UNIVERSITY OF AMSTERDAM

UvA-DARE (Digital Academic Repository)

Modelling flow-induced vibrations of gates in hydraulic structures

Erdbrink, C.D.

Publication date
2014

Citation for published version (APA):
Erdbrink, C. D. (2014). *Modelling flow-induced vibrations of gates in hydraulic structures*. [Thesis, fully internal, Universiteit van Amsterdam].

Download date:14 Feb 2025

3 Physics-based numerical modelling

3.1 Introduction

Suppose we are faced with the problem of finding the gate response signal from a set of given flow conditions, acting as input, and a physical model for performing experiments is not available. Physics-based numerical simulations with the purpose of system analysis is the traditional route for solving this problem. From a structural dynamics point of view this is an example of a direct method in the time-domain.

At the start of applying a numerical model, the main question is not which high-level, continuous-form differential equations describe the physics. For fluid-structure interaction (FSI), we have the Navier-Stokes for the flow and stress-strain relations for the structure – in general it is assumed that such a base already exists, derived via Figure 3 in section 1.2.1. The questions are rather what choices are sensible in the discretization and numerical solution of those equations and which boundary conditions are appropriate. In the interpretation of the model outcomes, important questions are how to judge model effects and scale effects and hence what the physical parameter ranges are where extrapolation is valid. Section 3.2 gives a framework for the application of computational fluid dynamics (CFD) models. The discussion is purposely biased towards FSI analysis. Section 3.3 then goes into a few specific aspects of FSI.

Although putting a detailed geometry of the barrier in a computer model is doable (albeit time-consuming), the modelling goal in chapters 3-6 is not to look at the structural response side in detail (i.e. by including trusses or beams and computing all stresses, etc.). This is because the assumption of a rigid body with a mass-spring suspension is a very reasonable and generalisable analogy for true structural flow-induced responses, while many simplifying assumptions for the modelling of the fluid flow are disastrous for capturing the flow field and therefore the flow load on the structure.

3.2 Applying CFD for problem solving in hydraulic engineering[1]

It is remarkable how different choices in numerical model set-up and use can shape and affect the way problems are solved in hydrodynamics. Some philosophy about applying models is necessary in order to get results that answer the right question and that pertain to the real world. When talking about solving a real-life physical problem, there is no single predefined algorithm that fits the job exactly. The result, after careful analysis and preparation, is more likely to be a combination of modelling tools. For ease of speech, this will still be referred to as a (numerical or computational) model. Figure 3.1 presents a framework aimed at preparing a model –or coming to an appropriate model choice– for solving a problem in hydraulic engineering. This diagram is walked through step by step from top to bottom.

---

[1] The framework and general part of the discussion in Section 3.2 is inspired by discussions about CFD use, research and development at Deltares, mostly with Mart Borsboom.
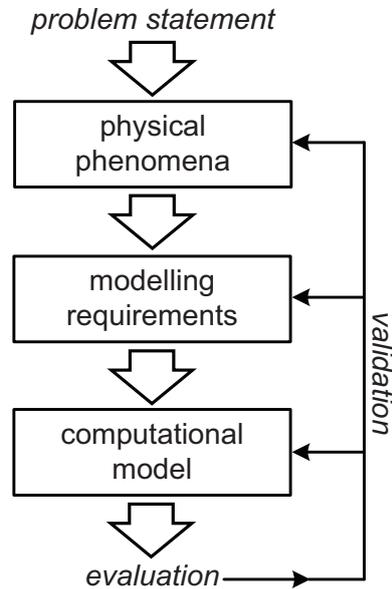
Figure 3.1. Application framework for CFD models.

Problem statement
It starts with the identification of a physical problem. There is usually one question that sums up the problem. If there are too many questions needed to pinpoint the problem, this is a hint that one model may not be enough. For the gate, three relevant questions are "What are the maximum dynamic forces experienced by the gate?", "Given a set of flow conditions and gate positions, will the gate start vibrating or not?" and "Which excitation mechanism can be expected to come about?".

Physical phenomena
In the first step, the task is to compile a list of the physical phenomena that play a role in the problem. Five classes of phenomena are mentioned: the influence of the free surface, turbulence, density effects, morphology and moving objects. For the vibrating gate problem, density effects and sediment transport are not important, but the other three phenomena are. The knowledge needed for judging the importance of these three aspects comes from Chapter 2.

This is also the moment to think about scales. When the fluid dynamics take place in a small volume and the smallest flow length scales have comparable sizes in all three dimensions, they can be called 'detail hydrodynamics'. Then turbulence is a factor and the flow is by definition non-hydrostatic, which means that the pressure does not vary proportionally with depth and the streamlines are not (approximately) parallel everywhere in the domain. Flow around a structure definitely files under detail hydrodynamics, but this does not mean that it is not also affected by conditions away from it – which are determined by systems of larger scale (called near-field and far field). Chapter 4 goes into this.

Modelling requirements
The now known physical phenomena can be modelled in various ways. It should be determined, in order to address the problem, which physical aspects should explicitly be represented by the model and to what level of detail. It is said that quantities that are computed explicitly (from fundamental principles) as a function of time are 'simulated', and quantities that are not, but do play a role in the problem and are part of the system, are 'modelled'. To model rather than simulate a physical quantity involves a certain degree of parametrization, i.e. representation by means of lumping, averaging or estimating of some kind. Practically all CFD and FSI models combine simulation and parametrization. Browsing through the list of relevant physical phenomena, it should be determined which need to be simulated and which can be modelled in a simplified way. A number of additional decisions are necessary too. For our problem the list is as follows.

(i). Time dependency
The flow around a static gate is quasi-stationary (Section 2.2) and could be modelled with some degree of accuracy by solving steady-state flow equations. The inclusion of a movable gate implies that the problem becomes fully transient (i.e. time-dependent). In fact, the model has to *simulate* the gate movement in order to do justice to the fluid-solid interaction.

(ii). What is the degree of non-hydrostaticity?
The vertical contraction of the underflow and the recirculation behind the gate call for a non-hydrostatic approach. Any simplification of the equations concerning vertical acceleration, e.g. as is the case in the shallow-water equations, will not suffice. Whether the model has to be three-dimensional is another matter. A two-dimensional gate section (uniform in width) provides a good model (Sections 2.2-2.4), but the turbulent kinetic energy may be underestimated, because $\overline{(v')^2} = 0$ in the model. The added value of a 3D flow simulation most likely does not outweigh the enormous extra computational costs.

(iii). How important is the free surface?
This depends on the discharge conditions that we want to examine. Moreover, the expected frequency of the gate motion determines if potential wave radiation is perceptible near the submerged zone of excitation at the gate bottom. If the problem is restricted to fully submerged flow with negligence of wave radiation, then a simulation of free-surface fluctuations seems unnecessary. As the downstream Froude number increases, this simplification becomes less and less suitable.

(iv). To which level of detail should turbulent effects be simulated?
It has been discussed that turbulence is inherently part of the flow around the gate. At the same time, the excitation mechanisms most likely to cause high-amplitude vibrations are not directly linked to turbulence. The turbulent eddies at the smallest (Kolmogorov) scales in the viscous range of the turbulence spectrum are surely important in the deceleration zone away from the gate and play a role in hydraulic damping due to the gate movement. But the excitation itself is presumably affected more strongly by the relatively large-scale fluctuations (in the inertial range of the spectrum) originating right away from those regions of the free shear layer with high stresses. Therefore, a parameterised modelling of turbulence is suggested that respects this. The difficulty of this point lies in the fact that a

pre-modelling dismissal of certain excitation mechanisms to some degree undermines the goal of the modelling: analysis of the process.

(v). Moving object

It has also already been established that the gate will act as a partly submerged oscillating object with one d.o.f.. The requirement of simulating a moving object strongly influences the model set-up with respect to mesh (see Section 3.3.3) and time discretization. Expected frequencies are estimated from the added mass of a rectangular object near a wall and the Archimedean-type hydraulic rigidity in the natural frequency formula; these indicate the minimum time time steps from a physics perspective.

(vi). What kind of boundary conditions are required?

The floor boundary of the flow is best modelled as a rough wall. The upper boundary of the flow domain is the free surface, which was already discussed. In flow models with more than one computational layer in the vertical, it is common to place a hydrostatic pressure boundary downstream and a velocity profile upstream. This is for good reason; there are many ill-defined or downright unstable boundary combinations (Stelling and Booij, 1999). The inclusion of waves would make especially the outflow boundary radically more complex (e.g. Wellens, 2012).

Computational model

The step from modelling requirements to computational model starts with an inventarisation of available computational tools. Here we are looking for a CFD model that can handle detail hydrodynamics and that can be extended to include the interaction between the flow and a moving object. Now we become concerned with numerical requirements such as discretisation schemes, grid type, solvers, etcetera. Trivial requirements such as "the model needs to be time-efficient, accurate and flexible" all fall in the category of finite computational capacity and available time. In summary, the everlasting trade-off between level of detail of modelled physics and simulation time (times number of required runs) needs to be identified in an early stage. It is good to think about how to spend your limited time as a model user. In CFD, preparing a run ('pre') and processing model results ('post') can also take a lot of time. How much do you want to do by hand (manual coding) and how much can be done with existing codes or software packages.

Evaluation and validation

After running the numerical model the results are visualised and evaluated. In the best scenario this leads to usable answers to the stated questions and recommendations for solving the (engineering) problem. The first series of runs never leads to the desired solution, however. There is critical feedback to check all previous choices – this process of building confidence in the model is called 'validation'. Validation works on all steps of Figure 3.1 and must be considered to be an iterative process, a loop, rather than a one-off action or a model property, in the spirit of Dee (1993). Validation actions are concerned with the question if the above steps are taken in the right direction, and not if they are implemented correctly; this is called 'verification'. Furthermore, a 'calibration' is a set of (additional) steps that make a physics-based computational model ready for a specific application. This is usually done with another data set (not used before in validation). So, as a result of these definitions, it is strange to speak about a "validated model", but it is possible to speak about a

"calibrated model". A computational model that has been tailor-made for solving a stated problem is called a 'model application'.

In conclusion, the central idea of these steps is to mold the model into a workable tool for answering the initial questions, and not the other way around (i.e. to search for questions that can be answered by your model). An interesting corollary is that a better description of the underlying physics or technical improvements to the algorithm may *or may not* contribute to better answers.

For the gate vibration problem, the crucial choices relate to above modelling requirements (iii), (iv) and (v). These are discussed further in the next section. From the onset our problem involved analysis by simulation and was not a prediction problem (Section 1.2), but with the inclusion of data for validation and calibration this distinction is starting to blur. If past output data is available and directly used in the process of computing future output values, then the model is used for prediction. However, in FSI context, it could well be the case that the validation data relates to the flow only – implying that in a strict sense the flow is predicted and the structural response is simulated.

3.3   Setting up a FIV model

3.3.1   Finite element method
The finite element method (FEM) is a numerical method for solving boundary value problems. It is based on a spatial discretization in a computational grid or 'mesh'. There are two other grid-based methods in CFD: finite difference method (FDM) and, the most widely used, finite volume method (FVM). Differences between these approaches are discussed in Van Kan et al. (2008). They still have plenty of computational features in common compared to CFD methods that are not grid-based. For instance, a shared issue is the importance of grid size and grid type, since we are dealing with a convection problem, see again Van Kan et al. (2008). Another typical aspect is that a *large* system of equations is being solved. The indication 'large' refers to the fact that standard solver algorithms will run into trouble when faced with systems of this size. A measure for the computational load is the grid size, i.e. the numbers of cells and nodes, times the number of computed time steps.

The FEM is also a classical tool for solving eigenvalue problems for the analysis of structures and elastic solids (Hughes, 2000). Eigenvalue problems (or 'eigenproblems') describe all sorts of frequency analyses such as the problem of determining natural frequencies and mode shapes of free vibrations of structural elements. Modal analysis, which involves solving eigenproblems, is usually performed in the design stage of structures, in order to make safety assessments. The ability to solve deformations and stresses of a solid (with a complex geometry) *and* the fluid flow makes the FEM a suitable approach for FSI problems.

3.3.2   Turbulence modelling
Direct numerical simulation (DNS) is a very costly technique that simulates all turbulent scales. All other methods apply parametrization of the turbulent dynamics to cut down on costs. Large eddy simulation (LES) divides the turbulent energy spectrum up in a parameterized part for the smaller scales and a simulated part for the larger scales, but this is still a computationally intensive approach. Application of DNS for real-life cases at high *Re*

brings great numerical challenges and is therefore very rare. LES is more accessible, but still far from a standard option for CFD applications. The household approach for real-life applications is to use a turbulent model. In this case the turbulence is approximated by averaging the Navier-Stokes equations after substitution of the Reynolds decomposition (Section 2.2) and to solve an extra set of equations to approximate the Reynolds stress term that describes the turbulence (see e.g. Zienkiewicz and Taylor, 2000).

For the present application three arguments are given why this last approach will be used. First of all, as argued in the previous section, there is reason to think that simulation of *small-scale* turbulence is not essential for representing the most important excitations. Secondly, combining LES with a moving object and possibly a moving free surface would become a very costly and experimental endeavour – not in line with the aims of this study. A third reason is that it is not sensible to compute small turbulence scales for concrete application purposes if there is no flow measurement data available at all – let alone data that cover a comparable level of detail.

The main flow equations will be given in Section 6.1. The two-equation k-ε model is used to model the Reynolds stresses, this is called the 'closure'. This turbulence model works by estimating the turbulent (eddy) viscosity ν_T using the formula

$$\nu_T = c_\mu \frac{k^2}{\varepsilon},$$

(3.1)

where c_μ is a fixed model constant and k, the turbulence kinetic energy (TKE), and ε, the turbulent dissipation, are both solved from their own transport equation. Details are found in Rodi (1993), a good reference for the analytical background and use of turbulent models in hydrodynamics. The two extra variables of the turbulent model need initial and boundary values. Moreover, their transport equations are not valid in the vicinity of walls. This is why a wall function is necessary to estimate the velocities and turbulent production and dissipation near walls.

3.3.3   The arbitrary Lagrangian-Eulerian mesh
Modelling a moving object interacting with fluid
For a grid-based computation of fluid-structure interaction we have to ensure that the body is able to move freely through the fluid domain and that it experiences static and dynamic water forces. To capture the active interaction, the feedback effect of a changed position of the body should also be included. This implies for the numerics that in addition to descriptions of the fluid and solid domain and their outside boundaries, conditions for the fluid-solid interface are necessary. These consist of dynamic and kinematic conditions, expressed in three parts (Wall et al., 2006):

  - force equilibirum, i.e. equility of stress vectors: $\boldsymbol{\sigma}_{\text{fluid}} \cdot \boldsymbol{n} = \boldsymbol{\sigma}_{\text{solid}} \cdot \boldsymbol{n}$,
  - no mass flow through the interface, i.e. equality of normal velocities: $\boldsymbol{u} \cdot \boldsymbol{n} = \frac{\partial \boldsymbol{y}}{\partial t} \cdot \boldsymbol{n}$,
  - the fluid viscosity also gives tangential components, resulting in the 'no-slip' condition
    $\boldsymbol{u} = \frac{\partial \boldsymbol{y}}{\partial t}$.

Here the bold symbols represent vectors, $\boldsymbol{n}$ is the normal vector, $\boldsymbol{u}$ is the flow velocity and $\boldsymbol{y}$ is the solid's displacement. If these conditions are met precisely, this implies a conservation of mass, momentum and energy at the interface (Wall et al., 2006).

The resulting system of equations can be solved in different ways. These have to do with a principal dilemma: most flow models use a Eulerian approach and most solid models use a Lagrangian approach, and both domains need information from each other at the interface to compute the next step (Dunne and Rannacher, 2006). The various classes of solver options are shown in Figure 3.2. The first option is a separation of the two domains. This is the 'partitioned' or 'segregated' approach. The domains are solved separately, i.e. in series, using an initial guess for the solid interface. In 'iterative' (or 'iteratively staggered') solving algorithms, in each time step, the serial steps are repeated until the solutions for both domains converge with respect to the interface conditions. A simpler and faster option is the 'sequential' (or 'sequentially staggered') scheme where you do the same with only one update step for the coupling information. That is, the initial prediction of the solid displacement is replaced once for each time step after the fluid and structural domains have been computed.
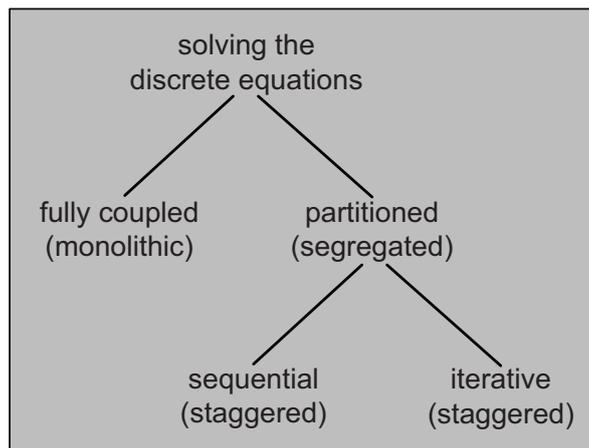

Figure 3.2. Computational approaches for fluid-structure interfaces, based on terminology by Wall et al. (2006) and Dunne and Rannacher (2006).

In a different setting, the equations of both domains are solved together as one system; this is the 'fully coupled' or 'monolithic' method. Two reasons why not to use fully coupled solving is that this prevents the use of dedicated solvers for each of the domains and the computational difficulties arising from the larger size of the system matrix. An advantage is of course the strong coupling type which –once solved– leads to a robust solution of the interface position.

One way to deal with the adaptation of the computational mesh to new interface positions at each time step is to introduce a mapping φ between the Eulerian fluid domain, with fixed or spatial coordinates, and a deforming reference system χ (following notation by Wall et al., 2006). All physical quantities find their actual position in the physical space $\boldsymbol{x}$ from the

transformation $x = \varphi(\chi,t)$ before they are actually computed in the fixed domain. Figure 3.3 illustrates this.
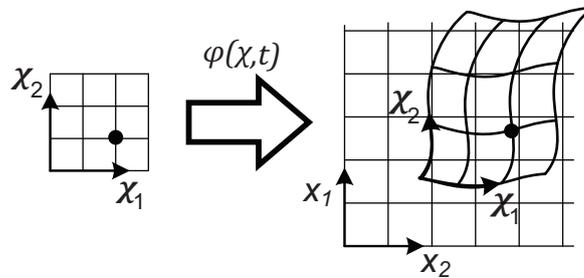

Figure 3.3. Mapping used in ALE mesh, diagram after Wall et al. (2006).

This deformation mapping has to be determined anew on each time step as part of the computations. Analytically, the transformation is achieved by inserting a time-dependent Jacobian of the mapping, $J_t = \det(\partial x/\partial \chi)$, into the Navier-Stokes equation. The method sketched here is called the arbitrary Lagrangian-Eulerian method (ALE). It basically adapts the fluid domain to changing boundaries and lets the structural domain keep its favorite Lagrangian form. Specific background in different modelling contexts is found in Ferziger and Perić (2002) and Donea et al. (2004).

The preceding discussion only tells how the mesh deals with changing domain boundaries. The interior nodes naturally also need to change their positions in time following these boundary changes and internal deformations (in the structural domain). This is organised via an extra set of mesh smoothing equations.

The ALE mesh adaptation only works if the node connectivities remain intact and grid cells do not turn inside out. This puts limits on what kind of squeezing and stretching is allowed, that is, on the (rate of) mesh deformation. One way to keep going when these limits are exceeded is to remesh the whole domain. This comes at a high cost when there are fine mesh regions, which there typically will be in the fluid domain. Of course, even without mesh deformations the mesh can contain substandard elements, e.g. tight cell angles or strongly elongated cells, so that the modeler needs to keep an eye on mesh quality before and during the simulation.

The free surface
Dynamic forces caused by the presence of water around a structure fall into two categories: waves and flow (sometimes called 'current'). These are two profoundly different classes of physical phenomena and therefore lead to drastically different modelling requirements. Simulation of (long-period) ocean waves or (e.g. ship-induced) short-period waves in time and space is already a computation-intensive task, so combining this with the simulation of flow effects in the same model at realistically high *Re* is nearly always avoided.

For simulation of the free surface near a structure there are several options. The ALE method is sometimes called an interface tracking method, as opposed to interface capturing methods.

An example of the latter is the volume of fluid method (VOF). It could be described as an Eulerian-Eulerian format, with a fixed Cartesian background grid and a special function that determines the density ratio of the two fluids inside each cell. This method can be applied for fluid density problems (fresh-salt) and free-surface problems (air-water). The package COMFLOW is an example of a VOF model (Veltman et al., 2007). It is used in hydraulic engineering to simulate local wave effects from which the heights of waves slamming into offshore structures such as wind turbines can be computed. See Wellens (2012) for background on VOF and numerical tests with COMFLOW. Other interface capturing methods are the 'level set' and the 'phase field' methods. Relatively recently other CFD approaches such as the smooth particle hydraulics (SPH) method and the lattice Boltzmann method (LBM) have also gained attention.

If high Froude numbers downstream of the gate, in particular hydraulic jumps, are excluded from the range of analysed conditions, it makes sense not to use interface capturing methods, because these are more difficult to combine with turbulent flow. Instead, the ALE mesh can be used for tracking mild flow-induced surface disruptions. The free surface is a special boundary in the sense that the pressure and tangential force ought to be zero there and that it corresponds to the outer edge of the material fluid particles for every time step (Zienkiewicz and Taylor, 2000). In essence the same interface conditions apply as listed at the start of this section, but without the influence of a solid.

It is decided to explore the ALE mesh in this study for modelling both foregoing phenomena: the moving gate and the free surface curvature behind the gate. Table 3.1 contains an overview of the two models.

Table 3.1. Characteristics of the two numerical physics-based models.

	chapter	time dependency	gate motion	turbulence modelling	free surface	validation data
model I	4	transient with steady pre-run	fixed	k-ε model	ALE mesh	Nago (1978, 1983)
model II	6	transient	1 d.o.f. (vertical translation)	k-ε model	rigid lid	experimental data gate motion (Chapter 5)