Shape optimisation tools for CFD analysis: ANSYS Fluent, RBF Morph and modeFRONTIER

Dr. Marco Evangelos Biancolini
Tor Vergata University, Rome, Italy
Outline

• Introduction
  • Problem definition
  • Goals
  • Research path
• RBF Morph tool presentation
  • Morphing & Smoothing
  • The Aim of RBF Morph
  • RBF Morph Features
  • Background
  • How It Works
• Industrial Application: a motorbike windshield optimisation
  • Calculation tools and hardware resources
  • Strategy used for driving CFD solution using modeFRONTIER
  • Industrial problem definition
  • Morphing Set Up
  • Results
Problem definition

• Product developers must quickly perform and test numerous design variations.
• Simulation Driven Product Development process requires fast methods to handle parametric shapes.
• Mesh morphing has emerged as a meaningful answer to this need.
• A powerful industrial solution is given by the use of Radial Basis Functions (RBF) Technology implemented in the ANSYS Fluent add-on RBF Morph.
Goals

- Defining a shape parametric CFD model using ANSYS Fluent and RBF Morph.
- Coupling of the parametric CFD model with the optimization tool modeFRONTIER
- Steering the solution to an optimal design
- Importing in the CAD the new design
Research Path

- *RBF Morph* development has been driven by a Formula 1 Top Team to fulfill their demanding morphing needs
- *RBF Morph* is an ANSYS Inc. Partner since 2009
- A research has been sponsored by EnginSoft (licenses and a lot of support!) to investigate if a coupling with modeFRONTIER was feasible
- Very good results were achieved using the proposed approach in a complex calculation environment (CFD solution running on a Linux cluster, MF running on a Windows laptop)
RBF Morph presentation outline

• Morphing & Smoothing
• The Aim of RBF Morph
• RBF Morph Features
• Background
• How It Works

RBF Morph is an embedded tool. This means that there is no need of saving several meshes. The original Fluent case becomes a truly shape parametric CFD model. A single command line in the journal file allows to update the mesh in the new configuration combining as much shape modifications as needed. The morphing module is fully integrated in the solving stage (including parallel runs) and allows to handle very large models (hundreds millions of cells).

Optimisation process can be automated using: internal DOE tools of RBF Morph, scripts, custom software, calculation Worksheets. MathCAD has proven to be an effective tool for post processing.

The coupling with the modeFRONTIER optimiser has been successfully tested in a complex environment, running MF on a Windows local machine and Fluent on a LINUX HPC cluster.

The integration with Workbench has been successfully implemented. Workbench shape parameters are defined and they can be steered using the module DesignXplorer.

2010 EnginSoft International Conference
21-22 October Montichiari (BS)
Morphing & Smoothing

- A mesh morpher is a tool capable to perform mesh modifications, in order to achieve arbitrary shape changes and related volume smoothing, without changing the mesh topology.

- In general a morphing operation can introduce a reduction of the mesh quality

- A good morpher has to minimize this effect, and maximize the possible shape modifications.

- If mesh quality is well preserved, then using the same mesh structure it’s a clear benefit.
The Aim of RBF Morph

- The aim of RBF Morph is to perform fast mesh morphing using a mesh-independent approach based on state-of-the-art RBF (Radial Basis Functions) techniques.

- The use of RBF Morph allows the CFD user to perform shape modifications, compatible with the mesh topology, directly in the solving stage, just adding a single command line in the input file.

- The final goal is to perform parametric studies of component shapes and positions typical of the fluid-dynamic design like:
  - Design Developments
  - Multi-configuration studies
  - Sensitivity Studies
  - DOE (Design Of Experiment)
  - Optimization

(rbf-morph '((("sol-1" amp-1) ("sol-2" amp-2)...("sol-n" amp-n)))
**RBF Morph Features**

- **Add on** fully integrated within Fluent (GUI, TUI & solving stage)

- **Mesh-independent** RBF fit used for surface mesh morphing and volume mesh smoothing

- **Parallel** calculation allows to morph large size models (many millions of cells) in a short time

- Management of **every kind of mesh** element type (tetrahedral, hexahedral, polyhedral, etc.)

- Support of the **CAD re-design** of the morphed surfaces

- **Multi fit** makes the Fluent case truly parametric (only 1 mesh is stored)

- **Precision**: exact nodal movement and exact feature preservation.
New upcoming RBF Morph features

- RBF allows to **exactly** prescribe surfaces movements, this opens several opportunities, among them:
  - **Target surfaces** (STL). The new shape can be directly defined in the CAD, meshed and used as a morphing target thanks to the **RBF algorithm** that allows to project a mesh onto another (non conformal) one.
  - FEM deformed shape (static, modal). The **structural solution** obtained on the (usually) non conformal FEM mesh is used to generate the **morphing field** applied to the CFD mesh. For complex shapes an advanced two steps approach is available for mapping FEM property IDs onto CFD threads IDs.
Some investigated applications

- F1 car aerodynamics
  - shape optimisation of wings and deflectors
  - wheel steering including tyre deformation
  - effect of adjustable surfaces
- Tuning of a motorbike windshield
  - deflector shape and set-up
  - driver size and position
- Shape optimisation of an air-box
  - air flow balancing
  - max volumetric efficiency
- More examples on the web:
  - http://www.rbf-morph.com/
  - http://www.youtube.com/user/RbfMorph
Background: RBF Theory

- A system of **radial functions** is used to **fit** a **solution** for the mesh movement/morphing, from a list of source points and their displacements. This approach is valid for both surface shape changes and volume mesh smoothing.

- The RBF problem definition does not depend on the mesh

- Radial Basis Function interpolation is used to derive the displacement in any location in the space, so it is also available in every grid node.

- An interpolation function composed by a radial basis and a polynomial is defined.

\[ s(x) = \sum_{i=1}^{N} \gamma_i \phi(\|x - x_i\|) + h(x) \]

\[ h(x) = \beta + \beta_1 x + \beta_3 y + \beta_4 z \]
Background: RBF Theory

- A radial basis fit exists if desired values are matched at source points with a null poly contribution.
- The fit problem is associated with the solution of a linear system.
- \( \mathbf{M} \) is the interpolation matrix.
- \( \mathbf{P} \) is the constraint matrix.
- \( \mathbf{g} \) are the scalar values prescribed at source points.
- \( \gamma \) and \( \beta \) are the fitting coefficients.

\[
\begin{align*}
  s(\mathbf{x}_{k_i}) &= g(\mathbf{x}_{k_i}) \quad 1 \leq i \leq N \\
  0 &= \sum_{i=1}^{N} \gamma_i q(\mathbf{x}_{k_i}) \\
  \begin{pmatrix} 
  \mathbf{M} & \mathbf{P} \\
  \mathbf{P}^T & 0 
  \end{pmatrix} 
  \begin{pmatrix} 
  \gamma \\
  \beta 
  \end{pmatrix} &= 
  \begin{pmatrix} 
  \mathbf{g} \\
  \mathbf{0} 
  \end{pmatrix} \\
  \mathbf{M}_{ij} &= \phi(\|\mathbf{x}_{k_i} - \mathbf{x}_{k_j}\|) \quad 1 \leq i \quad j \leq N \\
  \mathbf{P} &= 
  \begin{pmatrix} 
  1 & x_{k_1}^0 & y_{k_1}^0 & z_{k_1}^0 \\
  1 & x_{k_2}^0 & y_{k_2}^0 & z_{k_2}^0 \\
  \vdots & \vdots & \vdots & \vdots \\
  1 & x_{k_N}^0 & y_{k_N}^0 & z_{k_N}^0 
  \end{pmatrix} 
\end{align*}
\]
Background: RBF Theory

• The radial function can be fully or compactly supported. The bi-harmonic kernel fully supported gives the best results for smoothing.

• For the smoothing problem each component of the displacement prescribed at the source points is interpolated as a single scalar field.

Radial Basis Function

<table>
<thead>
<tr>
<th>Radial Basis Function</th>
<th>( \phi(r) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spline type ((R_n))</td>
<td>(</td>
</tr>
<tr>
<td>Thin plate spline ((TPS_n))</td>
<td>(</td>
</tr>
<tr>
<td>Multiquadric((MQ))</td>
<td>(\sqrt{1+r^2})</td>
</tr>
<tr>
<td>Inverse multiquadric ((IMQ))</td>
<td>(\frac{1}{\sqrt{1+r^2}})</td>
</tr>
<tr>
<td>Inverse quadratic ((IQ))</td>
<td>(\frac{1}{1+r^2})</td>
</tr>
<tr>
<td>Gaussian ((GS))</td>
<td>(e^{-r^2})</td>
</tr>
</tbody>
</table>

\[
\begin{align*}
v_x = s_x(x) &= \sum_{i=1}^{N} \gamma_i^x \phi(\|x-x_{k_i}\|) + \beta_1^x + \beta_2^x x + \beta_3^x y + \beta_4^x z \\
v_y = s_y(x) &= \sum_{i=1}^{N} \gamma_i^y \phi(\|x-x_{k_i}\|) + \beta_1^y + \beta_2^y x + \beta_3^y y + \beta_4^y z \\
v_z = s_z(x) &= \sum_{i=1}^{N} \gamma_i^z \phi(\|x-x_{k_i}\|) + \beta_1^z + \beta_2^z x + \beta_3^z y + \beta_4^z z
\end{align*}
\]
Background: accelerating the solver

• The evaluation of RBF at a point has a cost of order $N$

• The fit has a cost of order $N^3$ for a direct fit (full populated matrix); this limit to $\sim 10,000$ the number of source points that can be used in a practical problem

• Using an iterative solver (with a good pre-conditioner) the fit has a cost of order $N^2$; the number of points can be increased up to $\sim 70,000$

• Using also space partitioning to accelerate fit and evaluation the number of points can be increased up to $\sim 300,000$

• The method can be further accelerated using fast pre-conditioner building and FMM RBF evaluation…
Background: solver performances escalation

- 10,000 RBF centers FIT
  - 120 minutes Jan 2008
  - 5 seconds Jan 2010
- Largest fit 2,600,000 in 133 minutes
- Largest model morphed 300,000,000 cells
- Fit and Morph a 100,000,000 cells model using 500,000 RBF centers within 15 minutes
- Front wing flap rotation up to +/-6° (+/-8° enabling Fluent remeshing)

<table>
<thead>
<tr>
<th>#points</th>
<th>2010 (Minutes)</th>
<th>2008 (Minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3,000</td>
<td>0 (1s)</td>
<td>15</td>
</tr>
<tr>
<td>10,000</td>
<td>0 (5s)</td>
<td>120</td>
</tr>
<tr>
<td>40,000</td>
<td>1 (44s)</td>
<td>Not registered</td>
</tr>
<tr>
<td>160,000</td>
<td>4</td>
<td>Not registered</td>
</tr>
<tr>
<td>650,000</td>
<td>22</td>
<td>Not registered</td>
</tr>
<tr>
<td>2,600,000</td>
<td>133</td>
<td>Not registered</td>
</tr>
</tbody>
</table>
How it Works: the work-flow

• *RBF Morph* basically requires three different steps:

  • **Step 1 [SERIAL] setup** and definition of the problem (source points and displacements).

  • **Step 2 [SERIAL] fitting** of the RBF system.

  • **Step 3 [SERIAL or PARALLEL] morphing** of the surface and volume mesh (available also in the CFD solution stage).
How it Works: the problem setup

• The problem must describe correctly the **desired changes** and must **preserve exactly** the fixed part of the mesh.

• The prescription of the **source points** and their displacements fully defines the *RBF Morph* problem.

• The problem is **mesh-independent**, and could be defined using grid nodes as well as arbitrary point locations.

• Each problem and its fit define a mesh **modifier** or a **shape parameter**.
How it Works: the interface

- One of the key aspects of *RBF Morph*, in respect to FLUENT integration, is related to the ability of extracting information from the FLUENT mesh and to the user interface GUI.
License Start: Fri Oct 01 00:00:00 2010
License End: Fri Jan 28 23:00:00 2011
Current Date: Wed Oct 20 10:29:10 2010

RBF Morph is running in the licensing period.
Product license will expire in 101 days.

Done.
RBF Library loaded, Version 1.2
RBF Model reset done.
RBF Library closed.
Industrial Application: a motorbike windshield optimisation

• Outline
  • Calculation tools and hardware resources
  • Strategy used for driving CFD solution using modeFRONTIER
  • Industrial problem definition
  • Morphing Set Up
  • Results
Calculation tools and hardware resources

- CFD solver: ANSYS Fluent (Linux version)
- Mesh morpher: RBF Morph embedded in Fluent
- Optimizer: modeFRONTIER
- Hardware:
  - HPC calculation node with 2 AMD Opteron Dual Core Processors (4 cores), 16Gb RAM, Fedora 64 bit;
  - laptop with Windows XP 32 bit
Strategy used for driving CFD solution using modeFRONTIER

- MF creates a new folder for each new design; all required files has to be specified in Support_Fluent
- Input is controlled linking the input variable with a specific location in the Fluent journal file that is patched by MF to current value
- Output is extracted from output files and linked to an output variable
- The scheduler runs Fluent acting on the SSH (or DOS) node
- Fluent case (*.cas) and RBF solutions (*.rbf, *.sol) were removed by the support to avoid I/O bottlenecks; symbolic links to a remote copy of such files are generated by the ssh script
Increasing complexity test were used to set-up the coupling

1. A simple cube model controlling inlet velocity vs. drag force (no-shape, DOS)
2. As 1, but running fluent on the remote LINUX server using SSH node.
3. As 2 but changing the attitude angle vs. drag force (RBF Morph, LINUX)
4. As 3 but using 2 shape parameters (scale the top in the two transversal directions)

(rbf-morph '(("scale-top-x" 0.5)("scale-top-y" 0.5)))
Problem definition

- Variotouring windshield introduced in 2002 by the German company MRA to control the shape of the flux that acts on the driver.

- The system acts as a flux splitter and if properly tuned allows to obtain a substantial benefit in terms of riding comfort.

- Acquisition of the actual geometry of part of the motorbike by means of a reverse engineering tool

- Calculation mesh and CFD model of the baseline geometry
Explored modifiers

1. Changing of driver height 
   \([-5 \text{ cm}, 0 \text{ cm}, 5 \text{ cm}];\)

2. Changing of driver position 
   acting on the hunching angle \([0^\circ, 7.5^\circ, 15^\circ];\)

3. Adjustment of the vario-touring acting on the 
   deflector angle \([-10^\circ, -5^\circ, 0^\circ, 5^\circ, 10^\circ];\)
Set up of RBF Morph

- The morphed action is limited in the box region “domain 1”.
- The motion of the surfaces inside the encapsulation domain is imposed to the points on the windshield (fixed), the fairing (fixed) and the helmet (moving).
Set-up of modeFRONTIER Workflow

(rfb-morph '(("rotate-driver" 0.5)("rotate-deflector" 0.5)("height-driver" 0.5))')
Results
Results

- Driver height and position have a substantial effect on flow pattern.
- Deflector angle plays an important role on the flow pattern encountered by the driver.
Concluding remarks

- A **shape optimization** procedure for CFD problems has been successfully defined:
  - The CFD solver ANSYS Fluent powered by the mesh morphing add-on *RBF Morph* allows to get a truly shape parametric CFD model
  - The parametric CFD model can be **steered** using the optimization tool **modeFRONTIER**
- The approach has proven to be **very useful** for the presented **industrial application** demonstrating that a parametric motorbike model allows to define **optimal** configurations for **driver position** and windshield **deflector angle**.
- The approach can be easily extended to a **wide range of CFD** applications considering that is based on **general purpose** commercial tools (**modeFrontier**, ANSYS Fluent, RBF Morph)
- The **meshless** nature of RBF Morph allows to further extend the integration to **multi-objective multi-physics** problems (for instance the shape of FEM and CFD can be synchronized right now)
Have a look to the following posters:

Application of Radial Basis Functions to blood flow pattern interpolation

Biancolini ME¹, Ponzini R², Rizzo G³, Readelli A⁴, Morbiducci U⁵

1 Mechanical Engineering Dept., Università di Tor Vergata, Rome, Italy; 2 CILEA High Performance Computing Dept., Milan, Italy; 3 Institute of Molecular Bioimaging and Physiology (CNR), Milan, Italy; 4 Bioengineering Dept., Politecnico di Milano, Milan, Italy; 5 Mechanics Dept., Politecnico di Torino, Turin, Italy
MULTI OBJECTIVE OPTIMISATION BY MEANS OF MESH MORPHING: APPLICATION OF RBF MORPH TO A FSAE CAR AIRBOX DESIGN

E. Abbasciano, M.E. Biancolini, M. Urbinati

Università di Roma “Tor Vergata”, Dipartimento di Meccanica,
Via del Politecnico, 1 – 00133 Roma, e-mail: biancolini@ing.uniroma2.it

In this study a new mesh morphing approach for shape optimisation is presented through an industrial application: the structural and fluid dynamic optimization of a FSAE car airbox. The current configuration of the inlet system of the FSAE car TV-480R (STV Team, University of Rome Tor Vergata) suffers a consistent charging imbalance between internal and external runners. A new design needs to be optimised to find the optimal shape that minimize charging imbalance whilst preserving an overall volumetric efficiency of the engine. This challenging optimization task was made possible thanks to numerical CFD (Fluent) and FEM (Nastran) simulations parameterised by means of the mesh morphing tool RBF Morph.
RBF Morph driven by DesignXplorer

The fastest morphing add on for shape optimisation using Fluent can now be integrated in the ANSYS Workbench platform.

Product developers must quickly perform and test numerous design variations in an environment steeped in complex customer requirements and short development cycles. Faced with increasing competition, companies have to produce higher performing products and deliver in shorter time frames to remain competitive. The need to innovate has never been greater. Mesh morphing has emerged as a meaningful technology given its ability to accelerate the Simulation Driven Product Development process. RBF Morph is a unique morpher that combines a very accurate control of the geometrical parameters with an extremely fast mesh deformation, fully integrated in the CFD solving process.

The tool, initially released as RBF Morph 1.1 in July 2009, was presented at the European Automotive Simulation Conference, where it has been awarded as the "Most Advanced Approach using integrated and combined simulation methods". A further step forward has been made with the integration between RBF Morph and ANSYS Workbench. The major benefit is the coupling with any optimization tool, including the easy-to-use, workbench embedded, DesignXplorer, that makes the optimisation task effectively straightforward.

The new functionality is first described through two fundamental steps: the shape parameterisation set up within Fluent by means of the RBF Morph add on and the definition of shape parameters linked to Workbench variables. Then the shape parameters can be steered directly within Workbench and the CFD solution performed efficiently by Fluent with no I/O expenses, thanks to RBF Morph.

The CFD calculation performed by Fluent becomes a solver box capable to evaluate objective functions in response to the desired input parameters, from boundary conditions to geometrical changes. As illustrated in the last part of the example, such powerful “knobs” can be used for a “what if” analysis as well as for a fully automated and efficient optimisation process, using DesignXplorer.

The proposed approach is here demonstrated with an industrial application: shape optimisation of a motorbike windshield.
Thank you for your attention!

Dr. Marco Evangelos Biancolini

E-mail: biancolini@ing.uniroma2.it
Web: www.rbf-morph.com