ExaQUte

Exascale **Q**uantification of **U**ncertainties for **Te**chnology and Science Simulation

D7.2 Finalization of "deterministic" verification and validation tests

Document information table

Contract number:	800898
Project acronym:	ExaQUte
Project Coordinator:	CIMNE
Document Responsible Partner:	str.ucture
Deliverable Type:	REPORT
Dissemination Level:	Public
Related WP & Task:	WP7, Task 7.2
Status:	Final version

Authoring

Prepared by:				
Authors	Partner	Modified Page/Sections	Version	Comments
Andreas Apostolatos	TUM			
Contributors				

Change Log

Versions	Modified Page/Sections	Comments
V1.0	First version	
V1.1		
V2.0		

Approval

Approved by:				
	Name	Partner	Date	OK
Task leader	Roland Wüchner	TUM		
WP leader	Alexander Michaelski	str.ucture		
Coordinator	Riccardo Rossi	CIMNE		

Executive summary

This deliverable focus on the verification and validation of the solvers of Kratos Multiphysics which are used within ExaQUte. These solvers comprise standard body-fitted approaches and novel embedded approaches for the *Computational Fluid Dynamics* (CFD) simulations carried out within ExaQUte. Firstly, the standard body-fitted CFD solver is validated on a benchmark problem of high rise building - CAARC benchmark and subsequently the novel embedded CFD solver is verified against the solution of the body-fitted solver. Especially for the novel embedded approach, a workflow is presented on which the exact parameterized *Computer-Aided Design* (CAD) model is used in an efficient manner for the underlying CFD simulations.

It includes:

- A note on the space-time methods

- Verification results for the body-fitted solver based on the CAARC benchmark

- Workflow consisting of importing an exact CAD model, tessellating it and performing embedded CFD on it

- Verification results for the embedded solver based on a high-rise building

- API definition and usage

Contents

List of Figures

Nomenclature / Acronym list

Acronym	Meaning
API	Application Programming Interface
ExaQUte	EXAscale Quantification of Uncertainties for Technology and Science Simulation
CAD	Computer-Aided Design
CFD	Computational Fluid Dynamics
CAARC	Commonwealth Advisory Aeronautical Council
CWE	Computational Wind Engineering
QoI	Quantity of Interest
FEM	Finite Element Method
NURBS	Non-Uniform Rational B-Splines
API	Application Programming Interface

1 Introduction

This deliverable focuses on the verification of the employed *Computational Fluid Dynamics* (CFD) solvers in Kratos Multiphysics (A.K.A Kratos) for the application of interest in wind engineering. These solvers comprise standard body-fitted and novel embedded approaches. Especially for the novel embedded CFD approach, a workflow is herein presented which consists in the use of the *Computer-Aided Design* (CAD) and a tessellation algorithm for obtaining an optimal representation of the embedded object within the CFD mesh. The latter approach is especially advantageous as generating a body-fitted mesh for any free form CAD geometry is cumbersome. With the proposed workflow the latter challenge of generating a body-fitted mesh can be overcome, thus enabling the fast and efficient generation of CFD meshes for general settings.

At first the standard body-fitted solver is validated against the CAARC benchmark, which is well established in the literature.

Then, the novel embedded approach and the proposed workflow are validated for the CFD case of a real wind scenario around a straight and a twisted by 180° high-rise building. The results are validated in terms of the contour plots of the pressure and velocity magnitude resulting from both solvers.

2 A note on the space-time methods

Even though in the proposal it is stated that "novel time-parallel (or space-time-parallel) version will be benchmarked against the validation data, both in terms of runtime and of predictions" this evaluation can not be performed with the current state of the implementation. The main reason is that the nonlinear algorithms implemented in task 3.1 where not fully successful and, as it is discussed in the midterm report, further research is needed to obtain the minimum performance that deserves to be tested.

Wind engineering problems faced in this project require the simulation of turbulent flows around structures. The numerical simulation of turbulent flows is challenging for several reasons, among which we can mention, their strongly nonlinear, locally multiscale chaotic character. The traditional approach to the solution of nonlinear problems is the so-called Newton-Krylov method in which, a linearized problem is solved iteratively using a Krylov method to solve for each iterate and parallel computing is exploited using distributed preconditioning inside the Krylov loop. This approach is enough for traditional time-advancing methods as soon as the "roughness" of the nonlinear problem can be reduced by reducing the time-step size.

However, in a space-time context linearization is external to temporal evolution and the whole problem is solved (the whole time window is covered) at each nonlinear iteration, which has two effects: the roughness of the nonlinear problem cannot be changed through the discretization parameters and each iteration is quite expensive. The attempt made in task 3.1, reported in D3.1, was to develop a nonlinear preconditioner to replace the Newtown iteration, exploding parallel computing directly and not inside a Krylov method. However, although a reduction in the number of iterations is observed, this is

not enough to composite the extra cost of additional local solvers required.

Therefore, additional research is required to face the strongly nonlinear problem posed by space-time algorithms applied to turbulent flows. One possibility is to implement a composition of nonlinear solvers, as it is used for linear ones, see [4]. In fact, task 3.6 (ending M30) was original devised to perform "a careful calibration of the nonlinear algorithms" as it is required by the extraction of parallelism in a space-time context. The actual behaviour of the algorithms developed in task 3.1 actually poses in question even this development. We have recently learned from the literature [13] that the nonlinear space-time problem of turbulent flows is actually ill-conditioned. For these reasons, the numerical simulation of turbulent flows using a space-time approach is not feasible at this stage of the project and it is not reported herein.

3 High-rise benchmark building - CAARC

Computational Wind Engineering (CWE) has became an established research field with practical application, where CFD is employed to simulate wind effects on structures. The wind effects on long span bridges and tall buildings are traditionally done using wind tunnels. However, with advancement in CWE it is possible to numerically simulate these wind effects on structures. Kratos is used in the present work to study the wind effects on tall buildings. The Solver used for the project needs to be validated with comprehensive comparison with experimental results. For this purpose *Commonwealth Advisory Aeronautical Council* (CAARC) benchmark building B [8] is chosen to validate the CFD solver in Kratos.

Building B of the CAARC benchmark is widely studied and simulated in CWE and CFD [6] [8] [7] and [10]. The benchmark building B is a rectangular building with a classical bluff body shape having a flat top and rectangular cross section. The lateral walls are also flat without any parapets or other geometric features. The plan dimensions are Width W = 45 m and Length L = 30 m. The height of the building is H = 180 m.

3.1 Reference measurements and previous studies

Previous studies were conducted both in the wind tunnel and using numerical simulation. Braun 2009 [3] presented a numerical study on the aerodynamic and aero-elastic response of CAARC building B and the results are compared with previous studies. The current study will refer to [3] for validation purposes. The *Quantity of Interest* (QoI) reported are the pressure coefficients at a height of 2/3 H , the force and the moment coefficients. As these coefficients are normalized with the velocity it is easier to make a quantitative comparison for validation with the present work.

The pressure coefficient at any point is defined as

$$C_{P_i} = \frac{P_i - P_0}{1/2\rho(V_H)^2} \tag{1}$$

The force and moment coefficients used for validation in this study is listed below. The moment is computed about the centroid of the plan geometry of the building at ground

location.

$$C_{F_X} = \frac{\sum_{i=1}^{NTN} F_{X_i}}{1/2\rho W \int_0^H V^2 dZ}; \quad C_{\sigma F_X} = \frac{\sum_{i=1}^{NTN} \sigma F_{X_i}}{1/2\rho W \int_0^H V^2 dZ}$$

$$C_{F_Y} = \frac{\sum_{i=1}^{NTN} F_{Y_i}}{1/2\rho W \int_0^H V^2 dZ}; \quad C_{\sigma F_Y} = \frac{\sum_{i=1}^{NTN} \sigma F_{Y_i}}{1/2\rho W \int_0^H V^2 dZ} \tag{2}$$

$$C_{M_X} = \frac{\sum_{i=1}^{NTN} M_{X_i}}{1/2\rho \quad V_H^2 W H^2}; \quad C_{\sigma M_X} = \frac{\sum_{i=1}^{NTN} \sigma M_{X_i}}{1/2\rho \quad (V_H)^2 W H}$$

$$C_{M_Y} = \frac{\sum_{i=1}^{NTN} M_{Y_i}}{1/2\rho (V_H)^2 W H^2}; \quad C_{\sigma M_Y} = \frac{\sum_{i=1}^{NTN} \sigma M_{Y_i}}{1/2\rho (V_H)^2 W H^2}$$

where, ρ is the density of the fluid, V_H is the velocity at height H, and W is the building width. C_{F_X} represents the force coefficients in the direction X and M_{F_X} is the moment coefficient in the same direction. $C_{\sigma F_X}$ is the root mean square value of the fluctuation of forces in the X direction and $C_{\sigma M_X}$ is the root mean square value of the fluctuation of moment in the same direction.

3.2 Details of the numerical simulation

The properties of fluid and problem data are tabulated in Table 1. The high Reynolds

Table 1: Fluid properties and problem data

Density ρ $[kg/m^3]$	1.225
Viscosity μ m^2/S	$1.507E^{-05}$
Dynamic viscosity ν $N.s/m^2$	$1.846E^{-05}$
Characteristic length w $[m]$	45
Reynolds number Re	$9.7E^{07}$

number in the current simulation makes the flow very turbulent. Wind flow around the rectangular building is simulated using CFD analysis with the open-source Kratos Multiphysics tool. This involves a *finite element method* (FEM) formulation for flow problems based upon a VMS formulation [5]. The fluid domain is modeled with *Fractional step elements*. The computational domain setup is presented in the Figure 1. The extends of the fluid domain and the boundary conditions are shown in the figure. The blockage ratio is less than 0.8%.

3.3 Inlet wind condition

A steady inlet condition is used for the validation case. A logarithmic profile is adopted in the study.

$$\bar{u}(z) = \frac{u^*}{k} \ln(z/z_0) \tag{3}$$

where, k is von Karman's constant. $k \approx 0.4$
u^* is the friction velocity
and z_0 is the roughness length

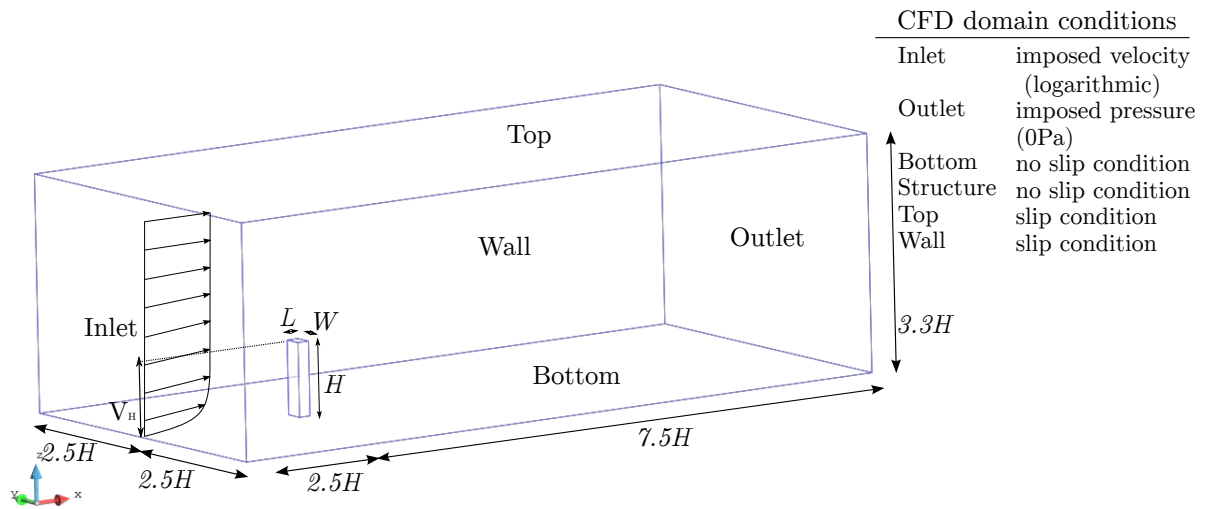| CFD domain conditions	
Inlet	imposed velocity (logarithmic)
Outlet	imposed pressure (0Pa)
Bottom	no slip condition
Structure	no slip condition
Top	slip condition
Wall	slip condition

Figure 1: Details of the CFD simulation

The reference mean wind velocity is measured/defined at a reference height z_{ref} A corrected logarithmic profile is used for heights more than 30 m as in [11]

$$\bar{u}(z) = \frac{u_*}{\kappa} \left(\ln \left(z/z_0 + 1 \right) + 34.5 fz/u_* \right) \tag{4}$$

with the Coriolis parameter $f = 2\Omega sin\phi$ where, Ω is the angular velocity of earth $(0.724.10^{-4}$ rad/s$)$ and ϕ is the geographical latitude. z_0 can be chosen from EN 1991-1-4 for the corresponding terrain category. Generally the wind is measured at a reference height and the wind velocity at the reference height is averaged over time period of 10 minutes (600s)

Wind Fluctuations Super imposed to the mean wind velocity are the fluctuating component of wind or wind gusts. These wind gusts causes the peak wind loads. The gust wind also depends majorly on the roughness length. The Mann model [11] will be used in the study. The model has three parameters, $\Gamma_{LT}, L_{iso}, E_0$. these three parameters control the statistics of the generated turbulence such as turbulence intensity and turbulence length scale. These three parameters are used to fit a Kaimal spectra in the study. The above mentioned model parameters of the spectra are as [11]

$$\begin{aligned} \Gamma &= 3.9 \\ L &= 0.59z \\ E_0 &= 3 \cdot 2 \frac{u_*^2}{z^{2/3}} \end{aligned} \tag{5}$$

where z is the height above ground for the point of interest and u_* is obtained from the log profile as in Eq. 3

A mean velocity of 40 m/s as in the original CAARC study at a reference height of 180 m is chosen for this study. The roughness length value of 2 m is chosen so that the logarithmic profile used for the current study matches with the exponential profile defined in [8] with exponent of 0.3.

3.4 Validation of the CFD results

The results obtained from the numerical simulation is compared with Braun (2009) [3]. A quantitative comparison is made with the force, moment coefficients defined earlier and the pressure coefficient at a height of 2/3H. A qualitative comparison of the pressure and velocity field is also made. s

3.4.1 Pressure Coefficients

Time mean and r.m.s. value of fluctuations of pressure coefficients at a height of 2/3 H is plotted for comparison in Figure 2. The results obtained from the present study is compared with previous experimental and numerical studies. The data included four experimental measurements from wind tunnels and 3 numerical simulations. The Cp plots are in good agreement with the established cp distribution. A very good agreement is
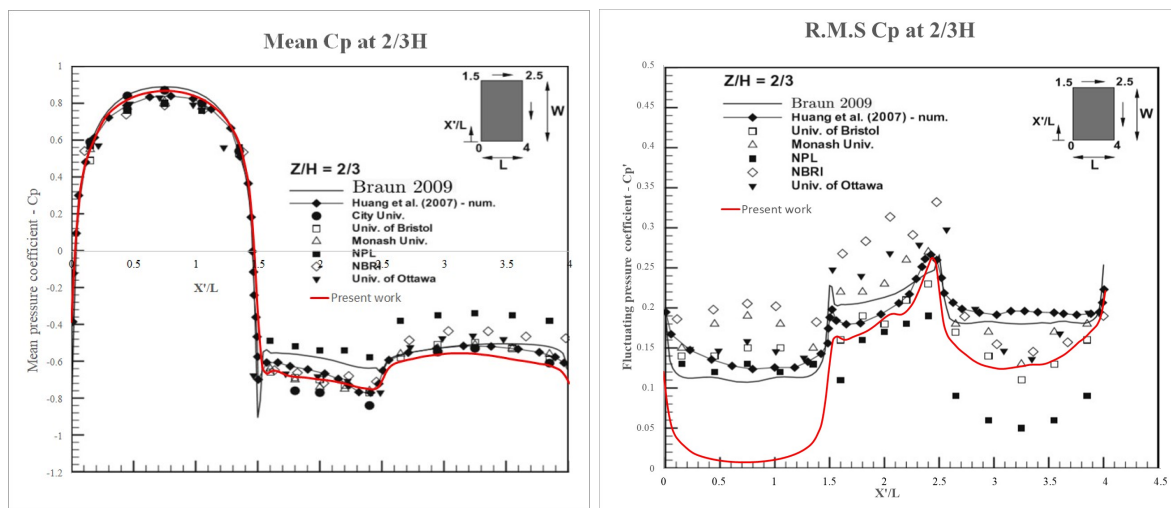
(a) Mean Coefficient of pressure at height of 2/3 H (b) R.M.S of fluctuation - Coefficient of pressure at height of 2/3 H

Figure 2: Coefficient of pressure validation - Mean and r.m.s of fluctuation

seen in the front face of the building and at the back side of the building for mean values. A jump seen in the Cp distribution at the front corner is not captured by the present study. A similar observation can be made about the r.m.s. of fluctuations as well. A good agreement of Cp values can be seen at the back face of the building. However, the r.m.s value at the front face of the building is under predicted in the present study. It is clear that the r.m.s of the pressure distribution is more dependent on the turbulent characteristics of the incoming wind, this might be the reason for the observed difference.
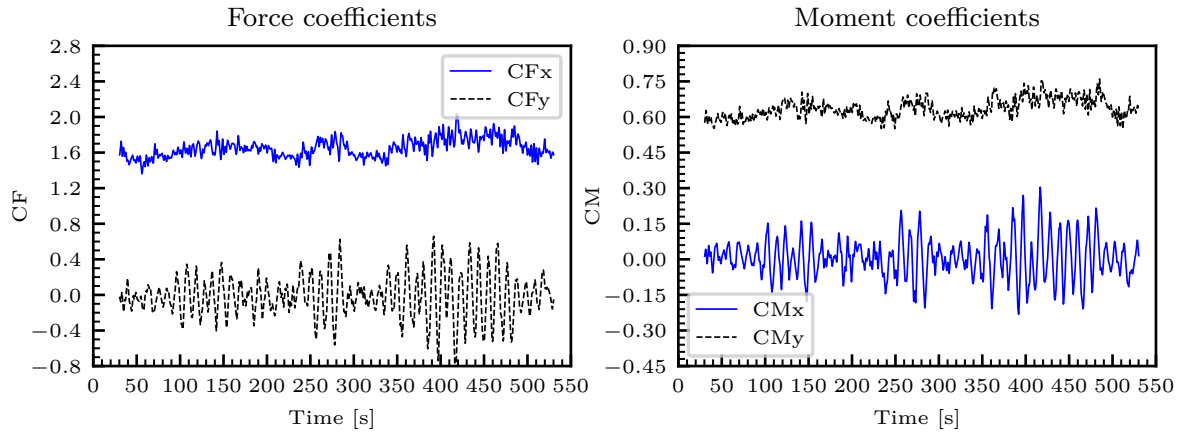
3.4.2 Forces and moment data

The aerodynamic coefficients are plotted with time for the present study and are compared with Braun 2009 in Figure 3. The time history plots are in good agreement with each other. The CFy plots have a larger variation in the present study. Time mean and time r.m.s of the force coefficients are tabulated in Table 2. The mean values of the force and moment coefficients have a closer agreement to Braun(2009)[3]. However the r.m.s

Table 2: Force and moment coefficients - validation

References	C_{F_X}	$C_{\sigma F_X}$	C_{F_Y}	$C_{\sigma F_Y}$
Present work	1.648	0.102	-0.015	0.238
Braun [3]	1.660	0.076	0.008	0.106
Obasaju [12]	1.490	0.060	-0.039	0.092
Huang et al. [9]	1.830	0.060	0.006	0.134
-	C_{M_X}	$C_{\sigma M_X}$	C_{M_Y}	$C_{\sigma M_Y}$
Present work	0.005	0.083	0.634	0.039
Braun [3]	0.004	0.048	0.570	0.038
Obasaju [12]	0.000	0.043	0.640	0.030
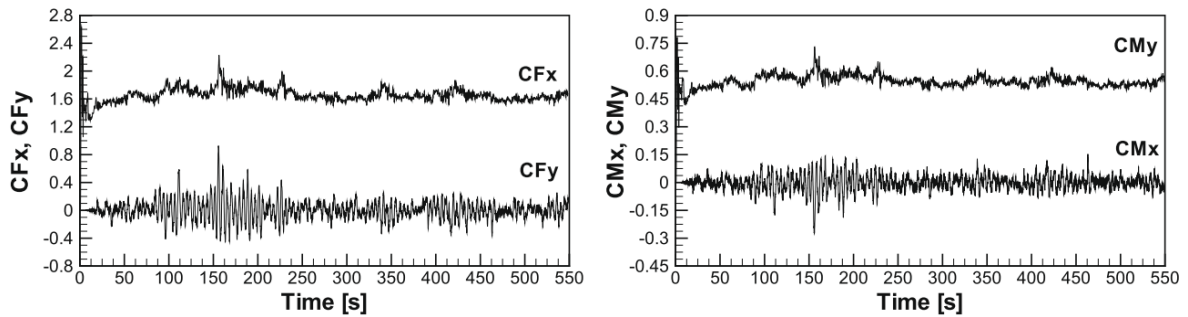
(a) Coefficient of forces and moments reported in Braun 2009 [3].

(b) Coefficient of forces and moments - current study

Figure 3: Coefficient of forces and moments validation - Time histories

values of the force and moment coefficients are found to have a larger error compared to the mean values. This may be due to the higher Reynolds number in the present study compared to the reference flow making the flow more turbulent. A good correlation is observed between the two data.

3.4.3 Flow field comparison

Figure 4 compares the time average pressure fields at a height of 2/3H. Both the pressure fields are in good agreement with each other. Since the inflow profile is different a quantitative comparison for the average pressure field is not possible. However, the flow field is capturing similar flow phenomenon. The time average of pressure field in the present work is done only with taking samples at 5 seconds intervals making the averaging of the field with some sampling errors.
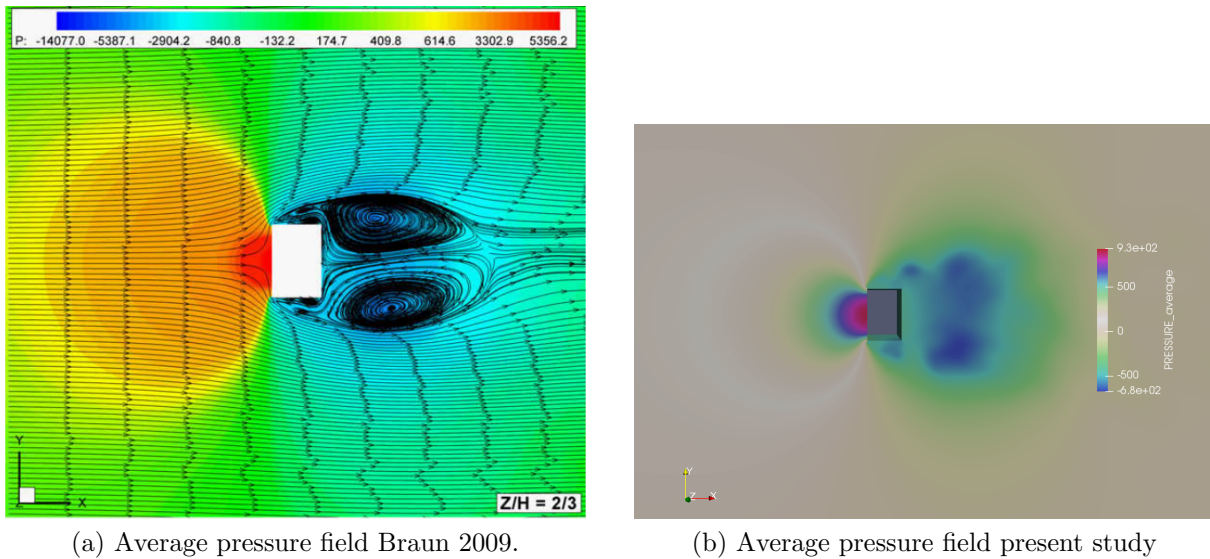

(a) Average pressure field Braun 2009.   (b) Average pressure field present study

Figure 4: Flow field comparison - at 2/3 H.

3.5 Validation results and conclusion

The CAARC benchmark building B is validated with the results presented in Braun 2009 [3] and a good agreement is found for pressure coefficients (Cp distribution), force coefficients, moment coefficients and in the characteristics of wind flow fields around the building.

4 Validation of Embedded Solveres : High-rise building

In this section the results of a series of CFD simulations over a straight and a twisted by 180° degrees tower is performed are presented, aiming in verifying the employed novel embedded CFD solver against the standard body-fitted approach. Moreover, a workflow developed in the frame of the ExaQUte project is also presented, on which the parameterized exact CAD model is used for the embedded CFD simulations.

The problem comprises in the prediction of the flow variable (velocity and pressure) around a building with a 0° and 180° twist angle when encountering a real wind as inflow condition. The corresponding problem setups of the tower with the 0° and 180° twist
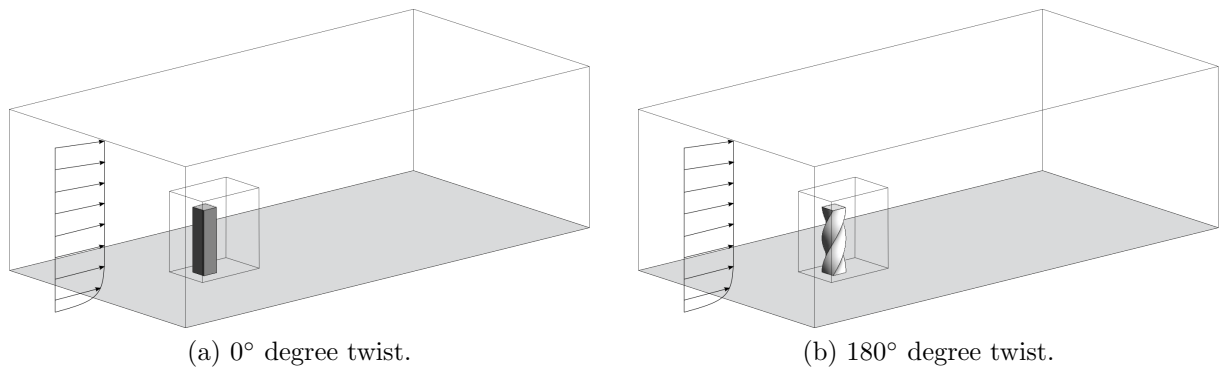
(a) 0° degree twist.   (b) 180° degree twist.

Figure 5: Problem settings of the flow around a tower with 0° and 180° degree twist.

(a) CAD model.   (b) $h_{\min} = 1E+1$.   (c) $h_{\min} = 1E+0$.   (d) $h_{\min} = 1E-1$.
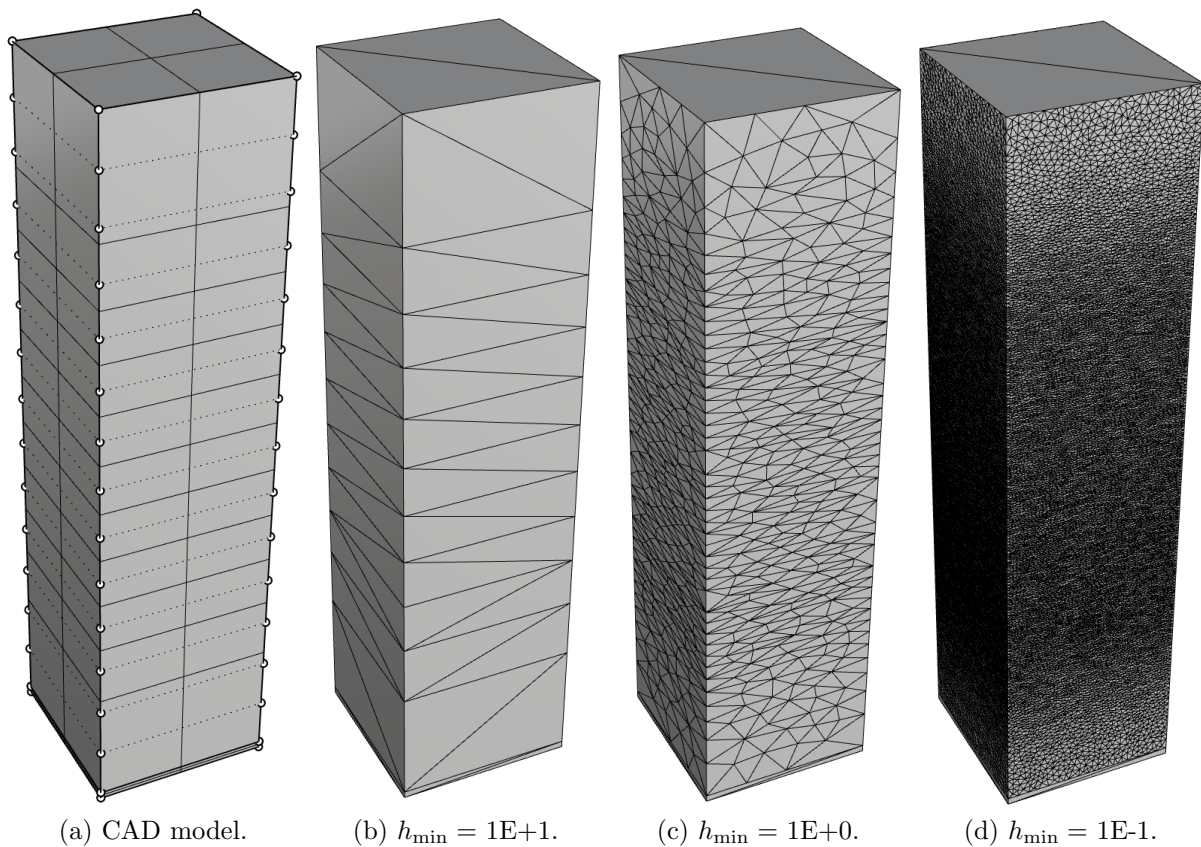
Figure 6: CAD model and tessellation of the tower with 0° angle twist.

angle are depicted in Figs. 5a and 5b, respectively. The computational domains comprise two parts, an outer and an inner block, as depicted in Fig. 5. a logarithmic profile $2.618/0.4 \cdot \log((z/0.4) + 1)$ is used as the inlet with roughness length of 0.4 and a mean velocity of 40 m/s at 180 m reference height. The computational mesh generated at the outer block is the same for both angles of twist, whereas the computational mesh within the inner block is adjusted for the different geometries in what concerns the body-fitted approach. As expected, the computational mesh in the inner block is the same for all embedded simulations, which is the main advantage of the novel embedded CFD method, that is, the simplicity of generating computational meshes for complex geometries.

(a) CAD model.   (b) $h_{\min} = 1\text{E}+1$.   (c) $h_{\min} = 1\text{E}+0$.   (d) $h_{\min} = 1\text{E}-1$.
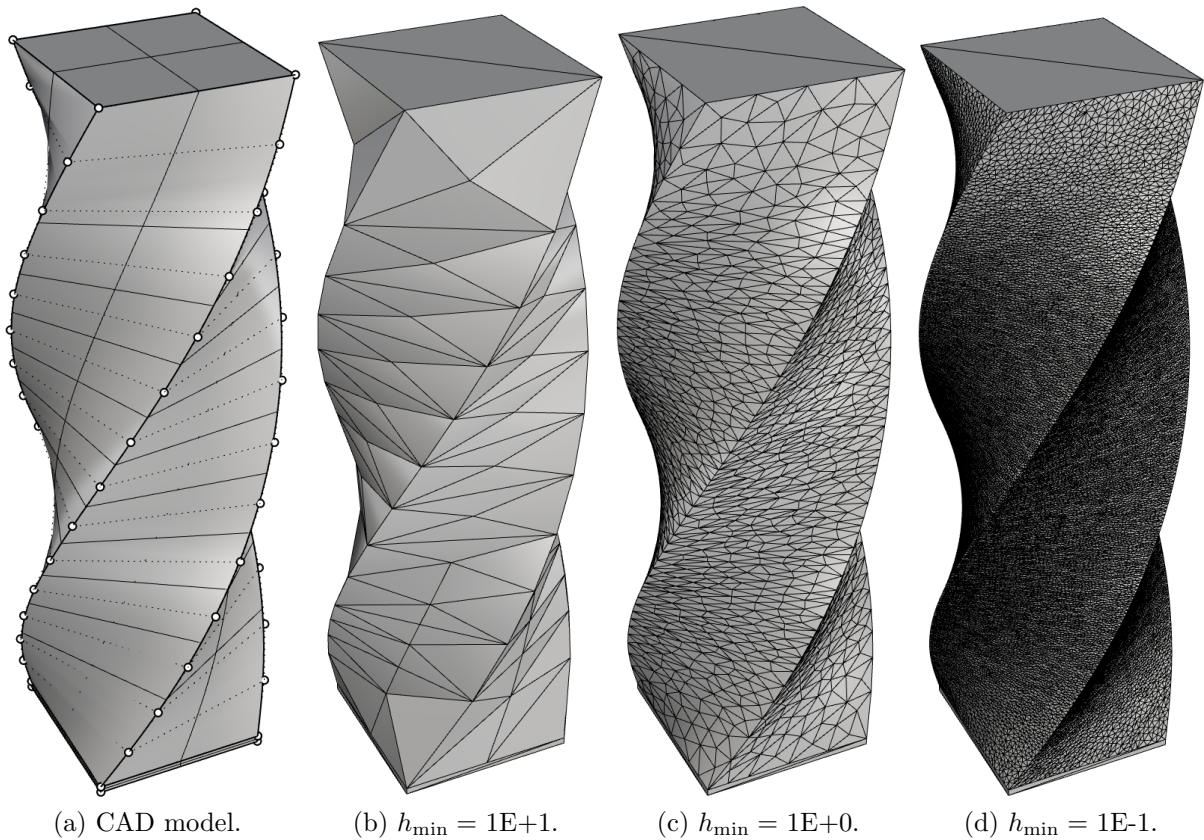
Figure 7: CAD model and tessellation of the tower with 180° angle twist.

The employed implementation of the embedded approach [14] using the AUSAS discontinuous basis functions [1] expects a tessellation describing the embedded object. Then a robust algorithm for finding the intersection of the skin of the embedded object with the background mesh within KRATOS MultiPhysics is used [2].

At first, a parameterized CAD model of a high-rise building is provided in terms of *Non-Uniform Rational B-Spline* (NURBS). The model is parameterized such that the ceiling of the the high-rise building can be rotated by any desirable degree whereas its foundation is fixed. In this way a square crossed sectioned high-rise building with an arbitrary twist angle can be generated. For the purposes of this deliverable two twist angles are employed: a 0° and a 180° twist angle, see Figs. 6a and 7a, respectively. A tessellation algorithm is implemented within KRATOS MultiPhysics which creates a tessellation of multipatch NURBS surface given its original CAD model in terms of NURBS. This tessellation algorithm generates a tessellation with a desirable accuracy in terms of the chordal error, that is, the maximum distance of the tessellated to the exact geometry. For both geometric settings, three levels of the tessellation are demonstrated, for $h_{\min} = 1E + 1$ m, $h_{\min} = 1E + 0$ m and $h_{\min} = 1E - 1$ m, see Figs. 6b, 7b, Figs. 6c, 7c and Figs. 6d, 7d, respectively. As it can be seen, the density of the generated tessellation increases as the accepted choral error is decreasing.

Next, the computational meshes are briefly discussed. Fig. 8 shows the computational
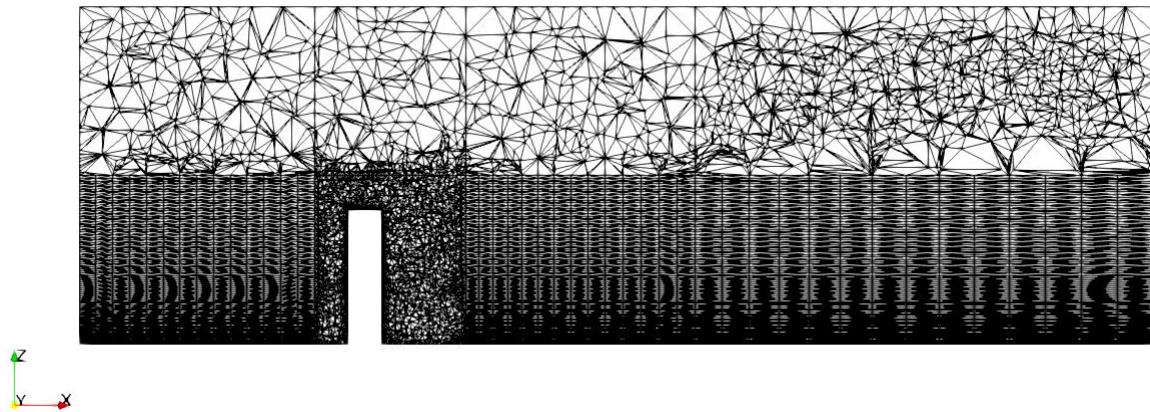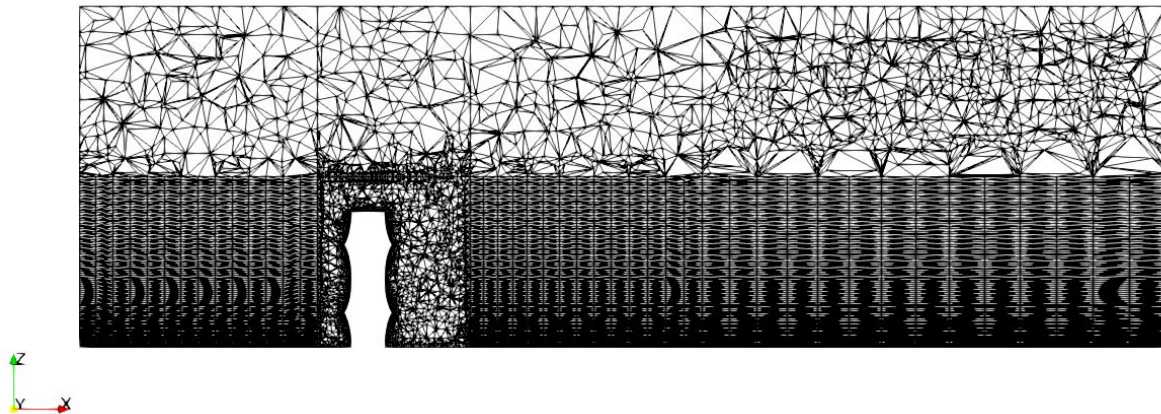
(a) Body-fitted CFD mesh.
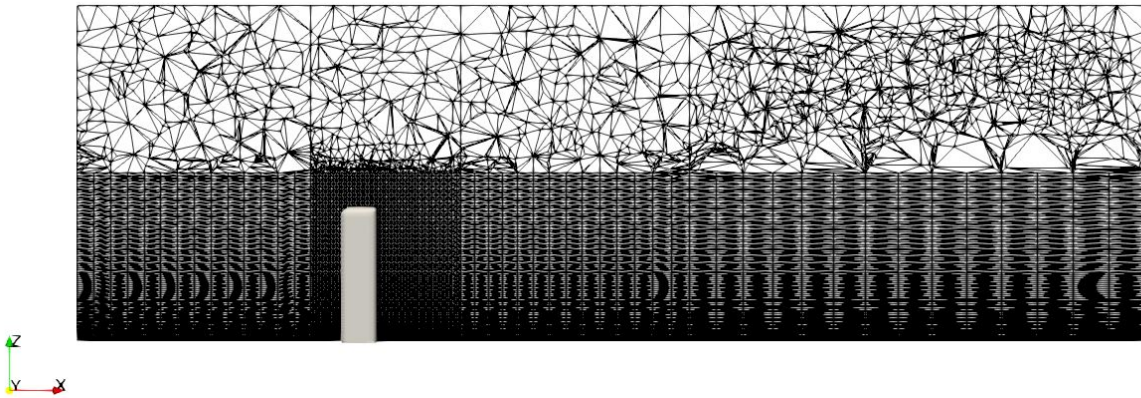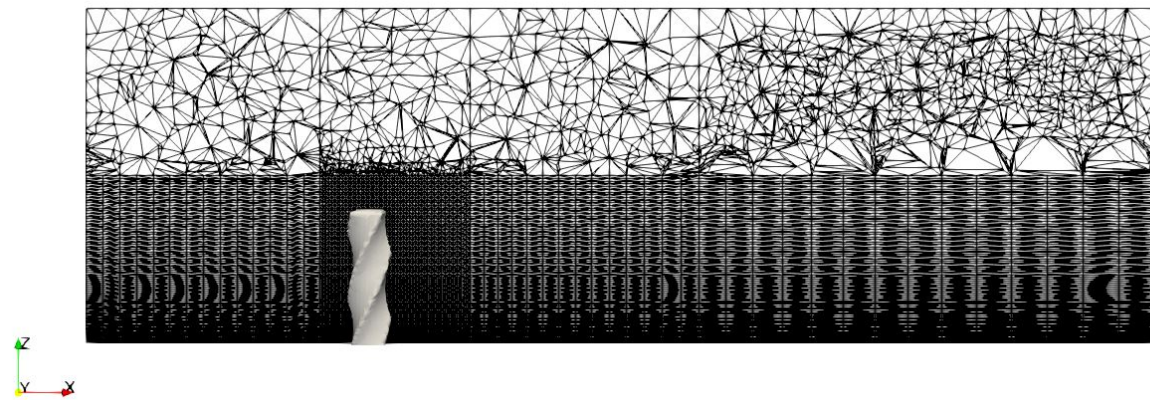

(b) Embedded CFD mesh.

Figure 8: CFD meshes for the tower with 0° degree twist.

CFD meshes for the straight tower, whereas Fig. 9 shows the computational CFD meshes for the tower with a twist angle of 180° degrees. It can be observed that although for the embedded approach, the exact computational mesh is used for both twist angle settings, see Figs. 8b, 9b, this is not the case for the body-fitted approach, see Fig. 8a, 9a where the mesh has to be manually adapted to exactly follow the embedded object. This is one of the important advantages of the embedded as opposed to the body-fitted approach, as the generation of a mesh exactly following a free form surface is often tedious, costly and inefficient.

In order to perform a fair comparison of the two solvers, the same computational meshes are used for the outer blocks, whereas the computational mesh is only adapted within the inner block where the tower is found as already mentioned in the preamble of this section. In contrast to the body-fitted approach where an unstructured mesh had to be generated within the inner block in order to obtain a body-fitted mesh around the tower, this is not the case for the embedded approach where a block structured mesh can be employed, see Figs. 6d, 7d. Having a block structured mesh is quite advantageous in terms of computational efficiency and accuracy as compared to an unstructured mesh

(a) Body-fitted CFD mesh.

(b) Embedded CFD mesh.

Figure 9: CFD meshes for the tower with 180° degree twist.

in that less and well-shaped elements are generated for the same level of accuracy, thus sparing time and gaining better accuracy levels.

The proposed workflow is as follows: A CAD model of the desirable object in flow is provided in terms of NURBS. The CAD model is tessellated up to any desirable level of accuracy based on the chordal error. This is very important as the tessellation can be made arbitrarily fine to capture features of any extent of the embedded object. Then, the level set function is used within KRATOS MultiPhysics to obtain the isosurface constructed by the intersections of the background fluid computational mesh and the provided tessellation of the embedded object. Subsequently, the novel embedded CFD solver is called and the simulation is performed on that initially created CFD mesh, which is highly efficient compared to the body-fitted approach where the computational CFD mesh has to be adapted by the user for any scale change on the initial CAD model of the object in flow.

Next the contours of the flow variables between the body-fitted and the embedded simulations at time instance $t = 20$ s are compared for the desirable verification purposes. The underlying contours are plotted on a lengthwise cut along the flow streamlines of the
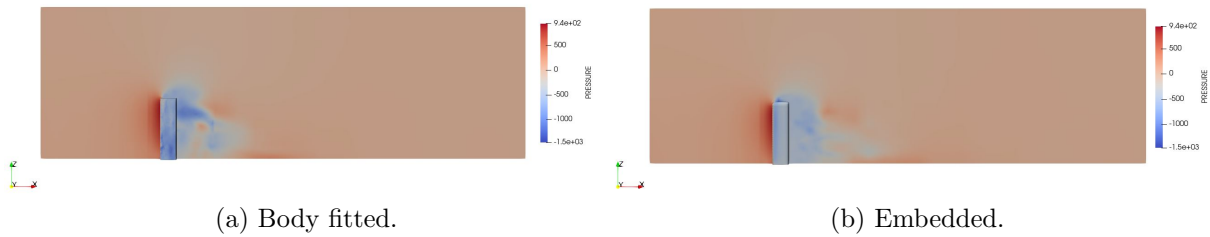
(a) Body fitted.

(b) Embedded.

Figure 10: Pressure field of the flow simulation around the tower with 0° angle twist at time instance $t = 20$ s.

(a) Body fitted.

(b) Embedded.

Figure 11: Velocity magnitude field of the flow simulation around the tower with 0° angle twist at time instance $t = 20$ s.

(a) Body fitted.

(b) Embedded.

Figure 12: Pressure field of the flow simulation around the tower with 180° angle twist at time instance $t = 20$ s.

(a) Body fitted.

(b) Embedded.
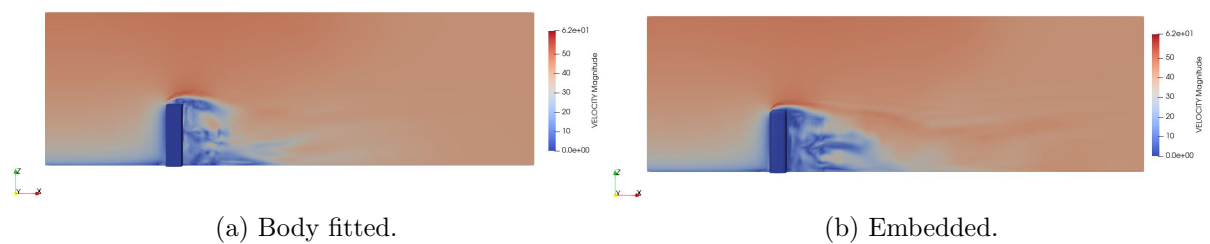
Figure 13: Velocity magnitude field of the flow simulation around the tower with 180° angle twist at time instance $t = 20$ s.
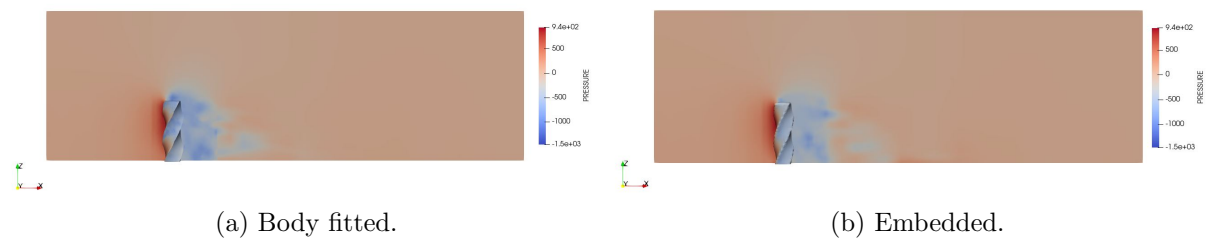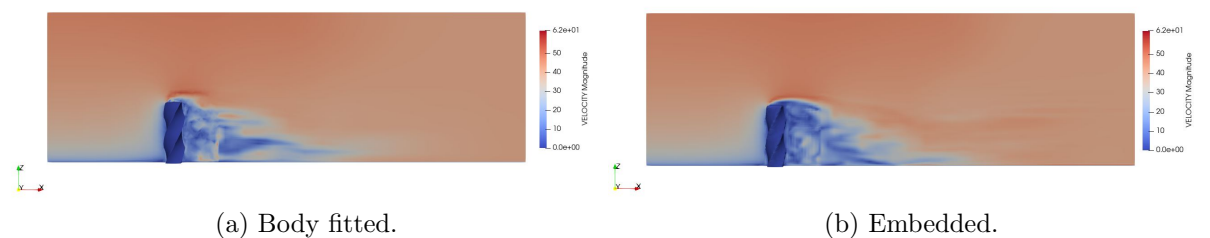
computational domain. First the results are presented for the setting with the straight tower and subsequently for the setting with the twisted tower by180° degrees.

Figs. 10a and 10b depict the pressure fields resulting from the body-fitted and the

embedded solver on the straight tower, respectively. The results show good agreement especially for the area behind the tower where the dominant part of the resulting forces acting on the tower. The next set of Figs. 11a, 11b depicts the distribution of the velocity magnitude field where once more a good agreement of the results can be deduced, also in the recirculation area past the tower.

Lastly, the pressure and velocity magnitude fields are plotted and compared for the case of the tower with the 180° twist angle. Figs. 12a and 12b depict the contour of the pressure field for the body-fitted and the embedded solver, respectively. As before a satisfactory agreement of the pressure fields on the tower surface can be observed between both cases, where the positive and negative pressure faces can be clearly distinguished, pre and post the tower, respectively, as expected. Finally, the velocity magnitude contours for both the body-fitted and the embedded solver are shown in Figs. 13a and 13b, respectively. The results show a highly convincing evidence that the embedded solver is able to capture all essential for the flow characteristics even for this high Reynolds number scenario.

A API definition and usage

```
191        nameOutputFile = 'geometry.json'
192        H = 180.0
193        Theta = 0
194        twisted_tower.generate_twisted_tower(parameters, H, Theta)
195
196        model = KratosMultiphysics.Model()
197        simulation = FluidDynamicsEmbeddedAnalysisWithFlush(model,parameters,nameOutputFile)
198        simulation.Run()
```

Figure 14: On the fly CAD geometry generation in terms of NURBS.

In this appendix section a brief demonstration of the implemented *Application Programming Interface* (API) and the underlying functions in the analysis stage is provided. As mentioned before, the exact CAD geometry in terms NURBS is generated on the fly using a geometry library in python developed in the scope of the ExaQUte project, see Fig. 14. In the latter API snippet it can be seen how the height and the twist angle can be specified as arguments in function `generate_twisted_tower` from the geometry library so that a `geometry.json` is written out.

```
33        self.skin_model_part = model.CreateModelPart(self.project_parameters["embedded_iga"]["model_part_name"].GetString())
34        iga_model_part = model.CreateModelPart("IgaModelPart")
35
36        self.skin_model_part.ProcessInfo.SetValue(KratosMultiphysics.DOMAIN_SIZE, self.project_parameters["solver_settings"]["domain_size"].GetInt())
37        self.skin_model_part.AddNodalSolutionStepVariable(KratosMultiphysics.VELOCITY)
38        self.drag_force_vector = np.zeros([0,4])
39
40        ## READ CAD GEOMTRY AND CREATE SKINMODELPART
41        with open(nameOutputFile,'r') as geometry_file:
42            iga_geometry_parameters = KratosMultiphysics.Parameters( geometry_file.read())
43
44        iga_geometry_reader = KratosMultiphysics.IgaApplication.BrepJsonIO()
45        embedded_iga_modeler = KratosMultiphysics.IgaApplication.EmbeddedIgaModeler(iga_model_part, self.project_parameters["embedded_iga"])
46        embedded_iga_modeler.ImportGeometry(iga_geometry_reader, iga_geometry_parameters)
47        embedded_iga_modeler.CreateElements3D(self.skin_model_part)
```

Figure 15: On the fly tessellation of the CAD geometry.

Next, the initialization phase of the analysis stage is enhanced by reading in the name

of the `geometry.json` and subsequently performing a tessellation based on the user-specified chordal error level, see Fig. 15.

```
99      ##COMPUTE DISTANCE GRADIENT AND NODAL_H
100     local_gradient = KratosMultiphysics.ComputeNodalGradientProcess3D(self._GetSolver().GetComputingModelPart().GetRootModelPart(),
101     KratosMultiphysics.DISTANCE,
102     KratosMultiphysics.DISTANCE_GRADIENT,
103     KratosMultiphysics.NODAL_AREA)
104
105     local_gradient.Execute()
106     find_nodal_h = KratosMultiphysics.FindNodalHNonHistoricalProcess(self._GetSolver().GetComputingModelPart().GetRootModelPart())
107     find_nodal_h.Execute()
```

Figure 16: Computing the level set function of the geometry tessellation.

Then, the level set function is called on the aforementioned generated tessellation, so that the isosurface of the intersection between the tessellation and the background mesh is obtained, which is essential for the embedded CFD solver, see Fig. 16. Then, the embedded CFD solver is called as usual and the simulation takes place. This is meant to demonstrate the efficiency of the proposed workflow consisting of an on-the-fly geometry generation, on-the-fly geometry tessellation and lastly embedded CFD simulation.

References

[1] R. F. Ausas, F. S. Sousa, and G. C. Buscaglia. An improved finite element space for discontinuous pressures. *Computer Methods in Applied Mechanics and Engineering*, 199(17):1019 – 1031, 2010. ISSN 0045-7825. doi:https://doi.org/10.1016/j.cma.2009.11.011. URL `http://www.sciencedirect.com/science/article/pii/S0045782509003867`.

[2] D. Baumgärtner, J. Wolf, R. Rossi, P. Dadvand, and R. Wüchner. A robust algorithm for implicit description of immersed geometries within a background mesh. *Advanced Modeling and Simulation in Engineering Sciences*, 5(1):21, 2018.

[3] A. L. Braun and A. M. Awruch. Aerodynamic and aeroelastic analyses on the CAARC standard tall building model using numerical simulation. *Computers and Structures*, 87(9-10):564–581, 2009. ISSN 00457949. doi:10.1016/j.compstruc.2009.02.002. URL `http://dx.doi.org/10.1016/j.compstruc.2009.02.002`.

[4] P. R. Brune, M. G. Knepley, B. F. Smith, and X. Tu. Composing scalable nonlinear algebraic solvers. *SIAM Review*, 57(4):535–565, 2015.

[5] J. Cotela-Dalmau, R. Rossi, and E. Oñate. A fic-based stabilized finite element formulation for turbulent flows. *Computer Methods in Applied Mechanics and Engineering*, 315, 11 2016. doi:10.1016/j.cma.2016.11.020.

[6] A. K. Dagnew and G. T. Bitsuamlak. LES evaluation of wind pressures on a standard tall building with and without a neighboring building. *The Fifth International Symosium on Computational Wind Engineering (CWE2010) Chapel Hill, North Carolina, USA*, (2007), 2010.

[7] A. Elshaer, H. Aboshosha, G. Bitsuamlak, A. El Damatty, and A. Dagnew. LES evaluation of wind-induced responses for an isolated and a surrounded tall building. *Engineering Structures*, 115(May):179–195, 2016. ISSN 18737323. doi:10.1016/j.engstruct.2016.02.026. URL `http://dx.doi.org/10.1016/j.engstruct.2016.02.026`.

[8] J. D. Holmes and T. K. Tse. International high-frequency base balance benchmark study. *Wind and Structures, An International Journal*, 18(4):457–471, 2014. ISSN 12266116. doi:10.12989/was.2014.18.4.457.

[9] S. Huang, Q. Li, and S. Xu. Numerical evaluation of wind effects on a tall steel building by cfd. *Journal of Constructional Steel Research*, 63(5):612 – 627, 2007. ISSN 0143-974X. doi:https://doi.org/10.1016/j.jcsr.2006.06.033. URL `http://www.sciencedirect.com/science/article/pii/S0143974X06001593`.

[10] G. N. Lygidakis, S. S. Sarakinos, and I. K. Nikolos. Imece2016-66399 Simulation of the Flow Over the Caarc Standard Tall Building. pages 1–10, 2018.

[11] J. Mann. Wind field simulation. 13(4):269–282, 1998.

[12] E. Obasaju. Measurement of forces and base overturning moments on the caarc tall building model in a simulated atmospheric boundary layer. *Journal of Wind Engineering and Industrial Aerodynamics*, 40(2):103 – 126, 1992. ISSN 0167-6105. doi:https://doi.org/10.1016/0167-6105(92)90361-D. URL `http://www.sciencedirect.com/science/article/pii/016761059290361D`.

[13] Q. Wang, S. A. Gomez, P. J. Blonigan, A. L. Gregory, and E. Y. Qian. Towards scalable parallel-in-time turbulent flow simulations. *Physics of Fluids*, 25(11):110818, 2013.

[14] R. Zorrilla, A. Larese, and R. Rossi. A modified finite element formulation for the imposition of the slip boundary condition over embedded volumeless geometries. *Computer Methods in Applied Mechanics and Engineering*, 353:123 – 157, 2019. ISSN 0045-7825. doi:https://doi.org/10.1016/j.cma.2019.05.007. URL `http://www.sciencedirect.com/science/article/pii/S004578251930266X`.