The Gradient and the Hessian of the Distance between Point and Triangle in 3D

Article *in* Algorithms · July 2018

DOI: 10.3390/a11070104

3 authors:

Igor Gribanov
Memorial University of Newfoundland
9 PUBLICATIONS **73** CITATIONS
SEE PROFILE

Rocky Taylor
Memorial University of Newfoundland
103 PUBLICATIONS **794** CITATIONS
SEE PROFILE

Robert Sarracino
Centre for Cold Ocean Resources Engineering (C-CORE)
37 PUBLICATIONS **258** CITATIONS
SEE PROFILE

algorithms

MDPI

Article

The Gradient and the Hessian of the Distance between Point and Triangle in 3D

Igor Gribanov *, Rocky Taylor and Robert Sarracino

Faculty of Engineering and Applied Science, Memorial University of Newfoundland, 40 Arctic Ave, St. John's, NL A1B 3X7, Canada; rstaylor@mun.ca (R.T.); rsarracino@mun.ca (R.S.)

* Correspondence: ig1453@mun.ca

Received: 5 June 2018; Accepted: 10 July 2018; Published: 12 July 2018

Abstract: Computation of the distance between point and triangle in 3D is a common task in numerical analysis. The input values of the algorithm are coordinates of three points of the triangle and one point from which the distance is determined. An existing algorithm is extended to compute the gradient and the Hessian of that distance with respect to coordinates of involved points. Derivation of exact expressions for gradient and Hessian is presented, and numerical accuracy is evaluated for various cases. The algorithm has O(1) time and space complexity. The included open-source code may be used in applications where derivatives of point-triangle distance are required.

Keywords: point-triangle distance; gradient; Hessian

1. Introduction

The algorithm developed by Eberly [1] evaluates the distance function between a point and a triangle in 3D. The input parameters are coordinates of four points, which are involved in the point-triangle setup. Three points are vertices of the triangle, with the remaining point being the one to which the distance is computed. The motivation for this work is the need to obtain the gradient and the Hessian of that distance with respect to the input variables. Each of the four input points possesses three coordinates, giving 12 independent variables for the distance function. The distance function, accordingly, has 12 first derivatives and 12×12 second derivatives, the components of the symmetric Hessian matrix.

Our approach is to differentiate the distance function provided by Eberly [1] and extend that algorithm with evaluation of first and second derivatives. The characteristics of the algorithm remain the same, i.e., the time and space complexity of the algorithm is O(1). A linear array of first derivatives and a 2D array of second derivatives are added to the output. The main contribution of the authors is the derivation of the expressions for the gradient and the Hessian of the distance. A potential application of this algorithm is in finding penetration penalty forces and their differentials for colliding polygonal objects in finite element (FE) simulations.

Numerical simulations of mechanical systems often involve contact interactions, and the penalty method is a common way of addressing this problem [2]. Implicit FE approaches rely on calculating derivatives of the forces generated in collisions, which in turn require first and second derivatives of the distance function. The FE simulation ARCSim [3] relies on penalty contact resolution for modeling deformation and fracture. ARCSim includes an approximation technique for obtaining the gradient and the Hessian of the distance function based on the surface normal of the interacting object. The approximation is not always accurate, which results in reduction of time steps, which sometimes halts the simulation.

In another method described by Fisher and Lin [4], the distance from the interior point to the surface of the object is precomputed on the nodes of interior polygonal mesh, representing a distance

field. This discretized field allows one to estimate the spacial gradient of the distance. The accuracy is limited by the resolution of the mesh. In general, such approximation techniques are inaccurate, and there is a need for developing and utilizing the exact formula.

2. Mathematical Formulation

Figure 1 shows the point $\mathbf{p}_0(x_0, x_1, x_2)$ and the triangle with vertices $\mathbf{p}_1(x_3, x_4, x_5)$, $\mathbf{p}_2(x_6, x_7, x_8)$ and $\mathbf{p}_3(x_9, x_{10}, x_{11})$. The projection point $\mathbf{p}_c$ has the barycentric coordinates $\zeta_1, \zeta_2, \zeta_3$:

$$\mathbf{p}_c = \zeta_1 \mathbf{p}_1 + \zeta_2 \mathbf{p}_2 + \zeta_3 \mathbf{p}_3. \tag{1}$$

The squared distance between the point and the triangle is

$$s = |\mathbf{p}_0 - \mathbf{p}_c|^2 = \sum_{k=0}^{2} \left(x_k - \sum_{m=1}^{3} \zeta_m x_{k+3m} \right)^2. \tag{2}$$

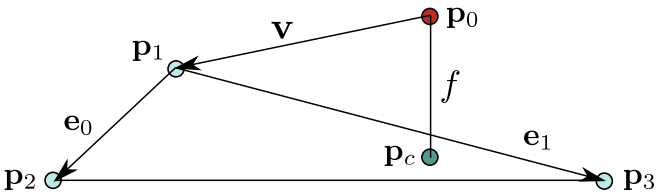


Figure 1. Distance f is determined between the point $\mathbf{p}_0$ and its projection $\mathbf{p}_c$ onto the triangle $\mathbf{p}_1\mathbf{p}_2\mathbf{p}_3$. Vectors $\mathbf{e}_0$, $\mathbf{e}_1$ and $\mathbf{v}$ are used in calculating f.

To obtain the gradient of s, Expression (2) is differentiated:

$$\frac{\partial s}{\partial x_i} = 2 \sum_{k=0}^{2} \left(x_k - \sum_{m=1}^{3} \zeta_m x_{k+3m} \right) \left(\delta_{(k)(i)} - \sum_{m=1}^{3} \left(\zeta_m \delta_{(k+3m)(i)} + x_{k+3m} \frac{\partial \zeta_m}{\partial x_i} \right) \right), \tag{3}$$

where δ_{ij} denotes the Kronecker symbol. Expressions for the barycentric coordinates ζ_m are given by Eberly [1], who introduces the following scalar coefficients:

$$a = \mathbf{e}_0 \cdot \mathbf{e}_0, b = \mathbf{e}_0 \cdot \mathbf{e}_1, c = \mathbf{e}_1 \cdot \mathbf{e}_1, d = \mathbf{e}_0 \cdot \mathbf{v}, e = \mathbf{e}_1 \cdot \mathbf{v}, \tag{4}$$

where $\mathbf{e}_0 = \mathbf{p}_2 - \mathbf{p}_1$, $\mathbf{e}_1 = \mathbf{p}_3 - \mathbf{p}_1$ and $\mathbf{v} = \mathbf{p}_1 - \mathbf{p}_0$. The barycentric coordinates are then

$$\zeta_1 = 1 - (\zeta_2 + \zeta_3); \zeta_2 = \frac{be - cd}{ac - b^2}; \zeta_3 = \frac{bd - ae}{ac - b^2}. \tag{5}$$

Expression (5) is used when $\mathbf{p}_c$ belongs to the interior region of the triangle. Cases when $\mathbf{p}_c$ belongs to an edge or coincides with one of the vertices are discussed in the next section. To obtain derivatives of ζ_2 and ζ_3, Expression (5) is differentiated:

$$\frac{\partial \zeta_2}{\partial x_i} = \frac{1}{\Delta} \left(e \frac{\partial b}{\partial x_i} + b \frac{\partial e}{\partial x_i} - d \frac{\partial c}{\partial x_i} - c \frac{\partial d}{\partial x_i} \right) - \frac{1}{\Delta^2} \frac{\partial \Delta}{\partial x_i} (be - cd), \tag{6}$$

$$\frac{\partial \zeta_3}{\partial x_i} = \frac{1}{\Delta} \left(d \frac{\partial b}{\partial x_i} + b \frac{\partial d}{\partial x_i} - e \frac{\partial a}{\partial x_i} - a \frac{\partial e}{\partial x_i} \right) - \frac{1}{\Delta^2} \frac{\partial \Delta}{\partial x_i} (bd - ae), \tag{7}$$

where $\Delta = ac - b^2$ and $\frac{\partial \Delta}{\partial x_i} = a\frac{\partial c}{\partial x_i} + c\frac{\partial a}{\partial x_i} - 2b\frac{\partial b}{\partial x_i}$. Derivatives of ζ_1 are

$$\frac{\partial \zeta_1}{\partial x_i} = -\left(\frac{\partial \zeta_2}{\partial x_i} + \frac{\partial \zeta_3}{\partial x_i}\right). \tag{8}$$

Coefficients a, b, c, d, e (4) can be expanded in terms of x_i:

$$\begin{aligned}
a &= (x_6 - x_3)^2 + (x_7 - x_4)^2 + (x_8 - x_5)^2,\\
b &= (x_9 - x_3)(x_6 - x_3) + (x_{10} - x_4)(x_7 - x_4) + (x_{11} - x_5)(x_8 - x_5),\\
c &= (x_9 - x_3)^2 + (x_{10} - x_4)^2 + (x_{11} - x_5)^2,\\
d &= (x_3 - x_0)(x_6 - x_3) + (x_4 - x_1)(x_7 - x_4) + (x_5 - x_2)(x_8 - x_5),\\
e &= (x_9 - x_3)(x_3 - x_0) + (x_{10} - x_4)(x_4 - x_1) + (x_{11} - x_5)(x_5 - x_2).
\end{aligned} \tag{9}$$

Then, gradients of Equation (9) are

$$\begin{aligned}
\left[\frac{\partial a}{\partial x_i}\right] &= 2 \times [0, 0, 0, x_3 - x_6, x_4 - x_7, x_5 - x_8, x_6 - x_3, x_7 - x_4, x_8 - x_5, 0, 0, 0],\\
\left[\frac{\partial b}{\partial x_i}\right] &= [0, 0, 0, 2x_3 - x_6 - x_9, 2x_4 - x_7 - x_{10}, 2x_5 - x_8 - x_{11}, x_9 - x_3, x_{10} - x_4,\\
&x_{11} - x_5, x_6 - x_3, x_7 - x_4, x_8 - x_5],\\
\left[\frac{\partial c}{\partial x_i}\right] &= 2 \times [0, 0, 0, x_3 - x_9, x_4 - x_{10}, x_5 - x_{11}, 0, 0, 0, x_9 - x_3, x_{10} - x_4, x_{11} - x_5],\\
\left[\frac{\partial d}{\partial x_i}\right] &= [x_3 - x_6, x_4 - x_7, x_5 - x_8, x_0 - 2x_3 + x_6, x_1 - 2x_4 + x_7, x_2 - 2x_5 + x_8,\\
&x_3 - x_0, x_4 - x_1, x_5 - x_2, 0, 0, 0],\\
\left[\frac{\partial e}{\partial x_i}\right] &= [x_3 - x_9, x_4 - x_{10}, x_5 - x_{11}, x_0 - 2x_3 + x_9, x_1 - 2x_4 + x_{10},\\
&x_2 - 2x_5 + x_{11}, 0, 0, 0, x_3 - x_0, x_4 - x_1, x_5 - x_2].
\end{aligned} \tag{10}$$

Substituting Equation (10) into Equations (6) and (7) allows for obtaining $\frac{\partial s}{\partial x_i}$ in terms of x_i.

The second derivatives of s are procured in the same manner. First, Expression (3) is differentiated to obtain

$$\frac{\partial^2 s}{\partial x_i \partial x_j} = 2\sum_{k=0}^{2}\left((\xi')^2 + 2\xi\xi''\right), \tag{11}$$

where

$$\xi = x_k - \sum_{m=1}^{3}\zeta_m x_{k+3m}, \tag{12}$$

$$\xi' = \delta_{(k)(i)} - \sum_{m=1}^{3}\left(\zeta_m \delta_{(k+3m)(i)} + x_{k+3m}\frac{\partial \zeta_m}{\partial x_i}\right), \tag{13}$$

$$\xi'' = -\sum_{m=1}^{3}\left(\frac{\partial \zeta_m}{\partial x_j}\delta_{(k+3m)(i)} + \frac{\partial \zeta_m}{\partial x_i}\delta_{(k+3m)(j)} + x_{k+3m}\frac{\partial^2 \zeta_m}{\partial x_i \partial x_j}\right). \tag{14}$$

Second derivatives of barycentric coordinates are obtained by differentiating Expressions (6) and (7):

$$\begin{aligned}\frac{\partial^2\zeta_2}{\partial x_i\partial x_j} &= \frac{1}{\Delta}\left(\frac{\partial b}{\partial x_j}\frac{\partial e}{\partial x_i}+\frac{\partial b}{\partial x_i}\frac{\partial e}{\partial x_j}-\frac{\partial c}{\partial x_j}\frac{\partial d}{\partial x_i}-\frac{\partial c}{\partial x_i}\frac{\partial d}{\partial x_j}\right)\\ &+\frac{1}{\Delta}\left(e\frac{\partial^2 b}{\partial x_j\partial x_i}+b\frac{\partial^2 e}{\partial x_j\partial x_i}-d\frac{\partial^2 c}{\partial x_j\partial x_i}-c\frac{\partial^2 d}{\partial x_j\partial x_i}\right)\\ &+\frac{1}{\Delta^2}\frac{\partial\Delta}{\partial x_j}\left(d\frac{\partial c}{\partial x_i}+c\frac{\partial d}{\partial x_i}-e\frac{\partial b}{\partial x_i}-b\frac{\partial e}{\partial x_i}\right)\\ &+\frac{1}{\Delta^2}\frac{\partial\Delta}{\partial x_i}\left(d\frac{\partial c}{\partial x_j}+c\frac{\partial d}{\partial x_j}-e\frac{\partial b}{\partial x_j}-b\frac{\partial e}{\partial x_j}\right)\\ &+\left(\frac{2}{\Delta^3}\frac{\partial\Delta}{\partial x_i}\frac{\partial\Delta}{\partial x_j}-\frac{1}{\Delta^2}\frac{\partial^2\Delta}{\partial x_i\partial x_j}\right)(be-cd),\end{aligned} \tag{15}$$

$$\begin{aligned}\frac{\partial^2\zeta_3}{\partial x_i\partial x_j} &= \frac{1}{\Delta}\left(\frac{\partial b}{\partial x_j}\frac{\partial d}{\partial x_i}+\frac{\partial b}{\partial x_i}\frac{\partial d}{\partial x_j}-\frac{\partial a}{\partial x_j}\frac{\partial e}{\partial x_i}-\frac{\partial a}{\partial x_i}\frac{\partial e}{\partial x_j}\right)\\ &+\frac{1}{\Delta}\left(d\frac{\partial^2 b}{\partial x_j\partial x_i}+b\frac{\partial^2 d}{\partial x_j\partial x_i}-e\frac{\partial^2 a}{\partial x_j\partial x_i}-a\frac{\partial^2 e}{\partial x_j\partial x_i}\right)\\ &+\frac{1}{\Delta^2}\frac{\partial\Delta}{\partial x_j}\left(e\frac{\partial a}{\partial x_i}+a\frac{\partial e}{\partial x_i}-d\frac{\partial b}{\partial x_i}-b\frac{\partial d}{\partial x_i}\right)\\ &+\frac{1}{\Delta^2}\frac{\partial\Delta}{\partial x_i}\left(e\frac{\partial a}{\partial x_j}+a\frac{\partial e}{\partial x_j}-d\frac{\partial b}{\partial x_j}-b\frac{\partial d}{\partial x_j}\right)\\ &+\left(\frac{2}{\Delta^3}\frac{\partial\Delta}{\partial x_i}\frac{\partial\Delta}{\partial x_j}-\frac{1}{\Delta^2}\frac{\partial^2\Delta}{\partial x_i\partial x_j}\right)(bd-ae),\end{aligned} \tag{16}$$

where $\frac{\partial^2\Delta}{\partial x_i\partial x_j}=\frac{\partial a}{\partial x_j}\frac{\partial c}{\partial x_i}+\frac{\partial a}{\partial x_i}\frac{\partial c}{\partial x_j}+c\frac{\partial^2 a}{\partial x_i\partial x_j}+a\frac{\partial^2 c}{\partial x_i\partial x_j}-2\frac{\partial b}{\partial x_i}\frac{\partial b}{\partial x_j}-2b\frac{\partial^2 b}{\partial x_i\partial x_j}$. Similarly to Equation (8):

$$\frac{\partial^2\zeta_1}{\partial x_i\partial x_j}=-\left(\frac{\partial^2\zeta_2}{\partial x_i\partial x_j}+\frac{\partial^2\zeta_3}{\partial x_i\partial x_j}\right). \tag{17}$$

Second derivatives of the coefficients a, b, c, d, e are constants that are included in Appendix A. The first and the second derivatives of the distance $f=\sqrt{s}$ are then expressed as:

$$\frac{\partial f}{\partial x_i}=\frac{1}{2\sqrt{s}}\frac{\partial s}{\partial x_i}, \tag{18}$$

$$\frac{\partial^2 f}{\partial x_i\partial x_j}=\frac{1}{2\sqrt{s}}\frac{\partial^2 s}{\partial x_i\partial x_j}-\frac{1}{4s^{3/2}}\frac{\partial s}{\partial x_i}\frac{\partial s}{\partial x_j}. \tag{19}$$

3. Point-Edge and Point-Point Cases

If the projection of $\mathbf{p}_0$ falls outside of the interior area of the triangle (Figure 2), the closest point $\mathbf{p}_c$ coincides with one of the vertices or belongs to one of the edges of the triangle. In the point–point case (Figure 3a), the squared distance is expressed as

$$s=|\mathbf{p}_0-\mathbf{p}_c|^2,$$

where $\mathbf{p}_c$ is one of $\mathbf{p}_1, \mathbf{p}_2$ or $\mathbf{p}_3$. The derivatives of s with respect to x_i are obtained trivially, and the details are not discussed here.

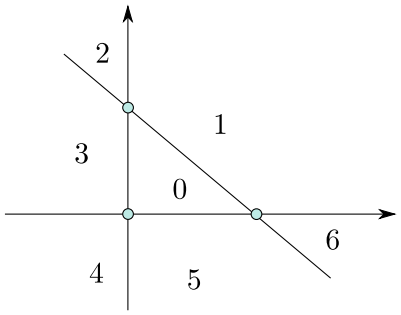
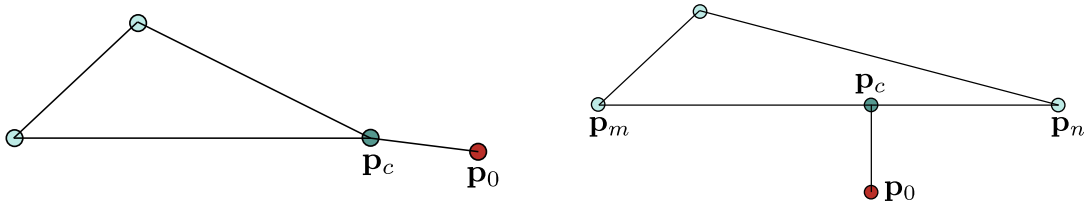


Figure 2. Partitioning of the plane by the triangle domain. Different domains are distinguished by the values of barycentric coordinates of the projection of $\mathbf{p}_0$ onto the plane of the triangle.



(**a**) point–point case (**b**) point-edge case

Figure 3. The closest point $\mathbf{p}_c$ may coincide with (**a**) one of triangle's vertices or (**b**) belong to one of the edges.

In the point-edge case, one of the barycentric coordinates of $\mathbf{p}_c$ is zero, which simplifies Expression (2). Let $\mathbf{p}_m$ and $\mathbf{p}_n$ be the vertices of the edge, to which the closest point $\mathbf{p}_c$ belongs (Figure 3b). Then $\mathbf{p}_c$ is expressed as the linear combination:

$$\mathbf{p}_c = (1 - \zeta_k)\mathbf{p}_m + \zeta_k \mathbf{p}_n,$$

where ζ_k is the non-zero barycentric coordinate of $\mathbf{p}_c$. This coordinate can be found as [5]:

$$\zeta_k = \frac{(\mathbf{p}_m - \mathbf{p}_0) \cdot (\mathbf{p}_n - \mathbf{p}_m)}{|\mathbf{p}_n - \mathbf{p}_m|^2}. \tag{20}$$

Similarly to the previous derivations, Expression (20) can be differentiated to obtain the first and the second derivatives of the ζ_k, which are then substituted into Expressions (3) and (11).

4. Algorithm and Testing

The calculations are implemented in double-precision floating-point arithmetic in C, and the tests are performed with squared distance s and its first and second derivatives. The first step is to determine the partition to which $\mathbf{p}_c$ belongs (Figure 2). This step coincides with the original point-triangle algorithm [1], but subsequent calculations are extended to evaluate $\frac{\partial s}{\partial x_i}$ and $\frac{\partial^2 s}{\partial x_i \partial x_j}$. If $\mathbf{p}_c$ belongs to partitions 1, 3, 5, then the point-line algorithm is invoked. For partitions 2, 4, 6, point–point calculations are performed. For partition 0, the point-plane algorithm is used.

For testing, 10 million cases were generated, about 2/3 of which are random coordinates that come from the uniform distribution in the range $[-1, 1]$. The remaining cases come from the finite element simulation of fracture, where the point $\mathbf{p}_c$ is often close to one of the triangle's vertices. Most of the cases that come from the simulation have a low ratio of the distance to the shortest edge of the triangle, in the range between 10^{-8} and 10^{-5}. Such test cases result in a lower accuracy of the final answer than the random arrangements of points.

To evaluate the accuracy of the proposed algorithm, calculations are first performed using arbitrary precision arithmetic with at least 17 digits calculated precisely. These results are denoted s_p, $\frac{\partial s_p}{\partial x_i}$, $\frac{\partial^2 s_p}{\partial x_i \partial x_j}$ and are compared to the results obtained in floating-point arithmetic s_c, $\frac{\partial s_c}{\partial x_i}$, $\frac{\partial^2 s_c}{\partial x_i \partial x_j}$. Relative errors are computed separately for the squared distance, its first derivatives and its second derivatives:

$$E_0 = \left|(s_c - s_p)/s_p\right|, \tag{21}$$

$$E_1 = \max_i \left| \left(\frac{\partial s_c}{\partial x_i} - \frac{\partial s_p}{\partial x_i} \right) / \frac{\partial s_p}{\partial x_i} \right|, \tag{22}$$

$$E_2 = \max_{i,j} \left| \left(\frac{\partial^2 s_c}{\partial x_i \partial x_j} - \frac{\partial^2 s_p}{\partial x_i \partial x_j} \right) / \frac{\partial^2 s_p}{\partial x_i \partial x_j} \right|. \tag{23}$$

E_0 is the relative error of the squared distance and is used as a baseline to compare with the relative errors of the first and second derivatives E_1 and E_2. Ideally, the values of E_0, E_1 and E_2 would have the same order of magnitude. However, due to the large number of algebraic operations, the accuracy of the calculation of the second derivative is lower. The values E_0, E_1 and E_2 cover a wide range of values. In about half of the cases, the precision for calculating second derivatives is better than 10^{-13}. However, the practical interest lies in investigating the worst cases because one incorrect calculation could affect a whole scientific study.

Discussion

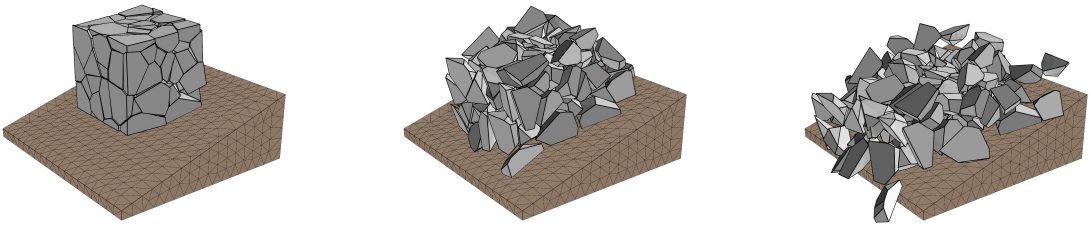
The highest errors among all test cases are shown in Table 1. The values come from separate test cases: $E_{0_{max}}$ is the maximum error for s, $E_{1_{max}}$ is the maximum error for $\frac{\partial s_c}{\partial x_i}$ and $E_{2_{max}}$ is the maximum error for $\frac{\partial^2 s_c}{\partial x_i \partial x_j}$. $E_{2_{max}}$ is higher than $E_{0_{max}}$ by two orders of magnitude, which is a good result, considering that it is the least accurate of 10 million tests. The tests show that the precision is adequate for all cases, including the ones form the collision simulation.

Table 1. Relative errors for squared distance and its first and second derivatives. The maximum values from 10 million test cases are shown.

Error Measure	Value
$E_{0_{max}}$	3.78×10^{-5}
$E_{1_{max}}$	2.75×10^{-5}
$E_{2_{max}}$	1.27×10^{-3}

Cases with the lowest accuracy usually correspond to the variables whose absolute values are very small, and computer simulations are often robust against such cases. For example, results of the grain interaction simulation where the current algorithm is applied are shown on Figure 4. The simulation advances with large time steps even when multiple fragments interact with each other.

The ratio between the distance and the largest edge of the triangle is one of the factors that affect the precision. When this ratio drops below 10^{-10}, the accuracy of the result is likely to deteriorate. The problem of the round-off error is common in numerical analysis and should be addressed properly. If the influence of the round-off error is suspected when applying this algorithm, additional testing should be performed. In some cases, calculations can be performed with arbitrary-precision arithmetic to yield accurate results.



(**a**) step 16, $t = 0.17$ (**b**) step 40, $t = 0.23$ (**c**) step 66, $t = 0.30$

Figure 4. Simulation of falling and colliding grains performed with implicit finite element method.

5. Conclusions

The presented algorithm has O(1) time and space complexity. Only static memory allocation takes place. The algorithm has several branches that evaluate algebraic expressions sequentially, with each branch completing in constant time. The main contribution of the authors is the derivation of the exact formulae for the gradient and the Hessian of the distance function. Additionally, testing of the reference implementation was performed to ensure that adequate precision is met. The proposed algorithm may be used in applications where point-triangle distance derivatives are required. The potential future work may include precision testing for more complex cases. The C code is available as open-source [6] and may be modified by the research community as needed.

Author Contributions: Conceptualization, I.G.; Software, I.G.; Writing—Original Draft Preparation, I.G.; Writing—Review and Editing, R.T. and R.S.; Supervision, R.T.

Funding: This research was funded by the Natural Sciences and Engineering Research Council (NSERC) of Canada and the Research Development Corporation of Newfoundland and Labrador (RDC).

Acknowledgments: The authors thank C-CORE for providing computing resources, office space, and creating productive working environment.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Second Derivatives of Coefficients a, b, c, d, e

$$\left[\frac{\partial^2 a}{\partial x_i \partial x_j}\right] = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{A1}$$

$$\left[\frac{\partial^2 b}{\partial x_i \partial x_j}\right] = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & -1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}, \tag{A2}$$

$$\left[\frac{\partial^2 c}{\partial x_i \partial x_j}\right] = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & -2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix}, \tag{A3}$$

$$\left[\frac{\partial^2 d}{\partial x_i \partial x_j}\right] = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -2 & 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{A4}$$

$$\left[\frac{\partial^2 e}{\partial x_i \partial x_j}\right] = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{A5}$$

References

1. Eberly, D. Distance between Point and Triangle in 3D. 1999. Available online: https://www.geometrictools.com/Documentation/DistancePoint3Triangle3.pdf.pdf (accessed on 9 July 2018).
2. Laursen, T.A. *Computational Contact and Impact Mechanics: Fundamentals of Modeling Interfacial Phenomena in Nonlinear Finite Element Analysis*; Springer Science & Business Media: Heidelberg/Berlin, Germany, 2013.
3. Pfaff, T.; Narain, R.; De Joya, J.M.; O'Brien, J.F. Adaptive tearing and cracking of thin sheets. *ACM Trans. Graph.* **2014**, *33*, 110. [CrossRef]
4. Fisher, S.; Lin, M.C. Fast penetration depth estimation for elastic bodies using deformed distance fields. In Proceedings of the Intelligent Robots and Systems, Maui, HI, USA, 29 October–3 November 2001; Volume 1, pp. 330–336.
5. Weisstein, E.W. Point-Line Distance–3-Dimensional. MathWorld–A Wolfram Web Resource. Available online: http://mathworld.wolfram.com/Point-LineDistance3-Dimensional.html (accessed on 10 July 2018).
6. Gribanov, I. Distance Derivatives, GitHub Repository. Available online: https://github.com/Spear520/dist/ (accessed on 10 July 2018).