Chapter 3
On the Performance of Drilling Elements in FEM-like Global DIC

Antonio Baldi and Pietro Maria Santucci

Abstract Digital Image Correlation (DIC) is usually implemented using a local description of the displacement field, resulting in the well-known subset approach. However, global displacement fields can also be used, leading to the iDIC method and the FEM-like global approach.

The latter is usually implemented using 3-node triangles or 4-node quads. Parabolic and cubic elements can also be used, but they must cover a large number of pixels to compensate for the significantly larger number of nodes (i.e., degrees of freedom).

In this work, we explore the performance of drilling elements, i.e., elements that possess an additional degree of freedom for each node to control the rotation around an axis perpendicular to the working plane.

Compared to a standard 3-node triangular element, the drilling variant allows for a significantly larger spectrum of deformation modes, but requires fewer degrees of freedom than a 6-node parabolic triangle, thus suggesting the potential for better performance in describing complex displacement fields.

Keywords Digital Image Correlation, Finite Element Formulation, Drilling Triangle

Introduction

Digital Image Correlation (DIC) is a well-known non-interferometric experimental techniques. It assumes that the intensity of the imaged pixels does not change during motion. Although this requirement appears to be of modest relevance, it is rarely satisfied; moreover, it leads to an ill-posed problem because the resulting Optical Flow equation

$$\frac{\partial I}{\partial x}\frac{\partial u}{\partial t} + \frac{\partial I}{\partial y}\frac{\partial v}{\partial t} + \frac{\partial I}{\partial t} = 0 \tag{1}$$

involves, apart from I (the pixel intensity), two variables (u and v).

Thus, the displacement field is usually estimated in the least-squares sense using the approach initially proposed by Lucas and Kanade [1]. In this formulation, the reference image intensity f(x) is compared to the current image intensity g(x) by Least-Squares fitting a local approximation of the displacement field. Indeed, the u_k and v_k displacement components are usually described using polynomials:

$$\begin{aligned} u_k &= s_0 + s_1\xi_k + s_2\eta_k + s_3\xi_k^2 + \dots \\ v_k &= t_0 + t_1\xi_k + t_2\eta_k + t_3\xi_k^2 + \dots \end{aligned} \tag{2}$$

Antonio Baldi
Dipartimento di Ingegneria Meccanica, Chimica e dei Meteriali
University of Cagliari
Via Marengo 2, IT09123, Cagliari, Italy
e-mail: antonio.baldi@unica.it

Pietro Maria Santucci
SISSA – Scula Internazionale Superiore di Studi Avanzati
Via Bonomea, 265, IT34136, Trieste, Italy
e-mail: pietro.santucci@sissa.it

© The Author(s), under exclusive license to River Publishers 2026

Motoharu Fujigaki et al. (eds.), *Advancement of Optical Methods & Digital Image Correlation in Experimental Mechanics, Volume 2, 2026*, Conference Proceedings of the Society for Experimental Mechanics Series, https://doi.org/10.13052/rp-9788743816256

Where ξ_k and η_k are local coordinates relative to the sampling point (x_0, y_0), (i.e.,$\xi_k = x_k - x_0$, $\eta_k = y_k - y_0$) and s_i and t_i are the sets of parameters to be identified.

By minimizing an error estimate over a local area

$$\sum_{k=1}^{N} \left[a + bf\left(x_x, y_k\right) - g\left(x_k + u_k, y_k + v_k\right)\right]^2 \tag{3}$$

where the summation extends over the set of pixels around the point of interest (usually known as the subset) and a and b account for mean intensity and contrast variations, the sets of parameters s and t can be identified by a Gauss-Newton algorithm. The full process requires linearizing the current intensity$g\left(x_k + u_k, y_k + v_k\right)$ and iteratively solving the linear system resulting from the search for the minimum. On convergence, the mapping sets are identified, and consequently, the displacement field is known.

Various alternatives to the sketched solution algorithm exist; whatever the algorithm, the imaged surface must be textured to ensure the uniqueness of the local pattern. Moreover, because of the implicit assumption of continuity of the intensity field (sampled at pixel locations), the intensity of the current or reference image (or both) must be interpolated at non-integer coordinates.

A full iteration of the described algorithm provides the displacement components at a single point. Indeed, the solution process is inherently local because the description (2) of the displacement field is valid in a limited area around the point of interest. Thus, to estimate a global field, the DIC analysis must be repeated to sample the imaged surface. However, two alternatives exist. In the integrated approach (iDIC) a problem-specific equation, that globally describes the displacement field, replaces (2). A single computation thus suffices to solve the problem completely; indeed, the summation appearing in the error functional (3) involves all the pixels of the Region of Interest (ROI). The iDIC approach appears to be the Columbus' Egg of DIC. However, it can only be used to solve a limited set of problems (those for which a theoretical displacement field is known and is simple enough to be embedded inside the DIC solution algorithm).

Apart from iDIC, a common approach to the development of a global DIC formulation involves using a Finite-Element (FE) like framework: the ROI is tiled using simple geometry patches (the elements) that are connected at specific points (the nodes, usually located at the corners of the elements). The displacement field inside each element is controlled by the displacement of its nodes by means of relatively simple shape functions [2]. Because nodes are shared by neighboring elements, the motion of a node propagates to the full mesh, thus making the description global.

The solution path of FE-DIC remains roughly the same of the Subset-based solution, with three notable differences:

- Because the solution is global, the set of parameters includes all displacement components of all nodes in the mesh, i.e., the number of unknowns involved in the solution process is equal to the number of nodes times the degrees of freedom of each node (usually 2, u and v).
- Since each nodal displacement affects only the elements connected to it, the solution matrix is sparse. By using a "smart" labeling of the nodes, the matrix band can be made very small, thus significantly reducing the CPU load.
- The assembly process involves each pixel only once. In the "standard" method, neighbor subsets usually overlap. Thus, intensity interpolation and derivative computation must be repeated several times per pixel, making the subset approach significantly slower[1] than FE-DIC.

It is worth noting that although FE-DIC appears to be very similar to a Finite Element Code (meshing infrastructure, shape functions, band-optimization algorithms and skyline solver are the same), it differs significantly because the DIC code continues to be a statistical process. While in a "standard" FE code increasing the number of elements improves accuracy, doing so in FE-DIC may be disruptive because the number of unknowns increases, but the experimental data (the number of pixels inside the ROI, i.e., the sample size) remains unchanged. Ultimately, the number of pixels per degree of freedom decreases, thus making the solution more uncertain.

The Drilling Triangle

The most commonly used element type in FE-DIC is the four-nodes quadrilateral. Although its use significantly limits the ability to mesh complex shapes, it is an optimal choice. Indeed, when comparing a four-node rectangular element and two

[1]Because each subset uses an independent set of parameters and subset-specific local coordinates, the displacement of the same pixel when viewed from different subsets is not the same, thus requiring re-performing intensity interpolation.

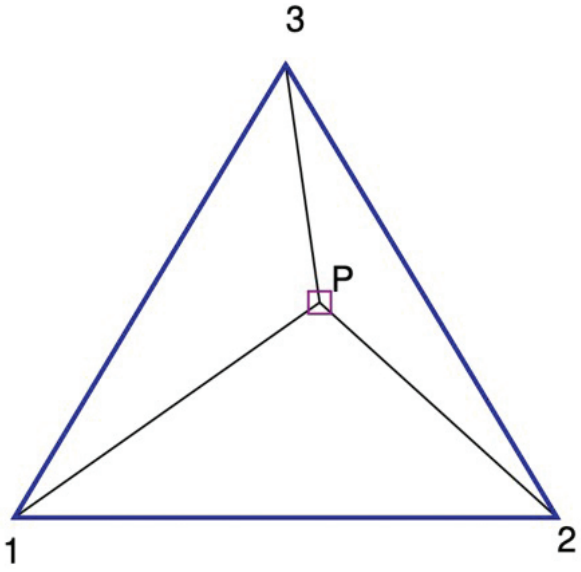

Fig. 1 Area coordinate L_1 is defined as the ratio of the area of triangle P23 to the area of triangle 123. L_2 and L_3 are defined in a similar way by index rotation. Note that $L_1 + L_2 + L_3 = 1$, thus, only two coordinates are independent.

same-total-area linear triangles, the number of degrees of freedom is the same (nodes on the common side of the triangles are shared). However, strains within each linear triangle are constant[2], whereas they change smoothly within the rectangle[3]. To obtain a similar behavior with triangular elements, a couple of parabolic elements is required, which, however, require nine nodes (i.e., 2.25 times the number of nodes of the rectangular one). Thus, the assembly of two parabolic triangles is much less reliable for the same area; alternatively, they require a much larger surface area to compensate for the higher number of degrees of freedom.

In 1984 Allman [3, 4] introduced a compatible triangular element that includes vertex rotation in addition to the standard in-plane degrees of freedom (figure 2). The proposed element allows for non-constant strains but requires significantly fewer degrees of freedom than a parabolic triangle. Because triangular elements allow much more flexibility when meshing complex ROIs, the proposed formulation appears to be a good intermediate choice between the requirement for meshing flexibility and the need for maximizing the number of pixels for each degree of freedom.

Skipping the detail of the derivation, in the cited article is shown that, inside the element,

$$u = u_1 L_1 + u_2 L_2 + u_3 L_3 + \frac{1}{2}\left[l_{12} cos\gamma_{12}\left(\omega_1 - \omega_2\right) L_1 L_2 + l_{23} cos\gamma_{23}\left(\omega_2 - \omega_3\right) L_2 L_3 + l_{31} cos\gamma_{31}\left(\omega_3 - \omega_1\right) L_3 L_1\right] \quad (4)$$

and

$$v = v_1 L_1 + v_2 L_2 + v_3 L_3 + \frac{1}{2}\left[l_{12} sin\gamma_{12}\left(\omega_1 - \omega_2\right) L_1 L_2 + l_{23} sin\gamma_{23}\left(\omega_2 - \omega_3\right) L_2 L_3 + l_{31} sin\gamma_{31}\left(\omega_3 - \omega_1\right) L_3 L_1\right] \quad (5)$$

where l_{ij} is the length of side *i-j*, γ_{ij} is the angle of side *i-j* with respect to the *x* axis, ω_i is the rotation of corner *i* and L_i are the area coordinates.

Looking at (4) and (5) it is apparent that displacement functions are the same as the linear triangle, augmented with extra terms controlled by the rotations at the corners of the sides. Moreover, by simple inspection it is apparent that the displacements along each side (e.g. 2-3) are controlled only by the related corners quantities (along the side 2-3 L_1 is null), thus, triangle is compatible (i.e., the displacement field is continuous).

For the sake of completeness, it should be noted that the element has two zero-energy modes: one related to rigid body motion, and a second appearing when $u_i = v_i = 0$ and $\omega_i = c$ (a constant). Both modes have to be avoided in "standard" Finite-Element analysis but give no problem in DIC (where no boundary condition is imposed).

[2]Shape function of a three-nodes triangles are $N_i = L_i$, where L_i are area coordinates (Fig. 1). Thus, strains are constants inside the element.

[3]Shape functions of four-nodes rectangle are $N_1 = (\xi - 1)(\eta - 1)/4$, $N_2 = (\xi + 1)(\eta - 1)/4$, $N_3 = (\xi + 1)(\eta + 1)/4$, and $N_4 = (\xi - 1)(\eta + 1)/4$. All of them include the $\xi\eta$ term, where ξ and η are local coordinates in the reference configuration

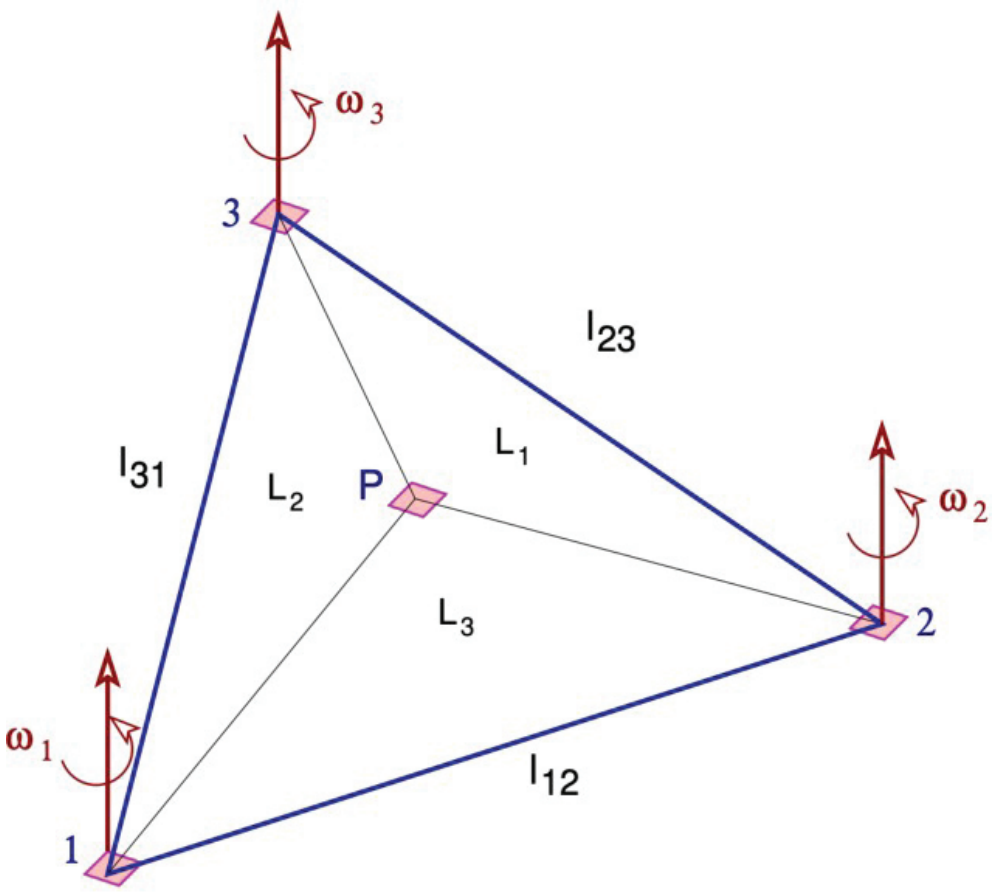

Fig. 2 The drilling triangle. Apart from standard *u* and *v* degrees of freedom, nodes can also rotate.

Integrating the Allman's triangle inside FE-DIC is not straightforward because of the non-standard number of degrees of freedom per node. Moreover, the Newton-Gauss fitting loop inside the DIC solver requires an initial value for all the variables (including corner rotations). To solve the problem, we adopted a standard coarse-fine approach for the displacement components, while we inherited the corner rotations from previous step, assuming a null rotational field at the beginning of the analysis.

Numerical Validation

To assess the performance of the new element, we opted to use numerically generated images because they allow us to estimate errors relative to the expected values. Using our in-house developed speckled-image generator, we created a set of images using parabolic displacements:

$$u = k_1 x^2 + k_2 y^2 \qquad v = k_2 x^2 + k_1 y^2$$

where k_1 moves from 10^{-6} to $20{\cdot}10^{-6}$ in twenty steps and k_2 ranges from $4{\cdot}10^{-6}$ to $80{\cdot}10^{-6}$.

Figure 3 shows the u displacement field at the last step of the analysis. The ROI is exactly 1000x1000 pixel^2 thus, the expected maximum displacement is 100 pixels in the x and y direction. The analysis was performed using both the proposed 3-node triangle with drilling degree of freedom and the standard 3-node triangle and 4-node quadrilateral.

Because we know the nodes' locations and their displacements, it is possible to estimate the mean displacement error (and its standard deviation) over the full image and across the full history. Figure 4 compares the mean *u* and *v* errors of the standard 3-node triangle (element code: T_3L), the 4-node quadrilateral (element code: Q_4L), and the drilling triangle (element code: T_3D). Analysis of the plots is inconclusive because mean errors are tiny and the clearly best element in terms of u displacement is the worst in the orthogonal direction.

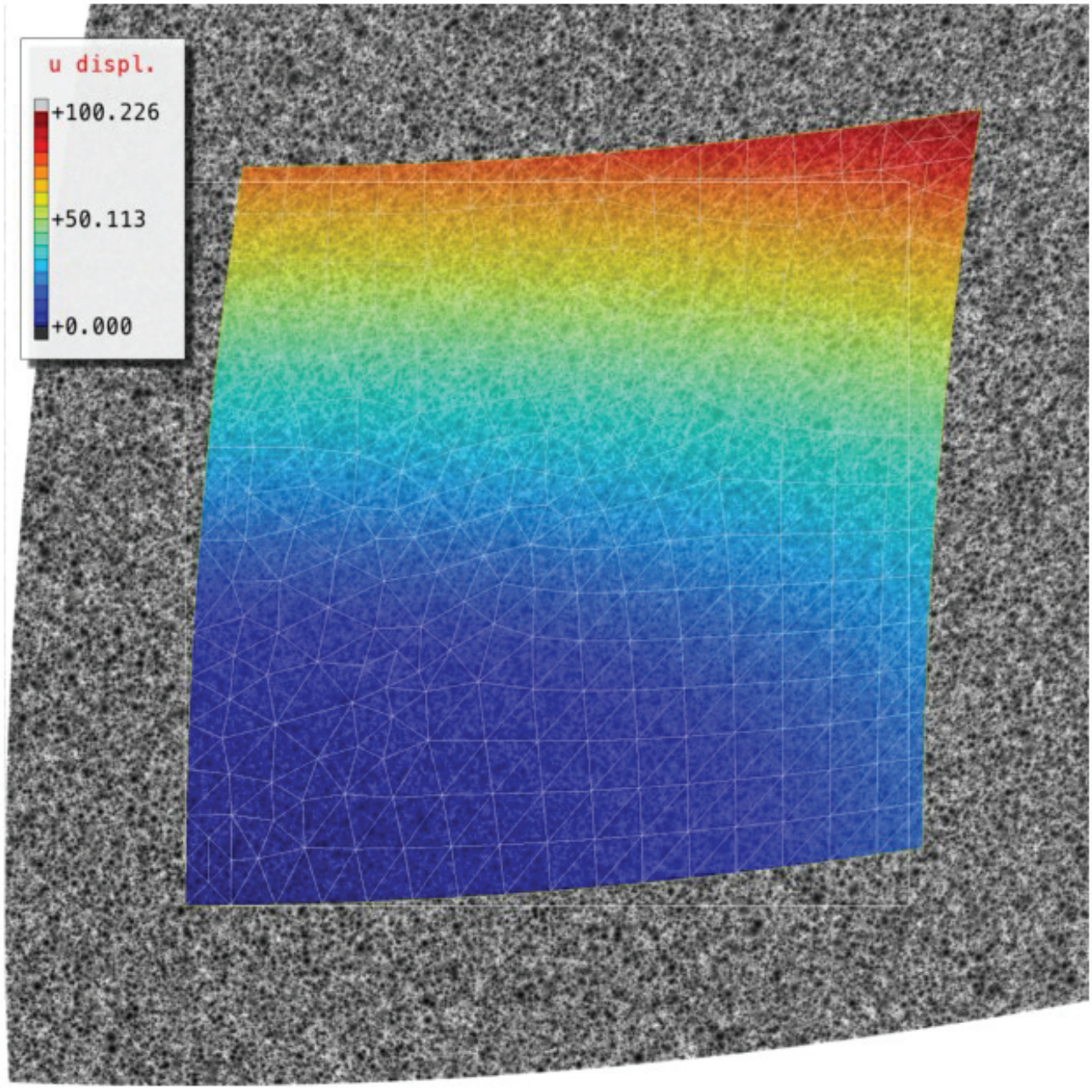

Fig. 3 Density plot of the u displacement at the end of the loading history. T3D element.

It is worth noting that the comparison is fair in the sense that we scaled the mean element size to keep the pixel-to-degree-of-freedom ratio constant.

Because our motivation was to improve performance in terms of strain components, we repeated the analysis looking at the mean strain error and standard deviation. Figure 5 shows the results for the ε_{yy} component. Results are unexpected: the T_3D triangle is outperformed by both the T_3L and Q_4L elements but shows the smaller standard deviation . In any case, differences are tiny, and there is no significant improvement.

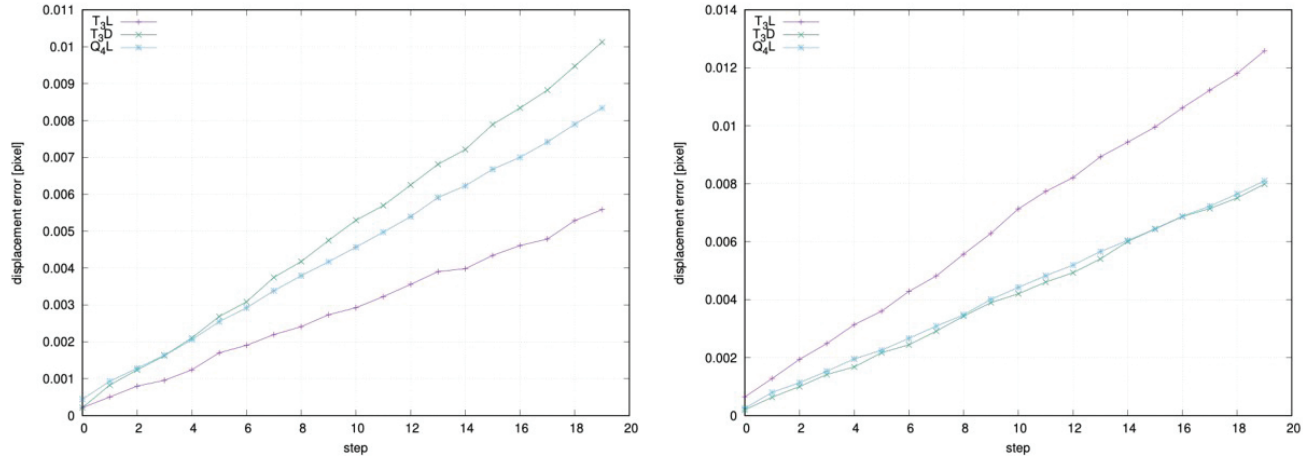

Fig. 4 History plot. Left: mean u error; right: mean v error.

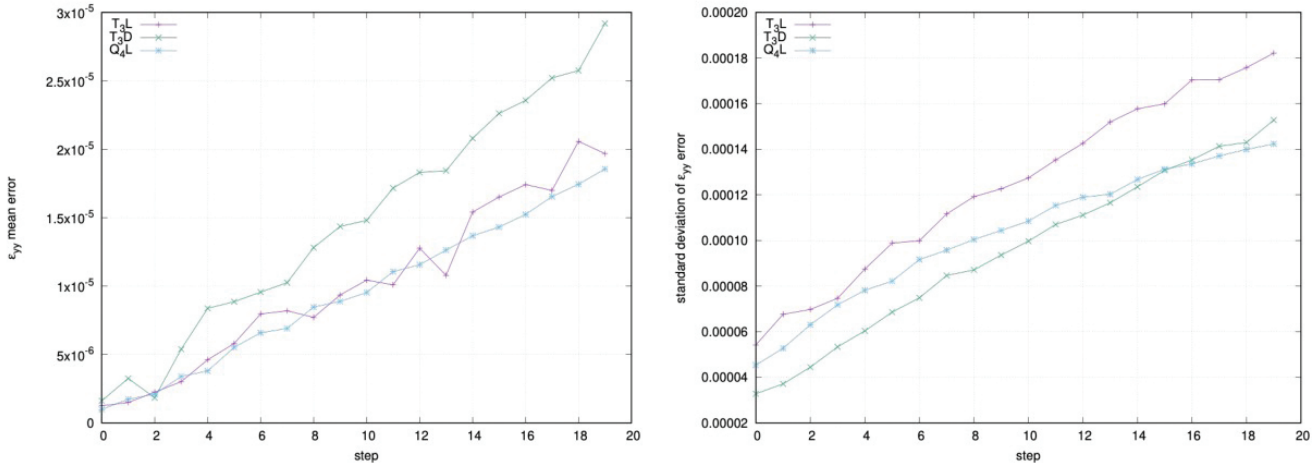

Fig. 5 History plot. left: mean strain error; right: standard deviation

Conclusion and Future Work

The numerical tests show that the expected performance improvements from the drilling triangle are only marginal. Considering the complexity involved in its use (almost all the FE infrastructure has to be adjusted to account for the increased number of degrees of freedom per node) its adoption is not recommended at this time.

References

1. B.D. Lucas and T.Kanade, “An Iterative Image Registration Technique with an Application to Stereo Vision”, Proceedings of Imaging Understanding Workshop, 121-130, 1981
2. O.C. Zienkiewicz, R.L. Taylor, J.Z.Zhu, “The Finite Element Method its Basis & Fundamentals”, Elsevier, Butterworth-Heinemann, 2005.
3. D.J. Allman, “A Compatible Triangular Element Including Vertex Rotations for Plane Elasticity Analysis”. Computers and Structures, 19 (1-2), 1-8, 1984.
4. D.J. Allman, “Evaluation of the Constant Strain triangle with Drilling Rotations”, International Journal for Numerical Methods in Engineering, 26, 2645-2655, 1988.