

Implementing Baraff & Witkin’s Cloth Simulation

David Pritchard

May 2003, with minor updates April 2012

1 Introduction

In April 2002, I implemented a cloth simulator based on Baraff & Witkin’s paper [BW98] for a graduate animation course taught by Michiel van de Panne at the University of British Columbia. I later released that simulator as an open-source project called Freecloth. Over the following year, I improved the performance of the simulator and fixed some major bugs.

The initial version of Freecloth had some major bugs, and did not implement all parts of the original paper. Newer versions have eliminated the bugs in the simulator, but do not yet include all parts of the original paper.

I will assume that you are familiar with Baraff & Witkin’s paper, and I will not try to re-explain concepts which are clearly presented there. For my own work, I relied heavily upon a paper by Dean Macri [Mac00], who attempted to interpret and implement Baraff & Witkin’s paper. I will make reference to Macri’s notation and some of his formulas, but his paper is not necessary reading. I rederived all of the formulas found in Macri’s paper, and verified my derivation against Macri’s. Initially, I found a few minor typographical errors in his paper, but my formulas were generally the same.

As my implementation progressed, I found more major errors in Macri’s paper. The largest error occurred when he calculated the derivatives of the bend condition. I give a corrected version of this equation in (54) and (57).

In this paper, I will attempt to explain obstacles that I encountered while implementing Baraff & Witkin’s paper.

2 Notation

In this paper, I will use the symbol \mathbf{I} to denote an identity matrix. The symbol \mathbf{I}_s refers to a vector containing the s th column of the identity matrix. To denote the skew-symmetric matrix form of a vector cross product, I will use the notation

$$S(\mathbf{v}) = \begin{pmatrix} 0 & -\mathbf{v}_z & \mathbf{v}_y \\ \mathbf{v}_z & 0 & -\mathbf{v}_x \\ -\mathbf{v}_y & \mathbf{v}_x & 0 \end{pmatrix}$$

I will also define the vector $S_s(\mathbf{v})$ as the transpose of the s th row of $S(\mathbf{v})$. S_s is a column vector, but represents a row in the original S matrix. I will use a hat (e.g. $\hat{\mathbf{w}}_u$) to refer to a normalised vector of unit length, and omit the hat (e.g. \mathbf{w}_u) for an unnormalised vector. This is the opposite convention to that used in my original project report.

I will retain Baraff & Witkin’s notation and use \mathbf{x} to refer to the positions of points, not \mathbf{p} as in Macri’s paper. Furthermore, I will refer to the points of a triangle as \mathbf{x}_0 , \mathbf{x}_1 and \mathbf{x}_2 , and not use the i , j and k subscripts used by the other two papers. For the points of an edge, I will likewise use subscripts of 0, 1, 2, and 3 rather than the h , i , j and k subscripts used by Macri.

When referring to points, I will use \mathbf{x}_m and \mathbf{x}_n to refer to two possibly distinct points. I will use the subscript m_s to refer to the s th component of \mathbf{x}_m , and likewise the subscript n_t for the t th component of \mathbf{x}_n . I will also sometimes use subsubscripts of x , y and z to refer to the components.

I like to think of the derivatives in this paper in terms of their components. Both Baraff & Witkin and Macri differentiate with respect to a vector in

their papers, giving quantities like

$$\frac{\partial C}{\partial \mathbf{x}_m} = \begin{pmatrix} \frac{\partial C}{\partial \mathbf{x}_{m,x}} \\ \frac{\partial C}{\partial \mathbf{x}_{m,y}} \\ \frac{\partial C}{\partial \mathbf{x}_{m,z}} \end{pmatrix}$$

or

$$\frac{\partial \mathbf{v}}{\partial \mathbf{x}_m} = \begin{pmatrix} \frac{\partial v_x}{\partial \mathbf{x}_{m,x}} & \frac{\partial v_y}{\partial \mathbf{x}_{m,x}} & \frac{\partial v_z}{\partial \mathbf{x}_{m,x}} \\ \frac{\partial v_x}{\partial \mathbf{x}_{m,y}} & \frac{\partial v_y}{\partial \mathbf{x}_{m,y}} & \frac{\partial v_z}{\partial \mathbf{x}_{m,y}} \\ \frac{\partial v_x}{\partial \mathbf{x}_{m,z}} & \frac{\partial v_y}{\partial \mathbf{x}_{m,z}} & \frac{\partial v_z}{\partial \mathbf{x}_{m,z}} \end{pmatrix}$$

This type of formula is common in some areas of graphics; for example, the Jacobian matrix has this form. However, what happens when we take the second derivative of \mathbf{v} ? It will yield a third-order tensor, a $3 \times 3 \times 3$ linear algebraic quantity. However, this is unnecessarily complicated, especially for people (like me) who never covered tensors in their undergraduate education. In this paper, I will try to avoid differentiation with respect to a vector. Instead, I will do most of the work on a component-by-component basis, using only differentiation with respect to a scalar. Using this approach, I will only ever need to use vectors and scalars, not matrices or higher-order tensors. This makes operations such as cross-products and dot-products straightforward.

Some of Baraff & Witkin’s formulas involve taking a second derivative with respect to two vectors. For these, just remember the layout of the resulting matrix. For example,

$$\frac{\partial^2 C}{\partial \mathbf{x}_m \partial \mathbf{x}_n} = \begin{pmatrix} \frac{\partial C}{\partial \mathbf{x}_{m,x}} \mathbf{x}_{n,x} & \frac{\partial C}{\partial \mathbf{x}_{m,y}} \mathbf{x}_{n,x} & \frac{\partial C}{\partial \mathbf{x}_{m,z}} \mathbf{x}_{n,x} \\ \frac{\partial C}{\partial \mathbf{x}_{m,x}} \mathbf{x}_{n,y} & \frac{\partial C}{\partial \mathbf{x}_{m,y}} \mathbf{x}_{n,y} & \frac{\partial C}{\partial \mathbf{x}_{m,z}} \mathbf{x}_{n,y} \\ \frac{\partial C}{\partial \mathbf{x}_{m,x}} \mathbf{x}_{n,z} & \frac{\partial C}{\partial \mathbf{x}_{m,y}} \mathbf{x}_{n,z} & \frac{\partial C}{\partial \mathbf{x}_{m,z}} \mathbf{x}_{n,z} \end{pmatrix}$$

3 Condition Functions

Baraff & Witkin used condition functions as an indirect way of defining energy and forces. Their condition functions are almost the same as energy formulas, since $E = \frac{k}{2} \mathbf{C}^T \mathbf{C}$, where \mathbf{C} is the condition function. For the scalar condition functions (shear

and bend), $E = \frac{k}{2} C^2$, and the condition function is hence basically just the sqrt-root of the energy, except that it can be positive or negative.

The condition functions were defined explicitly in the paper. Energy and forces can also be easily calculated, if the derivatives of the condition functions are known. However, the paper didn’t provide explicit equations for the derivatives or second derivatives.

Macri made a good attempt at deriving the equations for the derivatives and second derivatives. I found two major shortcomings in his approach, however. First, he made reference to tensors and other mathematical constructs that aren’t really necessary for the derivation. Second, I found an error in his derivation for the bend condition.

3.1 Per-Triangle Quantities

To begin, I will define a few quantities that will help with explanation of the stretch and shear conditions. These quantities are defined for a triangle using points \mathbf{x}_0 , \mathbf{x}_1 , and \mathbf{x}_2 . The triangle’s area in u/v space is given by

$$a = \frac{1}{2} \left\| \begin{pmatrix} \Delta u_1 \\ \Delta v_1 \\ 0 \end{pmatrix} \times \begin{pmatrix} \Delta u_2 \\ \Delta v_2 \\ 0 \end{pmatrix} \right\| \quad (1)$$

The $\hat{\mathbf{w}}_u$ and $\hat{\mathbf{w}}_v$ vectors represent the directions of the u and v axes in world space, as seen within a single triangle. They are defined by

$$\hat{\mathbf{w}}_u = \frac{\mathbf{w}_u}{\|\mathbf{w}_u\|} \quad \hat{\mathbf{w}}_v = \frac{\mathbf{w}_v}{\|\mathbf{w}_v\|} \quad (2)$$

$$\mathbf{w}_u = \frac{(\mathbf{x}_1 - \mathbf{x}_0) \Delta v_2 - (\mathbf{x}_2 - \mathbf{x}_0) \Delta v_1}{\Delta u_1 \Delta v_2 - \Delta u_2 \Delta v_1} \quad (3)$$

$$\mathbf{w}_v = \frac{-(\mathbf{x}_1 - \mathbf{x}_0) \Delta u_2 + (\mathbf{x}_2 - \mathbf{x}_0) \Delta u_1}{\Delta u_1 \Delta v_2 - \Delta u_2 \Delta v_1} \quad (4)$$

The first derivatives of the unnormalised vectors \mathbf{w}_u and \mathbf{w}_v are given by

$$\frac{\partial \mathbf{w}_u}{\partial \mathbf{x}_m} = \frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{m_x}} \mathbf{I} \quad (5)$$

$$\frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{0_x}} = \frac{\Delta v_1 - \Delta v_2}{\Delta u_1 \Delta v_2 - \Delta u_2 \Delta v_1} \quad (6)$$

$$\frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{1_x}} = \frac{\Delta v_2}{\Delta u_1 \Delta v_2 - \Delta u_2 \Delta v_1} \quad (7)$$

$$\frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{2_x}} = \frac{-\Delta v_1}{\Delta u_1 \Delta v_2 - \Delta u_2 \Delta v_1} \quad (8)$$

$$\frac{\partial \mathbf{w}_v}{\partial \mathbf{x}_m} = \frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{m_x}} \mathbf{I} \quad (9)$$

$$\frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{0_x}} = \frac{\Delta u_2 - \Delta u_1}{\Delta u_1 \Delta v_2 - \Delta u_2 \Delta v_1} \quad (10)$$

$$\frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{1_x}} = \frac{-\Delta u_2}{\Delta u_1 \Delta v_2 - \Delta u_2 \Delta v_1} \quad (11)$$

$$\frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{2_x}} = \frac{\Delta u_1}{\Delta u_1 \Delta v_2 - \Delta u_2 \Delta v_1} \quad (12)$$

The second derivatives of \mathbf{w}_u and \mathbf{w}_v are all zero.

3.2 Stretch Condition

Baraff & Witkin define the stretch condition function as a vector \mathbf{C} . Two special parameters are defined for stretch, b_u and b_v . These represent the desired stretchiness of the cloth in the u and v directions. A b_u value below one will result in the cloth trying to compress in the u direction, while a b_u value above one will result in the cloth trying to stretch in the u direction.

$$\mathbf{C} = \begin{pmatrix} C_u \\ C_v \end{pmatrix} = \alpha \begin{pmatrix} \|\mathbf{w}_u\| - b_u \\ \|\mathbf{w}_v\| - b_v \end{pmatrix} \quad (13)$$

Baraff & Witkin defined $\alpha = a$, the area of the triangle. We define $\alpha = a^{\frac{3}{4}}$, for reasons described later.

The first and second derivatives of the stretch condition function are

$$\frac{\partial C_u}{\partial \mathbf{x}_m} = \alpha \frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{m_x}} \hat{\mathbf{w}}_u \quad (14)$$

$$\frac{\partial^2 C_u}{\partial \mathbf{x}_m \partial \mathbf{x}_n} = \frac{\alpha}{\|\mathbf{w}_u\|} \frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{m_x}} \frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{n_x}} (\mathbf{I} - \hat{\mathbf{w}}_u \hat{\mathbf{w}}_u^T) \quad (15)$$

$$\frac{\partial C_v}{\partial \mathbf{x}_m} = \alpha \frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{m_x}} \hat{\mathbf{w}}_v \quad (16)$$

$$\frac{\partial^2 C_v}{\partial \mathbf{x}_m \partial \mathbf{x}_n} = \frac{\alpha}{\|\mathbf{w}_v\|} \frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{m_x}} \frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{n_x}} (\mathbf{I} - \hat{\mathbf{w}}_v \hat{\mathbf{w}}_v^T) \quad (17)$$

These second derivatives exhibit some symmetry, since $\frac{\partial^2 C_u}{\partial \mathbf{x}_{m_x} \partial \mathbf{x}_{n_x}} = \frac{\partial^2 C_u}{\partial \mathbf{x}_{n_x} \partial \mathbf{x}_{m_x}}$. Consequently, the 3×3 matrix $\frac{\partial^2 C_u}{\partial \mathbf{x}_m \partial \mathbf{x}_n} = \frac{\partial^2 C_u}{\partial \mathbf{x}_m \partial \mathbf{x}_n}^T$. To avoid any tensors when applying Baraff & Witkin's equation (8) to the shear equations, remember that

$$\frac{\partial^2 \mathbf{C}(\mathbf{x})}{\partial \mathbf{x}_m \partial \mathbf{x}_n} \mathbf{C}(\mathbf{x}) = \frac{\partial^2 C_u}{\partial \mathbf{x}_m \partial \mathbf{x}_n} C_u + \frac{\partial^2 C_v}{\partial \mathbf{x}_m \partial \mathbf{x}_n} C_v \quad (18)$$

3.3 Shear Condition

Baraff & Witkin define the shear condition function as a scalar. This scalar is essentially the dot product of the u axis with the v axis in world space. If no shear is occurring, the axes are perpendicular and the condition function is zero. If shear is occurring, the condition function is equivalent to the cosine of the angle between them, weighted by the triangle's area in u / v space.

$$C = \alpha \mathbf{w}_u \cdot \mathbf{w}_v \quad (19)$$

Baraff & Witkin's condition assumes that \mathbf{w}_u and \mathbf{w}_v are approximately unit vectors. If b_u and b_v are both equal to one, then this approximation is a fair one. If, however, b_u or b_v are not equal to one, the approximation may give rise to some strange behaviour. I have not had a chance to investigate the effects of varying b_u or b_v .

Since we know the derivatives of \mathbf{w}_u and \mathbf{w}_v , taking the derivative of this condition function is trivial.

$$\frac{\partial C}{\partial \mathbf{x}_{m_s}} = \alpha \left(\frac{\partial \mathbf{w}_u}{\partial \mathbf{x}_{m_s}} \cdot \mathbf{w}_v + \mathbf{w}_u \cdot \frac{\partial \mathbf{w}_v}{\partial \mathbf{x}_{m_s}} \right) \quad (20)$$

$$= \alpha \left(\frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{m_x}} \mathbf{w}_{v_s} + \mathbf{w}_{u_s} \frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{m_x}} \right) \quad (21)$$

$$\begin{aligned} \frac{\partial^2 C}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} &= \alpha \left(\frac{\partial^2 \mathbf{w}_u}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \cdot \mathbf{w}_v + \frac{\partial \mathbf{w}_u}{\partial \mathbf{x}_{m_s}} \cdot \frac{\partial \mathbf{w}_v}{\partial \mathbf{x}_{n_t}} + \right. \\ &\quad \left. \frac{\partial \mathbf{w}_u}{\partial \mathbf{x}_{n_t}} \cdot \frac{\partial \mathbf{w}_v}{\partial \mathbf{x}_{m_s}} + \mathbf{w}_u \cdot \frac{\partial^2 \mathbf{w}_v}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \right) \quad (22) \\ &= \begin{cases} \alpha \left(\frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{m_x}} \frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{n_x}} + \frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{n_x}} \frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{m_x}} \right) & \text{if } s = t, \\ \mathbf{0} & \text{otherwise.} \end{cases} \quad (23) \end{aligned}$$

These second derivatives exhibit even more symmetry than the stretch condition. Like the stretch condition, $\frac{\partial^2 C}{\partial \mathbf{x}_m \partial \mathbf{x}_n} = \frac{\partial^2 C}{\partial \mathbf{x}_n \partial \mathbf{x}_m}^T$ but beyond that, the $\frac{\partial^2 C}{\partial \mathbf{x}_m \partial \mathbf{x}_n}$ is actually just an identity matrix multiplied by a scalar.

3.4 Bend Condition

Baraff & Witkin define the bend condition in terms of two adjoining triangles. I have labelled the unit normals of these triangles as $\hat{\mathbf{n}}^A$ and $\hat{\mathbf{n}}^B$, rather than Baraff & Witkin's \mathbf{n}_1 or \mathbf{n}_2 . The unit vector parallel to their common edge is labelled $\hat{\mathbf{e}}$. Quantities associated with the normals or the common edge receive a superscript A , B or e .

$$\hat{\mathbf{n}}^A = \frac{\mathbf{n}^A}{\|\mathbf{n}^A\|} \quad \hat{\mathbf{n}}^B = \frac{\mathbf{n}^B}{\|\mathbf{n}^B\|} \quad \hat{\mathbf{e}} = \frac{\mathbf{e}}{\|\mathbf{e}\|} \quad (24)$$

$$\mathbf{n}^A = (\mathbf{x}_2 - \mathbf{x}_0) \times (\mathbf{x}_1 - \mathbf{x}_0) \quad (25)$$

$$\mathbf{n}^B = (\mathbf{x}_1 - \mathbf{x}_3) \times (\mathbf{x}_2 - \mathbf{x}_3) \quad (26)$$

$$\mathbf{e} = \mathbf{x}_1 - \mathbf{x}_2 \quad (27)$$

The scalar bend condition C is defined quite simply as the angle between the two edges, θ .

$$\cos \theta = \hat{\mathbf{n}}^A \cdot \hat{\mathbf{n}}^B \quad (28)$$

$$\sin \theta = (\hat{\mathbf{n}}^A \times \hat{\mathbf{n}}^B) \cdot \hat{\mathbf{e}} \quad (29)$$

$$C = \theta = \arctan \frac{\sin \theta}{\cos \theta} \quad (30)$$

Macri defined C using the arccos function, which is incorrect and will only yield positive values for θ . My definition of C will yield both positive and negative values if implemented using the `atan2` function.

In order to find the derivatives of C , we need the derivatives of the quantities seen so far. The first and second derivatives of \mathbf{n}^A , \mathbf{n}^B and \mathbf{e} are shown below, using auxiliary variables \mathbf{q}_m . The derivatives of \mathbf{q}_m are straightforward to derive.

$$\frac{\partial \mathbf{n}^A}{\partial \mathbf{x}_{m_s}} = S_s(\mathbf{q}_m^A) \quad (31)$$

$$\frac{\partial \mathbf{n}^B}{\partial \mathbf{x}_{m_s}} = S_s(\mathbf{q}_m^B) \quad (32)$$

$$\frac{\partial \mathbf{e}}{\partial \mathbf{x}_{m_s}} = q_m^e \mathbf{I}_s \quad (33)$$

$$\mathbf{q}^A = \{\mathbf{x}_2 - \mathbf{x}_1, \mathbf{x}_0 - \mathbf{x}_2, \mathbf{x}_1 - \mathbf{x}_0, \mathbf{0}\} \quad (34)$$

$$\mathbf{q}^B = \{\mathbf{0}, \mathbf{x}_2 - \mathbf{x}_3, \mathbf{x}_3 - \mathbf{x}_1, \mathbf{x}_1 - \mathbf{x}_2\} \quad (35)$$

$$q^e = \{0, 1, -1, 0\} \quad (36)$$

$$\frac{\partial^2 \mathbf{n}^A}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} = S_s \left(\frac{\partial \mathbf{q}_m^A}{\partial \mathbf{x}_{n_t}} \right) \quad (37)$$

$$\frac{\partial^2 \mathbf{n}^B}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} = S_s \left(\frac{\partial \mathbf{q}_m^B}{\partial \mathbf{x}_{n_t}} \right) \quad (38)$$

$$\frac{\partial^2 \mathbf{e}}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} = \mathbf{0} \quad (39)$$

Using Baraff & Witkin's assumption that the normal and edge vectors have constant magnitude,

$$\frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{m_s}} = \frac{1}{\|\mathbf{n}^A\|} \frac{\partial \mathbf{n}^A}{\partial \mathbf{x}_{m_s}} \quad (40)$$

$$\frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{m_s}} = \frac{1}{\|\mathbf{n}^B\|} \frac{\partial \mathbf{n}^B}{\partial \mathbf{x}_{m_s}} \quad (41)$$

$$\frac{\partial \hat{\mathbf{e}}}{\partial \mathbf{x}_{m_s}} = \frac{1}{\|\mathbf{e}\|} \frac{\partial \mathbf{e}}{\partial \mathbf{x}_{m_s}} \quad (42)$$

$$\frac{\partial^2 \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} = \frac{1}{\|\mathbf{n}^A\|} \frac{\partial^2 \mathbf{n}^A}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \quad (43)$$

$$\frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{m_s}} \mathbf{x}_{n_t} = \frac{1}{\|\mathbf{n}^B\|} \frac{\partial^2 \mathbf{n}^B}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \quad (44)$$

$$\frac{\partial^2 \hat{\mathbf{e}}}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} = \mathbf{0} \quad (45)$$

Given the derivatives of the normal and edge vectors, the first and second derivatives of $\cos \theta$ and $\sin \theta$ are trivial to calculate.

$$\frac{\partial \cos \theta}{\partial \mathbf{x}_{m_s}} = \frac{\partial \hat{\mathbf{n}}^A \cdot \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{m_s}} \quad (46)$$

$$= \frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{m_s}} \cdot \hat{\mathbf{n}}^B + \hat{\mathbf{n}}^A \cdot \frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{m_s}} \quad (47)$$

$$\frac{\partial \sin \theta}{\partial \mathbf{x}_{m_s}} = \frac{\partial (\hat{\mathbf{n}}^A \times \hat{\mathbf{n}}^B) \cdot \hat{\mathbf{e}}}{\partial \mathbf{x}_{m_s}} \quad (48)$$

$$= \left(\frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{m_s}} \times \hat{\mathbf{n}}^B + \hat{\mathbf{n}}^A \times \frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{m_s}} \right) \cdot \hat{\mathbf{e}} \\ + (\hat{\mathbf{n}}^A \times \hat{\mathbf{n}}^B) \cdot \frac{\partial \hat{\mathbf{e}}}{\partial \mathbf{x}_{m_s}} \quad (49)$$

$$\frac{\partial^2 \cos \theta}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} = \frac{\partial^2 \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \cdot \hat{\mathbf{n}}^B + \frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{n_t}} \cdot \frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{m_s}} \\ + \frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{n_t}} \cdot \frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{m_s}} + \hat{\mathbf{n}}^A \cdot \frac{\partial^2 \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \quad (50)$$

$$\frac{\partial^2 \sin \theta}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \\ = \left(\frac{\partial^2 \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \times \hat{\mathbf{n}}^B + \frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{m_s}} \times \frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{n_t}} \right. \\ \left. + \frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{n_t}} \times \frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{m_s}} + \hat{\mathbf{n}}^A \times \frac{\partial^2 \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \right) \cdot \hat{\mathbf{e}} \\ + \left(\frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{m_s}} \times \hat{\mathbf{n}}^B + \hat{\mathbf{n}}^A \times \frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{m_s}} \right) \cdot \frac{\partial \hat{\mathbf{e}}}{\partial \mathbf{x}_{n_t}} \\ + \left(\frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{n_t}} \times \hat{\mathbf{n}}^B + \hat{\mathbf{n}}^A \times \frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{n_t}} \right) \cdot \frac{\partial \hat{\mathbf{e}}}{\partial \mathbf{x}_{m_s}} \\ + (\hat{\mathbf{n}}^A \times \hat{\mathbf{n}}^B) \frac{\partial^2 \hat{\mathbf{e}}}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \quad (51)$$

Finally, we can differentiate equation (30) to obtain the derivatives of C .

$$\frac{\partial C}{\partial \mathbf{x}_{m_s}} = \frac{\partial \theta}{\partial \mathbf{x}_{m_s}} \quad (52)$$

$$= \frac{1}{\sin^2 \theta + \cos^2 \theta} \\ \times \left(\cos \theta \frac{\partial \sin \theta}{\partial \mathbf{x}_{m_s}} - \sin \theta \frac{\partial \cos \theta}{\partial \mathbf{x}_{m_s}} \right) \quad (53)$$

$$= \cos \theta \frac{\partial \sin \theta}{\partial \mathbf{x}_{m_s}} - \sin \theta \frac{\partial \cos \theta}{\partial \mathbf{x}_{m_s}} \quad (54)$$

Note that the second derivative formula below starts from (53), including the denominator term.

$$\begin{aligned}
\frac{\partial^2 C}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} &= \frac{\partial^2 \theta}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \\
&= \frac{1}{(\sin^2 \theta + \cos^2 \theta)^2} \\
&\times \left[(\sin^2 \theta + \cos^2 \theta) \left(\cos \theta \frac{\partial^2 \sin \theta}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \right. \right. \\
&\quad \left. \left. - \sin \theta \frac{\partial^2 \cos \theta}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \right) \right. \\
&\quad + (\sin^2 \theta - \cos^2 \theta) \left(\frac{\partial \sin \theta}{\partial \mathbf{x}_{m_s}} \frac{\partial \cos \theta}{\partial \mathbf{x}_{n_t}} + \frac{\partial \cos \theta}{\partial \mathbf{x}_{m_s}} \frac{\partial \sin \theta}{\partial \mathbf{x}_{n_t}} \right) \\
&\quad \left. + 2 \sin \theta \cos \theta \left(\frac{\partial \cos \theta}{\partial \mathbf{x}_{m_s}} \frac{\partial \cos \theta}{\partial \mathbf{x}_{n_t}} - \frac{\partial \sin \theta}{\partial \mathbf{x}_{m_s}} \frac{\partial \sin \theta}{\partial \mathbf{x}_{n_t}} \right) \right] \quad (55)
\end{aligned}$$

$$\begin{aligned}
&= \cos \theta \frac{\partial^2 \sin \theta}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} - \sin \theta \frac{\partial^2 \cos \theta}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \\
&\quad + (\sin^2 \theta - \cos^2 \theta) \left(\frac{\partial \sin \theta}{\partial \mathbf{x}_{m_s}} \frac{\partial \cos \theta}{\partial \mathbf{x}_{n_t}} + \frac{\partial \cos \theta}{\partial \mathbf{x}_{m_s}} \frac{\partial \sin \theta}{\partial \mathbf{x}_{n_t}} \right) \\
&\quad + 2 \sin \theta \cos \theta \left(\frac{\partial \cos \theta}{\partial \mathbf{x}_{m_s}} \frac{\partial \cos \theta}{\partial \mathbf{x}_{n_t}} - \frac{\partial \sin \theta}{\partial \mathbf{x}_{m_s}} \frac{\partial \sin \theta}{\partial \mathbf{x}_{n_t}} \right) \quad (57)
\end{aligned}$$

In a manner similar to the stretch condition, $\frac{\partial^2 C}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} = \frac{\partial^2 C}{\partial \mathbf{x}_{n_t} \partial \mathbf{x}_{m_s}}$ and the 3×3 matrix is symmetric. The second derivatives of $\cos \theta$ and $\sin \theta$ are likewise symmetric.

4 Verification

Macri didn't describe any way of verifying the correctness of his derivations or implementation. Once I had initially implemented the simulator, I found bugs but had no way of determining where an error was occurring. To address this, I used numeric derivatives.

For example, suppose that we need to verify the calculation of $\frac{\partial f(x)}{\partial x}$. Our program calculates the derivative using an analytic formula, which we'll call $f'_a(x)$. The derivative can also be calculated numerically:

$$f'_n(x) = \frac{f(x+h) - f(x)}{h}$$

If $f'_a(x) = f'_n(x)$, then the implementation and analytic formula give the correct results.

Verification is a little bit trickier for the bend condition. Here, we can't just use a normal derivative, since we have to differentiate under the assumption that the vectors have constant magnitude. Suppose that we want to verify the derivative $\frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{m_s}}$. Let $\mathbf{n}^A(\mathbf{x}_{m_s})$ represent the normal with point m at position \mathbf{x}_{m_s} , and $\mathbf{n}^A(\mathbf{x}_{m_s} + h\mathbf{I}_s)$ represent the normal once point m 's s th component is shifted by h . The unit normals are then given by

$$\begin{aligned}
\hat{\mathbf{n}}^A(\mathbf{x}_{m_s}) &= \frac{\mathbf{n}^A(\mathbf{x}_{m_s})}{\|\mathbf{n}^A(\mathbf{x}_{m_s})\|} \\
\hat{\mathbf{n}}^A(\mathbf{x}_{m_s} + h\mathbf{I}_s) &= \frac{\mathbf{n}^A(\mathbf{x}_{m_s} + h\mathbf{I}_s)}{\|\mathbf{n}^A(\mathbf{x}_{m_s} + h\mathbf{I}_s)\|}
\end{aligned}$$

And the numeric derivative is given by

$$(\hat{\mathbf{n}}_n^A)' = \frac{\hat{\mathbf{n}}^A(\mathbf{x}_{m_s} + h\mathbf{I}_s) - \hat{\mathbf{n}}^A(\mathbf{x}_{m_s})}{h}$$

The main trick is to normalise the shifted edge and normal vectors using the magnitude of the unshifted vector.

To be honest, I don't know why numeric derivatives can't be used for everything. I suspect that they may be less robust than analytic derivatives, and there may be a performance tradeoff involved, but I haven't confirmed these suspicions.

5 Discussion

5.1 Macri's Error

Equations (54) and (57) are very different from the equations obtained by Macri. Macri differentiated equations (28) and (29) and then rearranged to obtain two different formulas for $\frac{\partial C}{\partial \mathbf{x}_m}$:

$$\frac{\partial C}{\partial \mathbf{x}_m} = -\frac{1}{\sin \theta} \frac{\partial \cos \theta}{\partial \mathbf{x}_m}$$

and

$$\frac{\partial C}{\partial \mathbf{x}_m} = \frac{1}{\cos \theta} \frac{\partial \sin \theta}{\partial \mathbf{x}_m}$$

Likewise, Macri obtained two different formulas for $\frac{\partial^2 C}{\partial \mathbf{x}_m \partial \mathbf{x}_n}$. Macri suggested choosing the first set of equations if $|\sin \theta| > |\cos \theta|$. The rationale for this choice is presumably to avoid numerical problems and division by zero when $\sin \theta = 0$. In order for the transition from one regime to the other to be smooth, they should be equal at some changeover point. However, it is not evident that there exists any changeover point where the two formulas are equal. Additionally, neither of these equations matches the numeric derivative of C . My equations, however, give a result identical to the numeric derivative.

Macri's approach fails because he doesn't consider the transition between his two bend formulas. My approach succeeds in these cases because it relies on a single formula for all bend angles, instead of switching between formulas at critical bend points.

I derived equation (54) by the following logic. Define $y = \frac{\sin \theta}{\cos \theta}$, and $\theta = \arctan y$. Then, $\frac{\partial \theta}{\partial \mathbf{x}_{m_s}} = \cos^2 \theta \frac{\partial y}{\partial \mathbf{x}_{m_s}}$. Differentiating y , we obtain

$$\frac{\partial y}{\partial \mathbf{x}_{m_s}} = \frac{\cos \theta \frac{\partial \sin \theta}{\partial \mathbf{x}_{m_s}} - \sin \theta \frac{\partial \cos \theta}{\partial \mathbf{x}_{m_s}}}{\cos^2 \theta}$$

and thus,

$$\frac{\partial \theta}{\partial \mathbf{x}_{m_s}} = \cos \theta \frac{\partial \sin \theta}{\partial \mathbf{x}_{m_s}} - \sin \theta \frac{\partial \cos \theta}{\partial \mathbf{x}_{m_s}}$$

The derivation of (57) follows directly from differentiation of this equation. In some ways, this definition feels a little circular due to the introduction of the artificial arctan term. I believe that the derivation is nevertheless correct. One area of serious concern, however, is the fact that the second derivative does not exhibit typical symmetry: $\frac{\partial^2 \theta}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \neq \frac{\partial^2 \theta}{\partial \mathbf{x}_{n_t} \partial \mathbf{x}_{m_s}}$ in general.

5.2 Baraff & Witkin's Errors

I believe that there is a very small error in equation (14) of Baraff & Witkin's paper: the \mathbf{z} term should be scaled by h .

There is an error in their definition of $\frac{\partial \mathbf{C}}{\partial t}$, just above equation (11). They define it as

$$\frac{\partial \mathbf{C}}{\partial t} = \left(\frac{\partial \mathbf{C}}{\partial \mathbf{x}} \right)^T \frac{\partial \mathbf{x}}{\partial t}$$

However, the condition function \mathbf{C} is a function of the position of several particles \mathbf{x} , and the paper does not specify which particle to choose. For my early implementations, I defined a different $\frac{\partial \mathbf{C}}{\partial t}$ for each particle, using

$$\frac{\partial \mathbf{C}_m}{\partial t} = \left(\frac{\partial \mathbf{C}}{\partial \mathbf{x}_m} \right)^T \frac{\partial \mathbf{x}_m}{\partial t}.$$

However, this yielded incorrect results, especially when calculating the damping forces for the bend condition, resulting in frequent instability. Later, I realised that correct application of the chain rule should yield a single $\frac{\partial \mathbf{C}}{\partial t}$ for all particles, of the form

$$\frac{\partial \mathbf{C}}{\partial t} = \sum_m \left(\frac{\partial \mathbf{C}}{\partial \mathbf{x}_m} \right)^T \frac{\partial \mathbf{x}_m}{\partial t}. \quad (58)$$

The sum should be over the particles that influence \mathbf{C} . Visually, this yields vastly better results.

The final error in their paper is related to scale-invariance. Ideally, the parameters used for the simulation should be independent of the tessellation of the surface. For example, if you run the simulation with a 10×10 mesh and a 20×20 mesh and the same parameters, it should look basically the same. However, using Baraff & Witkin's choice of $\alpha = a$, this does not occur. I looked at the math for the specific case where the number of particles is doubled in u and v (e.g., from 10×10 to 20×20). In this case, the gravity force on each original particle decreases by a factor of four (since the mass of the surrounding triangles decreases by four). However, the (strong) stretch and shear forces decrease by a factor of eight. These forces are responsible for most of the resistance against gravity, and consequently the cloth sags substantially as the particle density is increased.

However, by using $\alpha = a^{\frac{3}{4}}$, the stretch and shear forces also decrease by a factor of four, as expected. The formula is not as intuitively obvious as $\alpha = a$, and I have not done tests with irregularly triangulated meshes, but it does work much better for regularly triangulated meshes.

5.3 Unit Magnitude Approximation

During my original development, I chased down one wrong alley while looking for the error in my (and Macri’s) derivations. I was highly suspicious of one of Baraff & Witkin’s approximations, where they assume that vectors’ magnitudes are constant. This occurs in both the shear condition derivation and the bend condition derivation. I went through and re-derived all of the formulas without this assumption, and obtained the formulas given in the appendix. After all of this work, there was still a problem, which proved to be the error in Macri’s versions of equations (54) and (57).

After finding this error in Macri’s work, I returned to the unit magnitude issue. I compared the results of my simulation using the formulas given in the body of the paper (using Baraff & Witkin’s approximation) to the results using the formulas given in the appendix (without the approximation). In my test case, I found that the approximation gave more pleasing results! In total, stretch energy was halved, and shear and bend energy dropped by a small amount. The formulas for both stretch and bend energy are unchanged between the two cases, so this is a valid comparison of the energy in the cloth. The formula for shear energy is slightly changed by removing the unit magnitude approximation, however, so comparisons of energy aren’t very meaningful.

To be honest, I was surprised that such a drastic change to the formulas had a relatively small effect on the final result. I suspect that there may be a lot of room for optimisation of Baraff & Witkin’s system, achieving comparable results at less computational cost.

In the interests of presenting a direct implementation of Baraff & Witkin’s paper, and on the basis of the better results, I have therefore presented the formulas with the constant magnitude approximation.

5.4 Parameters

For parameters, I adopted values that gave good results. I looked at Alias | Wavefront’s *Maya* cloth software, but I suspect that they scale their parameters internally by an unknown amount. For damp-

Name	Symbol	Value
Stretch force strength	$k_{stretch}$	5.0×10^3
Shear force strength	k_{shear}	0.5×10^3
Bend force strength (u)	k_{bend_u}	0.01×10^{-3}
Bend force strength (v)	k_{bend_v}	0.01×10^{-3}
Damping force strength	k_{damp}	0.2
Rest stretch (u)	b_u	1
Rest stretch (v)	b_v	1
Time step	h	0.02
Density	ρ	0.1
Gravity	g	9.81

Table 1: Parameters used for simulation.

ing forces, I multiplied the associated regular force’s constant by the damping constant (e.g. $k_{stretch}k_{damp}$ was used to damp stretch forces).

6 Results

To test out the simulator, I used a variety of cloth sizes and constraints. These can be seen at the Freecloth website <http://davidpritchard.org/freecloth>, in the screenshots section.

The simulator’s performance is reasonable, with a 66×66 cloth typically requiring about thirty to forty minutes of computation per second of animation on a 3 GHz Pentium IV system using a step size of 0.02. Smaller cloth sizes are much faster, and there is still a fair bit of room for optimisation, especially in terms of adaptive timesteps.

My implementation does exhibit some instability, but almost always works with a step size of 0.01 seconds; sometimes, a smaller stepsize is needed.

7 Conclusion

Baraff & Witkin’s paper does indeed provide a solid framework for cloth simulation. It appears to allow relatively large timesteps, and with suitable optimisation should run quite quickly.

Hopefully, this paper should clear up some of the confusions of both Macri’s paper, and the original

paper. Implementing Baraff & Witkin's paper should be straightforward using the equations shown here.

8 Acknowledgement

This work was supported in part by a scholarship from the Natural Sciences and Engineering Research Council of Canada, and by the British Columbia Advanced Systems Institute.

Thanks to Sebastian Lipponer for pointing out a correction to the bend condition formulas.

References

- [BW98] D. Baraff and A. Witkin. Large steps in cloth simulation. In *ACM SIGGRAPH Conference Proceedings*, pages 43–54, 1998.
- [Mac00] D. Macri. Real-time cloth. In *Game Developers Conference Proceedings*, 2000.

A Appendix

The equations given in this appendix are replacements for certain equations given in the body of the paper. By using these, several of Baraff & Witkin's approximations are avoided. However, my tests show that the results are less pleasing.

A.1 Per-Triangle Quantities

The first and second derivatives of the normalised vectors $\hat{\mathbf{w}}_u$ and $\hat{\mathbf{w}}_v$ are

$$\frac{\partial \hat{\mathbf{w}}_u}{\partial \mathbf{x}_{m_s}} = \frac{1}{\|\mathbf{w}_u\|} \frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{m_x}} (\mathbf{I}_s - \hat{\mathbf{w}}_{u_s} \hat{\mathbf{w}}_u) \quad (\text{A.1})$$

$$\frac{\partial \hat{\mathbf{w}}_v}{\partial \mathbf{x}_{m_s}} = \frac{1}{\|\mathbf{w}_v\|} \frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{m_x}} (\mathbf{I}_s - \hat{\mathbf{w}}_{v_s} \hat{\mathbf{w}}_v) \quad (\text{A.2})$$

$$\begin{aligned} \frac{\partial^2 \hat{\mathbf{w}}_u}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} = & - \frac{1}{\|\mathbf{w}_u\|} \frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{m_x}} \left(\hat{\mathbf{w}}_{u_s} \frac{\partial \hat{\mathbf{w}}_u}{\partial \mathbf{x}_{n_t}} + \frac{\partial \hat{\mathbf{w}}_{u_s}}{\partial \mathbf{x}_{n_t}} \hat{\mathbf{w}}_u \right) \\ & - \frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{m_x}} \frac{\partial \mathbf{w}_{u_x}}{\partial \mathbf{x}_{n_x}} \frac{(\mathbf{I}_s - \hat{\mathbf{w}}_{u_s} \hat{\mathbf{w}}_u) \hat{\mathbf{w}}_{u_t}}{\|\mathbf{w}_u\|^2} \end{aligned} \quad (\text{A.3})$$

$$\begin{aligned} \frac{\partial^2 \hat{\mathbf{w}}_v}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} = & - \frac{1}{\|\mathbf{w}_v\|} \frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{m_x}} \left(\hat{\mathbf{w}}_{v_s} \frac{\partial \hat{\mathbf{w}}_v}{\partial \mathbf{x}_{n_t}} + \frac{\partial \hat{\mathbf{w}}_{v_s}}{\partial \mathbf{x}_{n_t}} \hat{\mathbf{w}}_v \right) \\ & - \frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{m_x}} \frac{\partial \mathbf{w}_{v_x}}{\partial \mathbf{x}_{n_x}} \frac{(\mathbf{I}_s - \hat{\mathbf{w}}_{v_s} \hat{\mathbf{w}}_v) \hat{\mathbf{w}}_{v_t}}{\|\mathbf{w}_v\|^2} \end{aligned} \quad (\text{A.4})$$

A.2 Shear Condition

If we remove the assumption that the $\hat{\mathbf{w}}_u$ and $\hat{\mathbf{w}}_v$ vectors are unstretched, then equation (19) should be replaced with

$$C = \alpha \hat{\mathbf{w}}_u \cdot \hat{\mathbf{w}}_v \quad (\text{A.5})$$

$$\begin{aligned} \frac{\partial C}{\partial \mathbf{x}_{m_s}} = & \alpha \left(\frac{\partial \hat{\mathbf{w}}_u}{\partial \mathbf{x}_{m_s}} \cdot \hat{\mathbf{w}}_v + \hat{\mathbf{w}}_u \cdot \frac{\partial \hat{\mathbf{w}}_v}{\partial \mathbf{x}_{m_s}} \right) \quad (\text{A.6}) \\ \frac{\partial^2 C}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} = & \alpha \left(\frac{\partial^2 \hat{\mathbf{w}}_u}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \cdot \hat{\mathbf{w}}_v + \frac{\partial \hat{\mathbf{w}}_u}{\partial \mathbf{x}_{m_s}} \cdot \frac{\partial \hat{\mathbf{w}}_v}{\partial \mathbf{x}_{n_t}} \right. \\ & \left. + \frac{\partial \hat{\mathbf{w}}_u}{\partial \mathbf{x}_{n_t}} \cdot \frac{\partial \hat{\mathbf{w}}_v}{\partial \mathbf{x}_{m_s}} + \hat{\mathbf{w}}_u \cdot \frac{\partial^2 \hat{\mathbf{w}}_v}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \right) \quad (\text{A.7}) \end{aligned}$$

A.3 Bend Condition

If we remove the assumption that the normal and edge vectors have constant length, we must replace equations (40) and (43) with those shown below.

$$\frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{m_s}} = (\mathbf{I} - \hat{\mathbf{n}}^A (\hat{\mathbf{n}}^A)^T) \frac{S_s(\mathbf{q}_m^A)}{\|\mathbf{n}^A\|} \quad (\text{A.8})$$

$$\frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{m_s}} = (\mathbf{I} - \hat{\mathbf{n}}^B (\hat{\mathbf{n}}^B)^T) \frac{S_s(\mathbf{q}_m^B)}{\|\mathbf{n}^B\|} \quad (\text{A.9})$$

$$\frac{\partial \hat{\mathbf{e}}}{\partial \mathbf{x}_{m_s}} = (\mathbf{I} - \hat{\mathbf{e}} \hat{\mathbf{e}}^T) \frac{q_m^e \mathbf{I}_s}{\|\mathbf{e}\|} \quad (\text{A.10})$$

$$\begin{aligned} & \frac{\partial^2 \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \\ = & - \left(\frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{n_t}} (\hat{\mathbf{n}}^A)^T + \hat{\mathbf{n}}^A \left(\frac{\partial \hat{\mathbf{n}}^A}{\partial \mathbf{x}_{n_t}} \right)^T \right) \frac{S_s(\mathbf{q}_m^A)}{\|\mathbf{n}^A\|} \\ & + \frac{\mathbf{I} - \hat{\mathbf{n}}^A (\hat{\mathbf{n}}^A)^T}{\|\mathbf{n}^A\|^2} \left(\|\mathbf{n}^A\| S_s \left(\frac{\partial \mathbf{q}_m^A}{\partial \mathbf{x}_{n_t}} \right) \right. \\ & \left. - (S_t(\mathbf{q}_n^A) \cdot \hat{\mathbf{n}}^A) S_s(\mathbf{q}_m^A) \right) \end{aligned} \quad (\text{A.11})$$

$$\begin{aligned}
& \frac{\partial^2 \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \\
&= - \left(\frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{n_t}} (\hat{\mathbf{n}}^B)^T + \hat{\mathbf{n}}^B \left(\frac{\partial \hat{\mathbf{n}}^B}{\partial \mathbf{x}_{n_t}} \right)^T \right) \frac{S_s(\mathbf{q}_m^B)}{\|\mathbf{n}^B\|} \\
&+ \frac{\mathbf{I} - \hat{\mathbf{n}}^B (\hat{\mathbf{n}}^B)^T}{\|\mathbf{n}^B\|^2} \left(\|\mathbf{n}^B\| S_s \left(\frac{\partial \mathbf{q}_m^B}{\partial \mathbf{x}_{n_t}} \right) \right. \\
&\quad \left. - (S_t(\mathbf{q}_n^B) \cdot \hat{\mathbf{n}}^B) S_s(\mathbf{q}_m^B) \right) \tag{A.12}
\end{aligned}$$

$$\begin{aligned}
& \frac{\partial^2 \hat{\mathbf{e}}}{\partial \mathbf{x}_{m_s} \partial \mathbf{x}_{n_t}} \\
&= - \left(\frac{\partial \hat{\mathbf{e}}}{\partial \mathbf{x}_{n_t}} \hat{\mathbf{e}}^T + \hat{\mathbf{e}} \left(\frac{\partial \hat{\mathbf{e}}}{\partial \mathbf{x}_{n_t}} \right)^T \right) \frac{S_s(q_m^e)}{\|\mathbf{e}\|} \\
&- \frac{\mathbf{I} - \hat{\mathbf{e}} \hat{\mathbf{e}}^T}{\|\mathbf{e}\|^2} q_m^e q_n^e \hat{\mathbf{e}}_t \mathbf{I}_s \tag{A.13}
\end{aligned}$$