Isogeometric Shape Optimization in Fluid-Structure Interaction

Christoph Heinrich, Régis Duvigneau, Louis Blanchard

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Isogeometric Shape Optimization in Fluid-Structure Interaction

Ch. Heinrich — R. Duvigneau — L. Blanchard

N° 7639

June 2011

Domaine 1

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--7639--FR+ENG

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
SOPHIA ANTIPOLIS - MÉDITERRANÉE

Isogeometric Shape Optimization in Fluid-Structure Interaction

Ch. Heinrich*, R. Duvigneau †, L. Blanchard †

Domaine : Mathématiques appliquées, calcul et simulation
Équipes-Projets Opale

Rapport de recherche n° 7639 — June 2011 — 21 pages

Abstract: The objective of this work is to examine the potential of isogeometric methods in the context of multidisciplinary shape optimization.
We introduce a shape optimization problem based on a coupled fluid-structure system, whose geometry is defined by NURBS (Non-Uniform Rational B-Spline) curves. This shape optimization problem is then solved by using either an isogeometric approach, or a classical grid-based approach.
In spite of the fact that optimization results do not show any major differences, conceptional advantages of the new isogeometric method become apparent. In particular, control points of the spline can be directly handled as design variables without the need of a spline-fit and consequently geometry errors can be excluded at every stages of the optimization loop.

Key-words: Shape Optimization, NURBS, Isogeometric Analysis, Fluid-Structure Interaction

* Technische Universität München, Zentrum Mathematik, Boltzmannstraße 3, 85748 Garching, Germany
† INRIA Opale Project-Team

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

Optimisation de forme isogéométrique en interaction fluide-structure

Résumé : Le but de ce travail est d'examiner le potentiel des méthodes isogeometriques dans le contexte de l'optimisation de forme multidisciplinaire.
On introduit un problème d'optimisation de forme basé sur un système couplé fluide-structure, dont la géométrie est définie par des courbes NURBS (Non-Uniform Rational B-Spline). Ce problème d'optimisation de forme est ensuite résolu à l'aide soit d'une approche isogéométrique, soit d'une approche classique s'appuyant sur un maillage.
Bien que les résultats de l'optimisation ne présentent pas de différences majeures, les avantages conceptuels de l'approche isogéométrique apparaissent. Notamment, les points de contrôle des splines peuvent être directement manipulées en tant que variables de conception, sans devoir employer une approximation de la géométrie. Par conséquent, les erreurs géométriques sont exclues à toutes les étapes de la procédure d'optimisation.

Mots-clés : Optimisation de forme, NURBS, analyse isogéométrique, interaction fluide-structure

Introduction

For many years the interplay of Computer Aided Design (CAD) and numerical analysis suffered from the bottleneck of different representations of geometry. Whereas polynomials were predominant in the latter discipline, function classes like B-Splines and Non-Uniform Rational B-Splines (NURBS) were employed by CAD systems. With the advent of Isogeometric Analysis (IA) [7] in 2005, a methodology has been proposed to bridge this gap. In particular, IA describes both the geometry and the numerical solution in terms of NURBS basis functions and consequently can be seen as an isoparametric Finite Element Method (FEM).

Especially in shape optimization the benefits of a unified geometry representation become apparent. On the one hand geometries are represented exactly in the PDE (partial differential equation) solver which is called several times in the optimization loop. On the other hand NURBS data, i.e. control points (and weights), can be directly used to define design variables. In contrast to classical methods no spline fit has to be performed to reduce the number of unknowns on the boundary of interest.

This contribution discusses a multidisciplinary shape optimization problem in the isogeometric setting. In particular we consider a fluid-structure interaction (FSI) problem with the shape of a bent pipe and a flexible part of the boundary. Another part of the boundary is subject to optimization. The single field problems, fluid, structure and fluid mesh, are introduced and their numerical treatment is discussed. References for isogeometric FSI are [3, 2].

This article is organized as follows: The first section introduces NURBS in a nutshell. In Section 2 the fluid-structure interaction problem is presented. Therein we address the numerical solution of the single fields as well as a solution algorithm for the coupled FSI problem. Isogeometric shape optimization and its benefits are presented in Section 3. We introduce the test case of a bent pipe together with its linear approximation to be able to compare the results with the classical case in Section 4. The article closes with conclusions in Section 5.

1 Non-Uniform Rational B-Splines (NURBS)

In this contribution we concentrate on computational domains defined by a NURBS geometry function. Therefore we introduce in very short the basics on NURBS and refer to the standard work [9].
A NURBS is a function

$$\mathbf{F} : \Omega_0 \to \Omega, \quad \mathbf{F}(\boldsymbol{\xi}) = \mathbf{x}(\boldsymbol{\xi}), \tag{1}$$

which maps a parametric domain Ω_0 to a physical domain Ω. For the remainder of this article we restrict ourselves to the planar case such that the corresponding coordinates read $\mathbf{x} = (x, y)^T$ and $\boldsymbol{\xi} = (\xi, \eta)^T$, cf. Fig. 1.

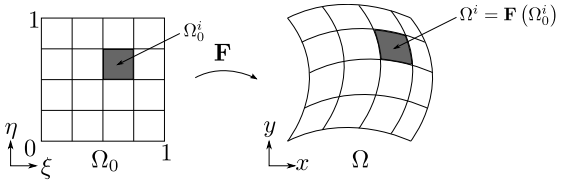

Figure 1: Mapping of the parametric domain to the physical space. A control volume Ω_0^i in parametric space and its image $\Omega^i = \mathbf{F}\left(\Omega_0^i\right)$ are highlighted in grey.

Let $\Xi = (\xi_0, \ldots, \xi_l) \in \mathbb{R}^{l+1}$ be the *knot vector* of a NURBS of degree p consisting of nondecreasing real numbers. We assume open knot vectors, i.e. the first and the last knot have multiplicity $p + 1$. Therefore the endpoints are interpolatory, which is important for representing the computational domain exactly. The ith *B-spline basis function* of p-degree $N_{i,p}$ is defined recursively as

$$N_{i,0}(\xi) = \begin{cases} 1 & \text{if } \xi_i \leq \xi < \xi_{i+1}, \\ 0 & \text{otherwise;} \end{cases} \tag{2}$$

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi). \tag{3}$$

Note that the quotient $0/0$ is assumed to be zero.

In one dimension, a NURBS of degree p is then given by

$$R_{i,p}(\xi) = \frac{w_i N_{i,p}(\xi)}{\sum_{j \in \mathcal{J}} w_j N_{j,p}(\xi)} \tag{4}$$

with B-splines $N_{i,p}$, *weights* $w_i \in \mathbb{R}$, and an index set $\mathcal{J} = \{0, \ldots, l - p - 1\}$.

Bivariate NURBS are constructed via (suppressing the degrees p_ξ and p_η)

$$R_{k\ell}(\xi, \eta) = \frac{w_{k\ell} N_k(\xi) N_\ell(\eta)}{\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} w_{ij} N_i(\xi) N_j(\eta)}. \tag{5}$$

This representation requires knot vectors $\Xi_\xi = (\xi_0, \ldots, \xi_{l_1})$ and $\Xi_\eta = (\eta_0, \ldots, \eta_{l_2})$ for each parameter direction. The index sets are $\mathcal{I} = \{0, \ldots, l_1 - p_\xi - 1\}$ and $\mathcal{J} = \{0, \ldots, l_2 - p_\eta - 1\}$.

A *single patch* NURBS parameterization of the physical domain consists of the geometry function

$$\mathbf{F}(\xi,\eta) = \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} R_{ij}(\xi,\eta)\mathbf{c}_{ij} \tag{6}$$

with bivariate NURBS R_{ij} defined on $\Omega_0 = [\xi_0, \xi_{l_1}] \times [\eta_0, \eta_{l_2}]$ and *control points* $\mathbf{c}_{ij} \in \mathbb{R}^2$. The shape of Ω is thus defined by the position of the control points, the weights, the knot vectors Ξ_ξ, Ξ_η, and the degrees p_ξ, p_η. Changing any of these results in a different geometry.

2 Isogeometric Fluid-Structure Interaction

This section introduces the fluid-structure interaction test case 'bent pipe'. We outline the used models to describe the fluid and structural behaviour as well as their discretization scheme. Furthermore we present a method on how to move the fluid mesh without gaps or overlaps at the interface. Finally a standard solution algorithm for the FSI problem is presented.

2.1 Test Case

Throughout the whole article we consider the 'bent pipe' problem which is depicted in Fig. 2. The structure (dark grey) is fixed at both ends and deforms according to the load exerted by the fluid (light grey) at the common interface. Concerning the fluid problem we impose a parabolic inlet profile on the left and an outlet condition at the bottom. The other two parts of the boundary, which are described by exact circles in the initial configuration, are equipped with 'no-slip' conditions. The inner circle will be subject to shape optimization, which will be dealt with in the following section. In appendix A the corresponding NURBS data can be found for both the fluid and the structural domain.

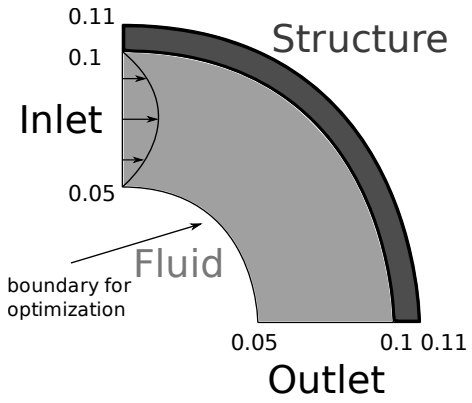

Figure 2: Initial geometry of FSI problem with fluid (light grey) and structural domain (dark grey)

In Section 4 we present results based on two different datasets of physical parameters. They are listed in 1(a) and 1(b), respectively, where u_{max} denotes the maximal inlet velocity, ν^f the kinematic viscosity, ρ the mass density of the fluid, Re the resulting Reynolds number, E Young's modulus and ν^s Poisson's ratio.

2.2 Fluid Problem

To describe the behavior of the fluid we employ the incompressible Navier-Stokes equations

$$-\nu^f \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla)\, \mathbf{u} + \frac{1}{\rho} \nabla p = \mathbf{0}, \tag{7}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{8}$$

Table 1: Parameters for the fluid and the structural problem

(a) Dataset 1

Fluid		**Structure**	
parameter	value	parameter	value
u_{max}	0.01	E	5
ν^f	0.01	ν^s	0.25
ρ	1		
Re	0.05		

(b) Dataset 2

Fluid		**Structure**	
parameter	value	parameter	value
u_{max}	0.01	E	10
ν^f	$5 \cdot 10^{-6}$	ν^s	0.25
ρ	1000		
Re	100		

the standard model in FSI. The balance of momentum (7) and the continuity equation (8) are defined on a domain $\Omega^f \subset \mathbb{R}^2$ by a NURBS geometry function. Here, $\mathbf{u} : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}^2$ represents the unknown velocity field and $p : \Omega \rightarrow \mathbb{R}$ the unknown pressure.
The fluid equations are solved numerically by means of a finite volume method which was extended to NURBS geometries [6]. It is based on a transformation of the arising integral formulations to the parametric domain. A mesh generation step can be omitted as the grid inherently defined by the knot vectors of the NURBS is used. Figure 1 depicts meshes in parametric and physical space and a resulting control volume in both domains. On the parametric domain we apply the midpoint rule and the central difference scheme whereas geometry information is integrated with higher-order quadrature rules. In that way CAD data is preserved exactly. After the incorporation of boundary conditions the arising non-linear system of equations is solved by an underrelaxed Newton's method. For an in-depth presentation of the method we refer to [6].

2.3 Linear Elasticity and Moving Mesh

The deformation of an elastic body is typically modelled in terms of the displacement field $\mathbf{d}(\mathbf{x}) \in \mathbb{R}^2$. It satisfies the balance equation

$$\text{div}\boldsymbol{\sigma}(\mathbf{d}) = \mathbf{0} \qquad \text{in } \Omega_s \subset \mathbb{R}^2 \tag{9}$$

with $\boldsymbol{\sigma}(\mathbf{d})$ denoting the Cauchy stress tensor in our case. The above equation is discretized by means of isogeometric analysis and we refer to [1, 10] concerning the details. In our context, it is important to realize that the numerical solution $\mathbf{d}_h$ of (9) by means of a NURBS-based Galerkin projection leads to a discrete displacement field that is a linear combination of the NURBS R^s_{ij} that

parameterize Ω_s (or a refinement thereof),

$$\mathbf{d}(\mathbf{x}) \mapsto \mathbf{d}_h(\mathbf{x}) = \sum_{i,j} R^s_{ij}(\mathbf{F}^s(\mathbf{x})^{-1})\mathbf{q}_{ij}, \tag{10}$$

with geometry function $\mathbf{F}^s$ of the structural domain and displacement coefficients $\mathbf{q}_{ij} \in \mathbb{R}^2$.

In FSI, the structure deforms according to the load exerted by the fluid, which requires adapting the fluid mesh to the new shape. A standard way to compute the moving fluid mesh is to solve the Laplace equation

$$\Delta \mathbf{u}^g = \mathbf{0} \qquad \text{in } \Omega_f \subset \mathbb{R}^2 \tag{11}$$

for the fluid mesh displacement $\mathbf{u}^g : \Omega_f \to \mathbb{R}^2$, where Ω_f is the fluid domain in physical space. Zero Dirichlet boundary conditions are set where the mesh remains fixed and combined with a prescribed coupling with the displacement of the structure at the common interface, see below.

We employ isogeometric analysis in order to obtain a new global fluid geometry function in terms of the already available NURBS basis functions. Again, the numerical solution of the mesh equation is then a linear combination of the same NURBS that were used to define the geometry. As it turns out, this approach leads to a particularly simple update procedure for the fluid domain.

Let the solution of the stationary moving mesh problem be

$$\mathbf{u}^g_h(\mathbf{x}) = \sum_{i,j} R_{ij}(\mathbf{F}^f(\mathbf{x})^{-1})\mathbf{w}_{ij} \tag{12}$$

with coefficients $\mathbf{w}_{ij} \in \mathbb{R}^2$. The updated fluid geometry function $\tilde{\mathbf{F}}^f$ is then computed from the old geometry function $\mathbf{F}^f$ and the solution of the moving mesh equation (12) via

$$\tilde{\mathbf{F}}^f = \mathbf{F}^f + \mathbf{u}^g_h = \sum_{i,j} R_{ij}\mathbf{c}_{ij} + \sum_{i,j} R_{ij}\mathbf{w}_{ij} = \sum_{i,j} R_{ij}\left(\mathbf{c}_{ij} + \mathbf{w}_{ij}\right), \tag{13}$$

which is performed by just adding the solution coefficients $\mathbf{w}_{ij}$ to the old control points $\mathbf{c}_{ij}$. In that way it is possible to prevent gaps or overlaps at the interface even in the case of non-matching grids if certain conditions are fulfilled(cf. [6]). This is a major advantage compared to classical methods.

2.4 The Solution Algorithm

The whole FSI problem can be formulated compactly with the non-linear FSI equation

$$\mathbf{d}|_{\text{if}} = S^s\left(S^f(\mathbf{d}|_{\text{if}})\right), \tag{14}$$

where $\mathbf{d}|_{\text{if}}$ denotes the restriction of $\mathbf{d}$ to the interface. The operators S^f and S^s represent the solution of the fluid and the structural problem, respectively, and are explained in detail in [6]. After discretization, (14) can be solved by fixed-point iteration, cf. [8]. To keep the notation simple, we do not mark here discretized quantities by an extra index h and write the iteration in the form

$$\bar{\boldsymbol{\lambda}}^{i+1}_c = S^s\left(S^f(\boldsymbol{\lambda}^i_c)\right), \tag{15}$$

where $\boldsymbol{\lambda}_c$ is the corresponding vector of solution coefficients of $\mathbf{d}|_{\text{if}}$.

In order to guarantee and accelerate convergence we employ a relaxation,

$$\boldsymbol{\lambda}_c^{i+1} = \omega_i \bar{\boldsymbol{\lambda}}_c^{i+1} + (1-\omega_i)\boldsymbol{\lambda}_c^i \tag{16}$$

with relaxation parameter ω_i which can be set fixed or dynamic. An established choice is the Aitken adaptive relaxation parameter

$$\omega_i = -\omega_{i-1} \frac{\left(\bar{\boldsymbol{\lambda}}_c^i - \boldsymbol{\lambda}_c^{i-1}\right)^T \left(\bar{\boldsymbol{\lambda}}_c^{i+1} - \boldsymbol{\lambda}_c^i - \bar{\boldsymbol{\lambda}}_c^i + \boldsymbol{\lambda}_c^{i-1}\right)}{|\bar{\boldsymbol{\lambda}}_c^{i+1} - \boldsymbol{\lambda}_c^i - \bar{\boldsymbol{\lambda}}_c^i + \boldsymbol{\lambda}_c^{i-1}|^2} \tag{17}$$

because it is easy to compute and efficient in practice [8]. Summing up, the Dirichlet-Neumann partitioned FSI coupling algorithm reads:

1. Move the fluid mesh according to $\boldsymbol{\lambda}_c^i$, solve the fluid equations $\left(S^f\right)$,
2. Transfer of the pressure forces at the interface,
3. Solve the structural problem $(S^s) \rightarrow \bar{\boldsymbol{\lambda}}_c^{i+1}$,
4. If $\frac{1}{\sqrt{\text{length}\left(\bar{\boldsymbol{\lambda}}_c^{i+1}\right)}} \cdot |\bar{\boldsymbol{\lambda}}_c^{i+1} - \boldsymbol{\lambda}_c^i| < \epsilon \rightarrow$ end,
5. Relaxation step according to (16) with Aitken relaxation (17),
6. $i = i+1$ and go to 1.

3 Isogeometric Shape Optimization

In this section we embed the FSI problem defined in Sec. 2 into an optimization loop. At first we discuss differences between classical optimization and optimization in the isogeometric setting. Then the parameters for optimization are defined, i.e. the design variables, the cost function and the optimizer.

3.1 Unified Geometry Representation

Figure 3 depicts the differences between classical and isogeometric shape optimization. In the classical case an approximate mesh is generated from the CAD data to make it suitable for the PDE (partial differential equation) solver, e.g. an FSI solver. From the results of the latter solver the current value of the cost function is computed and sent to the optimizer, which in turn suggests new design variables. These design variables consist in general of coefficients of a spline that approximates the boundary, which is subject to optimization. The reason for this is just to reduce the number of design variables drastically instead of using, e.g., the FEM nodes, which is impracticable even for coarse meshes. Afterwards the resulting new geometry has to be remeshed or can be generated from moving the old mesh according to the new shape. In the isogeometric case, by contrast, the CAD data can be directly used within the PDE solver and no mesh generation step has to be performed. Moreover, the isogeometric FSI solver additionally features a gap-free interface under certain conditions, which is not possible with classical methods in the case of non-matching grids. The cost function can again be evaluated from the solution of the FSI problem. If the fluid solver were based on isogeometric analysis the corresponding solution would also be a linear combination of NURBS basis functions. As the boundary is already given as a spline no spline fit has to be performed and thus the update of the geometry and the mesh is quite simple. In order to preserve mesh quality and to prevent self-intersections the interior control points can also be moved based on the solution of a mesh smoothing algorithm, which has already been described in connection with FSI in 2.3. So, all in all, the benefits of isogeometric design optimization are manifold.

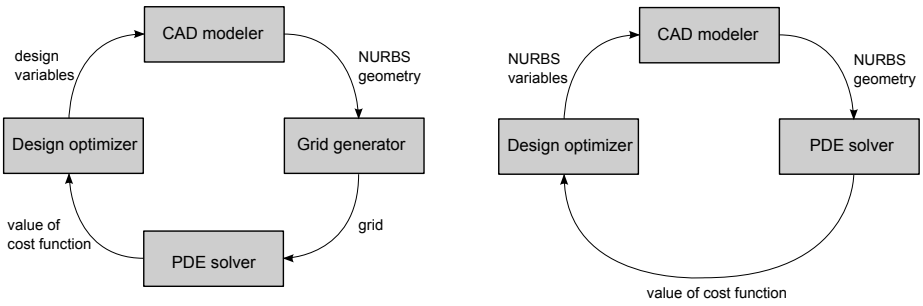

Figure 3: Optimization loop in classical (left) and isogeometric (right) setting

3.2 Parameters for Optimization

In order to define an optimization problem design variables, a cost function and an optimization algorithm have to be defined.

3.2.1 Design Variables

We define two different sets of design variables depending on control points. As on the coarsest parameterization according to appendix A we have only three control points describing the boundary to be optimized where two are fixed. Therefore we performed two uniform h-refinement steps two get more control points and consequently more flexibility in defining design variables. Moreover, a better solution can be expected.

The first case 'DV4' uses four design variables, namely the $x-$ and $y-$ components of the control points highlighted in Fig. 4 on the left. In the second case 'DV2' (see Fig. 4 the same control points are taken to define the design variables, but now we only accept movements along a line that is inclined at an angle of 45° which results in only two design variables. So the two control points of the optimized shape can be expressed as

$$\begin{pmatrix} x_i^{opt} \\ y_i^{opt} \end{pmatrix} = \begin{pmatrix} x_i^{init} \\ y_i^{init} \end{pmatrix} + \lambda_i \cdot \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}, \qquad i = 1, 2 \tag{18}$$

where λ_1 and λ_2 are the design variables.

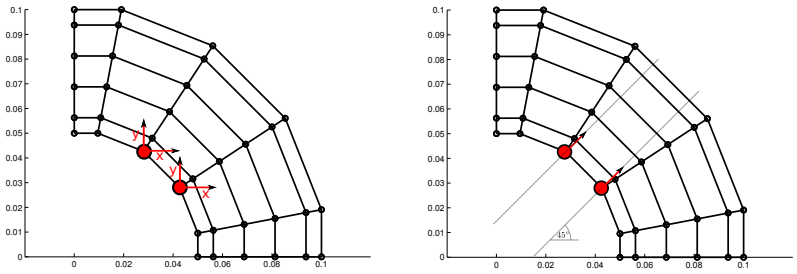

Figure 4: Control point grid after two h-refinement steps with highlighted design variables in the case 'DV4' (left) and 'DV2'

3.2.2 Cost function

The cost function f_{cost} can be formulated in terms of both the structural and the fluid solution. In the following we always use a cost function composed of the pressure drop and a constraint which bounds the length of the boundary to be optimized,i.e.

$$f_{\text{cost}} = \Delta p + \mu \cdot \max\left(0, l - l_{\text{init}}\right) \tag{19}$$

with an initial length $l_{\text{init}} = 0.025\pi$ and a choosable penalty factor μ. The pressure drop is defined as

$$\Delta p = |\overline{p}_{\text{in}} - \overline{p}_{\text{out}}| = |\overline{p}_{\text{in}}| = \left| \frac{\int_{\partial\Omega_{\text{inlet}}} p \, ds}{\int_{\partial\Omega_{\text{inlet}}} ds} \right|, \tag{20}$$

where $\overline{p}_{\text{in}}$ and $\overline{p}_{\text{out}}$ denote the mean pressure over the inlet and outlet boundary, respectively. As we model the outlet condition with $p = 0$ it follows that $\overline{p}_{\text{out}} = 0$. The current length l of the inner boundary $\partial\Omega_{\text{opt}}$ can be computed exactly via

$$l = \int_{\partial\Omega_{\text{opt}}} ds = \int_{\partial\Omega_{\text{opt},0}} |\dot{\mathbf{C}}| \, ds, \tag{21}$$

where $\mathbf{C}$ is the restriction of the fluid geometry function $\mathbf{F}$ to the boundary of interest and $\partial\Omega_{\text{opt},0}$ the corresponding boundary in parametric space.

3.2.3 Optimizer

Since the coupled state equations are highly non-linear, and because of the use of a penalization approach to take into account the constraint, one may expect to have a multi-modal cost function to minimize. Moreover, the computation of the gradient of the cost function, for instance by an adjoint method, is tedious in this case, because of the use of coupled state equations. Therefore, we have chosen to employ for this study a derivative-free optimization algorithm, able to avoid local minima.

The Covariance Matrix Adaption Evolution Strategy (CMA-ES) has been used, for its robustness and its capability to exhibit a quadratic converge rate asymptotically. This algorithm has been tested and validated on several mathematical or engineering problems[4].

This is a population-based evolutionary algorithm, that constructs iteratively a minimization path, by generating at each iteration trial points by the use of an anisotropic Gaussian perturbation. Each iteration k of the algorithm can be summarized as:

1. Generation of a population of trial points based on a covariance matrix C_k, a step size $\bar{\sigma}_k$ and a centroid $\bar{x}_k$:
$$x_i = \bar{x}_k + \bar{\sigma}_k \, N(0, C_k) \tag{22}$$
2. Update of the centroid $\bar{x}_{k+1}$ according to the best trial points ;
3. Update of the step size $\bar{\sigma}_{k+1}$ according to the cost function reduction ;
4. Update of the covariance matrix according to:
$$C_{k+1} = \underbrace{(1-c)C_k}_{\text{previous estimate}} + \underbrace{\frac{c}{m} p_k p_k^T}_{\text{1D update}} + \underbrace{c(1-\frac{1}{m}) \sum_{i=1}^{\mu} \omega^i (y^i)(y^i)^T}_{\text{covariance of parent population}}$$
with :

- p_k evolution path (moves performed during last iterations)
- $y^i = (x^i - \bar{x}_k)/\bar{\sigma}_k$

This algorithm allows the covariance matrix to converge to the inverse of the Hessian matrix, yielding an asymptotic quadratic convergence rate. All details of the implemented algorithm can be found in[5].

3.3 Comparison with Classical Methods

In order to compare the optimization procedure and the results with the classical case (cf. Fig. 3), but using the same solvers, we perform an approximation of the NURBS by linear B-splines. For that case of a bilinear element isogeometric analysis and classical FEM coincide. The process of the approximation step is depicted in Fig. 5 exemplarily for the fluid domain. We first refine the original NURBS geometry up to a desired level and then take the intersections of mesh lines (isoparametric lines) as control points to define the linear B-spline. The knot vectors have to be adapted accordingly.

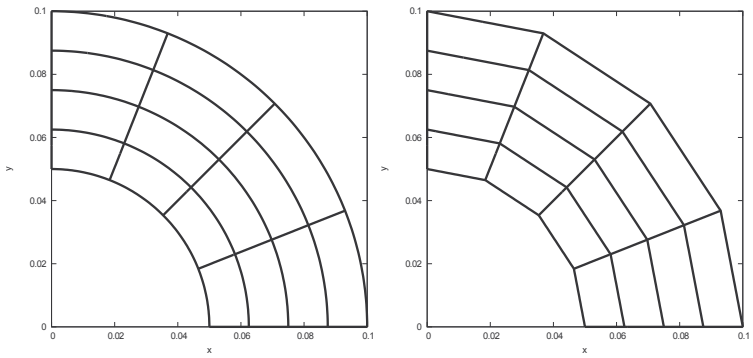

Figure 5: NURBS mesh (left) and its approximation by linear B-splines (right)

4 Results

In this section we present numerical results for the datasets defined in Sec. 2 and test cases presented in Sec. 3. We always compare the results based on the quadratic geometry function with the linear case described in Sec. 3.3, which imitates the classical procedure.

4.1 Dataset 1

We first consider dataset 1. In the quadratic case we refine the initial fluid domain five times by uniform h-refinement, which results in 3072 degrees of freedom (DOFs), and both the fluid mesh and the structural domain by four uniform h-refinement steps which leads to 648 DOFs each. For the linear case we employ 3072 DOFs for the fluid domain and 2178 DOFs for both the fluid mesh and the structural domain which is the result of five uniform h-refinement steps.

4.1.1 Test Case 'DV4'

In Fig. 6 we compare the evolution of the first two design variables (left) and the third and the forth design variable (right) for both the linear and the quadratic case. It gets apparent that the design variables coincide for the first 200 evaluations of the FSI problem but then diverge. Furthermore the convergence is slow and it is not obvious if the quadratic case has to be preferred. The corresponding cost functions are depicted in Fig. 7 on the left where only slight differences can be observed. On the right the length of the optimized boundary where it gets clear that the constraint is active in the end. Finally the FSI solution for the optimized shape in the quadratic case can be seen in Fig. 8.

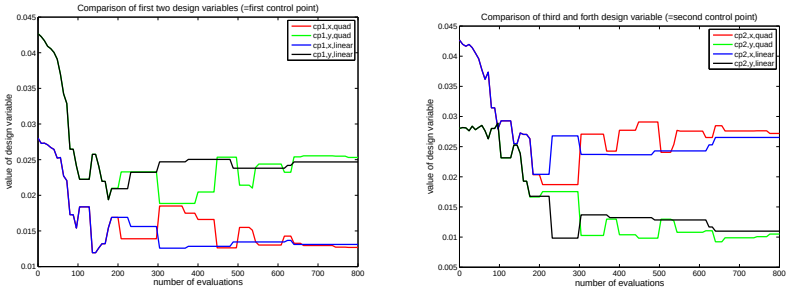

Figure 6: Comparison of first and second (left) and third and forth design variable (right) in the case of a penalty factor of 100

4.1.2 Test Case 'DV2'

Comparing the evolution of the design variables for different penalty factors $\mu = 1, 100$ (cf. Eq. (19)) in the test case 'DV2' we can observe almost identical values in the linear case (cf. Fig. 9 on the left) and exactly the same values in the quadratic case (cf. Fig. 9 on the right). The convergence is much faster compared to the test case 'DV4'. This can also be seen in Fig. 10

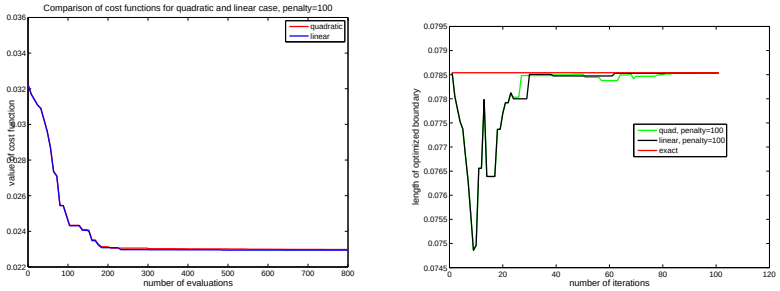

Figure 7: Comparison of cost functions for penalty factor 100 (left) and length of boundary to be optimized with constraint in yellow (right)

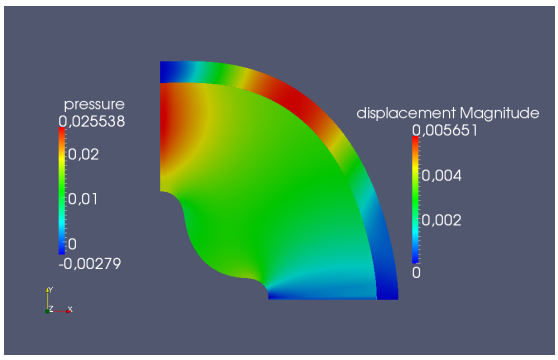

Figure 8: Pressure solution and displacement of optimized shape for penalty factor 100 in the quadratic case

where the evolution of the cost functions in both cases is depicted as well as the corresponding values of the design variables. In Fig. 11 again the length of the optimized boundary during the iteration process is shown and on the right the FSI result with the optimized shape of the inner boundary is depicted. Also here no major differences between the linear and the quadratic case can be observed.

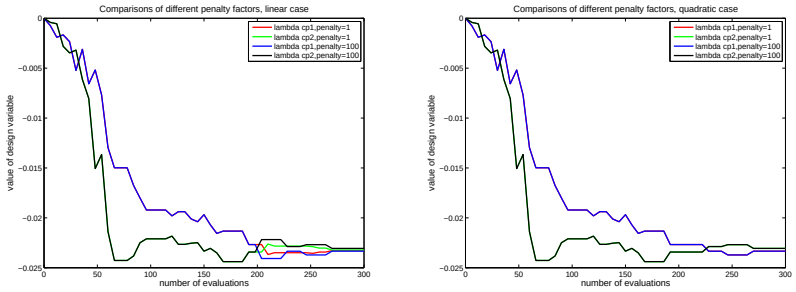

Figure 9: Design variables for penalty factors 1 and 100 in the linear (left) and the quadratic (right) case

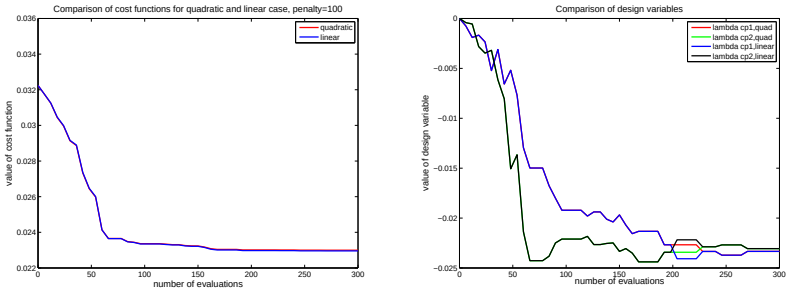

Figure 10: Cost function for linear and quadratic case (left) and the corresponding design variables (right) with a penalty factor of 100

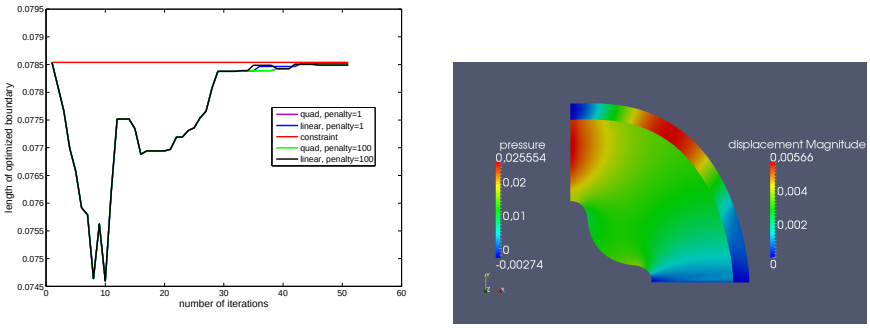

Figure 11: The length of the optimized boundary and the constraint in yellow (left) and the simulation result for the quadratic case and a penalty factor of 100 (right)

4.2 Dataset 2 with Test Case 'DV2'

As the Reynolds number is higher concerning dataset 2 and consequently the flow is more convective we employ a higher number of DOFs. In detail, we perform six uniform h-refinement steps on the fluid domain which corresponds to 12288 DOFs. Concerning the fluid mesh and the structural problem we use 2312 DOFs each in the quadratic and 8450 DOFs each in the linear case.

Again we compare the evolution of the design variables in the linear and the quadratic case for different penalty factors $\mu = 1, 100$ in Fig. 12. It can be observed that we get the same values for both penalty factors. Moreover, when regarding Fig. 13, the design variables exactly take on the same values in the linear and the quadratic case. Consequently the values of the cost functions coincide and, when we look at the length of the optimized boundary during the iteration process in Fig. 14, also identical values can be observed. On the right the FSI simulation result is depicted.

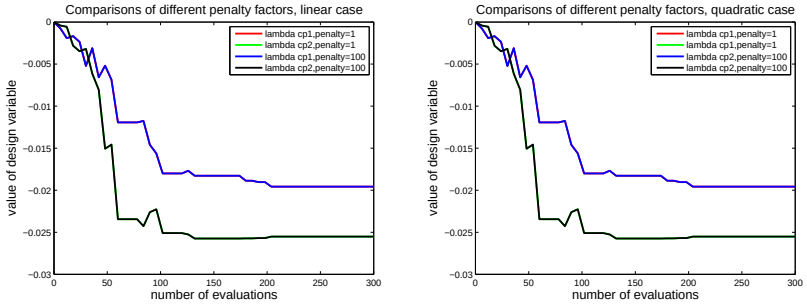

Figure 12: Design variables for penalty factors 1 and 100 in the linear (left) and the quadratic (right) case

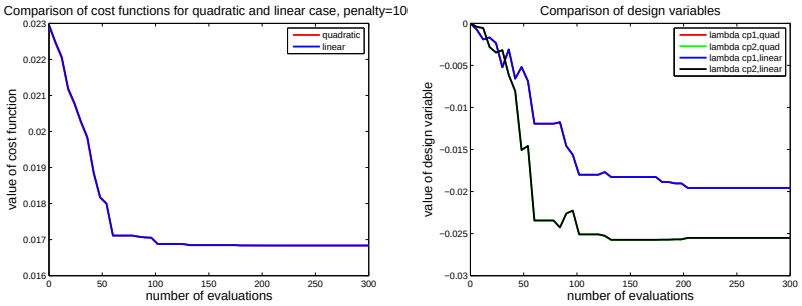

Figure 13: Cost function for linear and quadratic case (left) and the corresponding design variables (right) for a penalty factor of 100

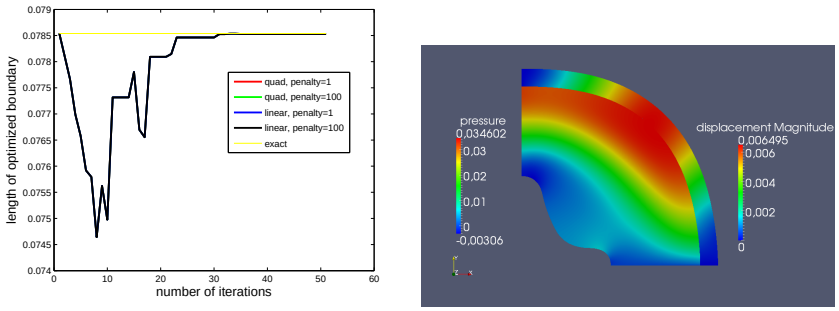

Figure 14: The length of the optimized boundary and the constraint in yellow (left) and the fsi solution for optimal shape (right)

5 Conclusions

In this article we presented a multidisciplinary design optimization framework in the isogeometric setting. With this approach geometry errors can be excluded from the very beginning within the PDE solver, i.e. a fluid-structure interaction solver in our case. In particular, the geometry given by a NURBS, a standard in CAD, can be directly used within the PDE solver without any approximation step. Moreover, the NURBS inherently defines a computational mesh by its knot vectors such that a mesh generation step can be omitted. Additionally we can ensure a matching FSI interface even in the case of non-matching grids, which cannot be fulfilled in classical methods. Concerning the definition of design variables in the optimization loop, we profit from the possibility of formulating them directly in terms of control points of the NURBS. Consequently no spline fit of the boundary, which is optimized, has to be performed.
We therefore introduced NURBS at a glance. Afterwards we presented the 'bent pipe' FSI problem together with some remarks on the used single field solvers. The incompressible Navier-Stokes equations are solved numerically with the help of a finite volume method which is adapted to NURBS geometries. Both the structural and the fluid mesh equations are discretized with isogeometric analysis. The FSI algorithm is then embedded in an optimization loop. To define the optimization problem we proposed a cost function and two different sets of design variables. In order to compare the optimization results with the classical case we approximated the geometry by linear B-splines.
The achieved results showed no major differences between the quadratic NURBS and the linear case concerning accuracy and convergence. So the main advantages of shape optimization in the isogeometric setting lies in the unified geometry representation such that no spline fit has to be performed and control points can be directly taken as design variables. The usage of a 'full' isogeometric FSI solver, i.e. with a fluid solver based on isogeometric analysis, could improve the results and is an interesting field of future research.

A NURBS Data for Bent Pipe Problem

A.1 Fluid Domain

```
knotVectorXi = ( 0, 0, 0, 1, 1, 1 );
knotVectorEta = ( 0, 0, 0, 1, 1, 1 );
weights = ( 1, 0.707106781, 1, 1, 0.707106781, 1, 1, 0.707106781, 1 );
controlpoints = [0.0, 0.05,
                 0.05, 0.05,
                 0.05, 0.0,
                 0.0, 0.075,
                 0.075, 0.075,
                 0.075, 0.0,
                 0.0, 0.1,
                 0.1, 0.1,
                 0.1, 0.0
                ];
```

A.2 Structural Domain

```
knotVectorXi =   ( 0, 0, 0, 1, 1, 1 );
knotVectorEta =  ( 0, 0, 0, 1, 1, 1 );
weights =    ( 1, 0.707106781, 1, 1, 0.707106781, 1, 1, 0.707106781, 1 )
controlpoints = [0, 0.1,
                 0.1, 0.1,
                 0.1, 0,
                 0, 0.105,
                 0.105, 0.105,
                 0.105, 0,
                 0, 0.11,
                 0.11, 0.11,
                 0.11, 0
                ];
```

References

[1] Aigner, M., Heinrich, C., Jüttler, B., Pilgerstorfer, E., Simeon, B., and Vuong, A.-V. Swept volume parametrization for isogeometric analysis. In *The Mathematics of Surfaces (MoS XIII 2009)* (2009), E. Hancock and R. Martin, Eds., Springer.

[2] Bazilevs, Y., Calo, V., Hughes, T., and Zhang, Y. Isogeometric Fluid-Structure Interaction: Theory, Algorithms and Computations. *Computational Mechanics 43* (2008), 3–37.

[3] Bazilevs, Y., Calo, V., Zhang, Y., and Hughes, T. Isogeometric Fluid-Structure Interaction Analysis with Applications to Arterial Blood Flow. *Computational Mechanics 38*, 4-5 (2006), 310–322.

[4] Hansen, N., and Kern, S. Evaluating the CMA evolution strategy on multimodal test functions. In *Parallel Problem Solving from Nature PPSN VIII* (2004), X. Yao et al., Eds., vol. 3242 of *LNCS*, Springer, pp. 282–291.

[5] Hansen, N., Muller, S., and Koumoutsakos, P. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). *Evolutionary Computation 11*, 1 (2003), 1–18.

[6] Heinrich, C., Simeon, B., and Boschert, S. A Finite Volume Method on NURBS Geometries and its Application in Isogeometric Fluid-Structure Interaction (submitted). 2011.

[7] Hughes, T., Cottrell, J., and Bazilevs, Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Comput. Methods Appl. Mech. Engrg. 194* (2005), 4135–4195.

[8] Küttler, U., and Wall, W. A. Fixed-point fluid-structure interaction solvers with dynamic relaxation. *Comp. Mech. 43*, 1 (2008), 61–72.

[9] Piegl, L., and Tiller, W. *The NURBS Book*, 2nd ed. Springer, 1996.

[10] Vuong, A.-V., and Simeon, B. On isogeometric analysis and its usage for stress calculation. (accepted), 2009.

Contents

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399