How To Simulate Quickly And Efficiently A Flow Over A Spillway?

L. Goffin[1] , S. Erpicum[1] , B.J. Dewals[1] , M. Pirotton[1] , P. Archambeau[1]
[1] University of Liege (ULg), Department ArGEnCo
Hydraulics in Environmental and Civil Engineering
Liege
BELGIUM
E-mail: l.goffin@ulg.ac.be

Abstract: *Flows over hydraulic structures, such as weirs or spillways, can be modelled using different techniques. New models such as SPH or PFEM are becoming more and more popular. These models are particle and/or meshless and consequently require a lot of computational power. Other methods such as VOF also require a lot of computational time (a few hours). In the frame of 2-D vertical flows, other techniques use much less computation time. For irrotationnal flows, solving the Laplace equation can be done very efficiently. The difficulty of this method lies in the definition of boundary conditions. The free-surface, which is naturally determined when using Lagrangian methods, needs a heavy iterative solving due to its non-linear nature when expressed in the frame of the Laplace equation. This paper will present an original technique that allows a quicker and easier determination of the free-surface. An irregular mesh for boundaries is used and discussed. The method is validated with analytical solutions and experimental measurements.*

Keywords: *Spillway flow, free surface, surface tracking, Laplace, boundary conditions, curvilinear coordinates*

1. INTRODUCTION

Many dam owners are re-evaluating the security level of their hydraulic structures which often use spillways and weirs as control structures. In order to evaluate efficiently their security level, experimental and numerical studies can be driven in parallel.

For the numerical part, many methods can be used. Meshless and/or particle methods such as SPH (Smoothed Particle Hydrodynamics) or PFEM (Particle Finite Element Method) are becoming more and more popular (Ferrari, 2010, Goffin et al., 2014, Larese et al., 2008). However they have a high computational cost and cannot be easily used nowadays in industrial applications. Moreover, their accuracy level is not as high as traditional methods due to boundary conditions they use. Other methods such as VOF or levelset combined with finite volumes (Detrembleur, 2011) can be applied. They still need more than an hour for a single simulation. This is why a light and efficient model is needed. It has been proven experimentally (Escande, 1937) that a flow over a spillway can be considered irrotationnal. Thus, the velocities derive from a potential and the Laplace equation can be solved: $\nabla^2\phi = 0$, where ϕ is the potential function and $\nabla^2 = \Delta$ the Laplace operator. This equation can be solved in a 2-D vertical finite difference grid. In order to fit as well as possible impervious borders and the free surface, irregular boundary conditions have been implemented. We present in this paper an original way to find the profile and position of the free surface for a given discharge and a given topography.

This paper is divided into three main parts. The first one is about the implementation of the code. We will first discuss the boundary conditions and how the free surface can be determined. Then, we will discuss briefly the opportunities to accelerate the execution of the code. The second part is about the validation of the code. Finally, we will show the adequacy between the numerical and experimental results of a spillway flow. This paper focuses on open channel spillways with 2-dimensional flow chracteristics.

2. IMPLEMENTATION

Spillway flow being irrotationnal, solving the Laplace equation over potential leads to a solution from which we can derive a velocity field:

$$u = \frac{\partial \phi}{\partial x} \quad ; \quad v = \frac{\partial \phi}{\partial y} \tag{1}$$

From this velocity field and a reference energy level E, we can compute pressures p over the entire domain:

$$E = y + \frac{p}{\rho g} + \frac{\left\|\vec{u}\right\|^2}{2g} \tag{2}$$

where y is the elevation and ρ the fluid density.

In this paper, the Laplace equation is solved using the finite difference method for discretizing the Laplace operator. The mesh in the inner domain is regular while it is irregular near boundaries. The discretized form of the Laplace equation can be written as

$$\nabla^2 \phi \approx \frac{\frac{\phi_{i+1,j}-\phi_{i,j}}{a\Delta x} - \frac{\phi_{i,j}-\phi_{i-1,j}}{c\Delta x}}{(a+c)\Delta x} + \frac{\frac{\phi_{i,j+1}-\phi_{i,j}}{b\Delta y} - \frac{\phi_{i,j}-\phi_{i,j-1}}{d\Delta y}}{(b+d)\Delta y} \tag{3}$$

Equation (3) leads to a second order of precision when the mesh is regular ($a = c$ and $b = d$) but a first order next to irregular boundaries.

2.1. Boundary Conditions

In the case of a free surface flow over a hydraulic structure, four kinds of boundary conditions (BC) are required: inflow BC, Dirichlet BC (imposition of a potential), free surface BC and impervious BC. Impervious and inflow BC are von Neumann conditions, i.e. $\vec{u} = \partial \phi / \partial \vec{n}$ is imposed.

2.1.1. Irregular Mesh at Boundaries

The calculation nodes are created according to a grid size and a vector contour. Boundary nodes are added on the intersections between the grid and the contour while regular nodes are added regularly on the grid (see Figure 1). This method fits curved boundaries with good geometrical accuracy. However, it is difficult to compute a classical finite difference derivative at these points.

From Green-Gauss theorem, the velocity on a control surface $\vec{u}_m$ can be written in 2-D as

$$\vec{u}_m = \frac{1}{S}\iint_S \nabla \phi \, \mathrm{d}S = \frac{1}{S}\oint_c \phi \, \vec{n} \, \mathrm{d}c \tag{4}$$

where c is the contour of a surface S and $\vec{n}$ the normal vector of contour c. For N nodes, Eq. (4) can be discretized as:

$$\vec{u}_m = \frac{\sum_i^N \left(\phi_i + \phi_{i+1}\right) \cdot \begin{bmatrix} y_{i+1} - y_i \\ x_i - x_{i+1} \end{bmatrix}}{\sum_i^N \left(x_i y_{i-1} - x_i y_{i+1}\right)} \tag{5}$$

When Δx and Δy tends to 0, the velocity on the control surface tends to the velocity of Green-Gauss nodes. For this reason, we used equation (5) to express the velocity at a boundary node (see Figure 1).

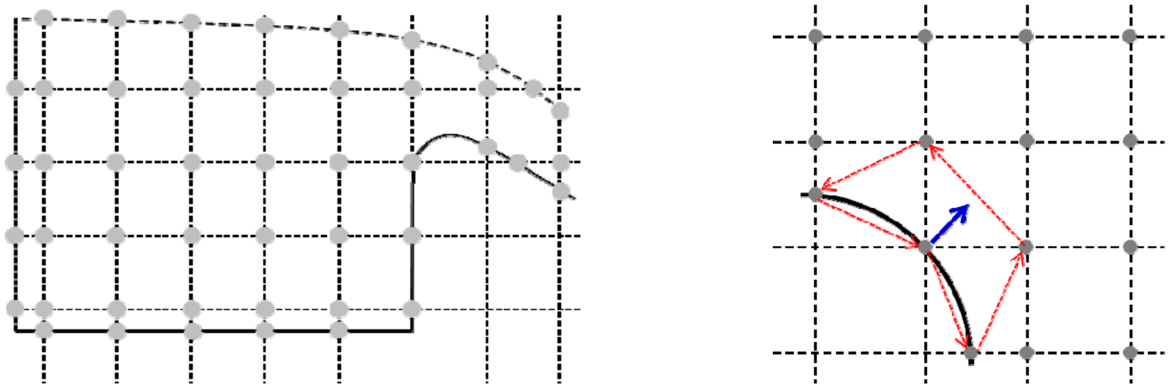

Figure 1 – Schematic view of regular and irregular zones in the mesh (left) and Green-Gauss principle (right)

2.1.2. Free Surface Boundary Condition

Some authors such as (Assy, 2001) or (Dias and Vanden-Broeck, 2011) use a nonlinear condition based on energy consideration for the free surface. Adding a nonlinear condition to a linear system leads to an iterative procedure. In order to avoid this, we define the free surface as an impervious border at first. Then, according to the pressure distribution, the free surface is moved in order to get closer to a uniform pressure distribution.

For trans-critical flows, the critical section minimizes the energy level for a given discharge. Thus, if an arbitrary energy level E is imposed, the pressure p_0 calculated at the critical section is the target pressure that must be reached on all free surface nodes.

Determining the critical section is the key of this problem. To do so, we have used the curvilinear theory exposed in (Stilmant et al., 2013). ξ and η axis are defined in the frame of spillways in Figure 2. The pressure in the domain can be written as

$$\frac{p(\eta)}{\rho g} = E - y_r - \eta \cos\theta - \frac{U^2}{2g} \tag{6}$$

where y_r is the vertical position of the reference curve and U the velocity profile along the η axis (linear section). The velocity profile can be deducted from the irrotationnal assumption:

$$U = \frac{\kappa q}{(1-\kappa\eta)\ln\dfrac{1+\alpha\kappa H}{1-(1-\alpha)\kappa H}} \tag{7}$$

with q the specific discharge, κ the curvature of the reference curve, H the length of the η axis between the free surface and the bottom and α the position of the reference curve (0: ξ is the upper boundary; 1: ξ is the lower boundary). At the free surface, Eq. (6) becomes:

$$\left.\frac{p}{\rho g}\right|_{\eta=\alpha H} = E - y_b - H\cos\theta - \frac{{U_s}^2}{2g} \quad \text{where } U_s = \frac{\kappa q}{(1-(1-\alpha)\kappa H)\ln\dfrac{1+\alpha\kappa H}{1-(1-\alpha)\kappa H}} \tag{8}$$

with y_b the altitude of the bottom. Then, the derivative of the pressure term according to the depth H is given by:

$$\frac{d\left(\dfrac{p}{\rho g}\right)_{\eta=\alpha H}}{dH} = -\cos\theta - \frac{U_s}{g}\frac{dU_s}{dH} = -\cos\theta - \frac{U_s^3}{gq}\left((1-\alpha)\ln\beta - \frac{1}{1+\alpha\kappa H}\right) \quad \text{where } \beta = \frac{1+\alpha\kappa H}{1-(1-\alpha)\kappa H} \tag{9}$$

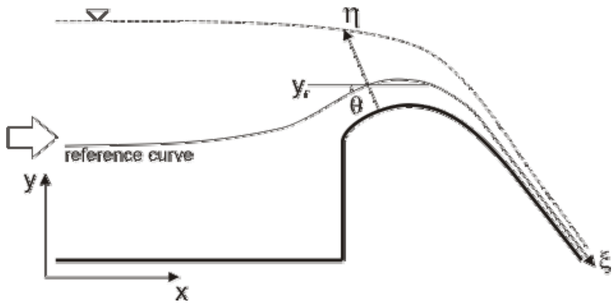

Figure 2 – Curvilinear coordinates for a spillway

Thanks to Eq. (9), it is now possible to determine the position of the critical section. When the sign of the derivative in (9) is positive, the flow regime is super-critical but when it is negative, the flow regime is sub-critical. The change of sign indicates the position of the critical section. Thanks to that information we are now able to determine the displacement ΔH that can be applied normally to the free-surface for every node:

$$\Delta H = \gamma \frac{p_{i+1} - p_i}{\dfrac{d\left(\dfrac{p}{\rho\, g}\right)_{\eta=\alpha H}}{dH}} \tag{10}$$

where γ is a relaxation coefficient that avoids instabilities, p_i the pressure at iteration i and $p_{i+1} = p_0$ the target pressure.

Since the derivative at denominator in Eq. (10) is close to 0 near the critical section, some instability may appear in that region. This problem can be solved by smoothing the free surface.

2.2. Efficient Solving

Solving the Laplace equation requires to solve a linear system $A{\cdot}x = b$. Since this task has a high computational cost, it is important to use efficient techniques. We have used two different solvers: (a) the PARDISO algorithm (Schenk and Gärtner, 2004) which is a parallel direct solver and (b) the GMRES algorithm (Saad and Schultz, 1986) which is an iterative solver that has been parallelized where possible. For better performances, the GMRES solver requires a first good approximate solution. Since the computational domain evolves slowly, the next solution is close to the current one. Then, the PARDISO solver can be used for the first iteration and the GMRES for all the following ones.

For a more efficient solving, the matrix *A*, which is sparse, should be organized with the narrowest bandwidth. To do so, the nodes should be numbered according to their neighborhood. The reverse Cuthill-McKee algorithm (RCM) is used in our code (Cuthill and McKee, 1969, George and Liu, 1981).

We implemented the code in Fortran, using an object-oriented paradigm. When the free-surface moves, the irregular nodes on the boundary change and the number of regular nodes may also change. In order to avoid expensive deallocation and reallocation processes (due to the object-oriented paradigm), a sufficiently large number of regular nodes is allocated at the beginning of the simulation and irregular nodes are not deallocated at every step but stored in a garbage list in order to be used later.

These measures allow reaching a mean computation time of approximately 2 s per iteration for 200,000 computed nodes (CPU: Intel i7-3770K, 3.5 GHz, compiler: ifort). The total computation time depends on the level of accuracy required, but good results can be obtained after less than 5 minutes for approximately 200,000 computed nodes.

3. VALIDATION

The code is validated thanks to three test cases of flow over a bump. The two first test cases are (a) subcritical and (b) supercritical. Since these flows do not present a critical section, the target pressure has been set arbitrarily to 0 m. The equation of the bump is $y = 0.1\,(x-5)^2 + 0.1$ for $x \in [4;6]$. The third test case (c) represents a transcritical flow. The target pressure is now determined by the critical section. The equation of the bump is $y = 1.5\,(x-5)^2 + 0.31375$ for $x \in [4.55;5.45]$. For each test case, an analytical solution based on Bernoulli principle is plotted. It makes the assumption that the pressure distribution is hydrostatic.

For the subcritical flow (Figure 3) a horizontal free surface is set as first guess (q=0.3 m²/s, head=0.45 m). Due to the presence of the bump, the free surface bends and finally corresponds to the analytical solution. The pressure is equal to 0 uniformly on the free surface. For the supercritical flow (Figure 3), the discharge is set to 0.3 m²/s and the head=0.44 m. The numerical solution fits well the analytical one. The pressure distribution on the free surface is uniform and equal to the target pressure. Test cases (a) and (b) validate the code for uniform regimes. The method is able to move the free surface in order to reach a uniform pressure distribution as required.

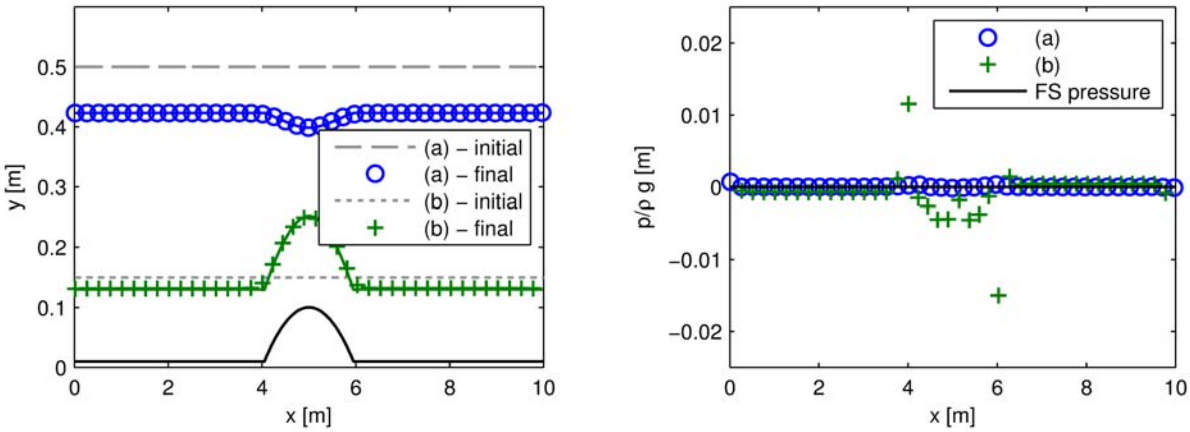

Figure 3 – Test cases (a and b): subcritical (a) and supercritical (b) flows over a parabolic bump (plain lines are for analytical solutions)

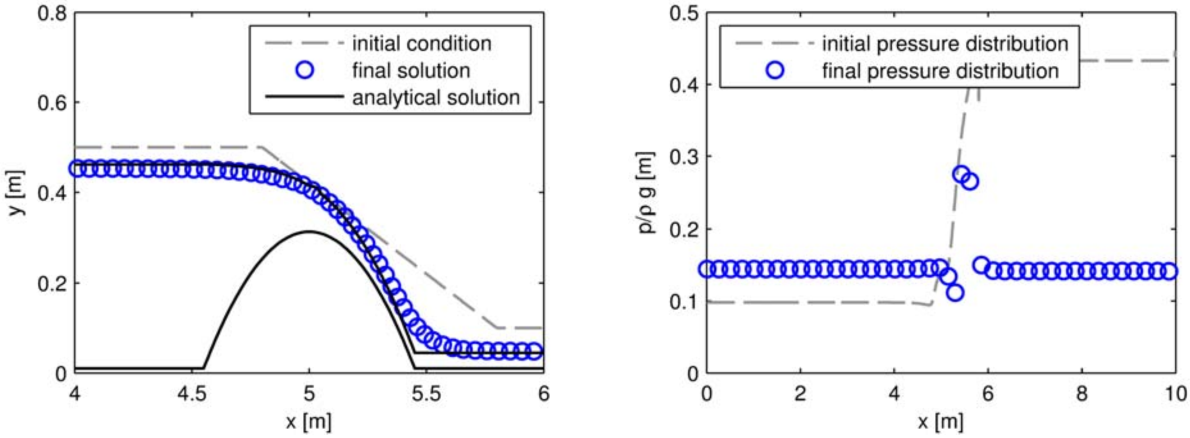

Figure 4 – Test case (c): transcritical flow over a parabolic bump

The last test case (Figure 4) used a first free surface that presents a slope in the middle on the simulation domain (q=0.1m²/s). The numerical solution is in good agreement with the analytical solution except at the downstream part of the bump. Due to the sudden change in the slope, the pressure distribution around that section cannot be hydrostatic. Since the analytical solution makes this assumption, the results cannot be compared around that section. Moreover, smoothing the free surface avoids deforming the surface too sharply. We notice also that the numerical results are a little bit shifted in comparison to the analytical solution. This is due to the fact the pressure oscillates around the critical section, which leads to a poorer pressure target.

4. APPLICATION TO SPILLWAYS

Flow over spillways can be modelled thanks to our code since the flow can be considered irrotationnal (Escande, 1937) and a critical section allows to determine the target pressure. In this section, we compare experimental measures to numerical results of a flow over a standard USACE WES (US Army Corps of Engineers Waterways Experiment Station) spillway (H_d = 0.15 m) (USACE, 1987) profile for 2 discharges: 0.1275m²/s (Figure 5 left) and 1.7632m²/s (Figure 5 right). Velocity profiles and pressures on the spillway are given in Figure 6.

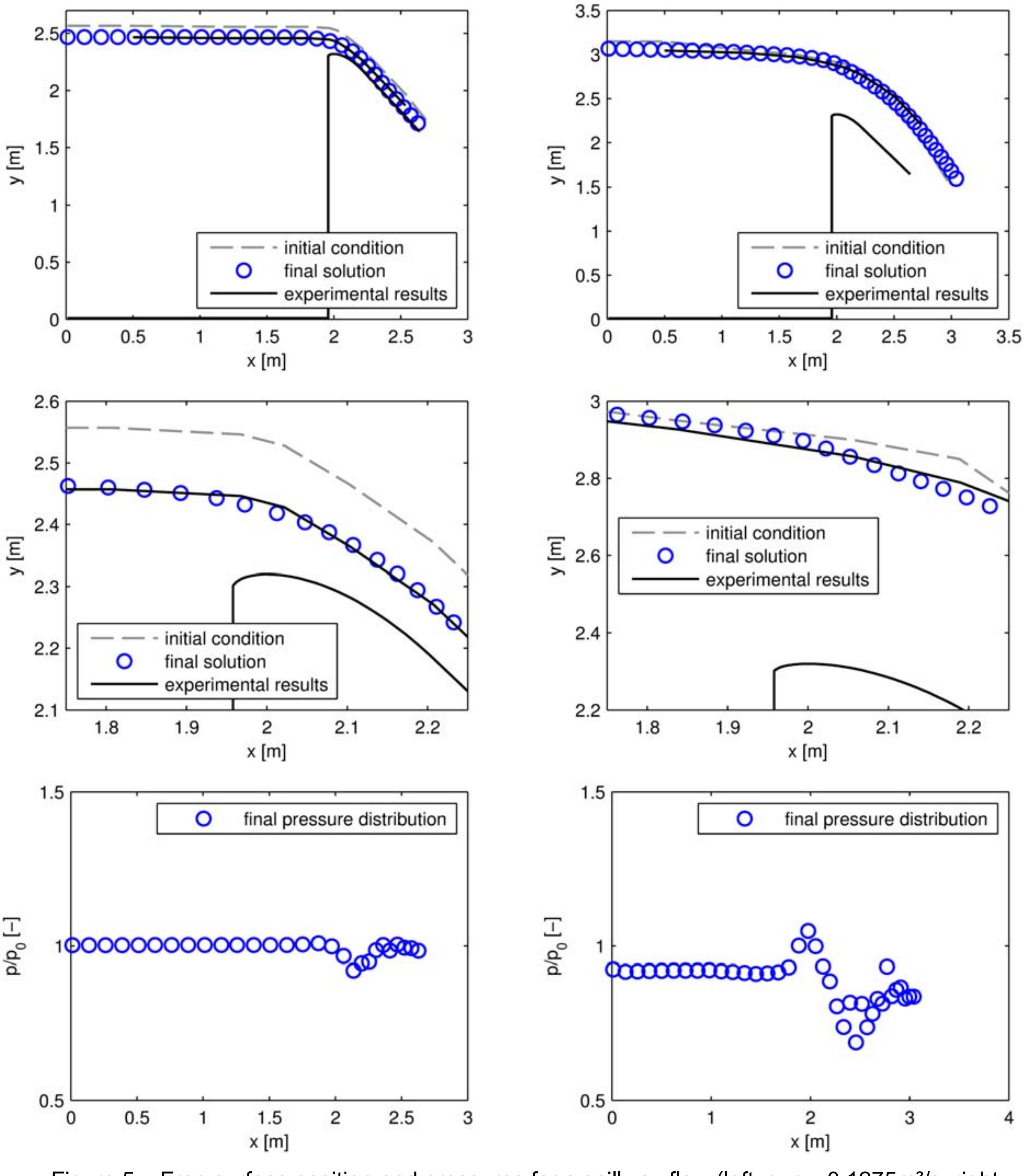

Figure 5 – Free surface position and pressures for a spillway flow (left: q=q_1=0.1275m²/s; right: q=q_2=1.7632m²/s)

For both discharges, the free surface computed numerically fits qualitatively well experimental measures. For q=1.7632m²/s, some imprecisions can be noticed around the crest. The pressure distribution on the free surface is also better for the lowest discharge. Better results could be reached for a lower smoothing length on the free surface (used here: 0.2m). However, releasing this parameter leads to oscillations. For the first discharge, we obtain numerically a head $H=1.01H_d$ and a discharge coefficient $C_d=0.506$ (experimentally: $H=1.01H_d$ and $C_d=0.505$). For the second discharge, we obtain numerically $H=4.97H_d$ and $C_d=0.618$ (experimentally: $H=5H_d$ and $C_d=0.615$).

Even if the free surface position is not determined very precisely, horizontal velocities and pressures are very close to experimental measures, which were made using LS-PIV (Large Scale Particle Image Velocimetry) (Peltier et al., 2014). The shape of the velocity profile is due to the irrotational nature of the flow, leading to a high increase of speed near the bottom boundary. The numerical velocity profiles match experimental measures both in shape and in amplitude.

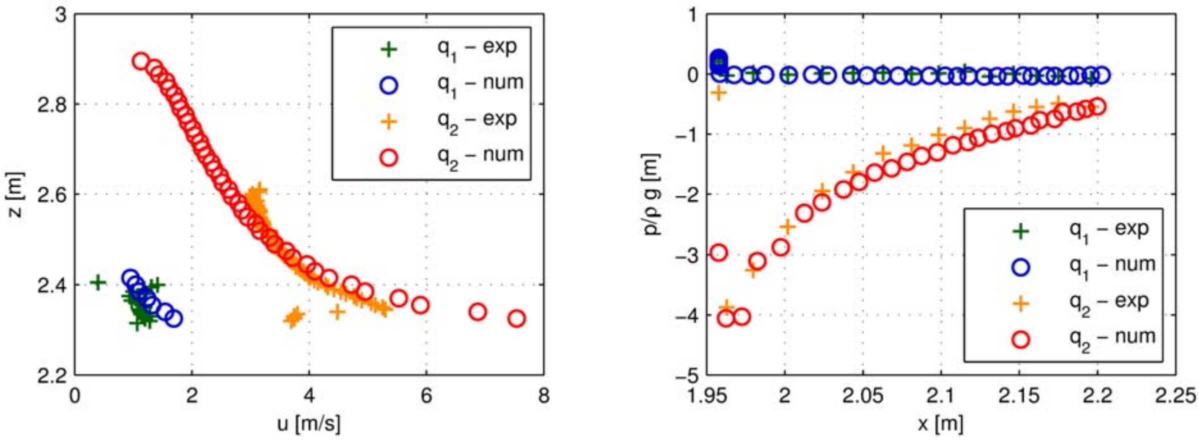

Figure 6 – Horizontal velocity profile at crest (left) and pressures on the bottom (right)

5. CONCLUSION

Computing a flow over a spillway can be very time consuming depending on the method that is used. We have proposed in this paper a technique that is able to deal with this kind of flow very efficiently. Indeed, for a problem containing approximately 200,000 computed nodes and with a first good approximation of the free surface, interesting results can be obtained after only 50 iterations, i.e. less than 2 minutes.

The code implemented is able to determine the free surface of a transcritical flow. For one-regime flows, the pressure target and the energy level must be set by the user. For transcritical flows, only the discharge and an arbitrary constant are needed. It was shown that the method implemented to iterate on the free surface is relevant and gives qualitatively good results. Solving the Laplace equation allows to compute velocity profiles and pressure distributions that correspond to experimental measures.

Concerning prospects, releasing the smoothing parameter for the free surface should lead to some improvement. To do so, a better treatment near the critical section should be found. Using curved sections for determining the pressure derivative is currently under investigation. Studying the numerical stability and computation time regarding the initial condition should also be done.

This code can be applied to industry or research area for the determination of new spillway profiles that maximizes the discharge for a high upstream head, avoids large negative pressures, etc.

6. ACKNOWLEGDMENTS

The experimental measures were acquired thanks to the collaboration with EDF-CIH.

7. REFERENCES

ASSY, T. M. 2001. Solution for spillway flow by finite difference method. *Journal of Hydraulic Research,* 39**,** 241-247.

CUTHILL, E. & MCKEE, J. 1969. Reducing the bandwidth of sparse symmetric matrices. *ACM '69 Proceedings of the 1969 24th national conference***,** 157-172.

DETREMBLEUR, S. 2011. *Modèles d'écoulements incompressibles en plan vertical en plan vertical appliqués aux structures du génie civil.* Univeristy of Liege (ULg).

DIAS, F. & VANDEN-BROECK, J. M. 2011. Potential-flow studies of steady two-dimensional jets, waterfalls, weirs and sprays. *Journal of Engineering Mathematics,* 70**,** 165-174.

ESCANDE, L. 1937. *Barrages*, Hermann & Cie.

FERRARI, A. 2010. SPH simulation of free surface flow over a sharp-crested weir. *Advances in Water Resources,* 33**,** 270-276.

GEORGE, A. & LIU, J. 1981. *Computer solution of large sparse positive definite systems*, Prentice-Hall.

GOFFIN, L., ERPICUM, S., DEWALS, B., PIROTTON, M. & ARCHAMBEAU, P. 2014. Validation of a SPH model for free surface flows. *Simhydro.* Nice, France.

LARESE, A., ROSSI, R., OÑATE, E. & IDELSOHN, S. R. 2008. Validation of the particle finite element method (PFEM) for simulation of free surface flows. *Engineering Computations (Swansea, Wales),* 25**,** 385-425.

PELTIER, Y., DEWALS, B., ERPICUM, S., PIROTTON, M. & ARCHAMBEAU, P. 2014. MAT_LSPIV 4.0: User Manual.

SAAD, Y. & SCHULTZ, M. H. 1986. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing,* 7**,** 856-869.

SCHENK, O. & ĞARTNER, K. 2004. Solving unsymmetric sparse systems of linear equations with PARDISO. *Future Generation Computer Systems,* 20**,** 475-487.

STILMANT, F., EGAN, R., ARCHAMBEAU, P., DEWALS, B., ERPICUM, S. & PIROTTON, M. Depth-averaged flow modeling in curvilinear coordinates. 2013 IAHR World Congress, 2013.

USACE 1987. Hydraulic design criteria. Waterways,Experimental Station, Vicksburg, Miss.