Mesh adaptation strategies using wall functions and low-Reynolds models

Loic Frazza, Adrien Loseille, Frédéric Alauzet

Mesh adaptation strategies using wall functions and low-Reynolds models

Loïc Frazza*, Adrien Loseille† and Frédéric Alauzet‡

Sorbonne Universités, UPMC, 4 place Jussieu 75252 Paris cedex 05, France

Gamma3 Team, Inria Saclay, 91120 Palaiseau, France

The scope of this paper is to determine an optimal mesh adaptation strategy to compute turbulent flows in presence of solid bodies using RANS models. To this end we propose to use additionally model specific wall functions when the low-Reynolds turbulence model is not sufficiently resolved. Such wall functions degenerate to the low-Reynolds turbulence model they mimic when the mesh size tends to 0. This significantly improves solutions on coarse initial grids and fasten computations toward the final solution.

I. Introduction

Low-Reynolds turbulence models have gained popularity over high-Reynolds turbulence models with wall functions as they can be integrated down to the wall, so that they do not require to actually chose an arbitrary wall distance. However, to do so, they require extremely fine meshes near walls to properly discretize the steep variations of the solution and precisely integrate the source terms involved by the low-Reynolds turbulence models near the walls (typically under $y^+ = 10$). This, dramatically slow down the convergence of flow solvers: most of the computational resources are concentrated in the boundary layers, deteriorating mean flow, the mesh sizes in boundary layers limits the overall CFL, strongly nonlinear terms of the turbulence model slows iterative solvers (Newton's method, ...), and the numerous layers also slows the convergence of linear solvers. Further more, in a mesh adaptation context, the initial grids are usually too coarse to allow a proper resolution of the boundary layers or have to be carefully hand-made, reducing the interest of mesh adaptation.

On the contrary, wall functions model the inner part of the boundary layers so that no low-Reynolds turbulence model is needed. Traditional high-Reynolds turbulence models are less stiff than their low-Reynolds counter parts which significantly eases the convergence. However, wall functions are trickier to use, they require to choose the distance to the wall as a parameter which changes the final result of the simulation. Traditional wall functions are based on the "logarithmic-law" and thus require to be used around $y^+ = 30$ below $y^+ = 20$ the "logarithmic-law" is not verified and over $y^+ \approx 50$ the solution is sensitive to the outer parameters (essentially the pressure gradients) and can significantly differ from the "logarithmic-law". This condition can be difficult to impose as the dimensionless distance y^+ depends on the flow conditions. Finally, the mesh and the implementation of the wall functions must be carefully handled to obtain a mesh convergence.[1]

We propose here a strategy to efficiently deal with RANS simulations in a mesh adaptation context using in conjunction, both wall functions and wall integration.

II. Numerical Implementation

A. Wall functions

Standard Wall Functions: Wall functions were initially developed for high-Reynolds number turbulence models which become invalid in the laminar part of the turbulent boundary layer. Standard logarithmic

*PhD student, Sorbonne Universités, UPMC Univ Paris 06, Gamma3 Team

†Researcher, Gamma3 Team

‡Researcher, Gamma3 Team

wall functions were derived from the turbulent Couette flow. It can be shown that in absence of pressure gradients the non-dimensional tangential velocity and distance to the wall expressed as:

$$u^+ = \frac{u}{u_\tau}, \quad \text{and} \quad y^+ = \frac{y u_\tau}{\nu}.$$

ν is the cinematic viscosity, $u_\tau = \sqrt{\tau_w/\rho}$ with τ_w the wall shear stress. u^+ and y^+ are related by a given function $u^+ = F^+(y^+)$ which is independent of the flow. It can be shown that this function behaves asymptotically near $y^+ = 0$ as

$$u^+ = F^+(y^+) \underset{y^+ \to 0^+}{\sim} y^+,$$

and to greater values

$$u^+ = F^+(y^+) \underset{y^+ \to +\infty}{\sim} \frac{1}{\kappa} ln(E y^+),$$

where κ is the Von Karman constant and E a roughness parameter (for a smooth wall we have $\kappa = 0.42$ and $E = 9.0$). These relations are theoretically valid for a high-Reynolds Couette flow with no pressure gradient. However for many flows, these relations still hold and the first relation in 0^+ can be extended up to $y^+ = 5$ which is called the viscous sublayer (as turbulent viscosity is negligible). The second relation is called logarithmic-layer and considered to be valid from $y^+ \approx 20 - 30$ up to a distance y^+ which depends on the intensity of the pressure gradient and the Reynolds number. It is commonly admitted that a reasonable upper bound for good result should be $y^+ \approx 50 - 100$.

As standard high-Reynolds turbulence models successfully model the logarithmic layer, they are usually matched with wall functions at a given distance d from the wall, ideally chosen so that the corresponding dimensionless wall-distance $d^+ \approx 30$. The wall function is then supposed to model the inner part of the boundary layer (below $y^+ = 30$) and provide a coherent boundary condition to the flow. This is achieved with a Robin type boundary condition. Given the distance d and u the tangential velocity of the flow on the boundary, the nonlinear equation

$$\frac{u}{u_\tau} = \frac{1}{\kappa} ln\left(E \frac{d u_\tau}{\nu}\right),$$

is solved for u_τ (with Newton iterations or dichotomy typically). The normal derivative of the flow field is imposed through the tangential stress tensor,

$$(\tau \cdot \mathbf{n}) \cdot \mathbf{t} = \tau_w = \rho u_\tau^2,$$

and turbulent quantities are deduced from u_τ and similar appropriate relations depending on the model considered.

These wall functions can similarly be used with low-Reynolds turbulence models. However, as usual flows always present pressure gradients, the logarithmic-law is never exactly verified so that standard wall functions provide unconsistent boundary conditions, leading to numerical results that actually depends on the chosen distance d. Moreover, although low-Reynolds turbulence models are valid at any distance of the wall, logarithmic wall functions cannot be used below $y^+ = 20$ as the logarithmic relation does not hold.

Universal and Model Specific wall functions: To overcome this problem, universal wall functions have been developed, providing an analytic profile valid throughout the boundary layer. However, as these analytic profiles do not match the exact profile of the turbulence model, they still provide inconsistent boundary conditions leading to wall distance dependent results (even with no pressure gradients). This motivation lead to the development of model specific wall functions.[1–3] The main idea of this approach is that the dependency of the solutions to the distance used in the wall function is due to the inconsistency of the boundary condition imposed by the wall function. It was shown[2] that using tabulated values of an actual low-Reynolds turbulence model as wall function allowed to obtain the same results as wall integration, with no dependency to the wall distance chosen. In our case, we will use the analytical solution of the Spalart-Allmaras turbulence model:[4]

$$
\begin{aligned}
u^+ &= 5.0333908790505579 \\
&+ 2.5496773539754747 \log((y^+ + 8.148221580024245)^2 + 7.4600876082527945^2) \\
&+ 1.3301651588535228 \log((y^+ - 6.9287093849022945)^2 + 7.468145790401841^2) \\
&- 3.599459109332379 \operatorname{Arctan}(y^+ + 8.148221580024245, 7.4600876082527945) \\
&- 3.6397531868684494 \operatorname{Arctan}(y^+ - 6.9287093849022945, 7.468145790401841)
\end{aligned}
$$

where Arctan(y,x) is the arctangent of y/x taking into account the quadrant in which the point (x, y) is located to yield the proper signed angle.

Both of these wall functions can be used at any wall distance below $y^+ = 50$. Both are also supposed to tend to wall integration as $d \to 0$.

B. Adaptive strategy

In a mesh adaptation context, we want to derive the mesh providing the best solution – in a given sense detailed here after – for a given number of nodes.[5,6] The iterative mesh adaptation process states as follow, starting from an initial solution S_0 on an initial mesh H_0:

1. Analyze the solution S_i with a given sensor and determine a new metric size
2. Generate new mesh H_{i+1} from the prescribed metric
3. Interpolate S_i on H_{i+1} to provide an initial solution
4. Compute new solution S_{i+1}
5. Go back to 1 until converged

Different sensors and error estimation tools have been derived[7–13] to determine the optimal mesh for a given application. In this work we will use a hessian mesh adaptation on the Mach field, *i.e.* we seek the mesh that minimizes the interpolation error of the Mach field. As the exact Mach field is not known, this error is estimated from the difference of the $P1$ solution and the reconstruction of a smoother field based on the hessian of the Mach. This approximation has strong implications on the results as we will see.

To fasten computations, the initial grid is usually chosen relatively coarse and the mesh complexity is progressively increased. Although this approach is very efficient for inviscid flows, it has a major disadvantage for RANS simulations. If the low-Reynolds turbulence model is under-resolved the resulting solution can be far from the exact one so that the iterative process may be slow. This is why, many RANS mesh adaptation rely on the addition of a prescribed mesh boundary layer which is not adapted. This leads to large initial grids, thus expensive to compute and reduces the overall performance of the mesh adaptation process.

This is why we propose here to develop an automatic approach using wall functions that degenerates to wall integration, depending on the mesh resolution.

C. Solver description

General Description: Our in house solver **Wolf** is a vertex centered RANS finite volume solver.[14] The solution is discretized using a $P1$ representation with median cells. **Wolf** combines a HLLC approximate Riemann solver[15] to compute the convective fluxes and the Galerkin centered method to evaluate the viscous terms. Second order space accuracy is achieved through a piecewise linear extrapolation based on the Monotonic Upwind Scheme for Conservation Law (MUSCL) procedure[16] which uses a particular edge-based formulation with upwind elements, stabilized with Piperno limiter.[17,18] RANS equations are closed with the Spalart-Allmaras turbulence model.[19]

Wall functions implementation: Numerically, the aim of wall functions is to modify the nodal residual near the boundaries in order to mimic the presence of the desired boundary layer. This is usually done either with an off wall implementation or a volume implementation as shown in Figure 1. In the off wall representation (left part of Figure 1) the computational domain is shifted at a distance d of the physical wall and the boundary layer region in red is computed by the wall function. To do so a consistent shear

stress is applied to the fluid in the volume through the numerical boundary edge in blue. This method is relatively simple to implement and the chosen distance d is independent of the mesh. This guaranties the ability to yield converging computations toward an unique solution with high-Reynolds turbulence models when the mesh size decreases.[20] Though, this off-wall distance d implicitly modifies the shape of the body as an additional flux virtually flows through this space, which can be undesired.

In the volume representation, the first layer of elements touching the boundary layer are flagged and an additional viscous flux is computed in the cells to take into account the turbulent boundary layer. In other words, the velocity field is assumed no to be linear in these element (for a $P1$ discretization) but instead to match the logarithmic profile of the turbulent boundary layer. The boundary edge in blue is thus coincident with the physical boundary and is a standard no-slip boundary condition. The major advantage of this approach is that it does not modifies the geometry of the body. However it yields severe restrictions on the mesh as the wall distance is fixed by the height of the first elements. For standard wall function this limits the minimal mesh size that can be used as the dimension less wall distance d^+ should not go below $d^+ = 30$. Moreover, this approach is relatively simple to implement for structured grids but it becomes cumbersome for unstructured grids (not to mention 3D cases).

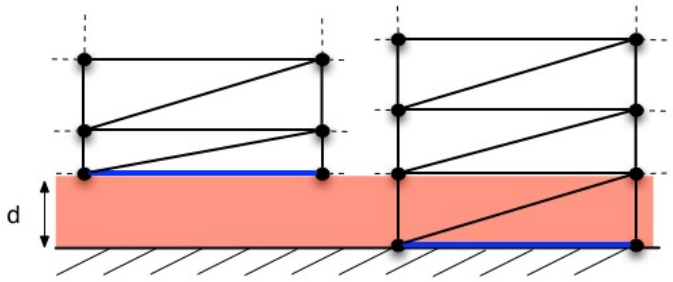

Figure 1. Off-wall implementation (left) and volume implementation (right) of wall functions

In this work we use an off-wall prescription of wall functions, so that the wall distance d can be fixed independently of the mesh. However, we will either fix the distance d to a given chosen value as in the traditional approach or fix it depending on the mesh size. In that case we compute the height of the elements connected to the surface and fix the distance d everywere to the mean value of the height of these elements (see Figure 2 right) or on each point to the minimal height of the elements to which it belongs (see Figure 2 left). Although this choice would be a constraint for standard wall functions, it allows here universal wall functions to degenerate to a wall-resolved computation as the mesh size decreases. Moreover this provides a relatively consistent treatment of the boundary condition depending on the mesh size: if the first elements size is about $y^+ = 1$ it is reasonable to use a standard wall-resolved model. On the contrary, if the first elements are to coarse it is better to use wall-functions.

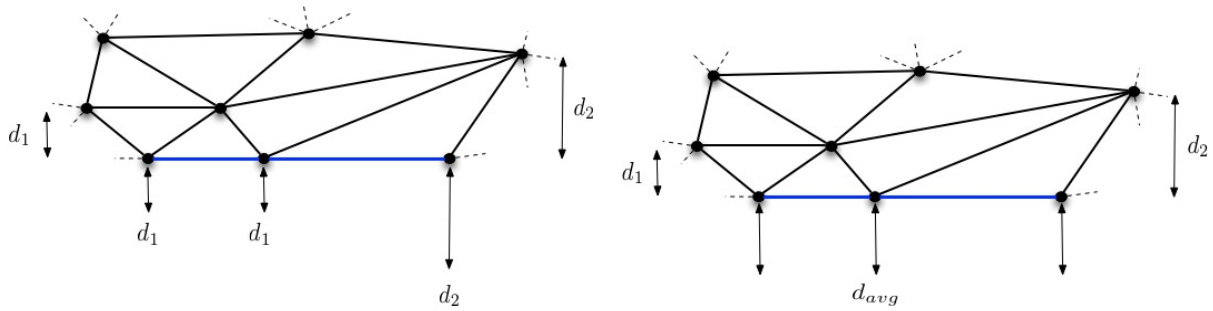

Figure 2. Local minimal off-wall distance computation (left) average off-wall distance computation(right)

We use here a two velocities scale[20] wall function: instead of solving $u^+ = F^+(y^+)$ for u_τ, we determine a first velocity scale from the boundary value of ν_{SA} as

$$u_1 = \frac{\nu_{SA}}{\kappa d},$$

from which y^+ is determined as

$$y^+ = \frac{du_1}{\nu}.$$

The second velocity scale is deduced from

$$u_2 = \frac{u_t}{F^+(y^+)},$$

with $u_t = \|\mathbf{u} - (\mathbf{u}\cdot\mathbf{n})\mathbf{n}\|$ the tangential velocity, from which we can compute the friction stress tensor as

$$\tau_w = \rho u_1 u_2.$$

This implementation was initially proposed for heat transfer cases as it improves predictions near stagnation points by preventing y^+ from going to 0 as $u_t \to 0$. It is also straight forward to implement and does not require to solve a non-linear system.

From these quantities we deduce the boundary fluxes through the wall function boundary condition

$$\begin{aligned}
\Phi_\rho &= 0 \\
\Phi_{\mathbf{u}} &= \tau_w \left(-\frac{\mathbf{u} - (\mathbf{u}\cdot\mathbf{n})\mathbf{n}}{\|\mathbf{u} - (\mathbf{u}\cdot\mathbf{n})\mathbf{n}\|} \right) \\
\Phi_e &= \tau_w \|\mathbf{u} - (\mathbf{u}\cdot\mathbf{n})\mathbf{n}\| \\
\Phi_{\nu_{SA}} &= \frac{2}{3}(\nu + \nu_{SA})\kappa u_1.
\end{aligned}$$

III. Numerical Results

A. NACA0012 validation

We first investigate the mesh adaptation computation of the flow around a NACA0012 using wall functions and standard wall resolution. The flow considered has the following characteristics: Mach number: 0.3, angle of attack: 1°, Reynolds number: 15×10^6, reference temperature: $300K$. Meshes are iteratively generated with **feflo.a**.[21,22] We use a hessian-based adaptation on the Mach number field which aims at minimizing the interpolation error of the Mach number field for a given number of vertices. For the present study we started with an initial mesh complexity of 2500 vertices and multiplied the complexity by two at each step. For each step we perform five sub-steps to converge the mesh/solution couple for the given complexity.

We compare the solutions obtained with the low-Reynolds Spalart-Allmaras turbulence model with:

- a standard wall resolution,
- wall functions with a fixed distance of $d = 1 \times 10^{-4}$,
- wall functions with a fixed distance of $d = 1 \times 10^{-5}$,
- wall functions with a fixed distance of $d = 1 \times 10^{-6}$,
- adaptive wall functions.

The meshes and solutions obtained with hessian-based mesh adaptation strategy using the standard wall resolution are shown in Figure 3. We clearly see the negative feedback between mesh adaptation and the low-Reynolds turbulence model. The insufficient discretization near the wall leads to a too viscous boundary layer that grows too rapidly compared to the reference, properly discretized solution (Figure 3 bottom). In turn, mesh adaptation tends to produce coarser grids to match the thicker boundary layer, leading to an even poorer discretization. As can be seen in Figure 3, the mesh adaptation process initially captures the too coarse boundary layer which progressively gets thiner. The process still converges toward an appropriate discretization but this negative feedback leads to an artificially slow convergence of the solution.

This negative feedback is alleviated using adaptive wall functions. The same grid/solution convergence is shown in Figure 4. Adaptive wall functions treat coarse grids with a large wall distance so that the inner, stiff, part of the boundary layer is not resolved. The mesh resolution is thus sufficient to discretize the boundary layer, which in turn is not too viscous. This leads to a thiner boundary layer on which the mesh adaptation concentrates the nodes leading to an even better discretization of the boundary layer. As the

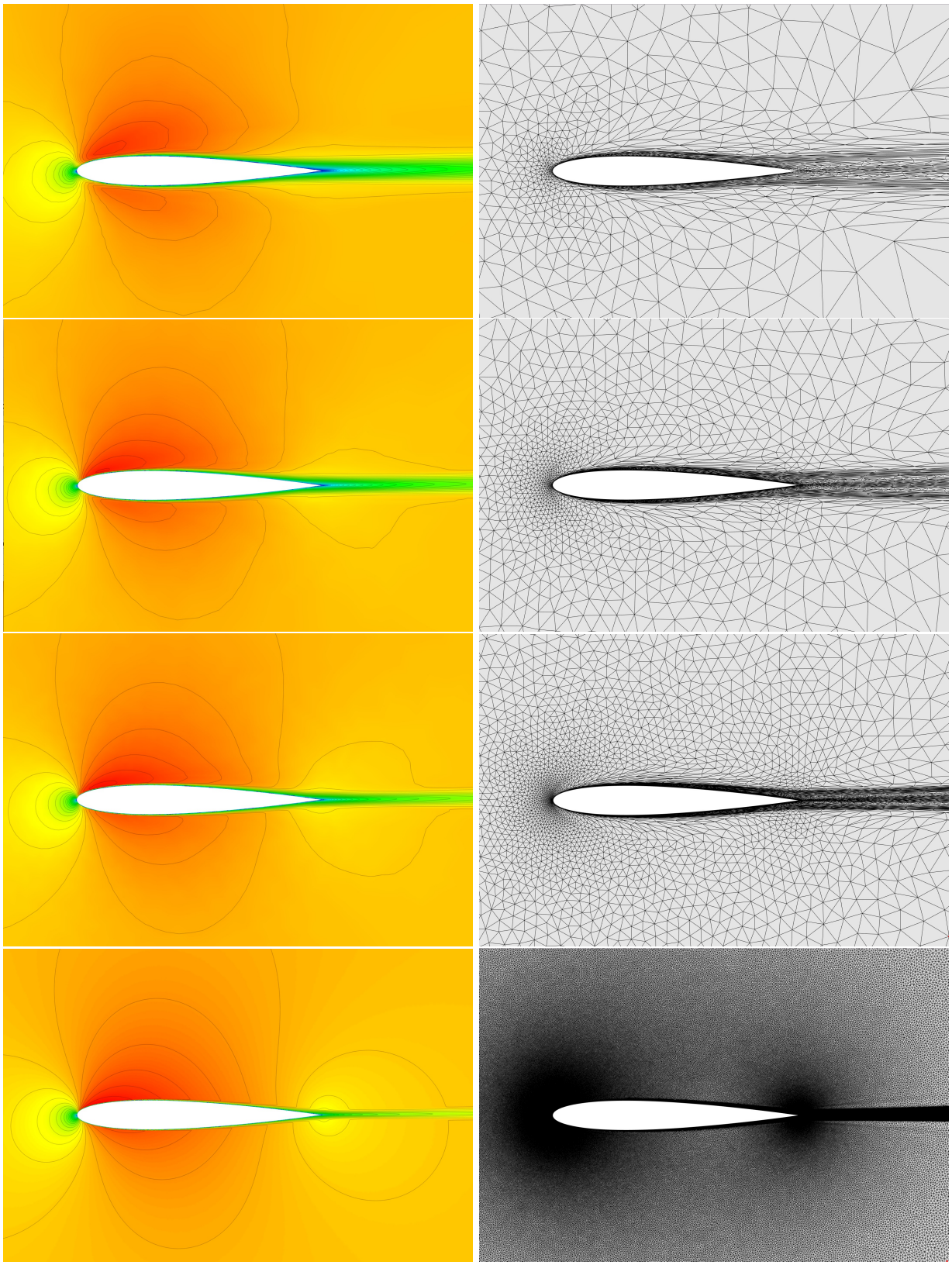

Figure 3. Mesh (right) and solution (left) convergence for the NACA0012 case with standard wall resolution: 2453, 4907, 9667 and 602715 nodes.

elements size near the wall decreases, the wall distance computed by wall function decreases leading to a stiffer boundary layer and a thiner mesh size requirement. An equilibrium is found when the decrease of the boundary elements size does not matches the requirements of the boundary layer prescribed by the wall functions with the corresponding decrease in the wall distance. Smaller elements would lead to a smaller wall distance and thus a much stiffer boundary layer profile for which their resolution would be insufficient. This would lead in an increase of the viscosity in the boundary layer leading to its coarsening leading to a coarser mesh prescription by mesh adaptation.

Figure 5 shows the velocity profiles predicted by wall integration and wall functions with fixed distances close to the trailing edge. The wall distances have been fixed to $d = 10^{-4}$, $d = 10^{-5}$ and $d = 10^{-6}$, corresponding to dimensionless wall distances of about respectively $d^+ \approx 50$, $d^+ \approx 5$, $d^+ \approx 0.5$. As shown in Figure 3, we can see that wall integration yields a thicker boundary layer on coarse meshes. Moreover, the velocity profile is smooth, preventing the mesh adaptation from efficiently refining the mesh in this region. Similarly, wall function with $d^+ \approx 0.5$ behaves like a wall integration, leading to a thick boundary layer. As expected, wall functions with $d^+ \approx 5$ and $d^+ \approx 50$ provide better results, the second converging faster than the first one.

This is a rather favorable case for wall functions as the flow is attached with mild pressure gradients, it is not surprising to have good results with larger d^+ especially on coarse grids. However, in presence of larger pressure gradients it is critical to minimize d^+ in order to minimize their effect. Figure 6 shows how adaptive wall functions achieve both. In the left figure we can see how the velocity profile converges rapidly toward wall integrated solution from coarse grids. On coarse grids they behave like wall-functions with about $d^+ = 100$, immediately providing good predictions. On finer grids they degenerate toward wall integration, as can be seen in the right figure.

Figure 6 suggests we can further optimize the choice of the wall distance with respect to the mesh size. On the 613064 nodes mesh, we can prescribe a smaller wall distance to be closer to wall integration as the discretization is sufficient. In facts it is also possible to switch to a standard wall integration on the boundary edges that reached a given minimal wall distance.

This approach gives satisfying results on both fine and coarse meshes. In particular, we saw that there was a negative feedback between the turbulence model and mesh adaptation process using wall integration. This is due to the fact that the mesh adaptation sensor is "unaware" of the existence of the boundary layer leading to its improper discretization. This is alleviated by adding some information about the boundary layer through the use of wall functions. From this observation, we developed a hessian boundary correction of the sensor for the wall integration cases in order to add the same positive feedback as with wall functions.

B. High-Lift 2D configuration

This is a classical three elements airfoil high-lift configuration considered here at Mach 0.175, with a Reynolds number of 15.1×10^6 and an angle of attack of $16.21°$. This is a very interesting case for mesh adaptation as the shear layer that develops behind the first element, above the boundary layer of the second element is difficult to predict. It is thus difficult to generate *a-priori* a proper structured mesh. Figures 7, 8, 9 and 10 show the influence of the discretization of the shear layer by hand-made meshes shown Figures 11, 12, 13 and 14 on the overall solution. We can see in Figures 7 and 8 that if the shear layer is not properly discretized as shown in Figures 11 and 12, it does not appears in the solution. This difference is relatively small on the second foil element (Figures 7 and 9) but it induces a large difference on the last element as can be seen in Figures 8 and10. However, we can easily realize how difficult it is to prescribe *a-priori* the mesh shown in Figures 13 and 14, with the only knowledge of the solution shown in Figures 7 and 8.

Mesh adaptation strategy automatically refines this region as can be seen in Figures 15 and 16. This mesh has been obtained with an iterative hessian based mesh adaptation procedure,[23] which aims at minimizing the interpolation error in L^2 norm of the Mach field and metric aligned method[24] to generate pseudo structured meshes. However, it has required the addition of a fixed boundary layer of 5 elements in order to prevent detachment of the flow as explained here after. Although necessary here, this treatment is inconsistent with mesh adaptation strategy which balances the distribution of degrees of freedom in order to get an optimal mesh. It also makes it extremely difficult or impossible to adapt the boundary surface (especially in 3D) as the boundary layer mesh has either to be frozen or adapted as a whole.

This is also a very interesting case for the present study as the treatment of the boundary layer is crucial as it has a tremendous effect on the overall result. We can see in Figure 17 and 18 that if the boundary layer

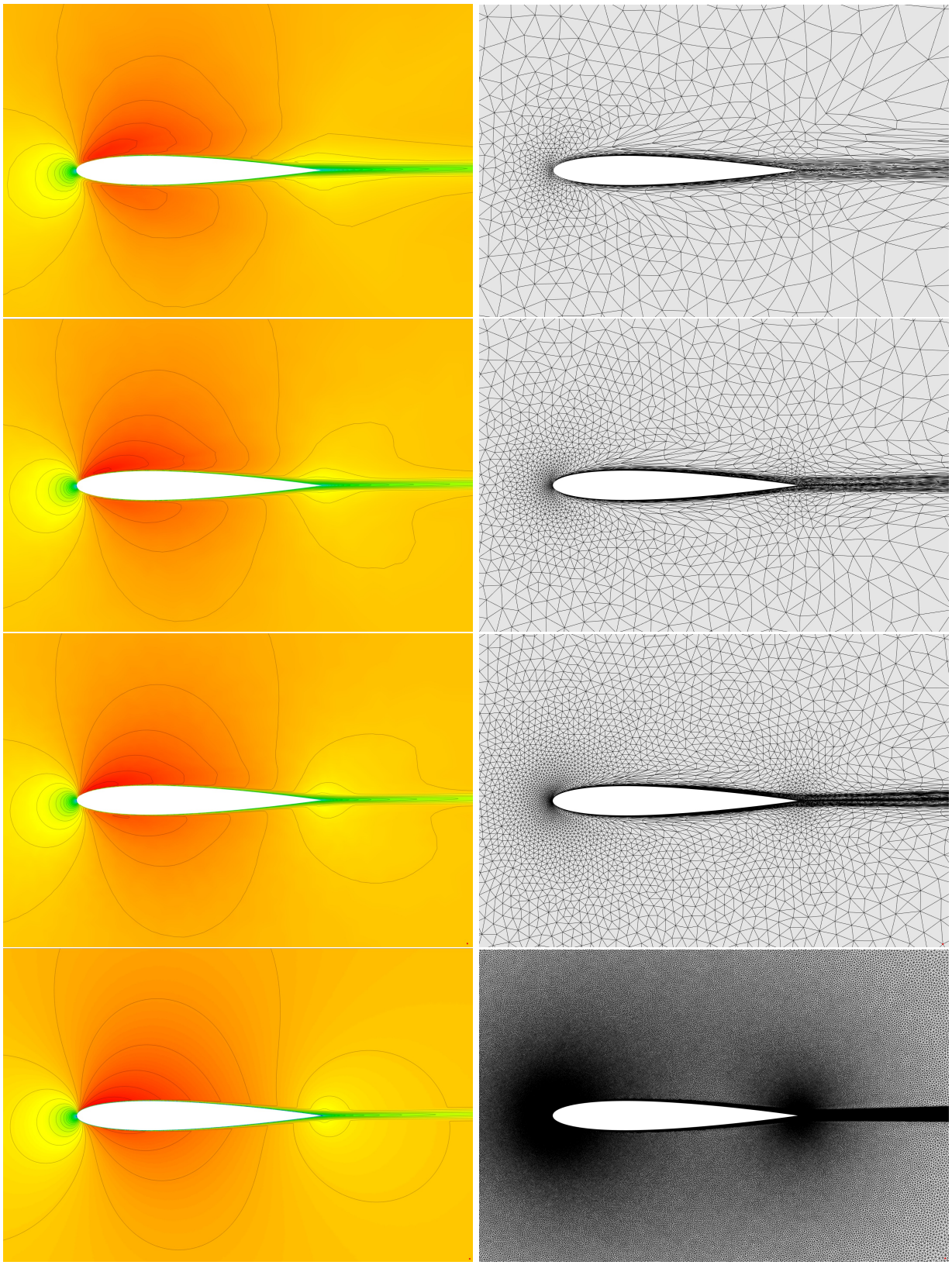

Figure 4. Mesh (right) and solution (left) convergence for the NACA0012 case with adaptive wall functions: 2522, 4813, 9536 **and** 613064 **nodes.**

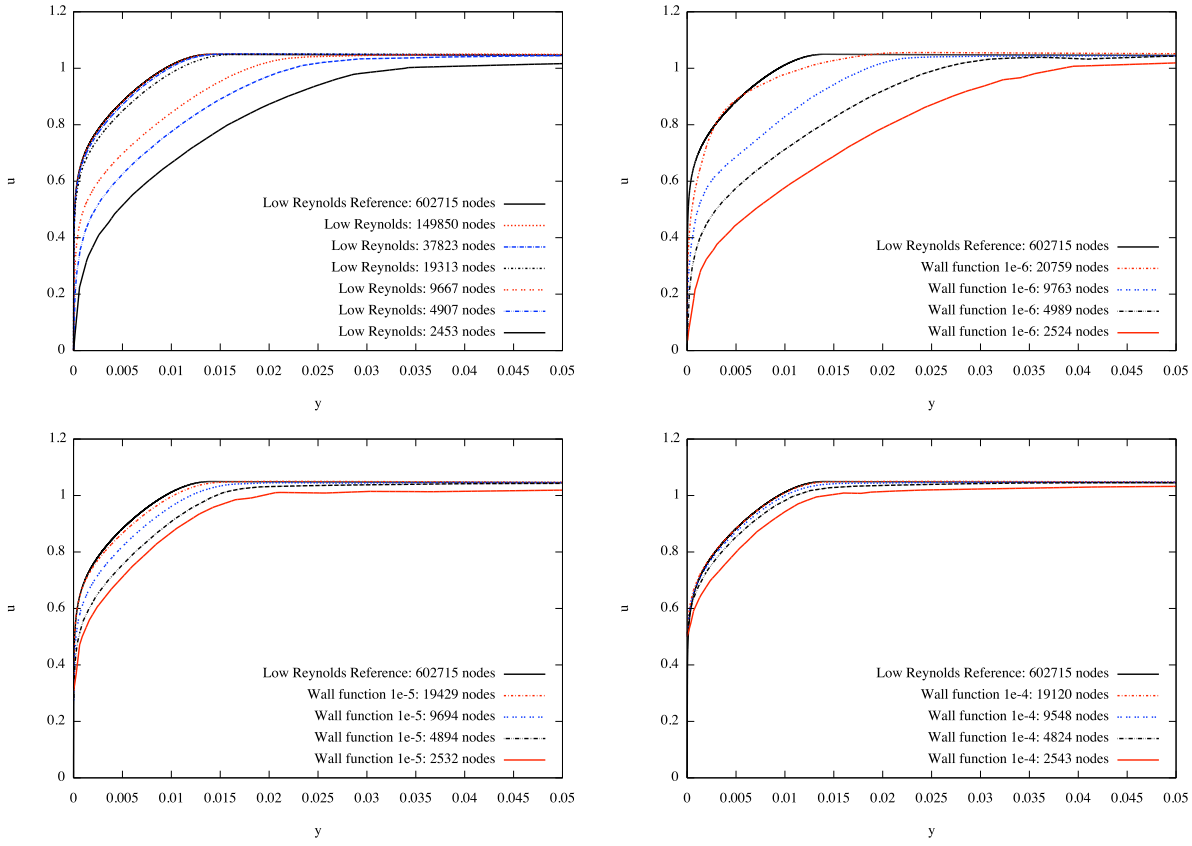

Figure 5. Convergence of velocity profiles above the NACA0012 trailing edge using wall integration (top left) and wall functions with fixed wall distances of $d = 10^{-6}$ (top right), $d = 10^{-5}$ (bottom left) and $d = 10^{-4}$ (bottom right).

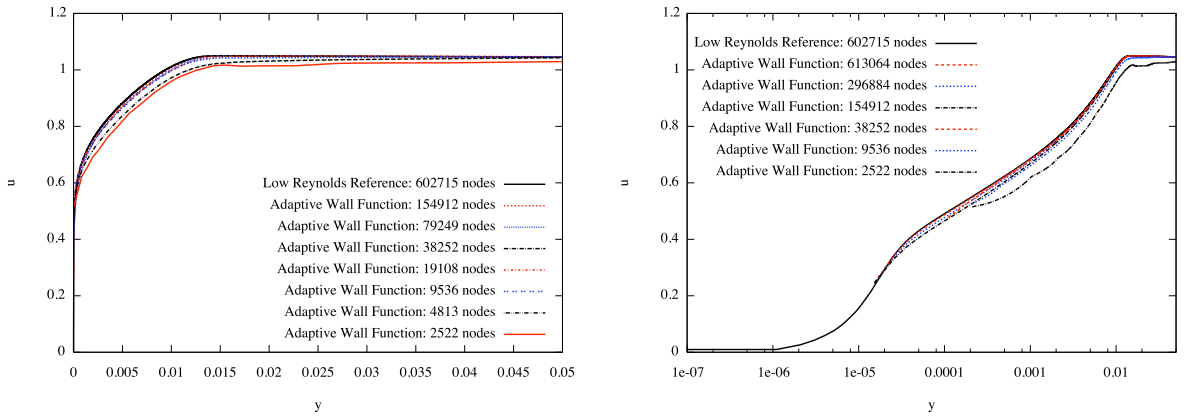

Figure 6. Convergence of velocity profiles above the NACA0012 trailing edge using adaptive wall functions. Linear (left) and log (right) scales.

resolution is not sufficient, the resulting flow is detached, contrary to the expected flow.

As previously, we performed a mesh adaptation with standard wall functions at a fixed distance, an adaptation with adaptive wall function, an adaptation with a wall integrated resolution and additionally an adaptation with wall integration using our boundary correction of the hessian in the error estimation. Meshes are iteratively generated with **feflo.a**[21,22] with a hessian adaptation on the Mach number field. Five meshes are generated for each complexity.

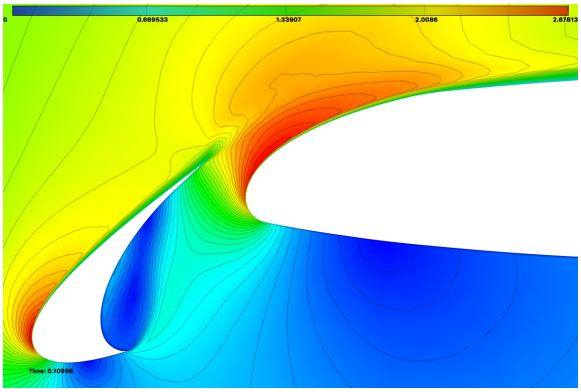

Figure 7. Zoom on the leading edge: velocity contours around the three elements high lift configuration computed on the first hand-made mesh without shear layer adaptation.

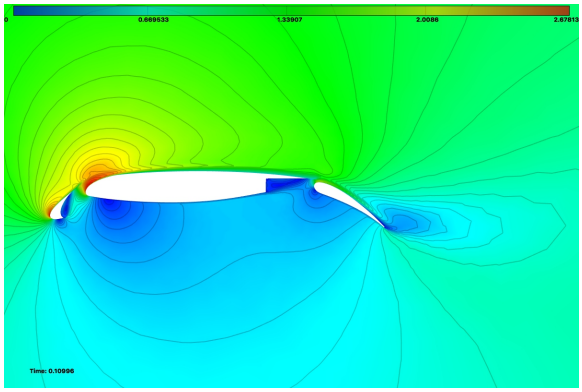

Figure 8. Velocity contours around the three elements high lift configuration computed on the first hand-made mesh without shear layer adaptation.

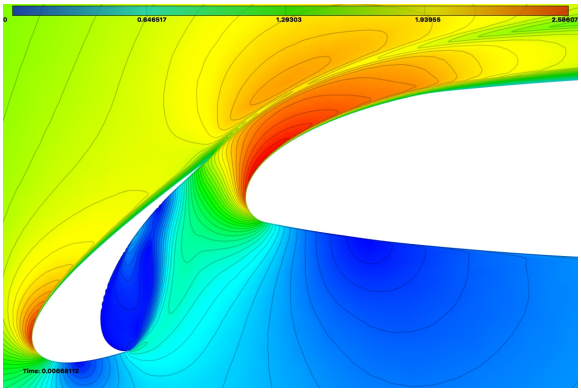

Figure 9. Zoom on the leading edge: velocity contours around the three elements high lift configuration computed on the second hand-made mesh with shear layer adaptation.

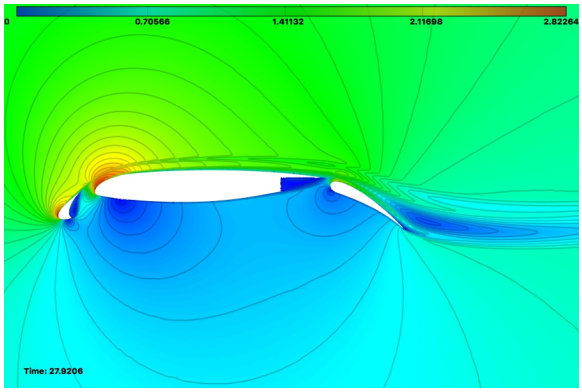

Figure 10. Velocity contours around the three elements high lift configuration computed on the second hand-made mesh with shear layer adaptation.

The resulting computed lift coefficient is shown in Figure 19. We can see that the detachment of the flow due to the under-resolution of the boundary layer leads to an under prediction of the lift in cases with wall integration. The resolution becomes sufficient around 10^5 nodes to re-attach the flow and converge to the expected solution. The initial iterations are thus useless with regard to the final solution.

Meanwhile, we can see that the standard wall functions also lead to an initial detached flow but converge earlier. Contrary to the NACA0012 case, the adaptive wall laws converge later than universal wall functions with fixed distance but still earlier than wall integration. This highlights again the need for a more clever choice of the wall distance. However, adaptive wall functions still behave like wall integration asymptotically.

Finally, the hessian boundary correction (BC) provides a much faster convergence and earlier reattachment. This is due to an early capture of the boundary layer as can be seen in Figure 20. On the left figure we can see how finer the mesh prescribed by the corrected sensor is near the wall, compared to the standard sensor. Note that contrary to a standard boundary layer insertion, this approach is naturally compatible with mesh adaptation. It does not affects the adaptation of the surfaces and the mesh complexity in the boundary layer is balanced with the main flow.

IV. Conclusion

Although naive, this first implementation of adaptive wall functions already shows satisfying results for mesh adaptation strategies. The use of universal wall functions allows to prescribe any wall distance even below $y^+ = 30$ providing much flexibility. The use of an adaptive wall distance ensures a proper description

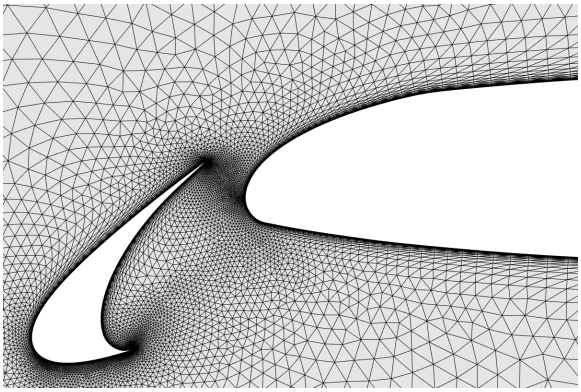

Figure 11. Zoom on the leading edge: hand-made mesh of the the three elements high lift geometry without shear layer adaptation.

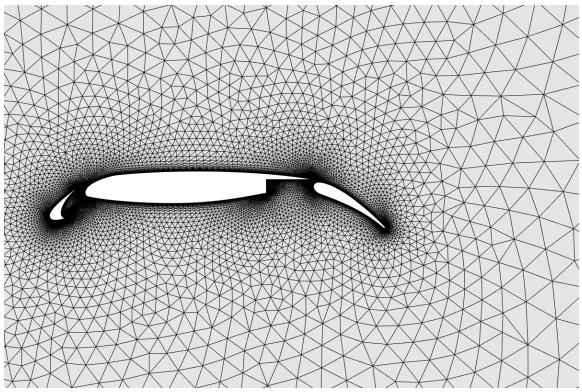

Figure 12. Hand-made mesh of the the three elements high lift geometry without shear layer adaptation.

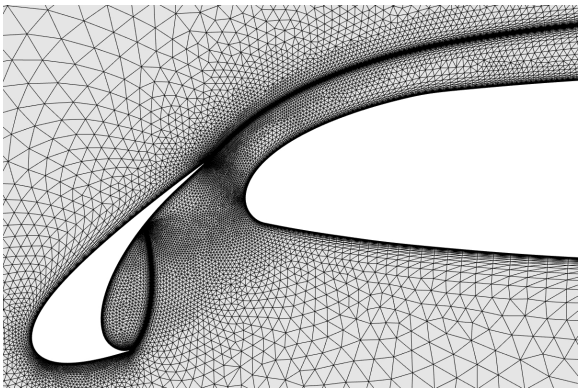

Figure 13. Zoom on the leading edge: hand-made mesh of the the three elements high lift geometry without shear layer adaptation. Hand-made mesh of the the three elements high lift geometry with shear layer adaptation.

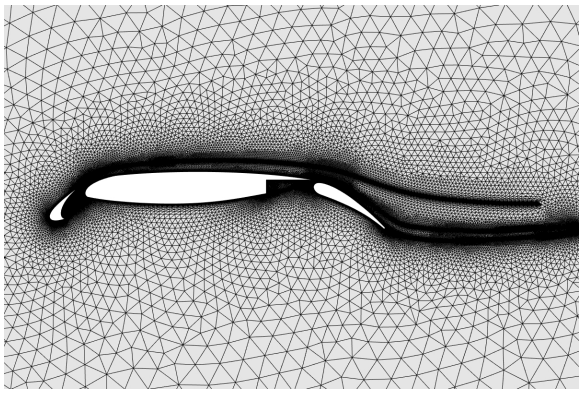

Figure 14. Hand-made mesh of the the three elements high lift geometry with shear layer adaptation.

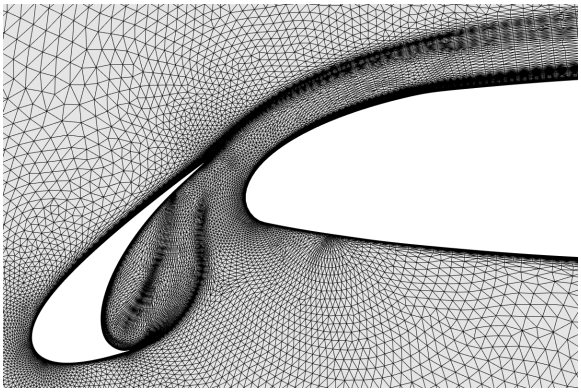

Figure 15. Zoom on the leading edge : adapted mesh of the the three elements high lift geometry with a hessian sensor.

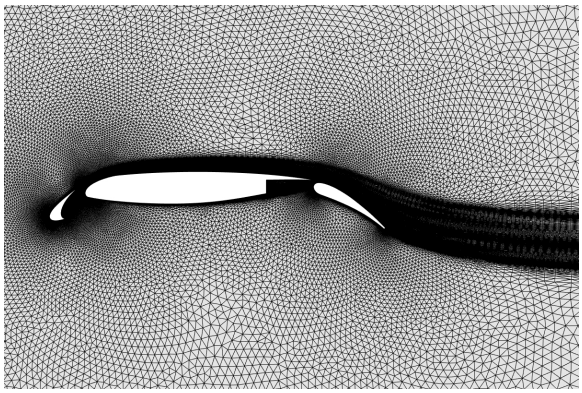

Figure 16. Adapted mesh of the the three elements high lift geometry with a hessian sensor.

of the boundary layer depending on the current mesh size. Ultimately it degenerates to a standard wall integration as the mesh size decreases.

However, the naive linear correlation between the wall distance and mesh size is insufficient as it does not takes fully account of the flow state. Moreover, we used here for simplicity a hessian based mesh adaptation

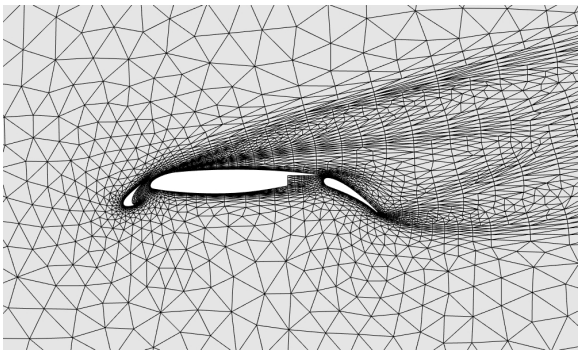

Figure 17. Adapted mesh generated with a hessian based approach on the mach field without boundary layer insertion.

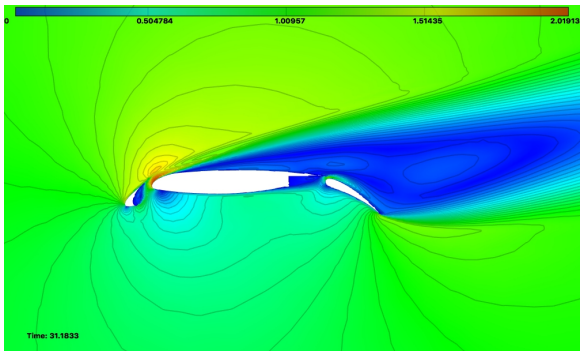

Figure 18. Velocity field of a computation on an adapted mesh generated with a hessian based approach on the mach field without boundary layer insertion.

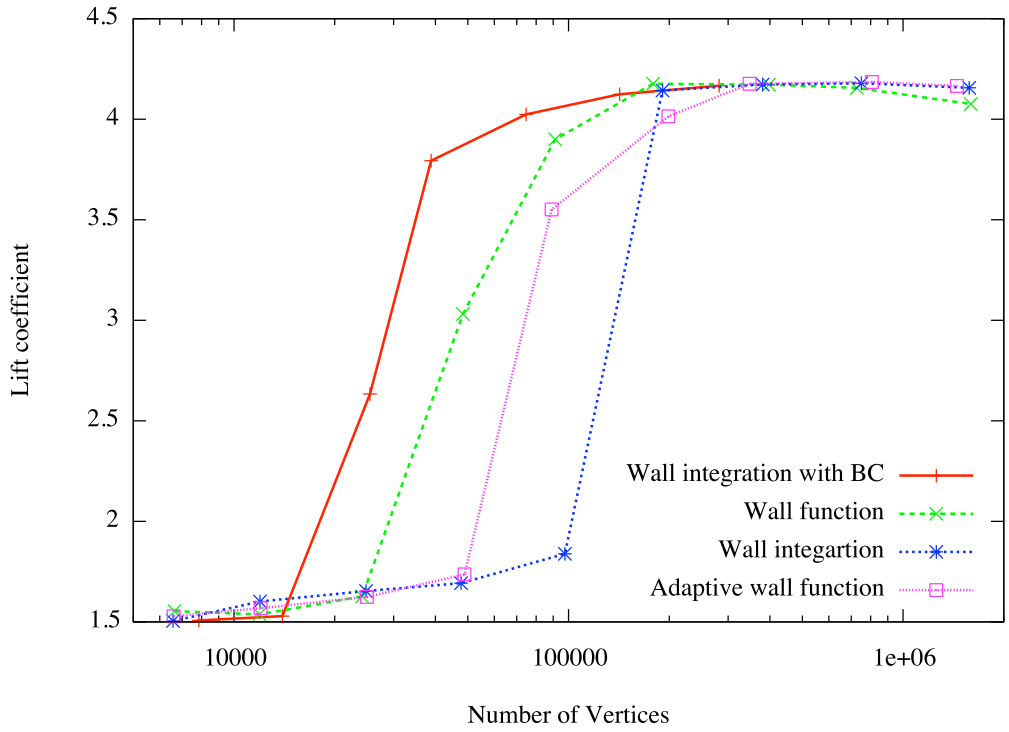

Figure 19. Lift convergence on the three element air foil geometry using wall functions and wall integration.

which is highly inefficient as it adapts the whole wake, wasting ressources. For those reasons an adjoint based approach taking into account boundary contributions would be more appropriate to perform the mesh adaptation and prescribe the wall distance.

The hessian boundary correction of the adaptation sensor also showed very promising results. Similarly, it should be extended to adjoint based sensors and 3D cases.

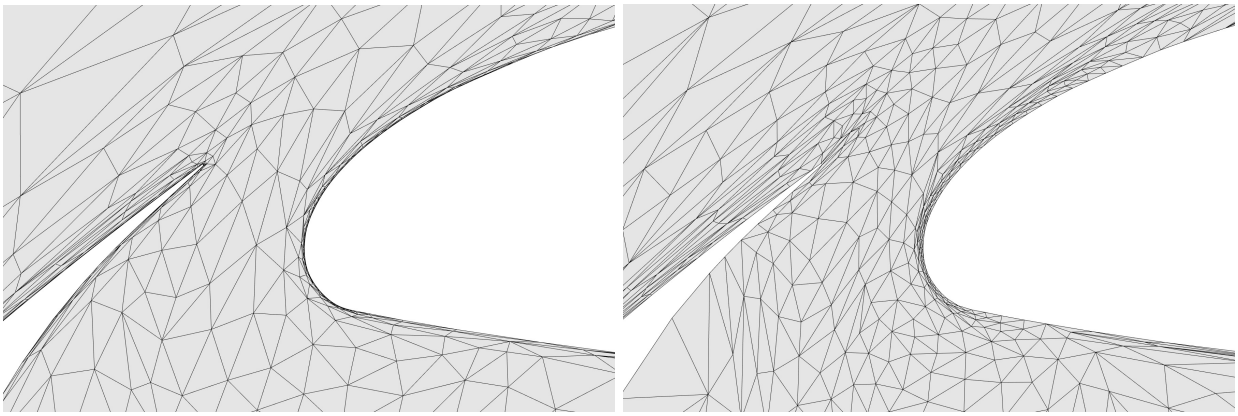

Figure 20. Meshes generated by hessian based mesh adaptation on the Mach field with hessian boundary correction (left, 7854 vertices) and without correction (right, 11957 vertices).

References

[1]Knopp, T., Alrutz, T., and Schwamborn, D., "A grid and flow adaptive wall-function method for RANS turbulence modelling," *Journal of Computational Physics*, Vol. 220, No. 1, 2006, pp. 19 – 40.

[2]Kalitzin, G., Medic, G., Iaccarino, G., and Durbin, P., "Near-wall behavior of RANS turbulence models and implications for wall functions," *Journal of Computational Physics*, Vol. 204, No. 1, 2005, pp. 265 – 291.

[3]Frazza, L., Hay, A., and Pelletier, D., *A logarithmic formulation for low-Reynolds number turbulence models with adaptive wall-functions*, American Institute of Aeronautics and Astronautics, 2017/11/09 2015.

[4]Allmaras, S. R. and Johnson, F. T., "Modifications and clarifications for the implementation of the Spalart-Allmaras turbulence model," *Seventh international conference on computational fluid dynamics (ICCFD7)*, 2012, pp. 1–11.

[5]Loseille, A., Dervieux, A., Frey, P., and Alauzet, F., "Achievement of global second-order mesh convergence for discontinuous flows with adapted unstructured meshes," *18th AIAA Computational Fluid Dynamics Conference*, AIAA Paper2007-4186, Miami, FL, USA, Jun 2007.

[6]Alauzet, F. and Loseille, A., "High Order Sonic Boom Modeling by Adaptive Methods," *J. Comp. Phys.*, Vol. 229, 2010, pp. 561–593.

[7]Frey, P. and George, P.-L., *Mesh generation. Application to finite elements*, ISTE Ltd and John Wiley & Sons, 2nd ed., 2008.

[8]George, P., Hecht, F., and Vallet, M., "Creation of internal points in Voronoi's type method. Control and adaptation," *Adv. Eng. Software*, Vol. 13, No. 5-6, 1991, pp. 303–312.

[9]Becker, R. and Rannacher, R., "A feed-back approach to error control in finite element methods: basic analysis and examples," *East-West J. Numer. Math.*, Vol. 4, 1996, pp. 237–264.

[10]Fidkowski, K. J. and Darmofal, D. L., "Review of Output-Based Error Estimation and Mesh Adaptation in Computational Fluid Dynamics," *AIAA Journal*, Vol. 49, No. 4, 2017/11/06 2011, pp. 673–694.

[11]A Venditti, D. and Darmofal, D., "Adjoint Error Estimation and Grid Adaptation for Functional Outputs: Application to Quasi-One-Dimensional Flow," Vol. 164, 10 2000, pp. 204–227.

[12]Jones, W., Nielsen, E., and Park, M., "Validation of 3D Adjoint Based Error Estimation and Mesh Adaptation for Sonic Boom Reduction," *44th AIAA Aerospace Sciences Meeting and Exhibit*, AIAA-2006-1150, Reno, NV, USA, Jan 2006.

[13]Loseille, A., Dervieux, A., and Alauzet, F., "Fully anisotropic goal-oriented mesh adaptation for 3D steady Euler equations," *J. Comp. Phys.*, Vol. 229, 2010, pp. 2866–2897.

[14]Frazza, L., Loseille, A., and Alauzet, F., "Anisotropic mesh adaptation for turbomachinery applications," *23rd AIAA Computational Fluid Dynamics Conference*, 2017, p. 3299.

[15]Batten, P., Clarke, N., Lambert, C., and Causon, D. M., "On the choice of wavespeeds for the HLLC Riemann solver," *SIAM J. Sci. Comput.*, Vol. 18, No. 6, 1997, pp. 1553–1570.

[16]Leer, B. V., "Towards the ultimate conservative difference scheme I. The quest of monotonicity," *Lecture notes in physics*, Vol. 18, 1973, pp. 163–168.

[17]Piperno, S. and Depeyre, S., "Criteria for the design of limiters yielding efficient high resolution TVD schemes," *Comput. & Fluids*, Vol. 27, No. 2, 1998, pp. 183–197.

[18]Piperno, S., "Schéma TVD d'ordre élevé pour la résolution de l'équation de Burgers," Rapport de recherche 1996-49, CERMICS, 1996.

[19]Spalart, P. and Allmaras, S., "A one-equation turbulence model for aerodynamic flows," *30th AIAA Aerospace Sciences Meeting and Exhibit*, AIAA-92-0439, Reno, NV, USA, Jan 1992.

[20]Lacasse, D., Turgeon, E., and Pelletier, D., "On the judicious use of the k–ε model, wall functions and adaptivity," *International Journal of Thermal Sciences*, Vol. 43, No. 10, 2004, pp. 925–938.

[21]Loseille, A. and Löhner, R., "Cavity-Based Operators for Mesh Adaptation," *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, AIAA Paper2013-0152, Dallas, TX, USA, Jan 2013.

[22]A. Loseille, F. A. and Menier, V., "Unique cavity-based operator and hierarchical domain partitioning for fast parallel generation of anisotropic meshes," *Computer-Aided Design*, Vol. 85, 2017, pp. 53–67.

[23]Loseille, A. and Löhner, R., "Adaptive anisotropic simulations in aerodynamics," *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, AIAA Paper2010-169, Orlando, FL, USA, Jan 2010.

[24]Marcum, D. and Alauzet, F., "Aligned metric-based anisotropic solution adaptive mesh generation," *Procedia Engineering*, Vol. 82, 2014, pp. 428–444.