AN OVERVIEW OF THE VIRTUAL STRAIN GAUGE FORMULATION IN DICE

D. Z. TURNER

ABSTRACT. This document outlines the virtual strain gauge formulation used in DICe, the Digital Image Correlation Engine (Sandia's open source DIC code). **[Report No. SAND2018-5463 R]**

1. Introduction

Computing strains using the virtual strain gauge method involves fitting a polynomial to a neighborhood of subset displacement values and using the coefficients of that polynomial fit to compute the strains. Strains are computed in the source code using a `DICe::Post_Processor` class. The code for the virtual strain gauge calculations is in the file `DICe\src\core\DICe_PostProcessor.cpp` under the derived class `DICe::VSG_Strain_Post_Processor`. Note that there are several other strain measures that can be used by adding another post processor to the `params.xml` file. The main difference between the strain measures is in how they handle noise in the displacement values. In this document, we only describe the virtual strain gauge as this is the most commonly used post processor.

Before proceeding to the formulation, we note that computing accurate strains requires some judgment from the user due to the trade-offs between noise reduction and capturing peak strain. Because the virtual strain gauge is predicated on a polynomial fit, the larger the neighborhood incorporated, the smoother the strain result will be. While increasing the size of the neighborhood reduces noise, it will also decrease the peak strain value. Whenever one is interested in the strain at a particular location, he or she should perform a gauge size study as well as a subset size and step size study to explore how the peak strain value changes with these parameters. A prescriptive process for completing a study like this is not provided here because this is an open area of research. We simply caution the user to think critically about the effect of the software parameters on the computed strains.

2. Formulation

After the subset displacements have been computed for a frame in the DIC analysis, all post processors (including the VSG strain post processor) are executed. The formulation below assumes that the x and y displacements are known for each subset. Prior to executing the DIC analysis, during a setup phase, the `DICe::VSG_Strain_Post_Processor` determines all of the subsets in the neighborhood of each subset. This occurs in the `pre_execution_tasks()` method of the class. A kd-tree is constructed with all of the subset coordinates and the neighborhood is determined using the `strain_window_size_in_pixels` parameter (defined in the post processor block of the `params.xml` file) as the diameter of the envelope. For a 2D analysis, the subset image coordinates are used to construct the neighborhood. For a stereo analysis, the physical or model coordinates are used.

In the next step, a polynomial fit of the displacement field over the neighborhood of each subset is constructed. Currently, only a linear polynomial fit is available. The scalar displacement values in x and y are independently fit to the following polynomial

Date: May 21, 2018.

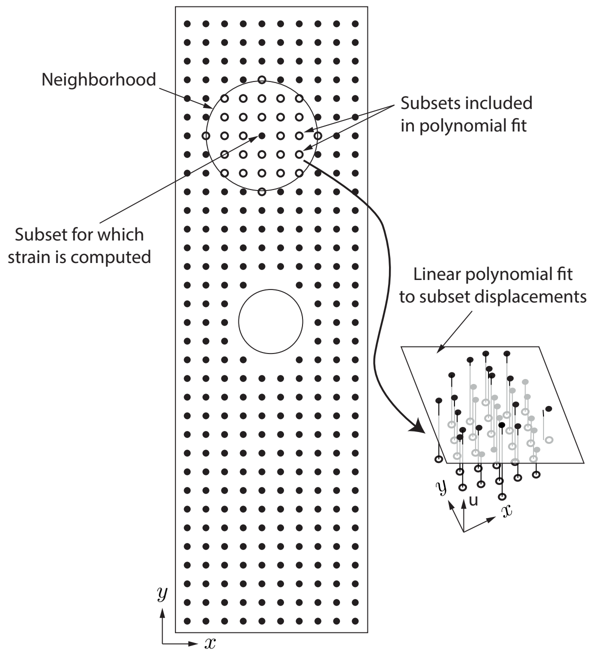

FIGURE 1. Depiction of the local polynomial fit of the displacements used in the virtual strain gauge calculation. The local polynomial fit is repeated for the neighborhood of each subset in the analysis.

$$\begin{aligned} u(x,y) &= a_0 + a_x x + a_y y \\ v(x,y) &= b_0 + b_x x + b_y y \,, \end{aligned} \tag{1}$$

where a_0 and b_0 are constants, a_x and b_x are the x coefficients and a_y and b_y are the y coefficients of the polynomial fit. Using a least-squares method leads to the following set of equations that can be solved to determine the coefficients

$$\begin{aligned} \mathbf{Xa} &= \mathbf{u} \\ \mathbf{Xb} &= \mathbf{v} \,, \end{aligned} \tag{2}$$

where for n neighbors, $\mathbf{X}$ is an n x 3 constant matrix involving the distances between the subset for which the strain is computed and its neighbors, $\mathbf{a}$ and $\mathbf{b}$ are the vectors of the three coefficients, and $\mathbf{u}$ and $\mathbf{v}$ are n x 1 vectors of the scalar displacement values for each neighbor in x and y, respectively. The first column of the matrix $\mathbf{X}$ is all ones. The second column is the signed x distance to neighbor i, and the third column is the signed y distance. Equation (3) is solved using a pseudo-inverse approach,

$$\begin{aligned} \mathbf{a} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{u} \\ \mathbf{b} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{v} \,. \end{aligned} \tag{3}$$

For a 2D analysis, the `SUBSET_DISPLACEMENT` field is used to construct $\mathbf{u}$ and $\mathbf{v}$. For a stereo analysis, the `MODEL_DISPLACEMENT` field is used. In a stereo analysis the distances between the subset locations are computed on the fitted plane.

The derivatives needed to compute the strain are determined using the coefficients as

$$\frac{du}{dx} = a_x \tag{4}$$
$$\frac{du}{dy} = a_y$$
$$\frac{dv}{dx} = b_x \tag{5}$$
$$\frac{dv}{dy} = b_y \,.$$

The Green-Lagrange strain components are then computed as

$$E_{xx} = \frac{1}{2}(2\frac{du}{dx} + \frac{du}{dx}\frac{du}{dx} + \frac{dv}{dx}\frac{dv}{dx}) \tag{6}$$
$$E_{yy} = \frac{1}{2}(2\frac{dv}{dy} + \frac{du}{dy}\frac{du}{dy} + \frac{dv}{dy}\frac{dv}{dy})$$
$$E_{xy} = \frac{1}{2}(\frac{du}{dy} + \frac{dv}{dx} + \frac{du}{dx}\frac{du}{dy} + \frac{dv}{dx}\frac{dv}{dy}) \,.$$

These quantities result in the `VSG_STRAIN_XX`, `VSG_STRAIN_YY`, and `VSG_STRAIN_XY` fields in the output files.

Acknowledgment

Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Correspondence to: Daniel Z. Turner, Center for Computing Research, Sandia National Laboratories, P.O. Box 5800; Albuquerque, New Mexico 87185. *E-mail address:* dzturne@sandia.gov, *Phone:* (505) 845-7446