

# Unified Visual Post Processing of Mesh and Meshless Data for Hydrodynamic Simulations

Yun Jang<sup>1</sup>, John Biddiscombe<sup>1</sup>, Jean-Christophe Marongiu<sup>2</sup>, and Etienne Parkinson<sup>3</sup>,

<sup>1</sup>Swiss National Supercomputing Centre, Manno, Switzerland

<sup>2</sup>Ecole Centrale de Lyon, France

<sup>3</sup>Andritz VA Tech Hydro, Vevey, Switzerland

---

## Abstract

*CFD researchers have a wide variety of interactive post-processing tools at their disposal for visualization and quantitative analysis, but these tools are generally designed to operate only on mesh based data. Recent developments in CFD have led to a growing field of particle based methods, in particular, Smoothed Particle Hydrodynamics (SPH), for which no such tools exist. To address this problem we present a framework for consistent visual and interactive analysis of both particle and mesh based data within a unified architecture. The goal of the framework is to enable quantitative visualization and analysis of either data type in an identical manner so that the user can use a single approach to post-processing. This unification makes the process of comparison between results from conventional CFD and experimental SPH directly possible, by making use of custom probes and filters which behave the same in either environment. The meshless nature of SPH data within a CFD environment also makes identification of distinct boundary surfaces more difficult as models generated for SPH simulations are not as well structured as those for conventional CFD, due to the lack of commercial support for these models. We therefore introduce the concept of probe projections onto surfaces to allow the selective positioning of regions of interest on boundaries for the purpose of integration of parameters and visualization of results.*

Categories and Subject Descriptors (according to ACM CCS): Computer Graphics [I.3.4]: Application packages—, Visualization

---

## 1. Introduction

Smoothed Particle Hydrodynamics (SPH) is a mesh-free method for simulation which was developed originally in the field of Astrophysics, however, it has recently been applied to areas as diverse as solid mechanics, molecular dynamics, biomechanics and fluid dynamics. Within the field of CFD, SPH has the potential to become widely adopted in the next generation of simulation tools for a range of applications, to a large extent because it does away with the need for volumetric meshes, which are time consuming to create, and are often difficult to manage and adapt to flow dynamics. SPH offers the ability to handle complex solid shapes using boundary representations alone, and also the capacity to handle mixed-fluids and fluid-structure interactions within a unified model. Analysis and visualization of SPH data is, however, challenging since the mesh-free nature of the data means that there is no connectivity information between data

points, and therefore no direct means of displaying a continuum representation of the computed fields.

Particle based simulations are still in their relative infancy with little significant support from commercial tools for either the generation of input data, or the post processing of results. The extraction of parameters from the data is generally left to individual researchers working with scripts or tools developed for a particular model. These customized tools are rarely portable to another problem domain or model implementation, resulting in a lack of standard practices and tools. Moreover, researchers wishing to compare the results of a new particle simulation against an equivalent grid based CFD result may not be able to obtain the same information from the conventional CFD data using the tools they have developed for their SPH models, this increases the amount of time and effort required to validate the model.

In this paper we describe our post processing *workflow*

and implementation of a toolkit developed for the purpose of visual and quantitative analysis of particle based data, along with direct comparison of results with their equivalents from more conventional mesh based CFD simulations. The primary goals of the material presented here are therefore

- Unify the process of extracting data from particle or mesh based results with a single set of probing and analysis tools, providing a consistent interface and architecture to the user.
- Provide enough flexibility in the design to allow experimentation, adaptation and customization of the tools to fit new simulation results as they become available.
- Ensure that accurate measurements of parameters are produced regardless of the data type supplied.
- Provide a means for the user to apply many existing mesh based visualizations to their particle data.

To make these goals possible we base our tools upon the flexible and widely used Visualization ToolKit (VTK) [Kit03], exposing specific modules as plugins for the ParaView visualization package [Hen05] which is used as the front end user environment. It is not possible to reimplement every possible visualization algorithm (such as contouring, streamlines generation, feature detection, flow structures etc) for particles within a reasonable timescale and in the same environment, so we focus instead on transforming particle data to a continuum representation by interpolation at the point of demand so that existing tools may be used.

In the following section we review the background and related work, then in Section 3 discuss our architecture and the specific types of probe and their usage including introduction to projection and animation of probes. Then we present our quantitative analysis in Section 4 and finally the validation and results of quantitative analysis on specific models in 5 followed by the conclusion of our work at the end.

## 2. Related Work

The vast majority of flow simulations use a Eulerian approach and many types of grid have been developed in order to obtain accurate simulation results. Similarly, many different flow visualization techniques have been developed for the appropriate display of results. Early visualizations used glyphs, such as arrows, to represent vector fields in the data. Recently more attractive techniques, such as line integral convolution (LIC) [CL93] and texture based approaches [LHD\*04], have been studied to provide more realistic flow depictions. Additionally, feature based visualization has been studied and the features extracted are directly used to analyze flow data, examples of accurate vortex core feature detection algorithms are presented in [JH95, LDS90, PVH\*03].

Recently particle based simulations, which are mesh free methods, has been introduced - SPH is one of the most popular mesh free approaches [Mon88] and it follows a La-

grangian formula in its implementation. Due to the relative simplicity of the equations, SPH has become popular in computer graphics, especially free surface of flow animations [MCG03]. In the visualization community, many particle based techniques have been presented, producing visually stunning images of particle based data. Most of these techniques focus on extracting isosurfaces of particle datasets [BDR98, MNKW07, RRL07, ZK06]. For the rendering of particles, Gribble et al. [GIK\*07] presents ray casting of particles and Ellsworth et al. [EGM04] presents rendering of terascale particles from curvilinear data by minimizing seeks. As an interactive exploration method, Co et al. [CFG\*05] present visualization of high dimensional data using scattered plots. Krüger et al. [KKKW05] shows a particle system for 3D flows and they present steady 3D flow field on uniform grids using the GPU and store particles on the GPU for interactive exploration. Zhou and Garland [ZG06] present particle based order-independent point rendering from a very large volume of tetrahedral meshes.

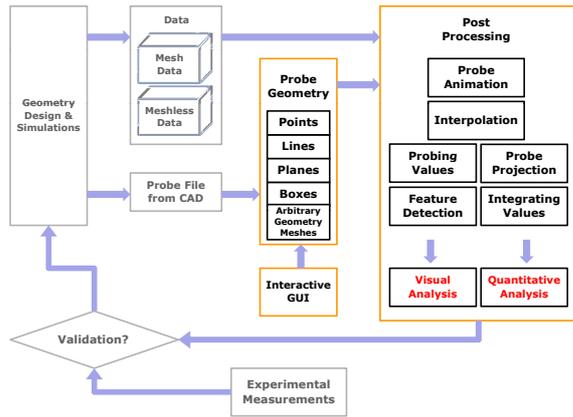
In astrophysics applications, Walker et al. [WKM05] and Navrátil et al. [NJB07] present visualization of SPH data from astrophysics simulations. Walker et al. show visualization of particles and contour plots on 2D slices. This work conveys more localized information using the specific slices. However, the work is in progress, therefore, many improvements are needed in terms of interpolation and rendering. Navrátil et al. present visualization of particle data using interpolation on a regular grid. They determine grid resolution automatically to capture sufficient information from the particle data and interpolate data with fixed number of particles to capture more local information.

Since there is no connectivity in particle data, interpolation is needed at a certain point which is not located in the particle data. Shepard [She68] introduces an interpolation function using weighted average with inverse distance. Later for scattered volumetric data, Nielson [Nie93] introduces several different interpolation methods and compare the methods. As an improvement of Shepard interpolation methods, Brodlie et al. [BAU05] introduces a modification with constraint in order to preserve positivity of data.

Most of the material presented within the visualization community has focused on the generation of pleasing images, at high frame rates. We are interested in extraction of parameters for visualization from simulations which are physically based and therefore require closer attention to the quantitative display and comparison of models, with both particles and geometries.

## 3. Probing Architecture and Pipeline

We aim to make it possible for users to extract information from their particle or mesh based data using a single set of tools. These tools are centered around the concept of probing, which is simply a means of providing a display of data at specific well defined locations.



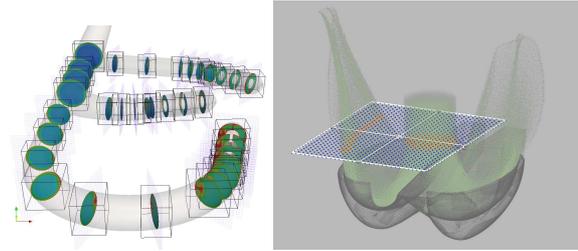
**Figure 1:** Overview of our system. Orange boxes indicate our post processing and the visual and quantitative analyses are the results of the post processing.

### 3.1. Probe Motivation

Probes can be any type of dataset, and can come from any source, but the main probe types are motivated by the needs of engineers using CFD simulations in a design process, improving local and global flow patterns, and of researchers developing numerical methods and comparing their results to reference data.

Numerical flow simulation are usually compared to experimental measurements, either in the development phase or in the production phase, when the numerical tool is actively used to define efficient designs. Experimental sensors sample field values at given locations, sometime arranged along a regular lattice (lines and/or arrays). Other experimental methods investigate the flow field on specific sections, like Laser Doppler Velocimetry. On the other hand during the development process, it is also necessary to benchmark the model against some ideal flow pattern or known test cases, where an analytical solution can be derived. Analysis of these flows is usually made at specific regions of the domain like boundary surfaces or control sections. Consequently numerical results should be accessed and analyzed at the same locations. It is therefore essential to provide a set of tools to extract data of interest from the numerical data set. We make use of point/line based probes for comparisons with measurements data as well as surface probes for integrations of parameters and comparisons with force measurements.

3D box probes are not so easily compared with either experimental or analytic results, but in an interactive environment, they allow the user to dynamically select regions of interest in the data which are selected for further analysis. Feature detection algorithms such as vortex core extraction, or flow topology visualization can be very slow to run on large data and we naturally wish to limit the amount of interpolation and data crunching that needs to be performed.



**Figure 2:** Left image shows an example of multiple plane probes generated by the CAD system for a water distributor. Right image presents our interactive plane GUI for the probe on Static Bucket case.

### 3.2. System Architecture

Within our toolkit, probe geometry can be accepted from custom probe files, by user interaction with on-screen widgets, or supplied by arbitrary geometry either loaded from file, or generated within the system. The probe itself is simply a collection of points at which we wish to interpolate or display our data. Our data-flow is outlined in Figure 1 and is centered around the interpolator which accepts geometry as input and outputs (the same, or new derived) geometry with probed values attached. The output of the probe interpolator can be visualized directly, passed into other visualization (or feature extraction) algorithms, or into our quantitative modules for detection/extraction of measured quantities.

### 3.3. Meshless and Meshed Coexistence

Interpolation of point based data is discussed in 4.1 but the data that is provided may in fact be from a standard mesh based simulation. To enable our software to automatically switch into a cell based interpolation mode, we tag all simple probes generated within the system with meta data describing the geometry. The meta data takes the form of implicit function descriptions of point, line, plane, box or sphere. These descriptions make it possible for the core probe module to correctly generate precise data slices such as those shown in Figure 2 (left). When the probe module receives a mesh based input, it must try to use cell based interpolations rather than nearest neighbor based ones as it would for particle data. When a 2D rectangular probe is supplied it contains a physical 2D plane made up of  $M \times N$  samples, and also an implicit function description of a plane. The plane also has a physical size along the  $x, y$  axes and so an implicit box function is also attached to the data. The meshed based probe code automatically clips the data to the box and slices it with a plane, using the standard clip and slice modules available in VTK/ParaView. This ensures that the point values on the generated data are exactly computed from the cells present in the original data. In a similar manner, discs can be generated by combining a plane and a sphere. We anticipate expansion of this technique to handle other implicit shapes in future. When arbitrary geometry is supplied

as a probing input for a mesh based dataset, the code cannot make use of implicit functions and resorts to the standard cell interpolation routines provided within VTK which use parametric coordinates for linear in cell interpolation.

It is worth noting that the example of Figure 2 uses a large number of probes all at the same time - our system treats collections of probes as multi-block datasets and automatically iterates over them when interpolating/integrating. This is one area where CAD generated probes have a significant advantage over GUI generated ones which are usually handled one by one (though they can be manually grouped together).

### 3.4. Limitation of Probes from CAD

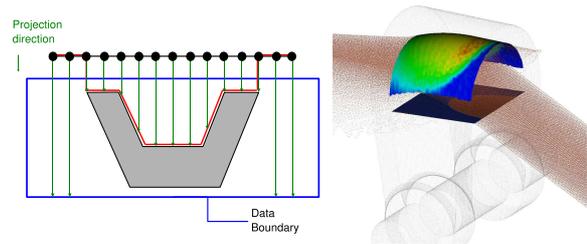
Although probe files generated directly by the CAD system are optimal from the point of view of design, there are circumstances which limit their applicability.

SPH models are in general less accurate than their conventional counterparts, and since they are still under development, we find that certain parts of a simulation may be erroneous. Problems particularly occur at sharp boundaries of geometric models where the SPH predictions of pressure (for example) may be wildly inaccurate. The experimenter may be aware of this limitation and want to perform an integration over 'most' of the surface, whilst leaving small border on one edge. To do this from the CAD system may involve considerable unwanted labor, particularly when the model is being continuously refined and the bad regions may be shifting. In this example, a probe file generated by hand, or interactively by manipulating a widget on screen is more desirable. The right image in Figure 2 shows an example of our interactive widget to probe the data that users wish to explore.

In a similar manner, it is possible to save some complex geometric shape which defines a boundary of the CAD model in a polygonal form and use this polygonal representation as a probe. However, once this has been done, it is not always easy to remove pieces from it or otherwise modify it. We make use of complex geometric probes for visualization of fields on surfaces, and integration of parameters over them when the model is well behaved.

### 3.5. Probe Projection

The limitations of Probes generated from CAD files makes it worthwhile to consider the projection of probes onto surfaces. Projecting a probe onto a surface is akin to shadow casting and it is possible to align a probe in such a way that it projects only onto a very specific part of a model, leaving unwanted regions untouched. The probe projection process is simple and illustrated by the left image of Figure 3. A ray intersection test is performed which uses the boundary particle sizes to identify those that lie on the surface. Once surface particles have been identified, they can be triangulated



**Figure 3:** A simple illustration of probe projection onto a surface (Left) High resolution probe geometry is projected onto a boundary to extract a subsection of the surface. Right, a projection of a rectangular probe patch onto the inner surface boundary particles of a water jet deflector.

to form a polygonal surface which is suitable for integration of parameters (i.e. having well defined surface areas for each patch). For mesh based geometry the surface cells intersecting the shadow region are extracted directly.

When geometry is generated from CAD systems for mesh based CFD simulations is usually possible to label surfaces, color them separately, manipulate boundary conditions and perform a myriad of other operations on them. Sadly, discretization software to convert these complex geometries into SPH compatible boundary particles is not yet mature, and all of the additional information is lost. For this reason, probe projections are also useful because individual surfaces which can be trivially identified using standard CFD tools are not available. Finally, we note that many SPH codes use boundaries generated by hard coded loops dumping particle lists, with no information about specific surfaces at all.

An example of surface projection is shown on the right image in Figure 3. The fluid free surface hits the inner solid surface and this inner surface has important physical properties such as force transition which we visualize and use for integration of torque.

Instead of projecting probes onto surfaces, one might consider clicking a point on the surface and running a local surface reconstruction algorithm over the points radiating outwards. We choose not to do this as we would like the reconstruction to stop at edges and well defined boundaries which are easier to implement using the simple probe interface. The further generation of surfaces will be explored as a future possibility.

### 3.6. Probe Animation

Many hydrodynamic simulations make use of rotating or moving domains which need to be tracked if some (surface or mid flow) parameter is to be measured. Our toolkit contains a Probe animation module which takes time from the simulation data and uses it to translate/rotate/scale the probe geometry by a user defined amount per time unit. The time unit may be the actual simulation time or a simple timestep

extracted from the data stream. We make use of VTK's temporal meta data information to extract exact simulation time (see [BGM\*07]) to ensure that the probe follows the geometry precisely. Probe animation may also be connected to probe projection, allowing complex feature following using a simple interface.

#### 4. Quantitative Analysis

The visualization is used to understand relative behavior on the probes. However, the visual analysis has limitation to compare the analysis with integrated quantities. Discharge in water flow, for example, is the volume of water transported in a certain amount of time [PSLR72]. The amount of water can be visualized but actual integrated amount cannot be compared in the visualization. Therefore, quantitative analysis is necessary for more accurate analysis and the value integration on the probes is introduced in the following.

##### 4.1. Interpolations

The values at sample points on a probe are calculated by interpolation with  $N$  nearest neighbors around the sampled points for the meshless data. Neighbors are searched using spatial subdivision with a regular or adaptive subdivision. Once the  $N$  neighbors are found, the values at the sampled points are interpolated with the neighbors. For particle data, we provide two types of interpolation method. One is the Shepard method [She68] with inverse distances between probe points and neighbors and the other is the Shepard method using kernels for the weights. The kernel based method requires the specific kernel functions used in the simulation and it is easy new ones as the need arises. The inverse distance based method can be used for any meshless data when we the types of kernels used in the original simulation are unknown. The interpolation equation with  $N$  neighbors is as the following.

$$F(x, y, z) = \sum_{i=1}^N w_i(x, y, z) \cdot f_i \quad (1)$$

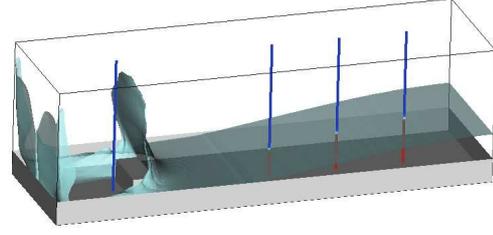
where  $f_i$  is a value of  $i^{th}$  neighbor and the weight  $w_i$  is calculated as

$$w_i(x, y, z) = \frac{v_i \times \sigma_i(x, y, z)}{\sum_{j=1}^N v_j \times \sigma_j(x, y, z)} \quad (2)$$

where  $\sigma_i(x, y, z) = 1/\|(x, y, z) - (x_{f_i}, y_{f_i}, z_{f_i})\|^2$  or  $\sigma_i(x, y, z) = \text{Kernel\_Function}(\|(x, y, z) - (x_{f_i}, y_{f_i}, z_{f_i})\|)$  and  $v_i$  is the  $i^{th}$  particle volume. For mesh based data, cell parametric coordinates are computed and used to weight the contributions of the cell nodes using a linear interpolation inside the cell.

##### 4.2. Flow Quantity Integrations

Any surfaces meshes with well defined 2D cells can be used for quantity integration: Probes (projected, animated or sim-



**Figure 4:** This image shows use of the line probe. Mesh based simulation of the dam breaking case is visualized and the line probes are used to measure water height at 4 specific positions. Red color indicates that the line segments are in the water and blue colored segments are in the air. Water surface is visualized with transparent blue colored surface.

ple), arbitrary surfaces extracted from the data, supplied by the CAD system, or generated by other means are suitable for surface integration, providing the cells are supported by VTK's underlying data structures.

$$I = \int_{Area} (\text{Value\_Integrated}) \vec{C} \cdot \vec{N} ds \quad (3)$$

with velocity ( $\vec{C}$ ). Note that  $\vec{N}$  indicates a normal vector of a probe. The *Value\_Integrated* includes discharge ( $d = \rho$ ), kinetic energy ( $e_k = \frac{1}{2} \rho \vec{C} \cdot \vec{C}$ ), total energy ( $e_t = e_k + p$ ) with pressure ( $p$ ), kinetic momentum ( $\vec{m}_k = \rho \vec{r} \times \vec{C}$ ) with the origin,  $\vec{O}$ , the location of point,  $\vec{x}$ , and  $\vec{r} = \vec{x} - \vec{O}$ .

Moreover, mass-averaged values ( $\bar{X}$ ) are computed with kinetic energy, and total energy. Let  $X$  be a value to be mass-averaged.  $X$  can be  $e_k$ ,  $e_t$  or  $\vec{m}_k$ . The mass-averaged value is computed as follows.

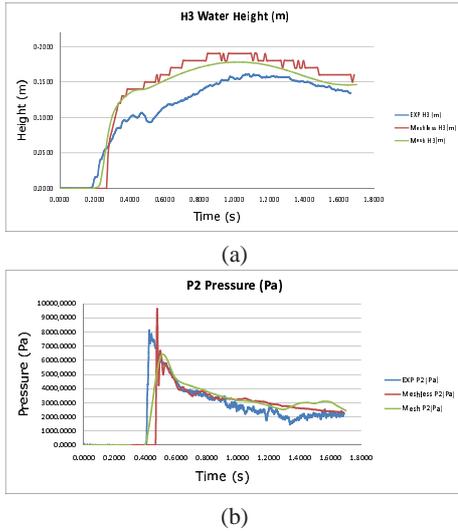
$$\bar{X} = \frac{\int_{Area} \rho X \vec{C} \cdot \vec{N} ds}{\int_{Area} \rho \vec{C} \cdot \vec{N} ds} \quad (4)$$

##### 4.3. Surface Quantity Integrations

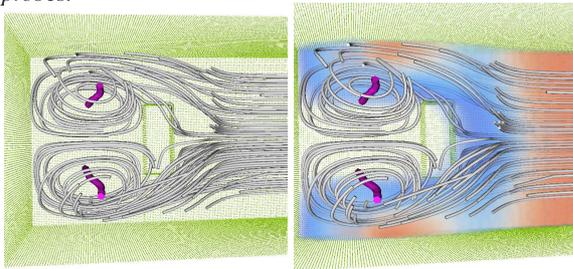
Surface quantities are integrated on a projected surface. After projecting probes, local force ( $\vec{f}$ ), total force ( $\vec{F}$ ), local torque ( $\vec{\tau}$ ), and total torque ( $\vec{T}$ ) are computed. The definitions of the forces and torques are shown as follows. Note that  $\vec{n}$  indicates a normal vector of a segment on a probe and it is different from  $\vec{N}$  described in Section 4.2. The local force ( $\vec{f}$ ) is defined as  $\vec{f} = p \vec{n} ds$  and the local torque ( $\vec{\tau}$ ) is defined as  $\vec{\tau} = \vec{r} \times p \vec{n} ds$ , where  $p$  is pressure and  $\vec{r}$  are defined as in Section 4.2. The total force ( $\vec{F}$ ) is defined as the following.

$$\vec{F} = \int_{Area} \vec{f} = \int_{Area} p \vec{n} ds \quad (5)$$

The total torque can be defined in similar way.



**Figure 5:** (a) Comparisons of dam-break water height using measurements, CFD reference and SPH from line probes. (b) Pressure comparisons on the obstacle front from point probes.



**Figure 6:** Visualization of SPH data resampled using the 3D box probe. Left shows streamlines (white) and vortex cores (pink). Volume rendering of water velocity magnitude is added to the Right image.

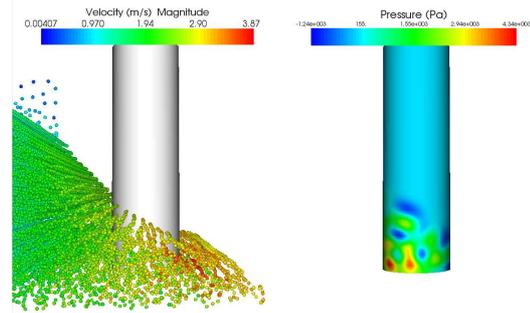
**5. Results and Discussion**

In the following, we present the application of our developments with two different test cases, a classic Dam Break simulation and a Turbomachinery case.

**5.1. Dam Breaking Simulation**

One popular model used by SPH researchers to validate their developments is the dam breaking case. The three dimensional case originally presented by Kleefsman et al. [KfV\*05] has been selected as a reference case by the SPHERIC group [SPH]. This case is very well documented by unsteady pressure measurements on an obstacle and unsteady water height measurements in the tank (see Figure 4). This case allows the validation of the simulation of gravity waves such as sea waves, and prediction of hydraulic forces on structures such as dykes or sea defenses.

Mesh and meshless numerical results are analyzed with



**Figure 7:** An example of arbitrary geometric probing using a cylinder to extract pressure values. The integrated pressure gives force which can be compared with measurements.

line probes to measure water heights. Comparisons with experiments are given in Figure 5 (a). Figure 5 (b) provides time evolution of pressure at one sensor location using a point probe in both SPH and standard CFD data - we are easily able to produce graphs comparing all three. For more advanced arbitrary geometry probing, Figure 7 shows a cylinder inserted into the flow . The pressure exerted on the cylinder can be integrated and compared with a real measurement device of the same dimensions.

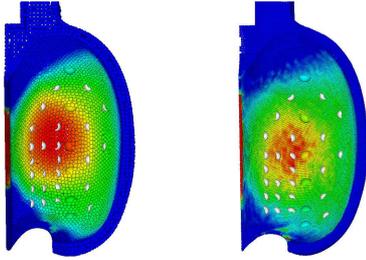
Figure 6 shows two images of feature detection and extraction on the dam-break SPH data after the box probe has been applied. Vortex core lines are shown with streamlines on the left, and on the right, a volume rendering of the velocity component of the resampled field.

**5.2. Turbomachinery Design**

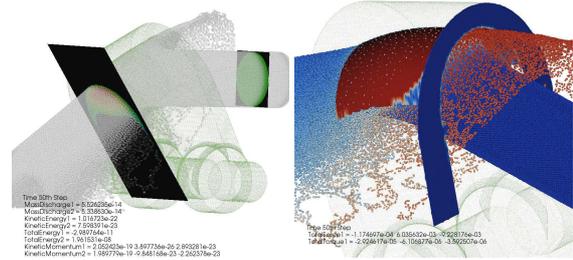
Turbomachinery design relies heavily on CFD, in particular, one application considered in this work is the modelling of a Pelton turbine - a tangential-flow impulse turbine. Despite the simplicity of its concept, it is characterized by complex free surface patterns where water, air and mixed water-air flows have their own key influences on performance [SKPV00]. The complete simulation of Pelton turbines needs considerable effort as there are very many parts, each requiring detailed grid structures. SPH is considered favorable for this kind of model since it requires no finely resolved internal meshes, and because of its simple handling of free surfaces.

The shape of the buckets in the turbine must be finely tuned to optimize the water shape as it enters and leaves the bucket. Pressure sensors are placed on the bucket surface which we use as probe locations and visually compare field strengths and plot values against measurement data. Figure 8 shows the bucket with pressure sensors for mesh (left) and meshless (right) data.

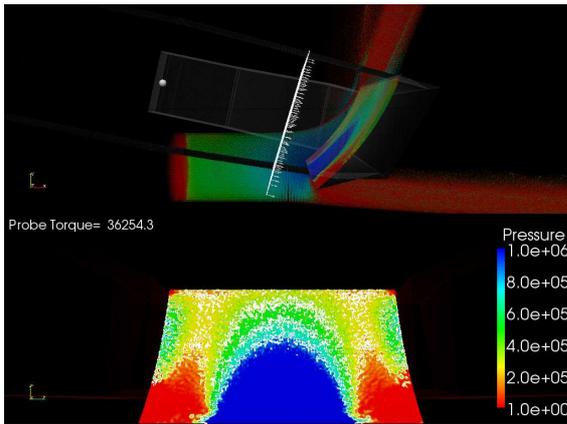
Two other components of the turbine we have used as tests for our tools are the deflector and cutter. These are security devices used to deviate water jets from the runner, enabling



**Figure 8:** Pressure sensors placed inside buckets and coloured according to pressure, additional probe locations shown as spheres. Reference CFD and SPH simulations are compared.



**Figure 10:** Two rectangular probes are shown on the left image. Flow quantities are displayed on the image. Surface projection probes are presented on the deflector boundaries on the right image. The Right shows the effect of backflow (more red particles) bounded from the deflector. With careful examination, we can see that the contact receives small force by the backward flow.



**Figure 9:** A rectangular projection of a probe onto the cutter surface. The probe is animated such that it follows the cutter rotation (about the sphere representing the pivot point) as it passes into the flow.

a rapid shutdown of the turbine. Among design criteria, the hydraulic torque generated by the jet impinging the deflector or the cutter is paramount, because the servomotor must be designed to operate the device in any condition. The meshless simulations of flows in these devices are analyzed in detail by projecting rectangular probes onto the solid body. As shown in Figures 9 and 10, numerical errors are responsible of non physical negative pressure values at the edge between the wet and dry surfaces, leading to errors in the torque computations. This is particularly true on the right image and is due to the small backflow (red particles) in Figure 10. We also make use of probes to derive integrated values at inlet and outlet sections, which allows the engineer to ensure that inflowing and outflowing water discharges are equal in the left image of Figure 10.

### 5.3. Validation

We have compared the results of our probe tools with measurements and reference simulations performed with commercial tools to ensure that the values reported are correct.

Figure 11 (a) shows the result of the integration over the surface of 37 probes for the distributor of figure 2. The results from a commercial CFD post-processor are plotted against those from our probe tool and we find a close agreement between the two. This gives us confidence that we can make predictions of quantities derived from the SPH simulations. Figure 11 (b) shows the measured Torque of a prototype cutter, plotted against the integrated torque from one of our animated projected probe patches. The agreement is sufficient that we can make concrete predictions about the effects of changes in simulation parameters on our results.

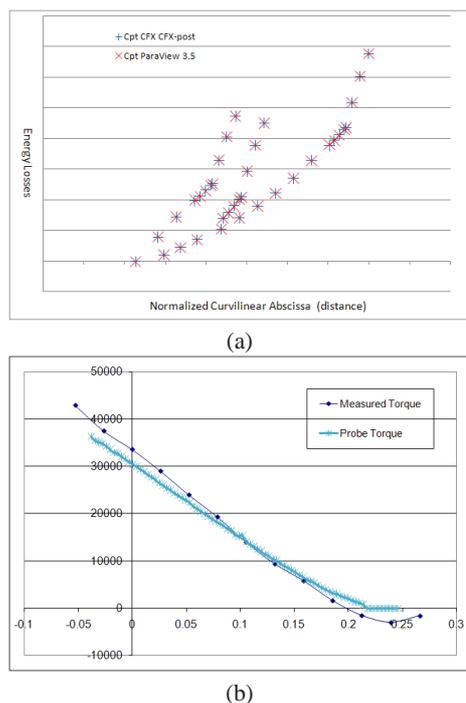
## 6. Conclusion and Future Work

We have presented a unified post processing approach for mesh and meshless simulation data which allows engineers to produce quick and reliable comparisons between different models in a way that it was not possible for them to do before with the tools available. Our design is motivated from actual CFD analysis and provides visual and quantitative information of the kind required by researchers. The work has been validated by comparison with measurement data and with commercial post processing software where applicable. We have produced a toolkit of probing modules which integrates directly into an established visualization package and thus allows great flexibility in its application. For the future work, we will study more adaptive sampling on the probes to obtain accurate values at the edge of the flow and to reduce the probing computation.

## References

[BAU05] BRODLIE K. W., ASIM M. R., UNSWORTH K.: Constrained visualization using the shepard interpolation family. *Computer Graphics Forum* 24 (December 2005), 809–820(12).

[BDR98] BACKES A., DAHR A., RUMPF M.: Interactive visualization of particle systems. In *CGI '98: Proceedings of the Computer Graphics International 1998* (1998), IEEE Computer Society, pp. 88–95.



**Figure 11:** (a) Validation of probe integration showing precise agreement between commercial CFD post-processing tool and our toolkit for the distributor of figure 2. (b) Measured vs computed Torque for the animated probe projected onto the SPH cutter boundary surface of figure 9.

- [BGM\*07] BIDDISCOMBE J., GEVECI B., MARTIN K., MORELAND K., THOMPSON D.: Time dependent processing in a parallel pipeline architecture. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (2007), 1376–1383.
- [CFG\*05] CO C. S., FRIEDMAN A., GROTE D. P., VAY J.-L., BETHEL E. W., JOY K. I.: Interactive methods for exploring particle simulation data. In *EuroVis* (2005), pp. 279–286.
- [CL93] CABRAL B., LEEDOM L.: Imaging vector fields using line integral convolution. In *Proceeding of SIGGRAPH-93: Computer Graphics* (Anaheim, CA, 1993), pp. 263–270.
- [EGM04] ELLSWORTH D., GREEN B., MORAN P.: Interactive terascale particle visualization. In *VIS '04: Proceedings of the conference on Visualization '04* (2004), IEEE Computer Society, pp. 353–360.
- [GIK\*07] GRIBBLE C. P., IZE T., KENSLER A., WALD I., PARKER S. G.: A coherent grid traversal approach to visualizing particle-based simulation data. *IEEE Transactions on Visualization and Computer Graphics* 13, 4 (2007), 758–768.
- [Hen05] HENDERSON A.: *ParaView Guide, A Parallel Visualization Application*. Kitware Inc., 2005.
- [JH95] JEONG J., HUSSAIN F.: On the identification of a vortex. *Journal of Fluid Mechanics* 285 (1995), 69–84.
- [KfV\*05] KLEEFMAN K. M. T., FEKKEN G., VELDMAN A. E. P., IWANOWSKI B., BUCHNER B.: A volume-of-fluid based simulation method for wave impact problems. *Journal of Computational Physics* 206, 1 (2005), 363–393.
- [Kit03] KITWARE, INC.: *The Visualization Toolkit User's Guide*, January 2003.

- [KKK05] KRÜGER J., KIPFER P., KONDRATIEVA P., WESTERMANN R.: A particle system for interactive visualization of 3D flows. *IEEE Transactions on Visualization and Computer Graphics* 11, 6 (11 2005), 744–756.
- [LDS90] LEVY Y., DEGANI D., SEGNER A.: Graphical visualization of vortical flows by means of helicity. *AIAA Journal* 28 (1990), 1347–1352.
- [LHD\*04] LARAMEE R., HAUSER H., DOLEISCH H., VROLIJK B. H., POST F., WEISKOPF D.: The state of the art in flow visualization: Dense and texture-based techniques. *Computer Graphics Forum* 23(2) (2004), 143–161.
- [MCG03] MÜLLER M., CHARYPAR D., GROSS M.: Particle-based fluid simulation for interactive applications. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation* (Aire-la-Ville, Switzerland, Switzerland, 2003), Eurographics Association, pp. 154–159.
- [MNK07] MEYER M., NELSON B., KIRBY R. M., WHITAKER R. T.: Particle systems for efficient and accurate high-order finite element visualization. *IEEE Transactions on Visualization and Computer Graphics* 13, 5 (2007), 1015–1026.
- [Mon88] MONAGHAN J.: An Introduction to SPH. *Computer Physics Communications* 48 (1988), 89–96.
- [Nie93] NIELSON G. M.: Scattered data modeling. *IEEE Computer Graphics and Applications* 13, 1 (1993), 60–70.
- [NJB07] NAVRÁTIL P. A., JOHNSON J., BROMM V.: Visualization of cosmological particle-based datasets. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (2007), 1712–1718.
- [PSLR72] PASLEY R., SNIDER D., LEE O. P., RIEKERT E. G.: National engineering handbook - chapter 14: Stage-discharge relationships, 1972.
- [PVH\*03] POST F., VROLIJK B., HAUSER H., LARAMEE R., DOLEISCH H.: The state of the art in flow visualization: Feature extraction and tracking. *Computer Graphics Forum* 22(4), 2 (2003), 775–792.
- [RRL07] ROSENTHAL P., ROSSWOG S., LINSEN L.: Direct surface extraction from smoothed particle hydrodynamics simulation data. In *Proceedings of the 4th High-End Visualization Workshop* (2007).
- [She68] SHEPARD D.: A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 23rd ACM national conference* (1968), ACM, pp. 517–524.
- [SKPV00] SICK M., KECK H., PARKINSON E., VULLIQUOUD G.: New challenges in pelton research. In *HYDRO 2000 Conference* (2000).
- [SPH] SPH European Research Interest Community. Website. <http://wiki.manchester.ac.uk/spheric>.
- [WKM05] WALKER R., KENNY P., MIAO J.: Visualization of Smoothed Particle Hydrodynamics for Astrophysics. In *Theory and Practice of Computer Graphics 2005* (June 2005), Lever L., McDermid M., (Eds.), Eurographics Association, pp. 133–138.
- [ZG06] ZHOU Y., GARLAND M.: Interactive point-based rendering of higher-order tetrahedral data. *IEEE Transactions on Visualization and Computer Graphics* 12, 5 (2006), 1229–1236.
- [ZK06] ZHANG H., KAUFMAN A.: Interactive point-based iso-surface exploration and high-quality rendering. *IEEE Transactions on Visualization and Computer Graphics* 12, 5 (2006), 1267–1274.