THE CPIV Software of IMFT - Institut de Mécanique des Fluides de Toulouse

for parallel processing of PIV data on MPI architectures*

By
Pierre Elyakime
IMFT

*Used for the experiments of the SMS EU Programme

4.2. CPIV-IMFT

The post-processing of the experiments results was processed by the free 2D-2C PIV software developed at the Institut de Mcanique des Fluides de Toulouse.
This multiplatform Windows, Linux software is based on the multi-grid iterative algorithm with image distortion described in [11,13]. The reader can refer to Fig. 3 [CFTL poster] for an overview of the CPIV-IMFT algorithm.
A first velocity field estimated from a large 64 by 64 pixel interrogation window is computed by a conventional PIV sub-pixel method with FFT correlation resolution. A correction of this velocity field is made on more spatial resolved interrogation windows, 32 by 32 pixels, following the iterative "Continuous Window Shifting correlation-based interrogation algorithm" (CCWS) presented in [10]. In practice, the interrogation window of the first image is shifting by a value of $-V(x)/2$ and the second image interrogation window by a value of $+V(x)/2$, where $V(x)$ is the velocity value.
This technique of a pixel fraction displacements using a bilinear interpolator reduce errors due to whole displacements well known as the peak-locking phenomenon. The correlation peak is determined with sub-pixel precision using 2nd order polynomial interpolation from the logarithm of a Gaussian function [7]. The velocity fields computed at the end of each iteration are then added to the initial field. Although this

iterative continuous window displacement algorithm reduces PIV measurement uncertainty emissions, as shown in [piV Review final Paper], the flow movement is not fully described by this algorithm.
According to Huang et al [xx] the movement of a flow can be decomposed into translation + rotation (a function of du/dy and dv/dx) + stretching (a function of du/dx and dv/dy) and consequently, the measurement of the velocity gradient may be poorly measured in some cases, especially for small scale velocity gradients encountered in turbulent flows. To take this into account, a technique for deforming particle images by interpolation is suggested by Huang et al. [xx].
This technique is used in the CPIV-IMFT algorithm following iterative shift. To do this, the interrogation windows, even more spatially resolved, 16 by 16 pixels, are iteratively distorted by the velocity fields and then shifted by half a vector. The deformation is computed at each point of the interrogation window by a second-order polynomial interpolation. After the shift, the interrogation windows are recomputed by cardinal sinus interpolation.
The interpolator used during the deformation step in the calculation of the velocity field at each point of the interrogation window is essential for the accuracy of the result. [10] believes that its high order can cancel the effects of velocity gradients without having to calculate them, but also introduce significant errors if parasitic vectors are not corrected. For this reason, in CPIV-IMFT a detection of false vectors is performed before each deformation by thresholding the peak ratio (main peak/secondary peak) and by the method of the residues of the median normalized by the median of the residues[3,11], followed by a polynomial interpolation correction. The process can be repeated on more resolved interrogation windows to increase spatial resolution.

An evaluation of the CPIV-IMFT algorithm in this present case is made using LaVision's commercial DaVis software as a reference value.
A sequence of 1000 images of the ”static” experimental case is computed with both software and their results are used to evaluate the CPIV-IMFT software.

The morphing studies conducted in this article lead to the study of the effects of

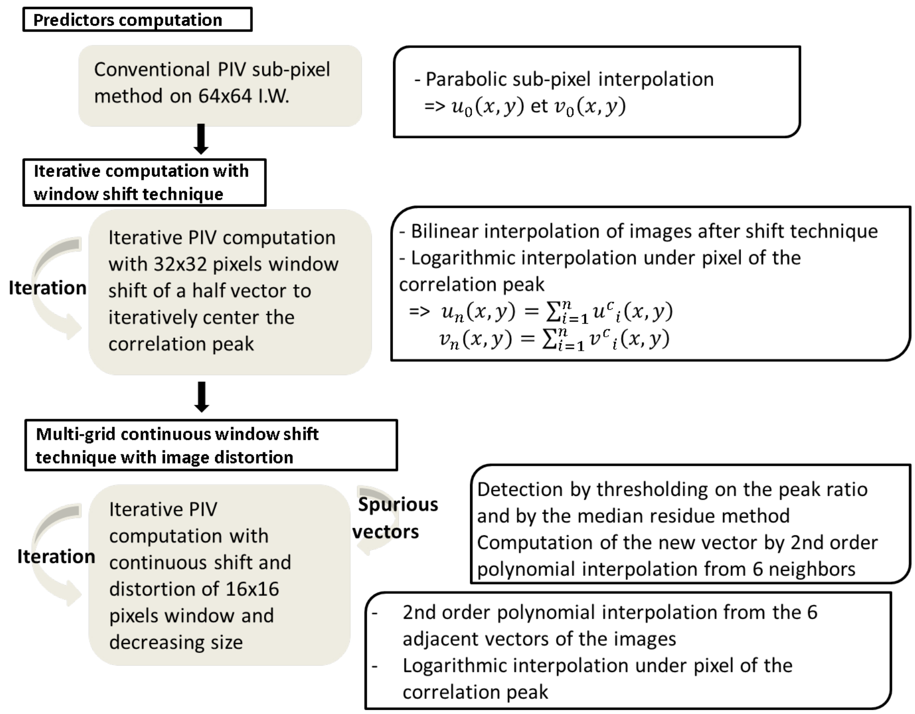


Figure 3: *Algorithm CPIV-IMFT schemes*

turbulence in the wake of a wing. For this purpose, a spectral analysis of the time signals of the vertical component of the velocity U is performed in three positions ($P1$, $P2$ and $P3$) downstream of the trailing edge, seen in Fig. 4 and computed by the Fast Fourier Transform method. The study is completed by the calculation of spectral density power (PSD) by the Wesh method, very useful here to describe the turbulence (to be improved).

Fig. 5 shows that both in the position near the trailing edge P1, a little further P2 than further P3, there is a very good correspondence of the time signal of the velocity vertical component U, with less than a decade values on a sample of 1000 with very different values.

The Fast Fourier Transform (FFT) of the time signals computed in these three po-

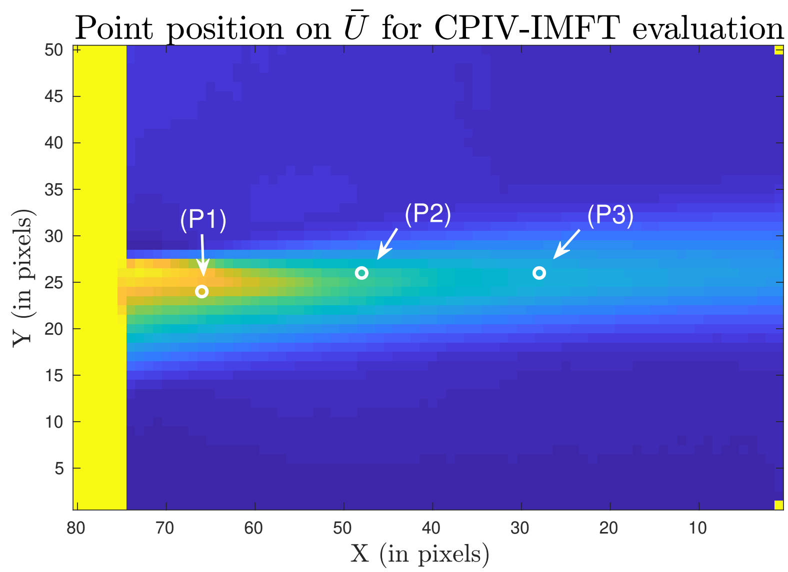


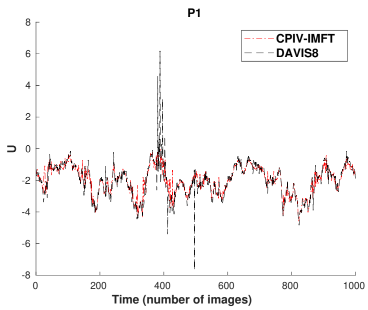
Figure 4: *Point position on* $\bar{U}$

sitions are presented on the graphs in Fig. 6 and shows a great similarity in the results obtained with the CPIV-IMFT and DaVis software as well.

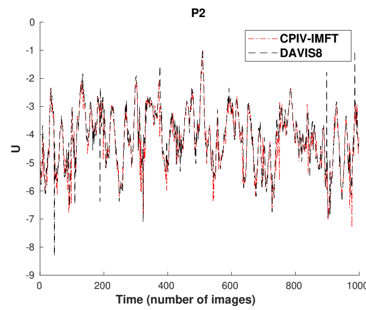
The proximity of the accuracy of the results computed by the two software programs CPIV-IMFT and DaVis is confirmed by the graphs of the spectral density power (PSD), seen in Fig. 7 which show curves very close to each other.

These three graphs show that the results processed by the CPIV-IMFT and DaVis software are very similar from a spectral analysis point of view, and thus allow the same physical interpretations to be obtained.

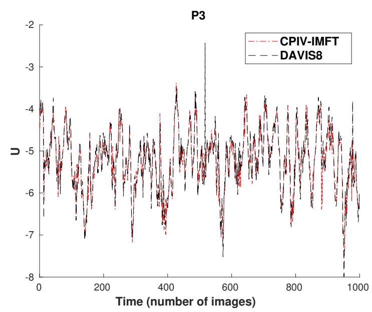
The study of turbulence in the wing wake by spectral analysis required the acquisition of large image data sequences (36 sets of 66775 images representing about 5TB) in order to have a high temporal resolution and a good statistical convergence. This was made possible by the use of the Time Resolved PIV experimental technique combined with the use of a high-speed acquisition camera with a large storage capacity.



(a) *Signal time in P1*



(b) *Signal time in P2*

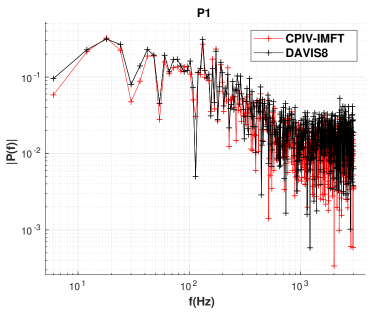


(c) *Signal time in P3*

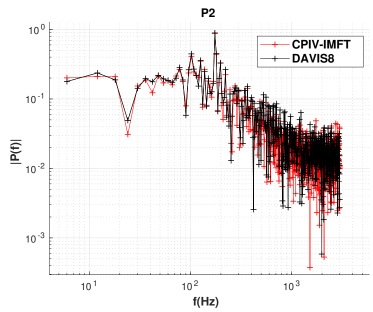
Figure 5: *Signal in time from CPIV-IMFT and DaVis in U results*

In addition, the processing time of a sequence of 66775 images with a resolution of 1280 by 800 pixels by a software without acceleration whose calculation is done on a single CPU core takes more than three days. The processing of the 36 sets would have taken more than a hundred days with continuous calculations.

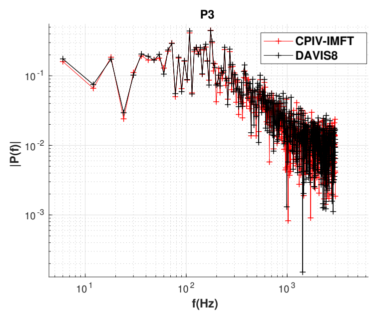
More generally, the evolution of acquisition systems, including the increase in camera acquisition frequency and laser pulse rate frequency, has made it possible to increase the number of data captured for a sequence and to improve the temporal resolution of the results. Moreover, some high-speed acquisition cameras nowadays make it possible to obtain sequences of 200,000 images with a spatial resolution of 1280 by 800 pixels. With such data, processing time has become the main problem in PIV, and even a limiting factor for the analysis of experimental results and therefore for the production of scientific results.



(a) *FFT of the signal time in P1*



(b) *FFT of the signal time in P2*



(c) *FFT of the signal time in P3*

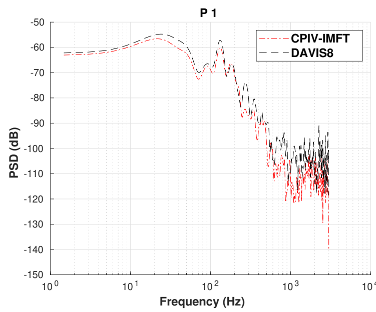
Figure 6: *FFT of the signal in time from CPIV-IMFT and DaVis in U results*

Despite the accelerations offered, by S. Tarashima [1] with GP-GPU technology for example, processing times remain long.

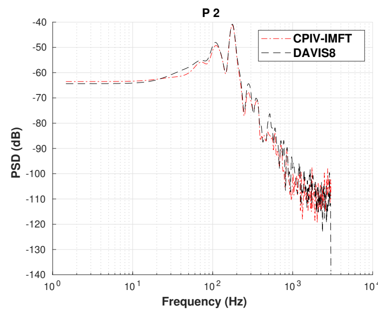
The CPIV-IMFT software offers acceleration with the distributed memory parallelization technique (SPMD) [2] applied to the image data sequences.

This parallelization implemented with the Message Passing Interface (MPI) library allows simultaneous processing on all the computation cores of a computer or a supercomputer of sub-sequences resulting from a preliminary, uniform and distributed breakdown on each computation core.

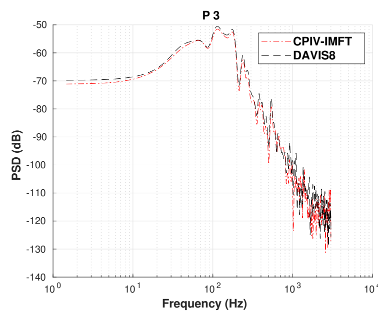
The accelerations obtained allow significant time gains: the calculation of a sequence of 66775 images on 40 nodes (800 cpu cores) of the French regional supercomputer EOS showed a time gain of 730.



(a) *PSD in P1*



(b) *PSD in P2*



(c) *PSD in P3*

Figure 7: *Power Spectral Density with CPIV-IMFT and DaVis U results*

The processing of PIV data here has benefited from the acceleration of the CPIV-IMFT software. The calculations launched on the EOS supercomputer have shown almost linear time savings with the number of cores used.

[1] S. Tarashima, M. T. , GPU accelerated direct cross-correlation PIV with window deformation, Int. Symp. on Appl. Laser Techniques to Fluid Mechanics, Lisbon, Portugal, 2010 [2] The SPMD Model: Past, Present and Future. Darema F. , In: Cotronis Y., Dongarra J. (eds) Recent Advances in Parallel Virtual Machine and Message Passing Interface. EuroPVM/MPI 2001. Lecture Notes in Computer Science, vol 2131. Springer, Berlin, Heidelberg, 2001