	193.0	3.7 E-14	214.0	3.2 E-14

Table C.1: Relative computational cost and average final accuracy for different PE approaches for GSM, testcase FSI-1 (TE = Taylor expansion and AB = adaptive bounds).

	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	3.4 E-15	100	2.4 E-16	100	8.2 E-16
SPE	152.6	2.5 E-14	∞	∞	∞	∞
Constr. SPE	103.5	6.9 E-15	105.5	4.8 E-15	108.9	4.2 E-15
Constr. SPE + TE	97.3	2.7 E-15	99.7	3.1 E-15	103.8	1.9 E-15
Constr. SPE + AB	103.7	4.6 E-16	111.1	2.6 E-16	115.4	3.4 E-16
MPE 3pt.	95.1	5.3 E-15	92.5	1.1 E-14	98.4	4.9 E-15
Constr. MPE 3pt.	111.8	2.3 E-15	108.6	5.4 E-15	109.8	4.3 E-15
MPE 4pt.	95.4	6.4 E-15	92.6	1.1 E-14	98.6	4.6 E-15
MPE 3pt. + TE	99.0	9.2 E-16	98.0	3.4 E-15	102.1	9.2 E-16
MPE 3pt. + AB	95.2	3.2 E-15	92.5	9.8 E-15	98.2	4.9 E-15
GPE 2pt.	136.4	2.2 E-14	164.3	1.5 E-14	206.2	3.5 E-14
Constr. GPE 2pt.	151.4	8.5 E-14	168.8	1.2 E-13	183.9	1.2 E-13
GPE 3pt.	120.6	2.9 E-15	139.4	5.8 E-15	185.6	6.1 E-15
GPE 2pt. + TE	99.2	1.1 E-15	97.6	6.4 E-15	100.3	2.6 E-15
GPE 2pt. + AB	136.4	2.2 E-14	164.3	1.5 E-14	206.2	3.5 E-14

Table C.2: Relative computational cost and average final accuracy for different PE approaches for GSM, testcase FSI-3 (TE = Taylor expansion and AB = adaptive bounds).

C.2 2D panel problem

	$\Delta t = P/10$		$\Delta t = P/30$		$\Delta t = P/50$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	2.0 E-13	100	9.2 E-14	100	2.7 E-13
Constr. SPE	98.4	3.3 E-13	96.8	2.0 E-13	108.2	2.7 E-13
Constr. SPE + TE	84.7	4.9 E-13	87.5	2.3 E-13	108.5	3.5 E-14
Constr. MPE	∞	∞	120.6	2.6 E-13	129.6	4.3 E-14
Constr. MPE + TE	∞	∞	107.2	2.5 E-13	130.6	3.5 E-14
Constr. GPE	123.5	3.4 E-13	128.2	1.6 E-13	144.9	8.1 E-14
Constr. GPE + TE	113.1	3.4 E-13	118.2	1.8 E-13	132.8	3.5 E-14

Table C.3: Computational cost and average final accuracy for the 2D panel problem applying GSM (testcase FSI-4, taking into account 2 periods).

	$\Delta t = P/10$		$\Delta t = P/30$		$\Delta t = P/50$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	2.5 E-13	100	2.3 E-13	100	6.6 E-14
Constr. SPE	122.7	6.0 e-13	129.9	2.1 E-13	108.6	2.4 E-13
Constr. SPE + TE	84.6	5.4 E-13	103.2	1.7 E-13	100.4	2.0 E-13
Constr. MPE	∞	∞	∞	∞	146.8	1.9 E-13
Constr. MPE + TE	∞	∞	123.7	2.4 E-13	123.2	2.5 E-13
Constr. GPE	133.2	4.9 E-13	150.2	1.9 E-13	143.9	1.9 E-13
Constr. GPE + TE	120.4	9.4 E-13	131.9	2.1 E-13	139.9	2.2 E-13

Table C.4: Computational cost and average final accuracy for the 2D panel problem applying GSM (testcase FSI-6, taking into account 2 periods).

Appendix D

Additional results for OSM

D.1 1D piston problem

	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	3.7 E-16	100	1.3 E-15	100	2.5 E-15
1: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}_{n-1}^*$	107.9	3.3 E-16	110.0	4.6 E-16	118.7	3.4 E-16
2: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}_{n-1}^* + \delta_n$	90.9	4.5 E-15	92.7	2.1 E-15	98.5	4.6 E-16
3: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}_n^*$	99.8	2.3 E-16	108.6	6.3 E-16	119.3	3.6 E-16
4: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}_{n-1}^*$	88.5	2.9 E-15	95.7	1.8 E-15	103.9	1.0 E-15
5: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}_{n-1}^* + \delta_n$	87.0	3.1 E-15	93.3	1.6 E-15	100.1	9.7 E-16
6: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}_n^* + \delta_n$	88.9	4.7 E-15	93.3	2.0 E-15	98.5	1.9 E-15

Table D.1: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, OSM test case FSI-1.

	$\Delta t = P/10$		$\Delta t = P/20$		$\Delta t = P/30$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	3.4 E-15	100	2.4 E-16	100	8.2 E-16
1: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}_{n-1}^*$	99.9	1.3 E-15	104.9	2.2 E-15	110.9	1.4 E-15
2: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}_{n-1}^* + \delta_n$	92.1	1.1 E-15	88.7	4.9 E-15	93.0	9.7 E-16
3: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}_n^*$	94.8	5.8 E-16	101.6	1.3 E-15	113.7	9.7 E-16
4: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}_{n-1}^*$	85.8	2.7 E-15	98.1	6.1 E-16	106.2	5.4 E-16
5: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}_{n-1}^* + \delta_n$	84.0	3.1 E-15	96.4	4.7 E-16	101.5	7.0 E-16
6: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}_n^* + \delta_n$	87.1	3.1 E-15	90.4	2.0 E-15	96.1	2.1 E-15

Table D.2: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, OSM test case FSI-3.

D.2 2D panel problem

	$\Delta t = P/10$		$\Delta t = P/30$		$\Delta t = P/50$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	1.8 E-13	100	1.8 E-13	100	3.0 E-13
1: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1}$	51.2	2.9 E-13	67.1	1.6 E-13	75.8	1.6 E-13
2: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1} + \delta_n$	51.6	2.7 E-13	64.6	1.4 E-13	70.5	2.3 E-13
3: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n$	52.7	2.9 E-13	72.4	2.1 E-13	85.8	1.1 E-13
4: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1}$	52.8	3.2 E-13	75.8	1.2 E-13	86.6	1.5 E-13
5: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1} + \delta_n$	52.4	3.3 E-13	72.2	1.4 E-13	83.7	1.3 E-13
6: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n + \delta_n$	52.3	2.5 E-13	66.4	1.7 E-13	81.3	1.2 E-13

Table D.3: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, OSM test case FSI-4.

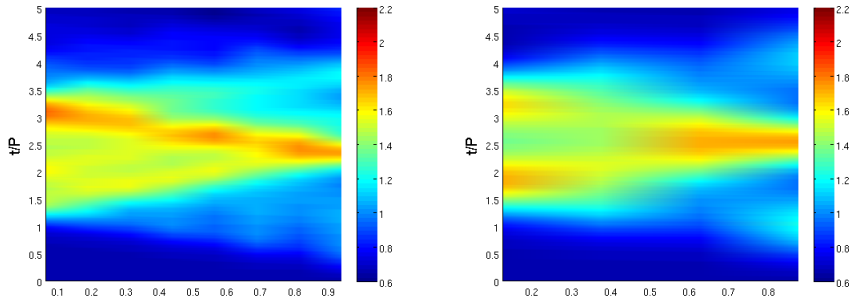
	$\Delta t = P/10$		$\Delta t = P/30$		$\Delta t = P/50$	
Approach	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$	Cost [%]	$\|\mathbf{R_f}\|_{av}$
QN	100	2.3 E-13	100	1.9 E-13	100	8.0 E-14
1: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1}$	46.8	3.4 E-13	70.6	2.4 E-13	75.9	1.7 E-13
2: $\mathbf{x_f}_n^{(1)} = \mathbf{x_f}^*_{n-1} + \delta_n$	44.7	3.3 E-13	66.0	1.5 E-13	69.0	3.5 E-13
3: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n$	50.1	4.1 E-13	79.5	1.7 E-13	88.9	1.6 E-13
4: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1}$	48.9	4.1 E-13	75.0	2.4 E-13	85.1	2.9 E-13
5: $\mathbf{x_f}_n^{(1)} = \mathbf{x_s}^*_{n-1} + \delta_n$	48.5	3.8 E-13	74.3	2.8 E-13	85.2	2.7 E-13
6: $\mathbf{x_f}_n^{(1)} = \mathbf{x_c}^*_n + \delta_n$	48.6	4.1 E-13	70.5	3.2 E-13	81.5	8.1 E-14

Table D.4: Relative computational cost and corresponding accuracy for different approaches to define the initial estimate, OSM test case FSI-6.

Appendix E

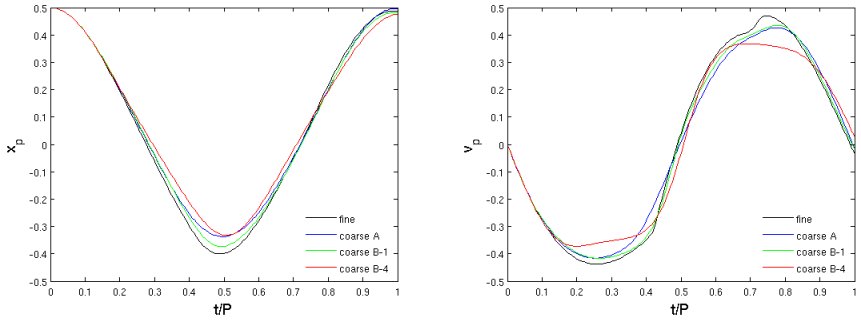
Additional validation results

E.1 Nonlinear piston problem, fixed domain



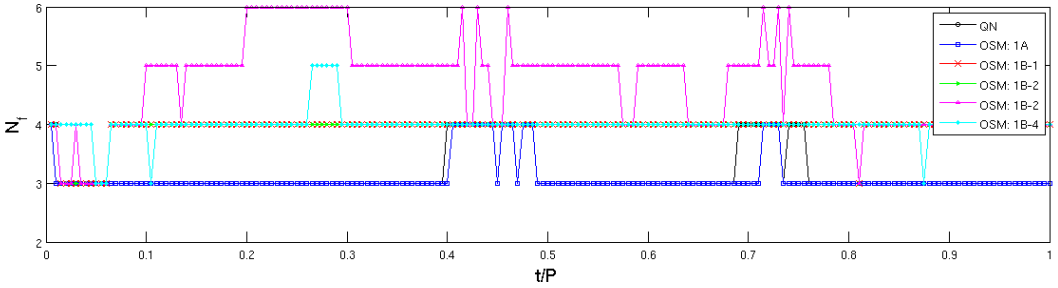
(a) Coarse model 1B-1, $N_v = 8$ (b) Coarse model 1B-4, $N_v = 4$

Figure E.1: Nondimensional fluid density on a coarse grid, FSI-2 with $\Delta t = P/200$.

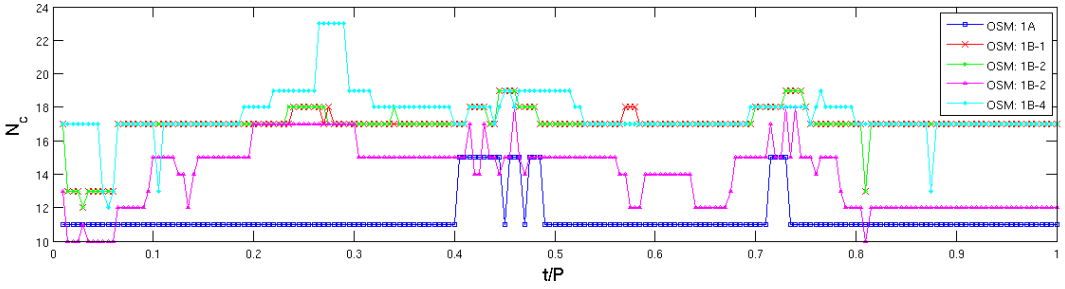


(a) Piston displacement (b) Piston velocity

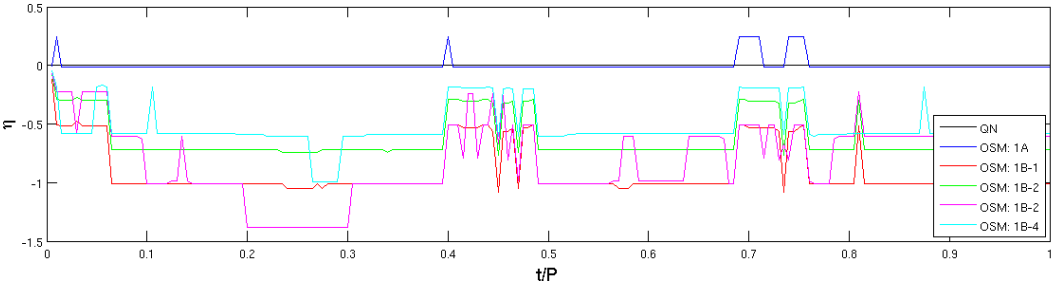
Figure E.2: Nonlinear piston response for different grid sizes, FSI-2 with $\Delta t = P/200$.



(a) Number of fine model evaluations.



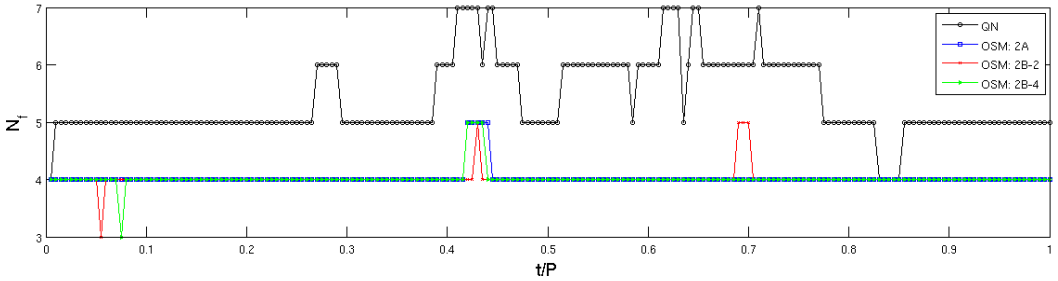
(b) Additional coarse model work.



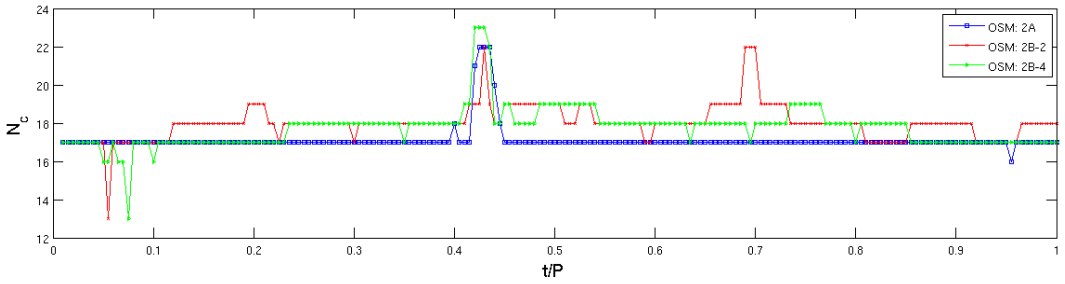
(c) Efficiency w.r.t. Quasi Newton

Figure E.3: Computational cost for the nonlinear piston problem on a fixed domain, OSM with different coarse models, FSI-2, $\Delta t = P/200$.

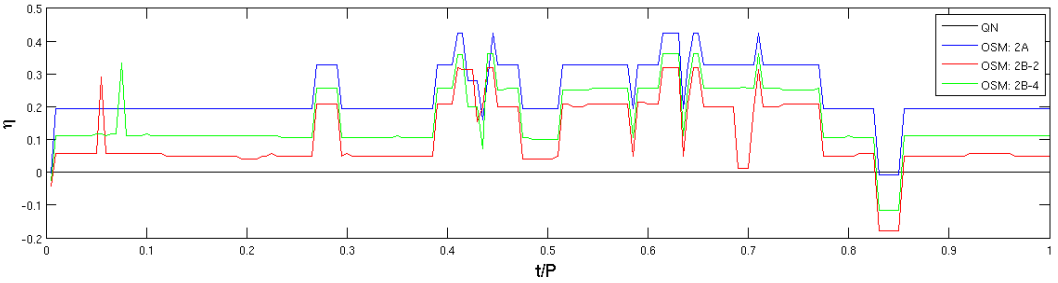
E.2 Nonlinear piston problem, deforming domain



(a) Number of fine model evaluations.



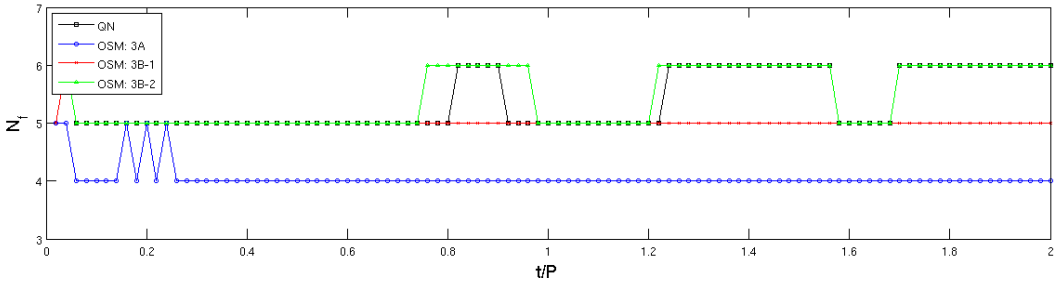
(b) Number of coarse model evaluations.



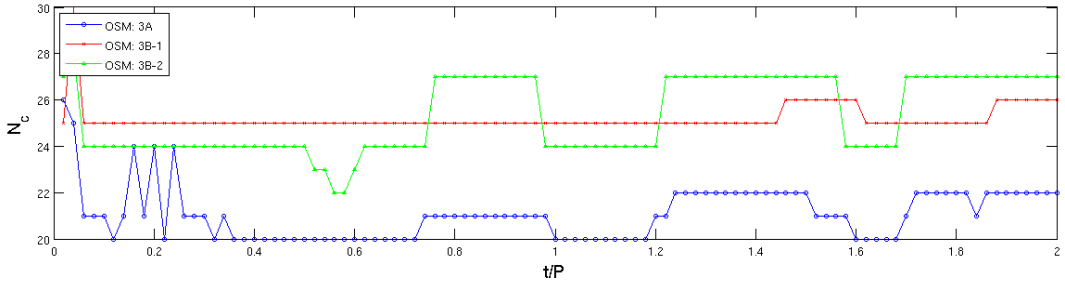
(c) Efficiency w.r.t. Quasi Newton

Figure E.4: Computational cost for the nonlinear piston problem on a deforming domain, OSM with different coarse models, FSI-2, $\Delta t = P/200$.

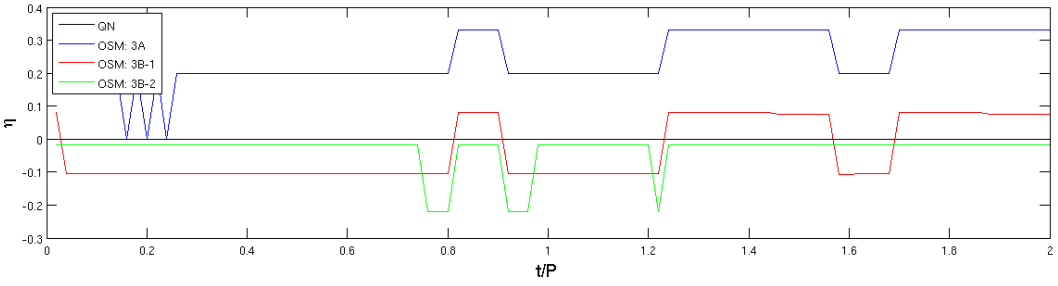
E.3 Panel problem



(a) Fine model evaluations



(b) Coarse model evaluations



(c) Efficiency w.r.t. Quasi-Newton

Figure E.5: Performance of the OSM method for the 2D panel problem when using coarse grid models, FSI-5 with $\Delta t = P/50$.