Finite Elements in Analysis and Design 116 (2016) 21–31

Contents lists available at ScienceDirect

Finite Elements in Analysis and Design

journal homepage: www.elsevier.com/locate/finel

A new approach to reduce the number of integration points in mass-matrix computations

Eli Hanukah, Sefi Givli *

Faculty of Mechanical Engineering, Technion – Israel Institute of Technology, Haifa 32000, Israel

ARTICLE INFO

Article history:
Received 14 December 2015
Received in revised form
17 March 2016
Accepted 27 March 2016
Available online 12 April 2016

Keywords:
Numerical integration
Improved degree of precision
Semi-analytical
Closed form
Mass matrix

ABSTRACT

We present a new approach to compute the mass matrix of solid finite elements which allows a significant reduction in the number of integration points. The method is based on exploiting information regarding the mathematical form of the integrand. This enables higher degree of precision for the same number of integration points compared to standard quadrature use. The approach is general and can be applied to both consistent and lumped matrices of all element types. Here, we focus on the consistent mass matrix of the widely used 10-node tetrahedral element, and demonstrate the superiority of the new approach over conventional quadrature use. For example, we show that the new integration scheme enables a degree of precision 4 with 1 integration point compared to 11 points with conventional numerical integration. Also, our 4-points integration rule is practically equivalent to conventional numerical integration with 15 points.

1. Introduction

FE analysis is an indispensable tool being used today in almost every field of engineering, design, and research. The need in simulating increasingly larger structures with many degrees of freedom leads to a constant race aiming at reducing the computation time. This is achieved by improvements in hardware capabilities, but also in raising the code efficiency.

Any dynamic or transient simulation as well as modal-analyses of solid continua requires the calculation of the mass matrix. Conventionally, the mass matrix is computed using numeric integration with standard quadrature [1,2]. The mass matrix computational cost is proportional to the number of integration points. Thus, the motivation to formulate a numerical scheme that enables the same accuracy with a smaller number of integration points is clear. In this paper, we propose a semi-analytical approach for calculating the mass matrix. The method is based on the observation that the integrand of the mass matrix is composted from two multiplicative terms. The first is a simple polynomial expression that does not depend on the mesh, thus can be treated analytically. The second term is a mesh-dependent function which is handled by means of a simple polynomial approximation. In other words, we use information regarding the structure of the integrand in order to enhance the degree of precision of the numerical integration.

Similar concepts have been considered previously, but in a different context. For example, it has been shown that symbolic computations combined with code generation allow significant reduction of computation time compared to traditional quadrature use (e.g. [3–6]). Also, closed form integration has been used in order to calculate more efficiently the stiffness matrix of specific elements, including plane elements [7–10], 3-D bricks [8,11,12], triangles [13,14], and 3-D elastic and electro-elastic tetrahedral elements [14–19]. Closed form formulation was also used to formulate an accurate and robust 10-node tetrahedral Cosserat-Point Element (CPE) that does not exhibit numerical stiffness for nearly incompressible material [20]. In another study, a systematic approximation combined with analytical integration of the weak form enabled high-accuracy closed-form expressions for the natural frequencies of skewed parallelepipeds [21]. Hinnant [22] proposed a technique of numerical quadrature especially suited for quadrilateral and hexahedron p-version finite element matrices. This technique is based on separating the integrand into two parts, and numerically operating on each part separately.

In this paper, we propose a new integration approach specifically designed to compute the mass matrix of solid elements. By exploiting information regarding the mathematical structure of the integrand, we significantly improve the degree of precision of the numerical integration compared to standard quadrature use, for the same number of integration points. In other words, our approach provides a similar accuracy to standard quadrature using a smaller number of integration points, enabling lower computation cost. Our new approach is not limited to a specific element type, and can be applied

* Corresponding author.
E-mail address: givli@technion.ac.il (S. Givli).

http://dx.doi.org/10.1016/j.finel.2016.03.004
0168-874X/

to the calculation of consistent and lumped mass matrices [23]. Here, we thoroughly examine the consistent mass matrix of the widely used 10-node tetrahedral element. Accordingly, we organize the paper as follows: Section 2presents the general theoretical considerations. Two new approaches are discussed, namely the Optimization-Based (OB) and Semi-Analytical (SA) approaches. In Section 2.1, we propose a systematic procedure to obtain the unknown coefficients of the OB and SA approaches for the 10-node tetrahedral element. In Sections 3 and 4, we provide the necessary details and considerations in order to implement the OB and SA approaches, respectively, with 1, 4, 5, 8, and 10 integration points. In Section 5, we study the performance of the OB and SA approaches and compare their accuracy with that of conventional quadrature. Summary and main conclusions are discussed in Section 6.

2. Theoretical considerations

The consistent mass-matrix of a solid finite element is defined as $M_{ij} = \int_V \rho_0 N_i N_j \, dV$, where ρ_0 is the initial mass density, N_i are the element shape functions, and the integration is carried out over the initial volume of the element V [24]. By introducing the local coordinates of the element ξ, η, ζ, the mass matrix takes the form

$$M_{ij} = \int_{V_\square} N_i N_j \, (\rho_0 J) \, d\xi \, d\eta \, d\zeta. \tag{1}$$

Above, J is the Jacobian determinant of the transformation from the global to the local coordinate system, and $V_\square$ is the integration domain associated with the local coordinate system. The common approach to approximate (1) is to employ a numerical integration schemes, i.e. $M_{ij} = \sum_{p=1}^{n_p} w_p N_{ip} N_{jp} \rho_{0p} J_p$. Here, the index p indicates evaluation at an integration point, w_p is the corresponding weight, and n_p is the number of *integration points*. Thus, based on this "standard" (ST) approach, $M_{ij} \approx M_{ij}^{ST}$ with

$$M_{ij}^{ST} = M_{ijp}^{ST} \, \rho_{0p} J_p; \quad M_{ijp}^{ST} = w_p N_{ip} N_{jp}. \tag{2}$$

In order to reduce the computational cost of the numerical integration, it is of interest to use the minimum number of integration points that still provides the sufficient accuracy. We propose to reduce the number of integration points by adopting a semi-analytical (SA) approach by rewriting (1) as follows. First, we note that the integrand can be separated into two multiplicative terms. The first, $N_i N_j$, is a simple polynomial expression that does not depend on the mesh (the locations of the element nodes), thus can be treated analytically. The second term, $\rho_0 J$ is a mesh-dependent function of the local element coordinates. Hence, we approximate $\rho_0 J$ with

$$\rho_0 J \approx \sum_{p=1}^{n_p} \hat{N}_p(\xi,\eta,\zeta) \cdot (\rho_0 J)|_{(\xi_p,\eta_p,\zeta_p)} = \sum_{p=1}^{n_p} \hat{N}_p \, \rho_{0p} J_p. \tag{3}$$

Note that the ansatz functions $\hat{N}_p$ and corresponding sampling points in (3) differ from the element shape functions N_i and integration points associated with the ST approach (1). Plugging (3) into (1) yields the SA approximation for the mass matrix $M_{ij} \approx M_{ij}^{SA}$ with

$$M_{ij}^{SA} = M_{ijp}^{SA} \rho_{0p} J_p; \quad M_{ijp}^{SA} = \int_{V_\square} N_i N_j \hat{N}_p \, d\xi \, d\eta \, d\zeta. \tag{4}$$

Note the similar mathematical form of the ST and SA approximations in (2)a and (4)a, which only differ in the way the coefficient matrix M_{ijp} is defined. Importantly, this matrix does not depend on the mesh, thus can be precomputed. In addition, the coefficients matrix M_{ijp} has to be computed only once. In other words, once these coefficients are set they are to be used for all elements (of the same type). In this sense, the use of the coefficients M_{ijp}^{SA} is similar to the implementation of weights w_p in the standard approach. The main difference in terms of implementation is that in the standard approach each entry of M_{ijp}^{ST} requires two multiplications of the pre-stored quantities w_p and N_{ip}, e.g. $M_{123}^{ST} = w_3 N_{13} N_{23}$; on the other hand, the entries of M_{ijp}^{SA} are pre-stored as is. Consequently, the SA approach requires slightly more memory but fewer multiplications. These differences in resources are minor, and, practically, the two approaches require a similar computational cost for the same number of points, n_p. However, we show in the following sections that the SA approach enables accuracy similar to that of the standard approach with significantly smaller number of integration points.

Following the mathematical structure of (2)a and (4)a, we propose a third approach which we term "optimization-based" (OB) approach, where $M_{ij} \approx M_{ij}^{OB}$ with

$$M_{ij}^{OB} = M_{ijp}^{OB} \rho_{0p} J_p. \tag{5}$$

The coefficient matrix M_{ijp}^{OB} is pre-computed, as in the SA approach. However, here, it is calculated by means of an optimization process which aims to maximize the accuracy of M_{ij}^{OB}. In-depth discussion of this procedure is provided in Section 3.

In summary, we present in this paper two new approaches, namely the SA and OB approaches. We develop a systematic method to obtain the coefficient matrices, and compare the accuracy of our new rule with that of the standard (ST) approach, which is the prevalent method in commercial finite-element software. Finally, we emphasize that the above considerations are general, and not specific to a particular element type. For specificity, we consider in what follows the well-known and widely used ten-node tetrahedral element.

2.1. Regression procedure: ten-node tetrahedral element

The OB and SA approaches require specification of the sampling points along with the coefficients M_{ijp}^{OB} (for the OB) and the ansatz functions $\hat{N}_p$ (for the SA). Here, we view these as "unknowns" and look for the most suitable ones, namely those that make the SA and OB approximations most accurate for a prescribed number of points. Below, we propose a systematic procedure aiming at finding these unknowns. This procedure involves the following steps: (i) Generation of a set of random elements, representing the element population generated by automatic mesh generators, which we term "learning set". (ii) Formulation of an objective function which reflects the average error with respect to the exact mass matrix of each element in the learning set. (iii) Determination of the unknowns by minimization of an objective function.

For specificity, we consider below the ten-node tetrahedral element. Standard definitions of this element are recalled in Appendix A. In addition, without loss of generality, we consider the case of constant initial density, $\rho_0 = 1$.

2.1.1. Generating the learning set

The learning set includes random elements which are generated as follows. We begin with the parent element, shown in Fig. 1. The nodes of the next element are randomly relocated by changing each degree of freedom by a random number uniformly distributed between $[-\delta, \delta]$. The value of δ is linearly increased from zero for the first element to $\delta_{\max}$ for the last element. Without loss of generality, the locations of nodes 1 and 5 are kept unchanged. This procedure repeats until the entire set is generated. In order to avoid unphysical elements we choose $\delta_{\max} = 0.12$, and exclude from the learning set elements having a negative metric, $J < 0$. Examples of such randomly generated elements are shown in Fig. 1b–d, and a flow chart summarizing the above procedure is