Chapter 4
Continuum versus Particle Representations

J. W. Eischen
Associate Professor
Department of Mechanical and Aerospace Engineering
NC State University
Raleigh, NC 27695
(919) 515-5263
eischen@eos.ncsu.edu

R. Bigliani
Visiting Scholar
Department of Mechanical and Aerospace Engineering
NC State University
Raleigh, NC 27695
(919) 515-5263
rbiglian@gateway.net

Cloth Modeling and Animation
A. K. Peters Publishing

1 Introduction

Simulation of drape and manipulation of cloth has only recently been successfully computed because proper physical models and/or computational strategies have not been previously available. The predominant method for simulating cloth has been the so-called "particle method," whereby equilibrium of external and internal forces is enforced at grid points on a particle mesh. The finite element method offers an alternative to the particle method in modeling fabric drape. Here we will describe the important differences in the two approaches by comparing the fundamental physical assumptions, solution algorithms, and results.

Following is a description of the previous computational drape work from the open literature. Imaoka, et al. [1] modeled fabrics using rod elements and bending springs. They carried out simulations of the draping of skirt shapes. Izumi and Niwa [2] studied the dynamic drape of ladies dress fabrics and correlated the results with mechanical properties such as bending rigidity and self weight. Weil [3] produced three-dimensional drape images by modeling the fabric as a constrained system of grid points, without consideration of material properties. Terzopoulos, et. al. [4] analyzed three-dimensional deformable surfaces using nonlinear elasticity theory, with a simplified stress-strain response. The stress-strain response did not employ standard engineering measures of rigidity like Young's modulus. Von Der Weeën [5] developed algorithms for draping of biaxially woven fabrics on a double-curved surface. Fabrics were modeled as two families of perpendicular interlocked inextensible fabrics. Gan, et. al. [6, 7] developed a finite element method to treat three-dimensional drape based on shell/plate theory. Both linear and nonlinear moment curvature relationships were treated, with different response in the warp and weft directions. Membrane response was taken to be nonlinear. Simulations of sheets draping over squares and circles were presented. Kim [8] showed how three-dimensional drape could be treated with the geometrically exact shell theory proposed recently by Simo, et. al. [9, 10, 11]. In this method, all engineering properties such as modulus, weight, thickness are accounted for in a consistent fashion. Collier, et. al. [12] treated draping behavior using orthotropic shell theory,, including the effect of geometric nonlinearity. The drape of an annular shape was studied in detail. Breen, et. al [13] developed a particle model for cloth with mechanical constraints to represent thread interactions. Deng [14] has extended the work of Kim [8] to treat buckling, nonlinear bending behavior, and contact. This approach models the fabric as a three-dimensional (shell) continuum. Kang and Wu [15] also used a continuum model (plate) for fabric drape, allowing for different mechanical response properties in the warp and weft directions. Drape response of circular parts over circular pedestals and rectangular parts over rectangular pedestals were shown. Chen and Govindaraj [16] presented further results on three dimensional drape using a degenerated solid shell theory. They present results showing square fabric parts draping over a flat square surface. The group at MIRAlab has been focused on dy-

namic drape of cloth on virtual humans, as described in the article by Volino, et.al [17]. Dynamic drape has also been treated recently by Baraff and Witkin [18], with an emphasis on novel numerical integration techniques. Recently, Hu and Teng [19] provided a recent literature review of computational fabric mechanics.

2 Finite Element Continuum Modeling of Cloth

The finite element approach described here is based on a "geometrically exact resultant shell theory" conceived by Simo, et. al. [9, 10, 11]. This approach follows from the well established discipline of continuum mechanics. As opposed to the particle method, there are no ad-hoc assumptions regarding calculation of stress and strain quantities. The finite element method results in a coupled system of nonlinear equations governing the displacements and surface orientation at discrete points on the cloth surface. These equations must be solved using an iterative technique such as the Newton-Raphson method. Equilibrium of the external applied forces and internal elastic forces is enforced globally, not locally as in the particle method. Generally a coarser spatial discretization of a fabric component is allowed for a given level of accuracy. Details of this work were provided recently by Eischen [20].

2.1 Physical Modeling

The steps involved in formulating the physically based model that lead to the nonlinear system of finite element equations are

- Kinematic description of the shell- how the shell is allowed to move
- Material stress-strain response- how internal forces/stress and deformations are related
- Momentum balance equations- how Newton's law is enforced
- Discretization of momentum balance equations- how the approximation to the balance equations is carried out
- Linearization of the finite element equations- how the governing equations are presented in a form for solution by the Newton-Raphson method

Details of each of these steps (including matrix expressions needed for coding) are included in the Appendix.

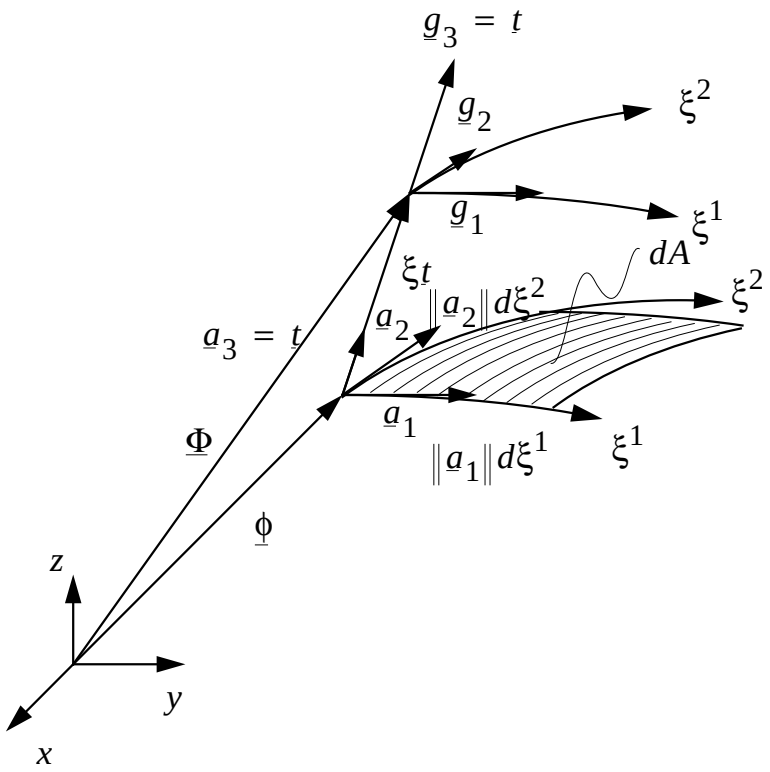

Figure 1: Configuration of the Shell

The kinematic description of the shell starts by parameterizing the position of points within the shell, both on and off the mid-surface. The mid-surface is a reference surface located midway between the two bounding surfaces of the thin shell. For a piece of cloth, it would be appropriate to think of the bounding surfaces as the "inside" and "outside" surfaces. Referring to Fig. 1, points off the mid-surface are located by a position vector $\underline{\Phi}$,

$$\underline{\Phi}(\xi^1, \xi^2, \xi) = \underline{\phi}(\xi^1, \xi^2) + \xi \underline{t}(\xi^1, \xi^2) \tag{1}$$

where $\underline{\phi}$ is a position vector that locates points on the mid-surface (reference surface) of the shell. The vector $\underline{t}$ is called the director and is a unit vector directed along fibers in the shell that are initially perpendicular to the reference surface. The coordinate ξ measures distance between the mid- surface and points off the mid-surface along $\underline{t}$. The coordinates (not necessarily arc-length) ξ^1 and ξ^2 serve to parameterize the midsurface. The book by Chung [21] provides further elaboration on this choice of coordinates from the point of view of classical shell theory. For this work where a finite element approach ultimately results, in fact, these coordinates

are selected to be the "parent element" coordinates for the standard isoparametric quadrilateral element (see Zienkiewicz and Taylor [22]). Note that the unit vector $\underline{t}$ is not necessarily normal to the mid-surface of the deformed shell, thus admitting the possibility of transverse shear strain. The thru-thickness coordinate ξ is in the range $-h/2 \leq \xi \leq h/2$, where h is the shell thickness. The undeformed configuration of the shell is given by

$$\underline{\Phi}_0(\xi^1, \xi^2, \xi) = \underline{\phi}_0(\xi^1, \xi^2) + \xi \underline{t}_0(\xi^1, \xi^2) \tag{2}$$

where $\underline{\phi}_0$ is a vector locating points on the undeformed mid-surface and $\underline{t}_0$ is the director in the initial configuration, assumed normal to the mid-surface.
For the shell model there are three types of deformation (strain) considered. First membrane strains measure the extension and/or compression of fibers in the plane of mid-surface as well as shear distortion of fibers in the plane of the mid-surface that are initially perpendicular. Transverse shear strain is a measure of the change in angle between fibers tangent to the mid-surface and fibers initially perpendicular to the mid-surface. Bending strains are a measure of the curvature and twist of the mid-surface. The strain measures are defined as follows and depend on the derivatives of position vector $\underline{\phi}$, together with the normal $\underline{t}$ and its derivative,

- Membrane strain :

$$\epsilon_{\alpha\beta} \equiv \frac{1}{2}(\underline{\phi}_{,\alpha} \cdot \underline{\phi}_{,\beta} - \underline{\phi}_{0,\alpha} \cdot \underline{\phi}_{0,\beta}) \tag{3}$$

- Transverse shear strain :

$$\gamma_\alpha \equiv \underline{\phi}_{,\alpha} \cdot \underline{t} - \underline{\phi}_{0,\alpha} \cdot \underline{t}_0 \tag{4}$$

- Bending Strain :

$$\kappa_{\alpha\beta} \equiv \underline{\phi}_{,\alpha} \cdot \underline{t}_{,\beta} - \underline{\phi}_{0,\alpha} \cdot \underline{t}_{0,\beta} \tag{5}$$

where

$$\underline{\phi}_{,\alpha} = \underline{a}_\alpha = \frac{\partial \underline{\phi}(\xi^1, \xi^2)}{\partial \xi^\alpha} \quad \text{and} \quad \underline{t}_{,\alpha} = \frac{\partial \underline{t}(\xi^1, \xi^2)}{\partial \xi^\alpha} \quad \text{for } \alpha = 1, 2 \tag{6}$$

are the covariant tangent base vectors and director derivatives on the mid-surface. The essential problem is to determine the evolution of $\underline{\phi}$ and $\underline{t}$ as the shell deforms under its own weight or is manipulated in some way.

The evolution of the unit director vectors $\underline{t}(\xi^1, \xi^2)$ during a motion of the shell depends on an orthogonal transformation (rotation) matrix $\underline{\Lambda}$. Let $\underline{t} = \underline{\Lambda}\underline{E}$, where $\underline{\Lambda}$ is an orthogonal matrix such that $\underline{\Lambda}\underline{\Lambda}^T = \underline{I}$ and $\underline{E}$ is an inertially fixed unit vector. Thus, to determine $\underline{t}$ during the shell motion, it is sufficient (or equivalent) to determine the matrix $\underline{\Lambda}$. At any point on the shell, this matrix is related to the unit director vector according to

$$\underline{\Lambda} = (\underline{E} \cdot \underline{t})\underline{I} + \widehat{\underline{E} \times} \underline{t} + \frac{1}{1 + \underline{E} \cdot \underline{t}} (\underline{E} \times \underline{t}) \otimes (\underline{E} \times \underline{t}) \tag{7}$$

where the $\widehat{\ }$ indicates the skew symmetric matrix associated with the indicated vector, and $\otimes$ represents a tensor outer product operator.

The relationship between the forces and moments acting on the shell and the strains is governed by constitutive equations. For shells, the internal forces and moments are referred to as stress and moment "resultants". Fig. 2 shows the directions and notation for these quantities. The constitutive equations for linear elastic isotropic materials in a non-orthogonal, non arc-length curvilinear coordinate system are given by

Membrane :

$$\tilde{n}^{\alpha\beta} = hC^{\alpha\beta\gamma\delta}\varepsilon_{\gamma\delta} \tag{8}$$

Bending :

$$\tilde{m}^{\alpha\beta} = \frac{h^3}{12} C^{\alpha\beta\gamma\delta}\kappa_{\gamma\delta} \tag{9}$$

Transverse Shear :

$$\tilde{q}^{\alpha} = \kappa h C^{\alpha 3 \gamma 3}\gamma_{\gamma} \tag{10}$$

where $\alpha, \beta, \gamma, \delta = 1, 2$, The elastic constants $C^{\alpha\beta\gamma\delta}$ are

$$C^{\alpha\beta\gamma\delta} = \frac{E}{1-\nu^2}\left[\nu a^{0^{\alpha\beta}} a^{0^{\gamma\delta}} + \frac{1-\nu}{2}(a^{0^{\alpha\gamma}} a^{0^{\beta\delta}} + a^{0^{\alpha\delta}} a^{0^{\gamma\beta}})\right] \tag{11}$$

$$C^{\alpha 3 \gamma 3} = \kappa G_s \begin{bmatrix} a^{0^{11}} & a^{0^{12}} \\ a^{0^{12}} & a^{0^{22}} \end{bmatrix} \tag{12}$$

and

$$a^{0^{\alpha\beta}} = \underline{a}^{0^{\alpha}} \cdot \underline{a}^{0^{\beta}} \tag{13}$$

$$G_s = \frac{E}{2(1+\nu)} \tag{14}$$

E is the modulus of elasticity, ν is Poisson's ratio, and κ is the shear correction factor. The Appendix contains further details and definitions of terms appearing in

these constitutive equations.

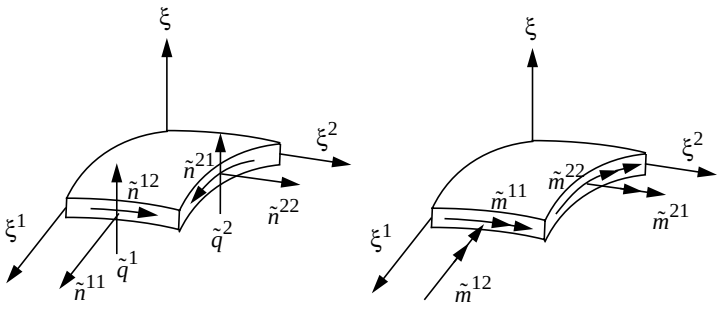

Figure 2: Effective Stress Resultants Acting on the Shell Mid-Surface

Generally fabric materials exhibit a nonlinear relationship between bending moments and bending strain. The nonlinear moment-curvature response for fabric is often measured by the Kawabata Bending Test System (Kato Tekko Co., Ltd - Kawabata [23]). A typical moment-curvature plot for a twill weave 100% cotton fabric is shown in Fig. 3.

In our work, this data is fit to a fifth order polynomial as depicted in Fig. 4 for subsequent use in the 3D shell theory. This yields a moment equal to

$$\begin{aligned} M &= f(K) \\ &= \alpha_1 K + \alpha_2 K^2 + \alpha_3 K^3 + \alpha_4 K^4 + \alpha_5 K^5 \end{aligned} \tag{15}$$

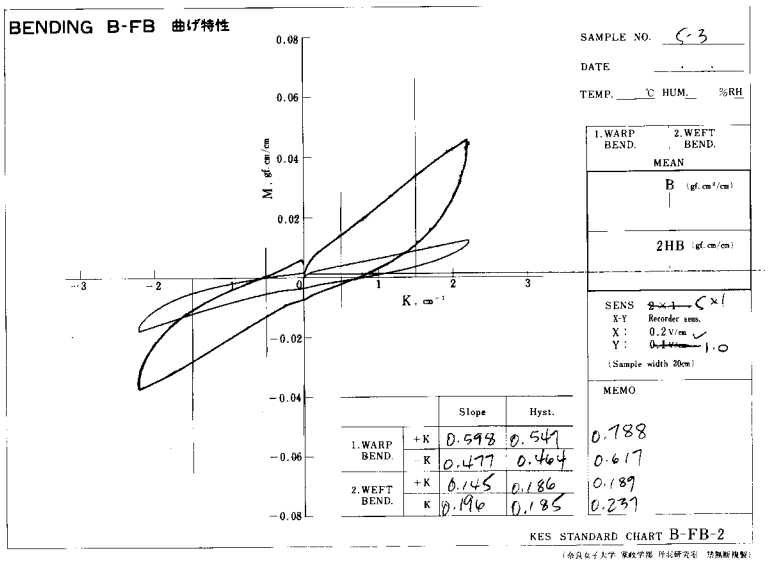

Figure 3: Kawabata Bending Test Output

where K is the curvature measured in units of cm^{-1}. For the cloth of Fig. 4, we found $\alpha_1 = 1.33387$, $\alpha_2 = -2.23070$, $\alpha_3 = 2.36510$, $\alpha_4 = -1.08471$, $\alpha_5 = 0.17876$.

The details of the modifications of the classical elastic constitutive equations to account for the nonlinear bending effect are contained in the Appendix.

Continuum mechanics then requires enforcing the balance of linear and angular momentum. This results in five nonlinear partial differential equations that hold at each point on the shell midsurface. Standard variational techniques then allow conversion of these PDE's to a weak or virtual work equation. A spatial discretization is then introduced following standard finite element methods leading to a system of coupled nonlinear equations. Upon linearization, a matrix equation of the following form results

$$\underline{K}(\underline{\phi}, \underline{t}) \left\{ \begin{array}{c} \Delta\underline{\phi} \\ \Delta\underline{t} \end{array} \right\} = \underline{F}_{ext} - \underline{P}(\underline{\phi}, \underline{t}) \tag{16}$$

where $\underline{K}(\underline{\phi}, \underline{t})$ is the tangent stiffness matrix, $\underline{P}(\underline{\phi}, \underline{t})$ is the internal force vector, and $\underline{F}_{ext}$ is the external force vector. Iterative solution of this matrix equation generates increments in the displacements and director vector, $\Delta\underline{\phi}$ and $\Delta\underline{t}$, that are used to update the position and orientation of the shell mid-surface.

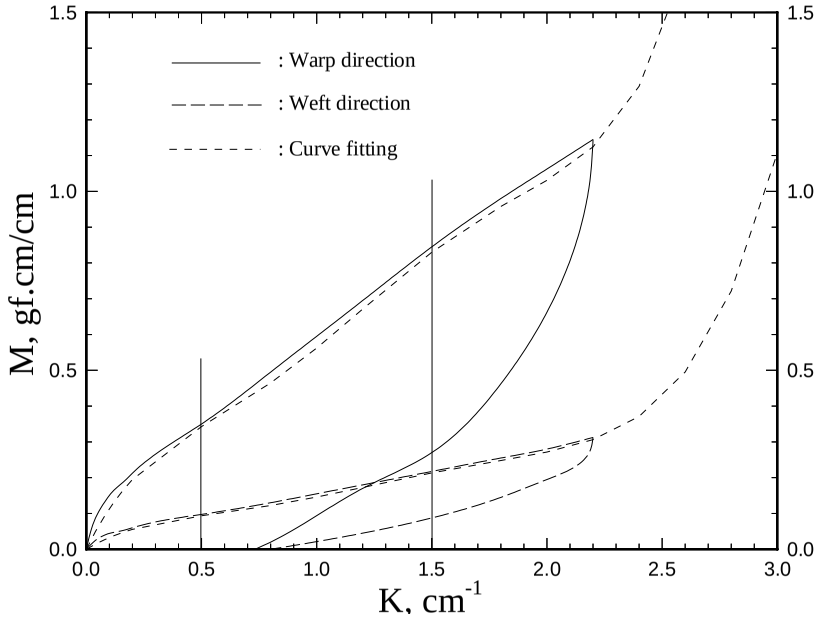

Figure 4: Curve Fitting for Kawabata Bending Test Output

2.2 Solution Algorithms

Supplementary algorithms are used in conjunction with the basic Newton Raphson procedure to solve the nonlinear system of equations. Arc-length control is typically necessary to account for buckling (or wrinkling) and a contact procedure is necessary to deal with fabric interactions with rigid objects or fabric-to-fabric contact (self-collision).

2.2.1 Adaptive Arc-length Control Algorithm

The finite element equations (Eq. 16) must be solved using a nonlinear iterative algorithm, such as Newton-Raphson. The basic algorithm fails when limit or bifurcation points are encountered along the load-deflection curve. Examples of structural configurations where limit points present problems are shown in Fig. 5. The figure on the left shows a straight beam acted on by a compressive axial force P. At a critical value of the axial force (buckling load), the beam will suddenly undergo a transverse displacement. The figure on the right shows a curved beam acted on by a transverse force. Here the relationship bewteen the force and the displacement traverses a complicated path, including two regions with zero slope, i.e. limit points.

The arc-length method of Schweizerhof and Wriggers' [24] treats these difficulty by limiting the increments in force and/or displacement near the limit points.

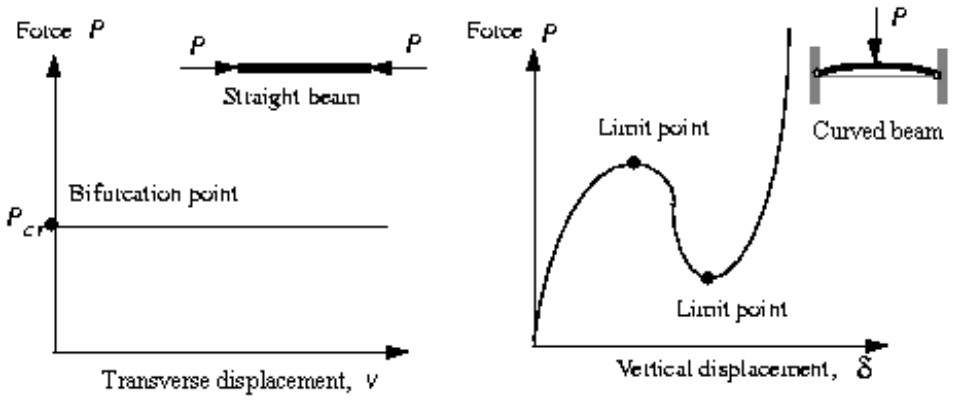

Figure 5: Typical Load-Displacement Curves in Structural Analysis

Referring to Fig. 6, the finite element equations (before linearization) for an equilibrium configuration at load step n are

$$\underline{F}_{ext}(\lambda_n) - \underline{P}(\underline{\Phi}_n, \lambda_n) = \underline{0} \tag{17}$$

where λ_n is a load factor defined by $\underline{F}_{ext} = \lambda \underline{F}_{ref}$, $\underline{F}_{ref}$ being a reference external force (or displacment) vector. The computational task is to advance the solution to a new equilibrium configuration $\underline{\Phi}_{n+1}, \lambda_{n+1}$. Note that $\underline{\Phi}_{n+1} = \underline{\Phi}_{n+1}(\underline{\phi}_{n+1}, \underline{t}_{n+1})$. The arc-length method limits the excursion along the load-deflection curve by an amount Δs, as shown. The equilibrium equations must be satisfied at load step at $n+1$,

$$\underline{F}_{ext}(\lambda_{n+1}) - \underline{P}(\underline{\Phi}_{n+1}, \lambda_{n+1}) = \underline{0} \tag{18}$$

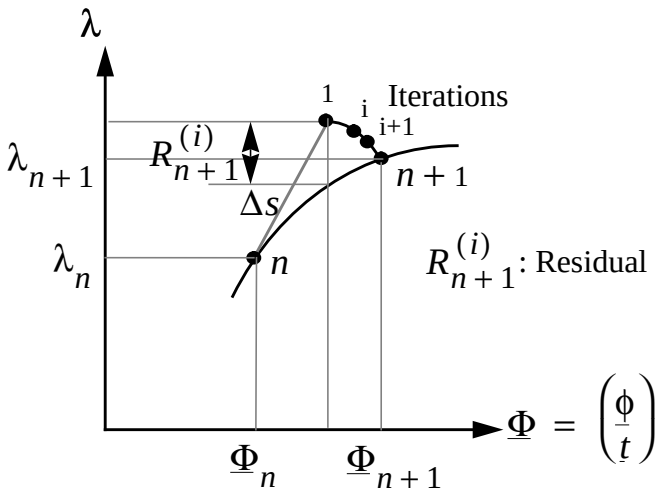

Figure 6: Load-Displacement curve

Schweizerhof and Wriggers [24] introduced a constraint equation

$$f = \left[(\underline{\Phi}_{n+1}^{(i)} - \underline{\Phi}_n) \cdot (\underline{\Phi}_{n+1}^{(i)} - \underline{\Phi}_n) + (\lambda_{n+1}^{(i)} \lambda_n)^2 \right]^{1/2} - \gamma_n \Delta s = 0 \tag{19}$$

where Δs is the arc-length control constant. An acceleration factor can be added to their algorithm to allow a variable arc-length based on the convergence behavior during previous load steps. Based on numerical experiments, successful simulation of highly nonlinear structures such as fabrics, is very dependent on the deformation during the first load step. Therefore, the initial load step (controlled by arc-length) must be very small to avoid divergence. An excessive amount of computer time is expended in subsequent load steps if the initial arc-length is held fixed. Therefore, after the first load step, the arc-length can be enlarged to accelerate computational speed. The newly introduced acceleration factor is denoted γ_n. The algorithm iteration counter is indicated with index i. The acceleration factor γ_n can be defined as a constant or any function. In the present work, γ_n is defined as

$$\gamma_n \equiv \begin{cases} 1 & \text{for } n = 1 \\ \dfrac{N_p}{N_{n-1}} & \text{for } n > 1 \end{cases} \tag{20}$$

where N_p is the target number of iterations per load step, and N_{n-1} is the actual number of iterations for last load step. If γ_n set to 1 for all time, then the constraint reverts to the form of Schweizerhof and Wriggers. Upon linearizing both the equilibrium and the constraint equations, the following formulas are found to govern the

increments in the kinematic quantities and the load factor

$$\Delta\lambda_{n+1}^{(i)} = -\frac{f^{(i)}(f^{(i)} + \gamma_n \Delta s) + (\underline{\Phi}_{n+1}^{(i)} - \underline{\Phi}_n)^T \Delta \underline{u}^{II}}{(\lambda_{n+1}^{(i)} - \lambda_n) + (\underline{\Phi}_{n+1}^{(i)} - \underline{\Phi}_n)^T \Delta \underline{u}^{I}} \tag{21}$$

$$\Delta\underline{\Phi}_{n+1}^{(i)} = \Delta\lambda_{n+1}^{(i)} \Delta \underline{u}^{I} + \Delta \underline{u}^{II} \tag{22}$$

and

$$\Delta \underline{u}^{I} \equiv \underline{K}^{(i)^{-1}} \underline{F}_0 \tag{23}$$

$$\Delta \underline{u}^{II} \equiv \underline{K}^{(i)^{-1}} \left(\underline{F}_{ext}^{(i)} - \underline{P}^{(i)} \right) \tag{24}$$

$$\underline{K}^{(i)} = \left. \frac{\partial \underline{P}}{\partial \underline{\Phi}_{n+1}} \right|^{(i)} \tag{25}$$

$$\underline{F}_0 = \left. \frac{\partial \underline{F}}{\partial \lambda_{n+1}} \right|^{(i)} - \left. \frac{\partial \underline{P}}{\partial \lambda_{n+1}} \right|^{(i)} \tag{26}$$

2.2.2 Contact Algorithm

The penalty method described in Cook, Malkus and Plesha [25] has been employed to treat fabric contact with rigid surfaces, due to its simplicity for incorporation in finite element analysis. Fig. 7 shows a simple one degree of freedom system that illustrates the essential features of the contact scheme.

When the displacement ϕ is less than the initial gap g, the equilibrium equation is

$$P(\phi) = F \tag{27}$$

where $P(\phi) = k\phi$ is the internal elastic force, F is the external force. Note that $\dfrac{\partial P}{\partial \phi} = k$. When $\phi > g$, equilibrium is achieved when

$$P(\phi) = F - F_c \tag{28}$$

where

$$P(\phi) = k\phi$$

$$F_c = k_c(\phi - g)$$

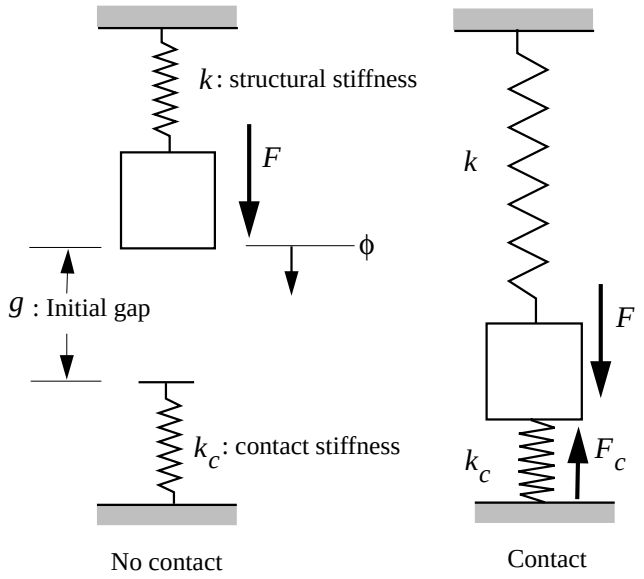

Figure 7: Contact Treatment Schematic

Upon linearization,

$$P(\phi) + \frac{\partial P}{\partial \phi}\Delta\phi + F_c(\phi) + \frac{\partial F_c}{\partial \phi}\Delta\phi = F \tag{29}$$

or

$$(k + k_c)\Delta\phi = F - P(\phi) - F_c(\phi) \tag{30}$$

Note that the incremental equilibrium equation used to solve for the displacement ϕ is

$$(k^{(i)} + k_c)\Delta\phi^{(i)} = F^{(i)} - P^{(i)}(\phi^{(i)}) - F_c^{(i)}(\phi^{(i)}) \tag{31}$$

where

$$F_c^{(i)}(\phi^{(i)}) = k_c(\phi^{(i)} - g) \tag{32}$$

The multi-dimensional analog of this simple example is then

$$\left[\underline{K}^{(i)} + \underline{K}_c^{(i)}\right] \Delta\underline{\Phi}_{n+1}^{(i)} = \underline{F}^{(i)} - \underline{P}^{(i)} - \underline{F}_c^{(i)} \tag{33}$$

where

$$\underline{K}_c = k_c \begin{bmatrix} 0 & 0 & 0 & 0 & \cdots & \cdots & 0 & 0 \\ & 1 & 0 & 0 & \cdots & \cdots & 0 & 0 \\ & & 1 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ & & & & & 0 & 0 & 0 \\ & & Sym. & & & & 1 & 0 \\ & & & & & & & 0 \end{bmatrix} \tag{34}$$

$$\underline{F}_c = \begin{Bmatrix} F_c^1 \\ F_c^2 \\ \vdots \\ F_c^N \end{Bmatrix}, \quad F_c^I = \begin{cases} k_c(\phi^I - g^I) & \text{for a contact node} \\ 0 & \text{for no contact} \end{cases} \tag{35}$$

where contact at a particular node is indicated by an 1 in the corresponding diagonal entry in matrix $\underline{K}_c$, and k_c is the penalty constant. N is the total number of nodal points. Note that the contact force F_c at node I is calculated according to Eq. 35. ϕ^I is the current displacement at contact node I and g^I is the initial gap between node I and the contact surface.

2.2.3 Numerical Implementation

The numerical implementation of the adaptive arc-length control algorithm and contact algorithm is listed below:

Program starting point

1. Input shell configuration
2. Choose $\Delta\lambda_0$ arbitrarily
3. Start load step
   - 3.1 Assemble $\underline{K}^{(0)}$
   - 3.2 Check contact
     - If contact violated
       - then add $\underline{K}_c$
     - Endif
   - 3.3 Compute $\Delta\underline{u}^I = (\underline{K}^{(0)})^{-1}\underline{F}_0$, set $\Delta\underline{u}^{II} = 0$
   - 3.4 If load step $n = 1$
     - then compute $\Delta s = \Delta\lambda_0 \|\Delta\underline{u}^I\|$
       - set $\gamma_0 = 1$
     - Endif
   - 3.5 Compute $\Delta\lambda_{n+1}^{(1)} = \dfrac{\gamma_n \Delta s}{\|\Delta\underline{u}^I\|}$
   - 3.6 Compute $\Delta\underline{\Phi}_{n+1}^{(1)} = \Delta\lambda_{n+1}^{(1)}\Delta\underline{u}^I$
   - 3.7 Update both shell configuration and load factor by

$$\underline{\Phi}_{n+1}^{(1)} = \underline{\Phi}_{n+1}^{(0)} + \Delta\underline{\Phi}_{n+1}^{(1)} = \underline{\Phi}_n + \Delta\underline{\Phi}_{n+1}^{(1)}$$
$$\lambda_{n+1}^{(1)} = \lambda_{n+1}^{(0)} + \Delta\lambda_{n+1}^{(1)} = \lambda_n + \Delta\lambda_{n+1}^{(1)}$$

4. Start iterative procedure

4.1 Check convergence

If $\|\underline{F}_{n+1}^{(i)} - \underline{P}_{n+1}^{(i)} - \underline{F}_c^{(i)}\| \leq (Tol)_c$

then end iteration, go to 5.

endif

If divergence has occurred

then cut initial arc-length by $\frac{1}{2}$

Resume shell configuration of last load step

Go to 3.

endif

4.2 Assemble $\underline{K}^{(i)}$

4.3 Check contact

If contact violated

then add $\underline{K}_c$

Endif

4.3 Compute $\Delta\lambda_{n+1}^{(i)}$ and $\Delta\underline{\Phi}_{n+1}^{(i)}$ by Equations 21 and 22, respectively

4.4 Compute $\Delta\underline{u}^I$ and $\Delta\underline{u}^{II}$ by Equations 11 and 12, respectively

4.5 Update shell configuration

$$\underline{\phi}_{n+1}^{(i+1)} = \underline{\phi}_{n+1}^{(i)} + \Delta\underline{\phi}_{n+1}^{(i)}$$
$$\Delta\underline{t}_{n+1}^{(i)} = \bar{\Lambda}_{n+1}^{(i)}\Delta\bar{\underline{T}}_{n+1}^{(i)}$$
$$\underline{t}_{n+1}^{(i+1)} = \cos\|\Delta\underline{t}_{n+1}^{(i)}\|\underline{t}_{n+1}^{(i)} + \frac{\sin\|\Delta\underline{t}_{n+1}^{(i)}\|}{\|\Delta\underline{t}_{n+1}^{(i)}\|}\Delta\underline{t}_{n+1}^{(i)}$$
$$\underline{\Lambda}_{n+1}^{(i+1)} = (\underline{E}\cdot\underline{t}_{n+1}^{(i+1)})\underline{I}_3 + \widehat{\underline{E}\times\underline{t}_{n+1}^{(i+i)}} + \frac{1}{1+\underline{E}\cdot\underline{t}_{n+1}^{(i+1)}}(\underline{E}\times\underline{t}_{n+1}^{(i+1)})\otimes(\underline{E}\times\underline{t}_{n+1}^{(i+1)})$$

4.6 Update the load factor

$$\lambda_{n+1}^{(i+1)} = \lambda_{n+1}^{(i)} + \Delta\lambda_{n+1}^{(i)}$$

4.7 Go to 4.1

4.8 End iteration loop

5. Check load factor λ_{n+1}

If $\lambda_{n+1} - 1 \leq (Tol)_l$

then go to 6

else compute new acceleration factor γ_n

go to 3

6. End program

Note:

1. $(Tol)_c$: the error tolerance for convergence.
   $(Tol)_l$: the error tolerance for load factor.

3 Particle Modeling of Cloth

Particle modeling of cloth, as illustrated in Fig. 8, is based on a network of particle nodes on the cloth surface. In our implementation of this approach the nodes are associated with square elements on the surface. During deformation of the cloth these elements are allowed to distort in and out of plane. Internal strains are calculated according to the laws of the classic mechanics related to beam and flat shell deformations. In order to calculate the internal stresses, the nodes of the network are assumed to be connected by linear(or possibly nonlinear) springs which simulate the elastic properties of the fabric material. The stiffness of these springs is related to the material and physical properties of the actual fabric. These properties are the elastic modulus, shear modulus, Poisson's ratio, and thickness. Equilibrium of forces is then enforced at each of the nodal points.

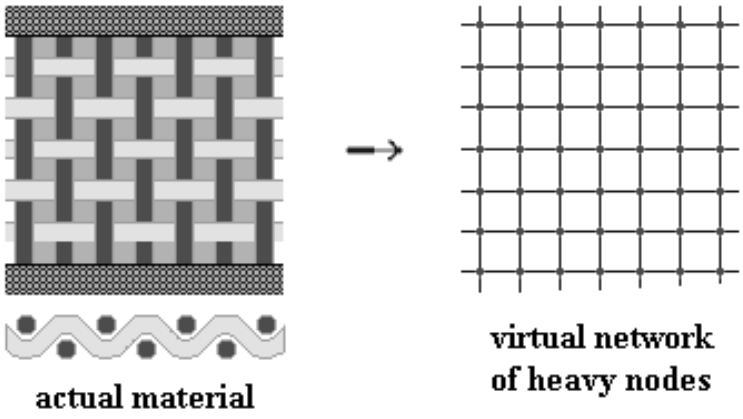

Figure 8: The Network Approximation

3.1 Physical Modeling

The steps involved in formulating the physically based model that leads to discrete equations of equilibrium at the nodal points are

- Kinematic description of the particle network
- Material stress-strain response
- Equilibrium of nodal forces

The network is a system of square elements where each vertex is a node. External forces and specified displacements are treated as input at each node. Internal reactions are calculated for each vertex considering the deformations (strains). In plane extension and shear strains are accounted for, as well as out-of-plane bending. Out-of-plane twisting and shearing is neglected.

For computing the in-plane stretching strain, the deformation of the edge of each cell of the network is considered. Elastic forces proportional to the unit elongation are applied to the opposite nodes of each edge. The distance b between two nodes of the undeformed network is taken as reference for the force calculation (see Fig. 9). The linear extension stress σ is defined as

$$\sigma = E\epsilon_{stretch} \tag{36}$$

where E=Young's modulus and $\epsilon_{stretch} = \frac{\Delta b}{b}$ is the unit elongation. The stretching forces $F_{stretch}$ subsequently applied to the nodes are related to the extension stress through the cross sectional area A

$$A = bh \tag{37}$$

where h is the cloth thickness. Then

$$F_{stretch} = D'EA\frac{\Delta b}{b} \tag{38}$$

where D' is a free parameter needed to allow fine tuning of the stretching strain (D'=1 for attraction forces and $D' < 1$ for repulsion forces).

The in-plane shearing forces applied to the nodes of the network are calculated based on some simplified assumptions. Only small in-plane shearing deformations are permitted and the calculation of strain assumes that all nodes within the element lie in the same plane. The actual fabric material has a more complex behavior compared

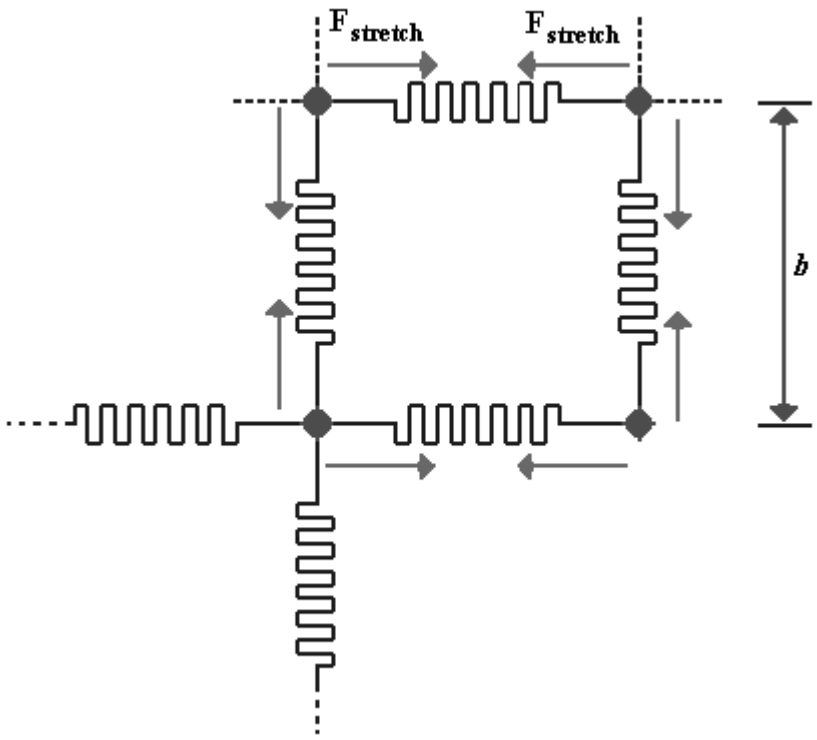

Figure 9: Stretching Forces

with the approximations made in our model. For instance, a simple weave of weft and warp thread reacts to a shearing stress, applied for example to a weft thread, with a torsional strain of the warp threads. Referring to Fig. 10, it can be shown that the shear angle ϕ is related to the change in length Δd of the diagonal according to

$$\phi = 2\frac{\Delta d}{d} \tag{39}$$

Since the shear stress-strain equation is

$$\tau = 2G\epsilon_{shear} = \frac{E}{1+\nu}\epsilon_{shear} \tag{40}$$

it follows that

$$\tau = \frac{E}{1+\nu}\frac{\Delta d}{d} \tag{41}$$

since $\epsilon_{shear} = \phi/2$.

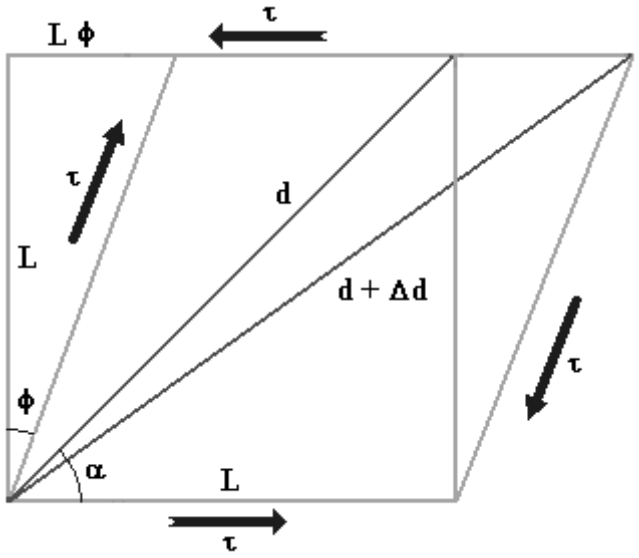

Figure 10: Shearing Strain

The elastic force F along the direction of the edge of the deformed cell is related to τ through the cross sectional area A (see Fig. 11).

$$F = \tau A = \frac{EA}{(1+\nu)}\frac{\Delta d}{d} \tag{42}$$

For small shear strains, a shearing force F_{shear} directed along the diagonal direction can be defined according to

$$F_{shear} = \frac{\sqrt{2}}{2}F = \frac{B'EA}{2b(1+\nu)}\Delta d \tag{43}$$

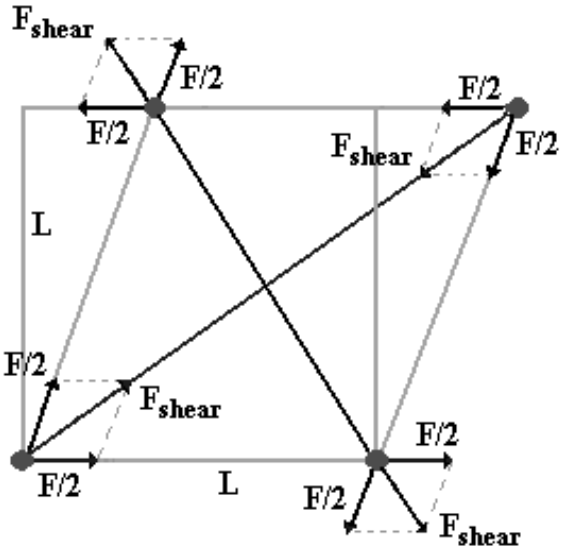

Figure 11: Shearing Forces

where B' is another free parameter needed to allow a fine tuning of the shearing strain (B'=1 for the longer diagonal and $B' < 1$ for the shorter diagonal).

Consideration of adjoining element edges is necessary to formulate bending strains. When two consecutive edges are not collinear after deformation, bending strain must exist. The angle $\Delta\phi$ between the two edges, as shown in Fig. 12, is a measure of the curvature. The bending moment M that results from the curvature is realized by applying two pairs of forces to the nodes. The relation between the bending moment M and the bending force F_{bend} is defined by

$$M = F_{bend} b \tag{44}$$

Then, it is possible using plate theory to construct the relationship

$$\frac{1}{R} = \frac{M(1-\nu^2)}{EI} = \frac{\Delta\phi}{b} \tag{45}$$

between the bending moment M and the angle $\Delta\phi$, where R is the radius of curvature, I is the second moment of area, and EI is the bending rigidity. The second

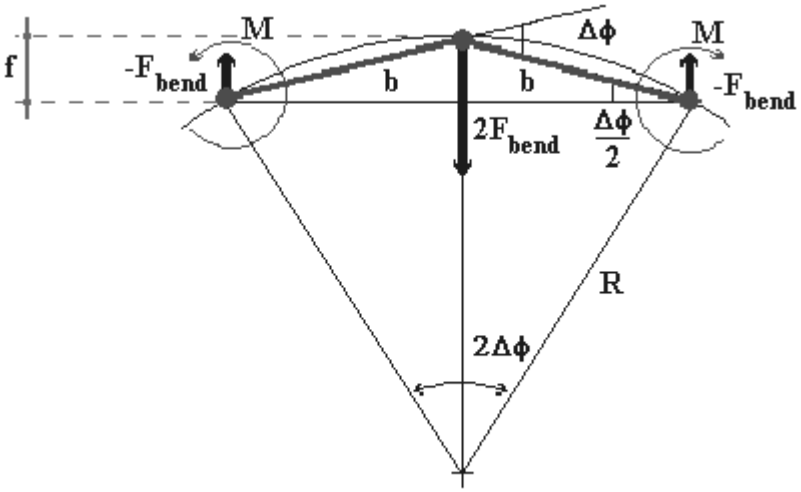

Figure 12: Bending Strain

moment of area I is

$$I = \frac{Ah^2}{12} \tag{46}$$

where A is again the cross sectional area and h is the material thickness. Using Eq.(44) it is then possible to obtain the needed expression for the bending force in terms of the deformation $\Delta\phi$

$$F_{bend} = \frac{EI}{b^2(1-\nu^2)}\Delta\phi \tag{47}$$

Similar treatment of stretching, shearing, and bending forces in all three coordinate directions (x, y, z) allows expressing equilibrium of forces at each of the nodes (i, j)within the network

$$\{W + F_{seams} + F_{ext} + F_{stretch} + F_{bend} + F_{shear}\}_{x,y,z}^{(i,j)} = 0 \tag{48}$$

where W is the gravity load at the node, F_{seams} is the sewing force (to be described later), and F_{ext} is the interaction force with external surfaces due to a potential collision.

3.2 Simulation Algorithm

The simulation algorithm computes the displacements of the nodes consistent with the above equilibrium equation. The algorithm pushes the network towards a configuration where the resultants of the forces applied to each node converge to zero.

There are no dynamic effects included. At every simulation step, the following operations are performed

- Gravity and external loads are applied to each node of the network and the strains due to stretching, bending and shearing are computed considering the deformations of the network
- Possible interactions with external surfaces are checked and repulsion forces are applied to the nodes if any collision is detected
- Displacements of the network nodes is generated in the direction of the resultant force acting at each node
- A correction in the new position of each node is eventually necessary according to sewing constraints or displacements imposed to some part of the fabric.

There is no self-collision algorithm in the particle model implementation described in this chapter. The example problems shown later for the purpose of comparing finite element and particle model methods do not require this feature. The self-collision algorithm we have developed is described in Chapter 9 of this book.
The deforming objects that we need to simulate can have a considerable surface area and require a substantial number of nodes to resolve this area accurately. An idea for accelerating the convergence process towards equilibrium is overloading the network with an extra load, i.e. pushing the nodes in the direction of their path at the previous step. The direction of the resultant displacement of each node at the previous step is used as the direction for the additional psuedo-load to apply to the node. If $Disp^{k-1}_{(i,j)}$ is the displacement of node (i,j) at step $k-1$, a coefficient C_{acc} with dimension $[force/length]$ is used for computing the psuedo-force

$$F^{k}_{acc(i,j)} = C_{acc} \cdot Disp^{k-1}_{(i,j)} \tag{49}$$

to apply to the node at step k. As equilibrium is approached, the displacement $Disp^{k-1}_{(i,j)}$ progressively decreases until it finally vanishes. At final equilibrium, the internal reactions perfectly balance the external loads and Eq. (48) is satisfied.

3.2.1 Sewing Process

The deforming cloth object can be constructed by assembling several flat 2D pattern pieces. These pieces must be "sewn" together before applying the complete simulation algorithm. During the first simulation steps no external loads are applied and no collision detection with external surfaces is performed. Instead sewing forces are applied to the nodes of the different patterns along lines that are guaranteed to result in the pattern pieces coming together at predetermined points (see Fig. 13). The internal reactions of the network are then calculated and the displacements of the

nodes are imposed. When the borders which have to be sewn are in mutual contact, the simulation algorithm starts to apply the external loads to the network and the borders, already sewed, are kept in contact by correcting the position of the involved nodes at every simulation step. Also collisions with external surfaces are checked. With this approach, the resistance of the seams can not be simulated but it is easy to modify the algorithm to take this into account by continuing to apply sewing forces during the evolution of the network.

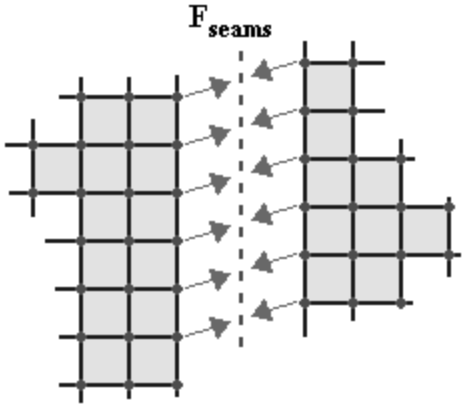

Figure 13: Sewing Algorithm

3.2.2 Collision Algorithm

A simple collision algorithm is performed in order to prevent the penetration of the fabric through external surfaces. The routine works on surfaces with simple geometry. When a collision between a node and an external surface is detected, a repulsion force is applied to the node involved in the impact. The direction of this force is normal to the surface at the contact point, while the magnitude is equal to the component of the total resultant force acting at the node before the collision in the direction of the normal vector to the surface. In this way, the algorithm drives to zero the component of the total resultant force normal to the surface after the collision. This process is depicted in Fig. 14. For cloth self-collision, a much more

involved algorithm is required. The one we have employed is covered in Chapter 9 of this book.

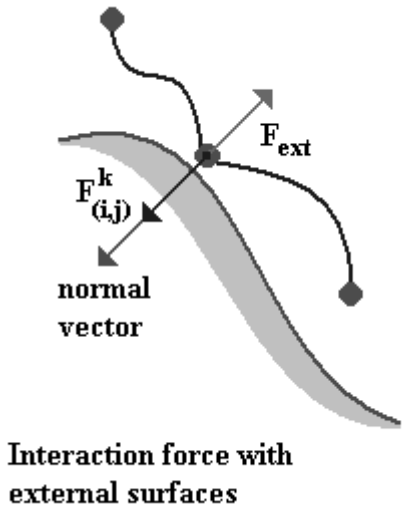

Figure 14: Collision Algorithm

3.2.3 Numerical Implementation

The overall particle model simulation algorithm is listed below

Program starting point

1. Load two-dimensional pattern shapes, seam configurations, and material properties
2. Each node is loaded with its weight W and the acceleration force $F^k_{acc(i,j)}$
   $F^k_{(i,j)} = W + C_{acc} \cdot Disp^{k-1}_{(i,j)}$
3. Internal reactions of the network are computed and added
   $F^k_{(i,j)} \Leftarrow F^k_{(i,j)} + F_{stretch} + F_{shear} + F_{bending}$
4. Sewing algorithm is performed if necessary and these forces are added
   $F^k_{(i,j)} \Leftarrow F^k_{(i,j)} + F_{seams}$
5. Collision algorithm with external surfaces is performed and these forces are added
   $F^k_{(i,j)} \Leftarrow F^k_{(i,j)} + F_{ext}$
6. Displacements are calculated for each node of the network

$Disp^k_{(i,j)} = C_{disp} \cdot F^k_{(i,j)}$
where C_{disp} is the displacement coefficient with $[C_{disp}] = length/force$

7. Convergence criteria
   If $Disp^k_{(i,j)} < e$ with $e = \frac{b}{10{,}000}$ then go to 8.
   else $k = k + 1$, go to 2.
   Endif
8. End program.

3.3 Dynamic Particle Model

With the particle model described in this chapter, the discretization of the deforming material is provided before applying any external load or physical property. Then the internal strains of the network are computed locally considering the deformations of the network. When the displacements are calculated, the nodes of the network are treated as independent and not coupled to other nodes. With this approach it is straightforward to extend the static particle model to a dynamic model where accelerations and velocities are considered and integrated using a numerical method. After the calculation of the internal strains is performed and each particles is acted upon by a resultant force, a direct numerical integration of Newton's second law can be applied. We adopted a fourth order accurate Runge-Kutta method. Runge-Kutta is a conditionally stable method, and an appropriate time step must be selected. The equation of motion is now

$$\{F_{dyn}(\underline{x},t) - W - F_{seams}(\underline{x},t) - F_{ext}(\underline{x},t) - F_{stretch}(\underline{x},t) - F_{bend}(\underline{x},t) - F_{shear}(\underline{x},t)\}^{i,j}_{x,y,z} = 0 \tag{50}$$

where $\underline{x} = (x, v)$ can be viewed as a state vector storing x=position and v=velocity at each node and $F_{dyn}(\underline{x},t) = m\frac{dv}{dt}$, where m=mass at the node and $\frac{dv}{dt}$ is the acceleration of the node. The dynamic particle method involves nodal velocity, thus allowing dissipative forces. Furthermore it is also possible to simulate collisions involving velocity changes that are calculated according to classical impact laws via coefficients of restitution.

4 Results: Comparing Finite Element and Particle Methods

One purpose of this section is to provide some benchmark results that researchers writing new codes can use to validate their models. Several classical results from beam theory serve this purpose very well. Beam problems are essentially one-dimensional as contrasted to the two-dimensional surface theory we have outlined. To force the finite element or particle model to follow the one-dimensional theory, it

is sufficient to consider just one row of finite elements or two lines of particles. Nodes on the boundary are constrained to have zero displacement and rotation. We also show a draping tablecloth and cloth drape on a mannequin. The reader is referred to Chapter 9 for further simulation results.

4.1 Tip Loaded Cantilever Beam

A simple beam with a single force applied at the end is an excellent test case for a cloth model. The initial configuration of the beam is shown in Fig. 15. Axial and bending rigidity are the elastic properties specified for the problem. A variable load P is applied at the end of the beam. The load P is a linear function of the non-dimensional parameter λ_P with the $0.0 \leq \lambda_P \leq 10.0$. The baseline exact solution is provided by elastica theory. The results for the tip loaded cantilever are in terms of elliptic integrals and cannot be presented explicitly.

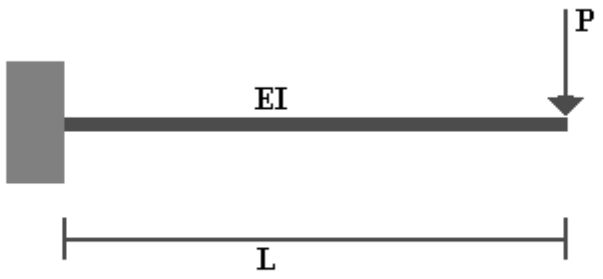

Figure 15: Tip Loaded Cantilever Beam: Initial Configuration

The elastic and geometric properties are defined as: $L = 1.0\ length \Rightarrow$ Length of the beam
$EA = 2\sqrt{3} \times 10^{-3}\ force \Rightarrow$ Axial rigidity
$EI = 2\sqrt{3} \times 10^{-9}\ force \cdot length^2 \Rightarrow$ Bending rigidity
$P = \lambda_P \frac{EI}{L^2}\ force \Rightarrow$ Force applied at the end of the beam
$1.0 \leq \lambda_P \leq 10.0$

Number of elements/particles = 8

The graph in Fig. 17 shows the results for the dimensionless tip deflections and rotation obtained with the two methods. In this example, the finite element method provides essentially exact results, while the particle method produces some error.

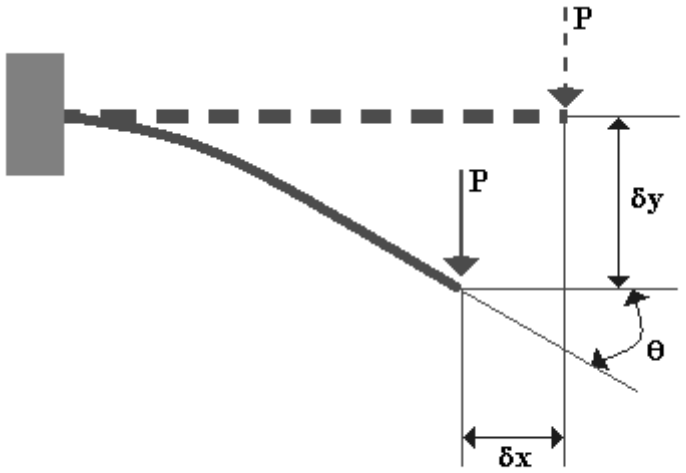

Figure 16: Tip Loaded Cantilever Beam: Final Configuration

4.2 Moment Loaded Cantilever Beam

In this case a variable moment M is applied at the end of the beam (see Fig. 18) and the simulation is performed using the following mechanical and geometric properties:
$L = 1.0\ length \Rightarrow$ Length of the beam
$EA = 2\sqrt{3} \cdot 10^{-3}\ force \Rightarrow$ Axial rigidity

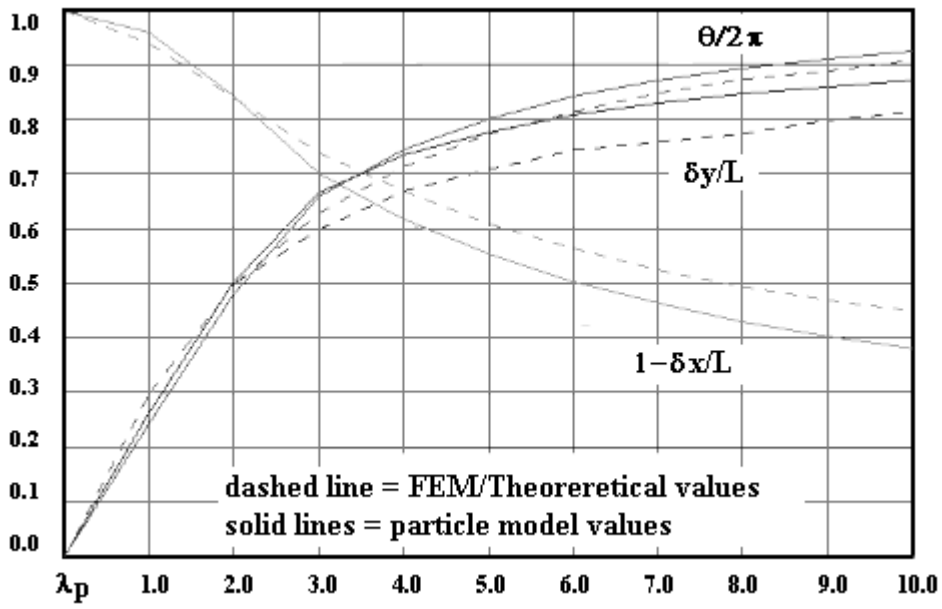

Figure 17: Tip Loaded Cantilever Beam: Results

$EI = 2\sqrt{3} \cdot 10^{-9}\ force \cdot length^2 \Rightarrow$ Bending rigidity
$M = \lambda_M \pi \frac{EI}{L}\ force \Rightarrow$ Force applied at the end of the beam
$1.0 \leq \lambda_M \leq 2.0$
Number of elements/particles = 8

The graph in Fig. 19 shows the results for the dimensionless tip deflections and rotation obtained with the two methods. In this example, again the finite element method provides essentially exact results, while the particle method produces some error.

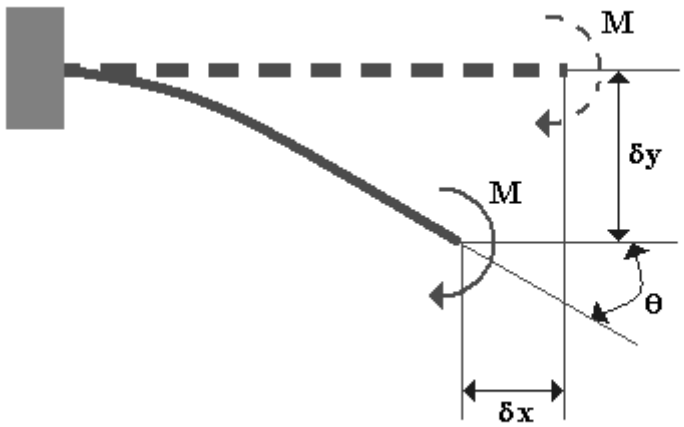

Figure 18: Moment Loaded Cantilever Beam: Final Configuration

4.3 Overhanging Beam

Fig. 20 shows a beam that partially overhangs an edge. Before deformation, a span of the beam is in contact with the rigid horizontal contact surface, while the remaining span a is free to displace vertically. When the self-weight of the beam acts, a part of the beam (length p) is free to lift off the contact surface. This distance is not known a-priori and must be determined during the simulation. The deflection δy (see Fig. 21) and the reactions forces (R_b and R_c) at points B and C are the unknowns that are chosen to compare the two approaches.

The elastic and geometric properties for the overhanging beam are:
$L = 6.0\ cm \Rightarrow$ Length of the beam
$a = 1.5\ cm \Rightarrow$ Distance past the edge
$EA = 2809.0\ gf \Rightarrow$ Axial rigidity

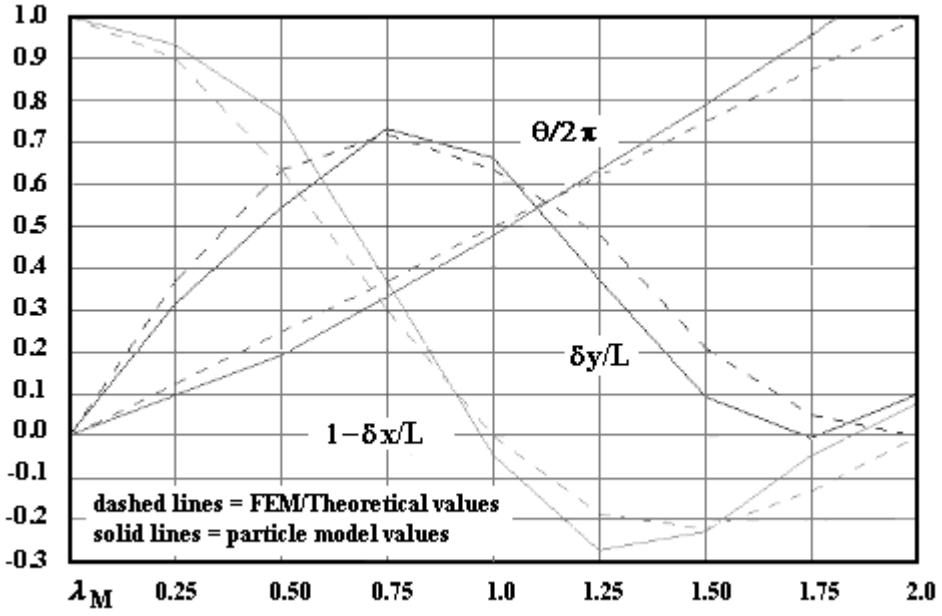

Figure 19: Moment Loaded Cantilever Beam: Results

$EI = 0.386\ gf \cdot cm^2 \Rightarrow$ Bending rigidity
$w = 0.0255\ gf/cm \Rightarrow$ Weight density
Number of elements/particles = 24

Classic mechanics provides a theoretical solution for this problem which we compare with the response of our simulations in the following table

24 elements	$p[cm]$	$\delta_y[cm]$	$R_b[gf]$	$R_c[gf]$
Theoretical	2.1213	0.0812	0.0135	0.0788
F.E.M.	2.0999	0.0810	0.0022	0.0785
% error	1.01	0.25	83.7	0.38
Particle Model	2.2499	0.0966	0.0156	0.0813
% error	6.06	18.0	15.52	3.15

The finite element method predicts the displacement and contact distance quite accurately, but fails to produce an accurate answer for the reaction at the lift-off point. Generally, for these one-dimensional problems the finite element method gives better results for the deflections. This is due mainly to the fact that the finite element method enforces continuity of slope on the surface, whereas the particle method does not.

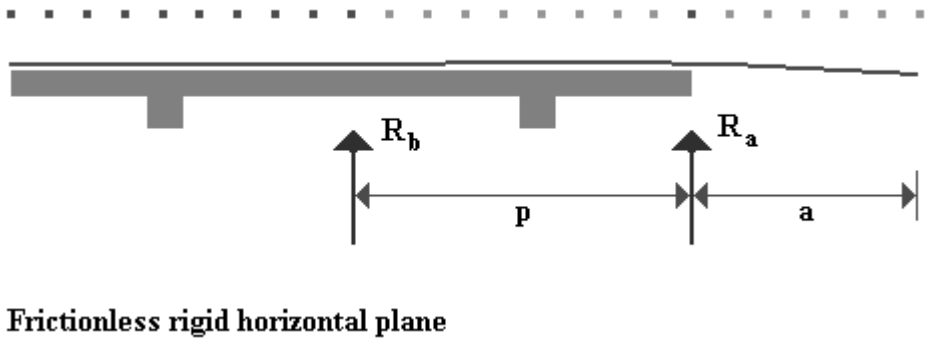

Figure 20: Overhanging Beam: Initial Configuration

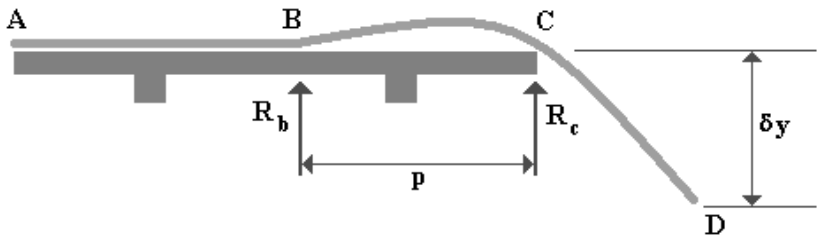

Figure 21: Overhanging Beam: Final Configuration

4.4 Tablecloth on a Square Table

The simulated tablecloth is a square sample of fabric which has surface dimensions 20 cm by 20 cm. For both simulation methods we have used a 20 by 20 grid of 400 elements and 441 nodes. The tablecloth is centered on an 8 cm by 8 cm rigid frictionless table. The physical properties listed below correspond to a cotton woven material.
$L = 20.0\ cm \Rightarrow$ Edge of the tablecloth
$E_{eff} = 2.688 \cdot 10^3\ oz/inch^2 = 118.1176\ gf/mm^2 \Rightarrow$ Young's Modulus
$w = 5.425 \cdot 10^{-3}\ oz/inch^2 = 0.2384 \cdot 10^{-3}\ gf/mm^2 \Rightarrow$ Weight density
$h = 0.02135\ inch = 0.54229\ mm \Rightarrow$ Thickness
$\nu = 0.3 \Rightarrow$ Poisson's ratio

The final result of the particle method simulation (finite element result looks the same on the scale of the figure) is shown in Fig. 22. The computational time required for reaching the static equilibrium condition is 15 minutes on a 200Mhz Pentium machine. The lower section of the draped tablecloth, projected on the xz plane (see Fig. 23), is used for comparing the simulated response with experimental results. An actual cotton cloth was draped over a cube in order to allow comparison with the simulations. The final positions of the midside and corner nodes were compared.

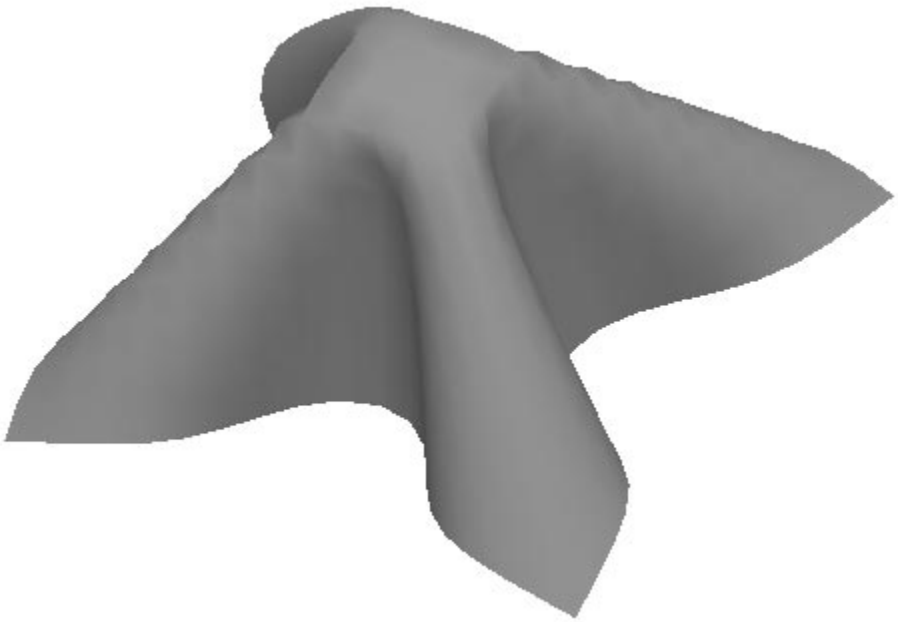

Figure 22: Tablecloth on a Square Table: Final Configuration, Particle Method

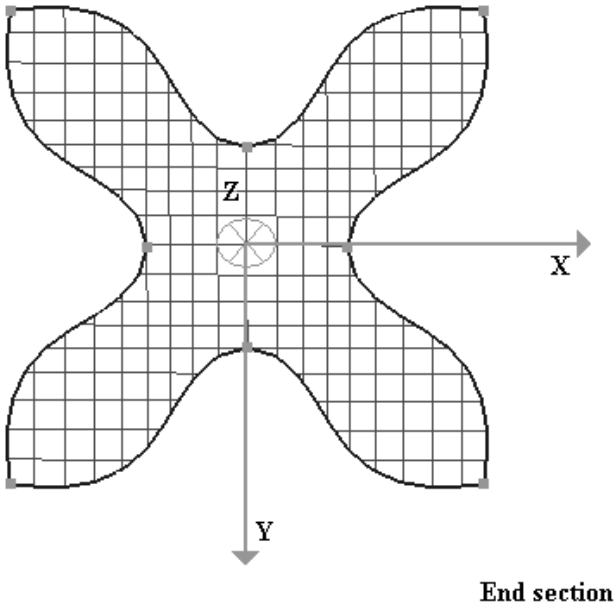

Figure 23: Tablecloth on a Square Table: Lower Section

The following table shows that both methods accurately predict the final drape shape.

	Corner Node			*Midside Node*	
	x/L	y/L	z/L	y/L	z/L
Experimental	0.39	0.38	0.39	0.18	0.40
Particle Model	0.39	0.39	0.38	0.16	0.37
% error	0.45	3.12	1.79	7.90	5.34
F.E.M.	0.41	0.41	0.36	0.19	0.38
% error	5.12	7.32	7.69	5.26	5.00

5 Clothing Drape Results

The possibility to simulate complex fabric part shapes using the physical and mechanical properties of the actual material allows modeling of important problems in both clothing manufacturing and in fashion, such as

- Manufacturing process simulation;

- Garment design and simulation;
- Virtual clothing evaluation;
- Virtual fashion.

Fig. 24 and Fig. 25 show examples of clothes draping on a mannequin. The draping of the dress shown in Fig. . 24 requires only a few minutes of computational time on a 200 MHz pentium machine. When the configuration results in a high degree of wrinkling, as in the case of the skirt in Fig. 25, the simulation time increases substantially. Both the simulations are realized with a static particle model on a square network. The mesh has been triangularized for the rendering of the surfaces.

6 Conclusions

The finite element and particle methods of computing fabric drape have been presented in sufficient detail to allow implementation by new researchers in the field. Numerical results show that the methods compare relatively well, with the finite element method exhibiting a slight edge in benchmarks against theoretical solutions. Where high accuracy is required, the finite element method should be given serious consideration. Where computational speed and qualitatively correct results are required, the particle method is recommended. It should be noted that the particle approach is substantially simpler to implement.

7 Acknowledgments

Financial support for this work was provided by the National Textile Center which is funded by the US Department of Commerce.

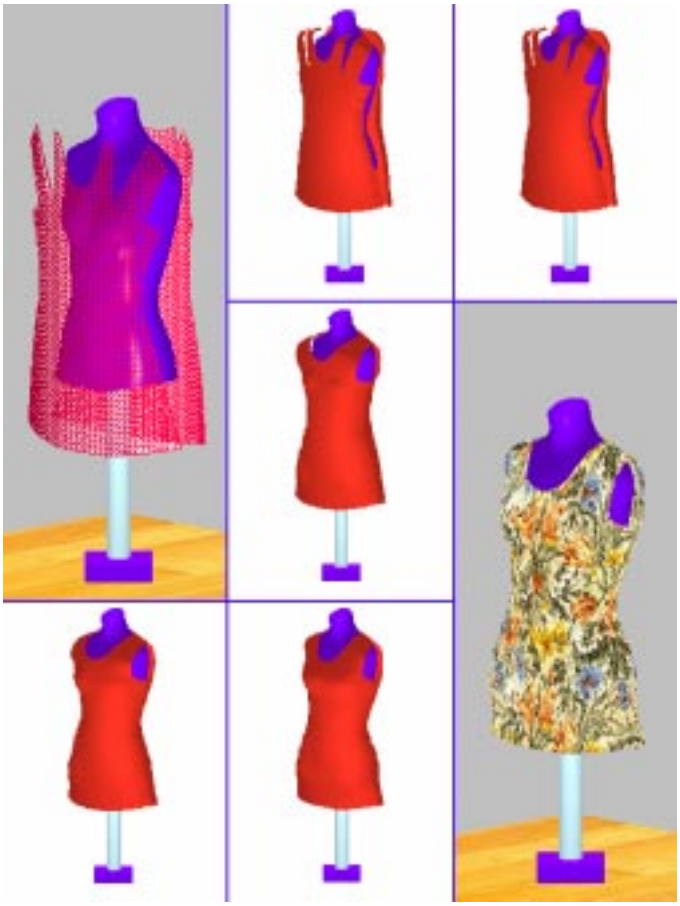

Figure 24: Clothing Drape

References

[1] Imaoka H., at. el., (1981), "Analysis of Drapeability by Computer," 1981 Annual Conference Preprints, Research Institute for Polymers and Textiles, Tsukuba City, Japan, pp. 1-9.

[2] Izumi, K. and Niwa, W. ,(1985), "Evaluation of Dynamic Drape of Ladies' Dress Fabrics,"Objective Measurement: Applications to Product Design and Process Control, The Textile Machinery Society of Japan, pp. 725-734.

[3] Weil, J. ,(1986), "The Analysis of Cloth Objects," Proceedings of the Association for Computing Machinery Conference (SIGGRAPH '86), Dallas, TX, pp.49-53.

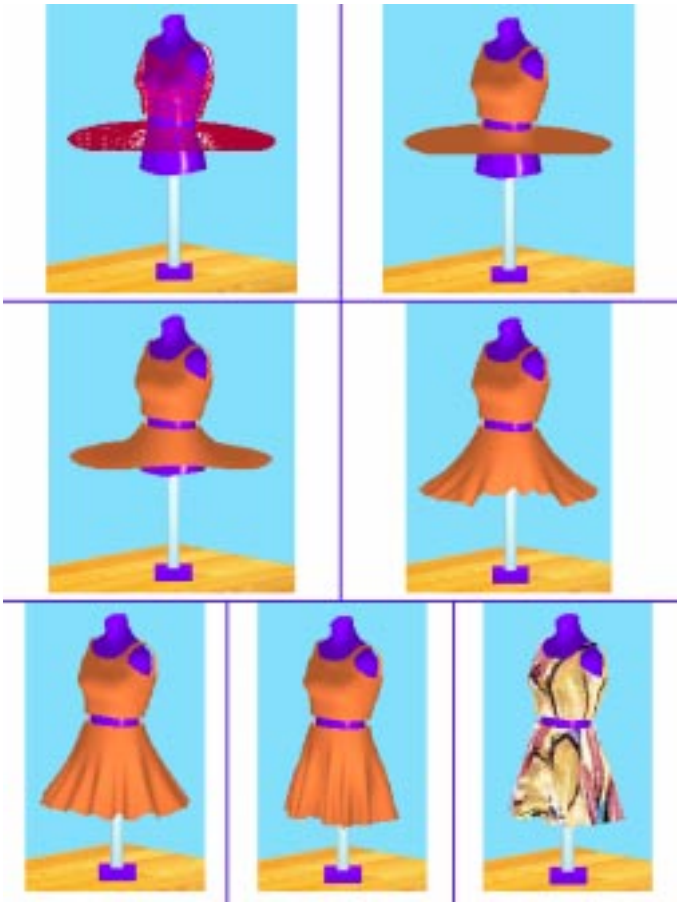

Figure 25: Clothing Drape

[4] Terzopoulos, D., et. al., (1987), "Elastically Deformable Models," Proceedings of the Association for Computing Machinery Conference (SIGGRAPH '87), Anaheim, CA, pp. 205-214.

[5] Von Der Weeën F., (1991) "Algorithms for Draping Fabrics on Double-Curved Surfaces," International Journal for Numerical Methods in Engineering, Vol. 32, pp. 1415-1426.

[6] Gan, L., et. al., (1991), "A Finite Element Analysis of the Draping of Fabrics," The 6th International Conference in Australia on Finite Element Methods, pp. 402-414.

[7] Gan, L., et. al. (1995), "A Study of Fabric Deformation Using Nonlinear Finite Elements," *Textile Research Journal*, Vol. 65, No. 11, pp. 660-668.

[8] Kim, J.,(1991), "Fabric Mechanics Analysis Using Large Deformation Orthotropic Shell Theory," Ph.D. Thesis, Mechanical and Aerospace Engineering Department, North Carolina State University.

[9] Simo, J. C., and Fox, D. D. ,(1989),"On a Stress Resultant Geometrically Exact Shell Model. Part I: Formulation and Optimal Parameterization," *Computer Methods in Applied Mechanics and Engineering*, Vol. 72, pp. 267-304.

[10] Simo, J. C., Fox, D. D., and Rifai, M. S., (1989),"On a Stress Resultant Geometrically Exact Shell Model. Part II:The Linear Theory; Computational Aspects," *Computer Methods in Applied Mechanics and Engineering*, Vol. 73, pp. 53-92.

[11] Simo, J. C., Fox, D. D., and Rifai, M. S., (1990),"On a Stress Resultant Geometrically Exact Shell Model. Part III:Aspects of the Nonlinear Theory," *Computer Methods in Applied Mechanics and Engineering*, Vol. 79, pp. 21-70.

[12] Collier, J. R., et. al., (1991), "Drape Prediction by Means of Finite-Element Analysis," *Journal of the Textile Institute*, Vol. 82, No. 1, pp. 96-107.

[13] Breen, D., House, D. H., and Wozny, M. J., (1994), "A Particle-Based Model for Simulating the Draping Behavior of Woven Cloth," *Textile Research Journal*, Vol. 64, No. 11, pp. 663-685.

[14] Deng, S., (1994), "Nonlinear Fabric Mechanics Including Material Nonlinearity, Contact, and An Adaptive Global Solution Algorithm," Ph.D. Thesis, Mechanical and Aerospace Engineering Department, North Carolina State University.

[15] Kang, T. J. and Yu, W. R.,(1995), "Drape Simulation of Woven Fabric by Using the Finite-Element Method," *Journal of the Textile Institute*, Vol. 86, No. 4, pp. 635-648.

[16] Chen, B., and Govindaraj, M., (1995), "A Physically Based Model of Fabric Drape Using Flexible Shell Theory" *Textile Research Journal*, Vol. 65, No. 6, pp. 324-330.

[17] Volino, P., Courchesne, M., and Thalmann, N. M., (1998), "Versatile and Efficient Techniques for Simulating Cloth and Other Deformable Objects," Course 31- Cloth and Clothing in Computer Graphics, 25th SIGGRAPH Conference, pp. D-41 - D-50.

[18] Baraff, D., and Witkin, A., (1998), "Large Steps in Cloth Animation, Proceedings of the 1998 SIGGRAPH Conference, pp. 43-54.

[19] Hu, J. L., and Teng, J. G.., (1996), "Computational Fabric Mechanics: Present Status and Future Trends" *Finite Elements in Analysis and Design*, Vol. 21, pp. 225-237.

[20] Eischen, J. W., (1998), "Technical Issues, Approaches, and Challenges," Course 31- Cloth and Clothing in Computer Graphics, 25th SIGGRAPH Conference.

[21] Chung, T. J.,(1988), Continuum Mechanics, Prentice -Hall.

[22] Zienkiewicz,
O. C., and Taylor, R. L., (1989), The Finite Element Method- Vol. 1, Fourth Edition, McGraw-Hill.

[23] Kawabata, S., (1975), "The Standardization and Analysis of Hand Evaluation," The Hand Evaluation and Standardization Committee. The Textile Machinery Society of Japan, Osaka, Japan.

[24] Schweizerhof, K. H., and Wriggers, P., (1986),"Consistent Linearization of Path Following Methods in Nonlinear FE Analysis," *Computer Methods in Applied Mechanics and Engineering*, Vol. 72, pp. 267-304.

[25] Cook, R. D., Malkus, D. S. ,and Plesha, M. E., (1989), Concepts and Applications of Finite Element Analysis, Third edition, John Wiley & Sons.

A Appendix- Details of the Nonlinear Shell Theory

A.1 Kinematic Description of the Shell

The covariant tangent base vectors on the mid-surface are defined by the partial derivatives of $\underline{\Phi}$ with respect to ξ^α (with $\xi = 0$).

$$\underline{a}_\alpha \equiv \frac{\partial \underline{\Phi}(\xi^1, \xi^2, 0)}{\partial \xi^\alpha} = \underline{\phi}_{,\alpha} \qquad \text{for } \alpha = 1, 2 \tag{1}$$

Likewise, the covariant tangent base vectors off the reference surface are represented by

$$\underline{g}_\alpha = \frac{\partial \underline{\Phi}(\xi^1, \xi^2, \xi)}{\partial \xi^\alpha} = \underline{\phi}_{,\alpha} + \xi \underline{t}_{,\alpha} \tag{2}$$

or

$$\underline{g}_\alpha = \underline{a}_\alpha + \xi \underline{t}_{,\alpha} \tag{3}$$

Differentiating Eq. 1 by ξ, we obtain

$$\underline{g}_3 \equiv \underline{a}_3 = \underline{t} \tag{4}$$

The contravariant vector $\underline{a}^i$ has the property

$$\underline{a}^i \cdot \underline{a}_j = \delta^i_j \qquad \text{for } \; i, j = 1, 2, 3 \tag{5}$$

$$\underline{a}^{\alpha} = a^{\alpha\beta}\underline{a}_{\beta}, \qquad \underline{a}_{\alpha} = a_{\alpha\beta}\underline{a}^{\beta} \tag{6}$$

where $a_{\alpha\beta} = \underline{a}_{\alpha} \cdot \underline{a}_{\beta}$ and $a^{\alpha\beta} = \underline{a}^{\alpha} \cdot \underline{a}^{\beta}$ are the covariant and contravariant components of the metric tensor, respectively. δ^i_j is the Kronecker delta. It also follows that

$$a^{\alpha\beta}a_{\beta\gamma} = \delta^{\alpha}_{\gamma} \tag{7}$$

$$|a_{\alpha\beta}| = a_{11}a_{22} - a_{12}a_{21} = a \tag{8}$$

$$|a^{\alpha\beta}| = a^{11}a^{22} - a^{12}a^{21} = \frac{1}{a} \tag{9}$$

$|\bullet|$ indicates a determinant operation. A similar construction holds for obtaining $\underline{g}^{\alpha}$. The cross product of the covariant tangent vectors takes the form

$$\underline{a}_i \times \underline{a}_j = \sqrt{a}e_{ijk}\underline{a}^k \tag{10}$$

where e_{ijk} is the permutation tensor. Similarly, the cross product of the contravariant tangent vectors takes the form

$$\underline{a}^i \times \underline{a}^j = \frac{1}{\sqrt{a}}e^{ijk}\underline{a}_k \tag{11}$$

Here, e^{ijk} functions identically as e_{ijk}. From Eq. 10,

$$\underline{a}^3 = \frac{1}{\sqrt{a}}(\underline{a}_1 \times \underline{a}_2)$$

Using Eq. 5,

$$\sqrt{a} = (\underline{a}_1 \times \underline{a}_2) \cdot \underline{a}_3 \tag{12}$$

The element of area of a parallelogram (see Fig. 1) with two adjacent edges $\underline{a}_1 d\xi^1$ and $\underline{a}_2 d\xi^2$ for any configuration of the shell is given by

$$d\mathcal{A} = \|\underline{a}_1 \times \underline{a}_2\| d\xi^1 d\xi^2 = \overline{j} d\xi^1 d\xi^2 \tag{13}$$

The notation $\|\bullet\|$ indicates a vector Euclidean norm. Likewise, the element of mid-surface area for the reference configuration is given by

$$d\mathcal{A}_0 = \|\underline{a}^0_1 \times \underline{a}^0_2\| d\xi^1 d\xi^2 = \overline{j}^0 d\xi^1 d\xi^2 \tag{14}$$

A.1.1 Strain measures

The relative strain measures are defined as

Membrane strain :

$$\varepsilon_{\alpha\beta} \equiv \frac{1}{2}(\underline{\phi}_{,\alpha} \cdot \underline{\phi}_{,\beta} - \underline{\phi}_{0,\alpha} \cdot \underline{\phi}_{0,\beta}) \tag{15}$$

Transverse shear strain :

$$\gamma_{\alpha} \equiv \underline{\phi}_{,\alpha} \cdot \underline{t} - \underline{\phi}_{0,\alpha} \cdot \underline{t}_0 \tag{16}$$

Bending Strain :

$$\kappa_{\alpha\beta} \equiv \underline{\phi}_{,\alpha} \cdot \underline{t}_{,\beta} - \underline{\phi}_{0,\alpha} \cdot \underline{t}_{0,\beta} \tag{17}$$

A.2 Constitutive Equations

From Chung [21], the constitutive equations for linear elastic isotropic materials in a non-orthogonal, non arclength *curvilinear* coordinate system are given by

Membrane :

$$\tilde{n}^{\alpha\beta} = hC^{\alpha\beta\gamma\delta}\varepsilon_{\gamma\delta} \tag{18}$$

Bending :

$$\tilde{m}^{\alpha\beta} = \frac{h^3}{12}C^{\alpha\beta\gamma\delta}\kappa_{\gamma\delta} \tag{19}$$

Transverse Shear :

$$\tilde{q}^{\alpha} = \kappa h C^{\alpha 3\gamma 3}\gamma_{\gamma} \tag{20}$$

where $\alpha, \beta, \gamma, \delta = 1, 2$, The elastic constants $C^{\alpha\beta\gamma\delta}$ are

$$C^{\alpha\beta\gamma\delta} = \frac{E}{1-\nu^2}\left[\nu a^{0^{\alpha\beta}} a^{0^{\gamma\delta}} + \frac{1-\nu}{2}(a^{0^{\alpha\gamma}} a^{0^{\beta\delta}} + a^{0^{\alpha\delta}} a^{0^{\gamma\beta}})\right] \tag{21}$$

$$C^{\alpha 3\gamma 3} = \kappa G_s \begin{bmatrix} a^{0^{11}} & a^{0^{12}} \\ a^{0^{12}} & a^{0^{22}} \end{bmatrix} \tag{22}$$

and

$$a^{0^{\alpha\beta}} = \underline{a}^{0^{\alpha}} \cdot \underline{a}^{0^{\beta}} \tag{23}$$

$$G_s = \frac{E}{2(1+\nu)} \tag{24}$$

E is the modulus of elasticity, ν is Poisson's ratio, and κ is the shear correction factor.

For the nonlinear bending response, it is assumed that

$$\tilde{m}^{11} = f_1(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) \tag{25}$$

$$\tilde{m}^{22} = f_2(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) \tag{26}$$

$$\tilde{m}^{12} = f_3(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) \tag{27}$$

where f_1, f_2, f_3 are as yet to be determined functions of the curvatures κ_{11}, κ_{22} and the twist $\kappa_{(12)}$. Based on the Kawabata bending test data, these functions can conveniently be expressed as 5th order polynomials

$$\begin{aligned} f_i(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) = {} & a_{i1}\kappa_{11} + a_{i2}\kappa_{11}^2 + a_{i3}\kappa_{11}^3 + a_{i4}\kappa_{11}^4 + a_{i5}\kappa_{11}^5 \\ & + b_{i1}\kappa_{22} + b_{i2}\kappa_{22}^2 + b_{i3}\kappa_{22}^3 + b_{i3}\kappa_{22}^3 + b_{i5}\kappa_{22}^5 \\ & + c_{i1}\kappa_{(12)} + c_{i2}\kappa_{(12)}^2 + c_{i3}\kappa_{(12)}^3 + c_{i4}\kappa_{(12)}^4 + c_{i5}\kappa_{(12)}^5 \end{aligned} \tag{28}$$

where $a_{i1}, \cdots, c_{i5}$ are constants and $i = 1, 2, 3$. Eqs. 25 - 27 can also be expressed in matrix form

$$\begin{Bmatrix} \tilde{m}^{11} \\ \tilde{m}^{22} \\ \tilde{m}^{12} \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} f_1(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) \\ f_2(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) \\ f_3(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) \end{Bmatrix} = \underline{\mathbf{1}}\underline{F}(\underline{\kappa}) \tag{29}$$

where

$$\underline{F}(\underline{\kappa}) \equiv \left\{ \begin{array}{c} f_1(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) \\ f_2(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) \\ f_3(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) \end{array} \right\} \tag{30}$$

The nonlinear moment-curvature response obtained from the Kawabata test on a 1cm wide strip of fabric is in the form

$$\begin{aligned} M &= f(K) \\ &= \alpha_1 K + \alpha_2 K^2 + \alpha_3 K^3 + \alpha_4 K^4 + \alpha_5 K^5 \end{aligned} \tag{31}$$

Assuming isotropic behavior and that Poisson's ratio is independent of the curvature, it is possible to expand this one-dimensional response to the three-dimensional shell configuration, then

$$\begin{aligned} f_1(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) &= \frac{1}{1-\nu^2} \Big\{ a^{011} a^{011} f(\kappa_{11}) + \Big[\nu a^{011} a^{022} + (1-\nu) a^{012} a^{012} \Big] f(\kappa_{22}) \\ &\quad + a^{011} a^{012} f(\kappa_{(12)}) \Big\} \end{aligned} \tag{32}$$

$$\begin{aligned} f_2(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) &= \frac{1}{1-\nu^2} \Big\{ \Big[\nu a^{011} a^{022} + (1-\nu) a^{012} a^{012} \Big] f(\kappa_{11}) \\ &\quad + a^{022} a^{022} f(\kappa_{22}) + a^{022} a^{012} f(\kappa_{(12)}) \Big\} \end{aligned} \tag{33}$$

$$\begin{aligned} f_3(\kappa_{11}, \kappa_{22}, \kappa_{(12)}) &= \frac{1}{1-\nu^2} \Big\{ a^{011} a^{012} f(\kappa_{(11)}) + a^{022} a^{012} f(\kappa_{(22)}) \\ &\quad + \left[\frac{1-\nu}{2} a^{011} a^{022} + \frac{1+\nu}{2} a^{012} a^{012} \right] f(\kappa_{(12)}) \Big\} \end{aligned} \tag{34}$$

The relationships between the constants $a_{i1}, \cdots, c_{i5}$ and $\alpha_1, \cdots, \alpha_5$ should be clear.

A.3 Static Weak Form of the Equilibrium Equations

The static weak form or the statement of virtual work for the geometrically exact shell theory is, according to Simo, et. al. [9]

$$G(\underline{\Phi}, \delta\underline{\Phi}) = \int_{\mathcal{A}} [\tilde{n}^{\alpha\beta} \delta\varepsilon_{\alpha\beta} + \tilde{q}^{\alpha} \delta\gamma_{\alpha} + \tilde{m}^{\alpha\beta} \delta\kappa_{\alpha\beta}] d\mathcal{A} - G_{ext}(\delta\underline{\Phi}) \tag{35}$$

or in matrix form

$$G(\underline{\Phi}, \delta\underline{\Phi}) = \int_{\mathcal{A}} \underline{B} \left\{ \begin{array}{c} \delta\phi \\ \delta\bar{\underline{T}} \end{array} \right\} \cdot \underline{R} d\mathcal{A}_0 - G_{ext}(\delta\underline{\Phi}) \tag{36}$$

where $\underline{B}$ is a differential operator and $\underline{R}$ is the total resultant stress vector. They are defined as

$$\underline{B} = \left[\begin{array}{cc} \underline{B}_m & \underline{0}_{3\times 2} \\ \underline{B}_{sm} & \underline{B}_{sb} \\ \underline{B}_{bm} & \underline{B}_{bb} \end{array} \right]_{8\times 5} \qquad \underline{R} \equiv \left\{ \begin{array}{c} \tilde{\underline{N}} \\ \tilde{\underline{Q}} \\ \tilde{\underline{M}} \end{array} \right\} = \frac{\bar{j}}{\bar{j}^0} \left\{ \begin{array}{c} \tilde{\underline{n}} \\ \tilde{\underline{q}} \\ \tilde{\underline{m}} \end{array} \right\} \tag{37}$$

G_{ext} is the external force operator.

A.4 Finite Element Formulation

The static weak form of the equilibrium equations represents a highly nonlinear system of integral equations. In order to solve these equations efficiently, one must linearize them to take advantage of the convergence characteristics of the Newton-Raphson method. The calculation of the tangent stiffness is essential for this solution method.

A.4.1 Directional Derivative

The concept of the directional derivative leads to a systematic procedure for carrying out this linearization. Based on the definition of the directional derivative, the static weak form $G(\underline{\Phi}, \delta\underline{\Phi})$ can be approximated by retaining the first two terms of the Taylor's series expansion about a fixed configuration $\underline{\bar{\Phi}}$ as follows

$$G(\underline{\Phi}, \delta\underline{\Phi}) \approx G(\underline{\bar{\Phi}}, \delta\underline{\Phi}) + DG(\underline{\Phi}, \delta\underline{\Phi}) \cdot \Delta\underline{\Phi}|_{\underline{\Phi}=\underline{\bar{\Phi}}} \tag{38}$$

The quantity $G(\underline{\bar{\Phi}}, \delta\underline{\Phi})$ is called the internal force operator, while the quantity $DG(\underline{\Phi}, \delta\underline{\Phi}) \cdot \Delta\underline{\Phi}$ is called the tangent stiffness operator. Based on the definition of the directional derivative(Simo, Fox and Rifai(1989)), $DG \cdot \Delta\underline{\Phi}$ is expressed as

$$DG(\underline{\Phi}, \delta\underline{\Phi}) \cdot \Delta\underline{\Phi} \equiv \left. \frac{d}{d\epsilon} \right|_{\epsilon=0} G(\underline{\Phi}_\epsilon) \tag{39}$$

where

$$\underline{\Phi}_\epsilon = \underline{\phi} + \epsilon\Delta\underline{\phi} + \xi \left(\cos \|\epsilon\Delta\underline{t}\| \underline{t} + \frac{\sin \|\epsilon\Delta\underline{t}\|}{\|\epsilon\Delta\underline{t}\|} \epsilon\Delta\underline{t} \right) \tag{40}$$

The tangent stiffness operator is conveniently separated into two parts: the material part and the geometric part, respectively. This decomposition is expressed as

$$DG(\underline{\Phi}, \delta\underline{\Phi}) \cdot \Delta\underline{\Phi} = (D_M G + D_G G) \cdot \Delta\underline{\Phi} \tag{41}$$

A.4.2 Finite Element Discretization

The finite element discretization is introduced by interpolating the position vector $\underline{\phi}$ between nodal values $\underline{\phi}_I$ using shape functions $N^I(\xi^1, \xi^2)$. The discrete form of $\underline{\phi}$ is denoted by $\underline{\phi}^h$

$$\underline{\phi}^h = \sum_{i=1}^{4} N^I(\xi^1, \xi^2) \underline{\phi}_I \qquad \text{where} \qquad \underline{\phi}_I = \left\{ \begin{array}{c} \phi_I^1 \\ \phi_I^2 \\ \phi_I^3 \end{array} \right\} \tag{42}$$

The unit directors $\underline{t}$ are interpolated with the same shape functions using the nodal values $\underline{t}_I$. The discrete version of $\underline{t}$ is then $\underline{t}^h$

$$\underline{t}^h = \frac{\sum_{i=1}^4 N^I \underline{t}_I}{\left\| \sum_{i=1}^4 N^I \underline{t}_I \right\|} \equiv \frac{\hat{\underline{t}}^h}{\left\| \hat{\underline{t}}^h \right\|} \qquad \text{where} \qquad \underline{t}_I = \left\{ \begin{array}{c} t_I^1 \\ t_I^2 \\ t_I^3 \end{array} \right\} \tag{43}$$

and $N^I(\xi^1, \xi^2)$ are the standard bilinear isoparametric shape functions defined by

$$N^I(\xi^1, \xi^2) = \frac{1}{4}(1 + \xi^1 \xi_I^1)(1 + \xi^2 \xi_I^2) \tag{44}$$

where the coordinates at the four nodal points are $(\xi_I^1, \xi_I^2) \in \{(-1,-1), (1,-1), (1,1), (-1,1)\}$. The derivatives of both Equations 42 and 43 are

$$\underline{\phi}^h_{,\alpha} = \sum_{i=1}^4 N^I_{,\alpha} \underline{\phi}_I \tag{45}$$

$$\underline{t}^h_{,\alpha} = \frac{1}{\left\| \tilde{\underline{t}}^h \right\|} \left[\tilde{\underline{t}}^h_{,\alpha} - (\underline{t}^h \cdot \tilde{\underline{t}}^h_{,\alpha}) \underline{t}^h \right] \tag{46}$$

where

$$\tilde{\underline{t}}^h_{,\alpha} = (\underline{t}^h \cdot \tilde{\underline{t}}^h_{,\alpha}) \underline{t}^h + \left\| \tilde{\underline{t}}^h \right\| \underline{t}^h_{,\alpha}$$

The discretization of incremental quantities stored in $\Delta\underline{\phi}$ and $\Delta\underline{t}$ and the weighting functions stored in $\delta\underline{\phi}$ and $\delta\underline{t}$ are introduced by setting

$$\Delta\underline{\phi}^h = \sum_{i=1}^4 N^I(\xi^1, \xi^2) \Delta\underline{\phi}_I \qquad \text{where } \Delta\underline{\phi}_I = \left\{ \begin{array}{c} \Delta\phi_I^1 \\ \Delta\phi_I^2 \\ \Delta\phi_I^3 \end{array} \right\} \tag{47}$$

$$\begin{aligned} \Delta\underline{t}^h &= \sum_{i=1}^4 N^I(\xi^1, \xi^2) \Delta\underline{t}_I \\ &= \sum_{i=1}^4 N^I(\xi^1, \xi^2) \bar{\underline{\Lambda}}_I \Delta\bar{\underline{T}} \qquad \text{where } \Delta\bar{\underline{T}} = \left\{ \begin{array}{c} \Delta T_I^1 \\ \Delta T_I^2 \end{array} \right\} \end{aligned} \tag{48}$$

$$\delta\underline{\phi}^h = \sum_{i=1}^4 N^I(\xi^1, \xi^2) \delta\underline{\phi}_I \qquad \text{where } \delta\underline{\phi}_I = \left\{ \begin{array}{c} \delta\phi_I^1 \\ \delta\phi_I^2 \\ \delta\phi_I^3 \end{array} \right\} \tag{49}$$

$$\begin{aligned} \delta\underline{t}^h &= \sum_{i=1}^4 N^I(\xi^1, \xi^2) \delta\underline{t}_I \\ &= \sum_{i=1}^4 N^I(\xi^1, \xi^2) \bar{\underline{\Lambda}}_I \delta\bar{\underline{T}} \qquad \text{where } \delta\bar{\underline{T}} = \left\{ \begin{array}{c} \delta T_I^1 \\ \delta T_I^2 \end{array} \right\} \end{aligned} \tag{50}$$

where $\bar{\underline{\Lambda}}_I$ is the 3 by 2 matrix formed by eliminating the third column of the 3 by 3 transformation matrix $\bar{\underline{\Lambda}}$.

A.4.3 Nonlinear Material Tangent Stiffness Operator

The nonlinear material tangent stiffness operator is obtained by operating on the weak form given in Eq. 36. Taking the directional derivative of the stress resultant $\underline{R}$ term while holding the $\underline{B}$ matrix constant leads to the desired result

$$\begin{aligned} D_M G \cdot \Delta\underline{\Phi} &= \int_{\mathcal{A}} \underline{B}\delta\underline{\Phi} \cdot \left.\frac{d\underline{R}(\underline{\Phi}_\epsilon)}{d\epsilon}\right|_{\epsilon=0} d\mathcal{A}^0 \\ &= \int_{\mathcal{A}} \underline{B}\delta\underline{\Phi} \cdot \left\{ \begin{array}{c} \underline{C}_m \dfrac{d\underline{\varepsilon}}{d\epsilon} \\ \underline{C}_s \dfrac{d\underline{\gamma}}{d\epsilon} \\ \underline{\mathbf{1}} \dfrac{\partial \underline{F}(\underline{\kappa})}{\partial \underline{\kappa}} \dfrac{\partial \underline{\kappa}}{\partial \epsilon} \end{array} \right\}_{\epsilon=0} d\mathcal{A}^0 \\ &= \int_{\mathcal{A}} \underline{B}\delta\underline{\Phi} \cdot \underline{C} \left\{ \begin{array}{c} \underline{B}_m \Delta\underline{\phi} \\ \underline{B}_{sm}\Delta\underline{\phi} + \underline{B}_{sd}\Delta\underline{\bar{T}} \\ \underline{B}_{bm}\Delta\underline{\phi} + \underline{B}_{bb}\Delta\underline{\bar{T}} \end{array} \right\} d\mathcal{A}^0 \\ &= \int_{\mathcal{A}} \underline{B}\delta\underline{\Phi} \cdot \underline{C}\underline{B}\Delta\underline{\Phi} d\mathcal{A}^0 \end{aligned} \tag{51}$$

where

$$\underline{C} = \left[\begin{array}{ccc} \underline{C}_m & \underline{0} & \underline{0} \\ \underline{0} & \underline{C}_s & \underline{0} \\ \underline{0} & \underline{0} & \underline{f}_r \end{array} \right] \tag{52}$$

$$\delta\underline{\Phi} = \left\{ \begin{array}{c} \delta\underline{\phi} \\ \delta\underline{\bar{T}} \end{array} \right\}, \qquad \Delta\underline{\Phi} = \left\{ \begin{array}{c} \Delta\underline{\phi} \\ \Delta\underline{\bar{T}} \end{array} \right\} \tag{53}$$

$$\delta\underline{t} = \underline{\bar{\Lambda}}\delta\underline{\bar{T}}, \qquad \Delta\underline{t} = \underline{\bar{\Lambda}}\Delta\underline{\bar{T}} \tag{54}$$

and

$$\underline{f}_r = \frac{\partial \underline{F}(\underline{\kappa})}{\partial \underline{\kappa}} = \left[\begin{array}{ccc} \dfrac{\partial f_1}{\partial \kappa_{11}} & \dfrac{\partial f_1}{\partial \kappa_{22}} & \dfrac{\partial f_1}{\partial \kappa_{(12)}} \\ \dfrac{\partial f_2}{\partial \kappa_{11}} & \dfrac{\partial f_2}{\partial \kappa_{22}} & \dfrac{\partial f_2}{\partial \kappa_{(12)}} \\ \dfrac{\partial f_3}{\partial \kappa_{11}} & \dfrac{\partial f_3}{\partial \kappa_{22}} & \dfrac{\partial f_3}{\partial \kappa_{(12)}} \end{array} \right] \tag{55}$$

For linear elastic isotropic materials and curvilinear coordinates, the functions in Eq. 28 are reduced to

$$f_1(\kappa_{11},\kappa_{22},\kappa_{(12)}) = \frac{h^3}{12}(C^{1111}\kappa_{11} + C^{1122}\kappa_{22} + 2C^{1112}\kappa_{(12)}) \tag{56}$$

$$f_2(\kappa_{11},\kappa_{22},\kappa_{(12)}) = \frac{h^3}{12}(C^{1122}\kappa_{11} + C^{2222}\kappa_{22} + 2C^{2212}\kappa_{(12)}) \tag{57}$$

$$f_3(\kappa_{11},\kappa_{22},\kappa_{(12)}) = \frac{h^3}{12}(C^{1112}\kappa_{11} + C^{2212}\kappa_{22} + 2C^{1212}\kappa_{(12)}) \tag{58}$$

Taking the derivatives with respect to κ_{ij}, then $\underline{f}_r$, it follows that

$$\underline{f}_r = \frac{h^3}{12}\begin{bmatrix} C^{1111} & C^{1122} & 2C^{1112} \\ C^{1122} & C^{2222} & 2C^{2212} \\ C^{1112} & C^{2212} & 2C^{1212} \end{bmatrix} = \underline{C}_b \tag{59}$$

For nonlinear elastic isotropic materials with Poisson's ratio ν Assumed independent of curvatures, the required partial derivatives of Eq. 55

$$\begin{array}{lll} \frac{\partial f_1}{\partial \kappa_{11}} = \mu_{11}\frac{\partial f}{\partial \kappa_{11}}, & \frac{\partial f_1}{\partial \kappa_{22}} = \mu_{12}\frac{\partial f}{\partial \kappa_{22}}, & \frac{\partial f_1}{\partial \kappa_{(12)}} = \mu_{13}\frac{\partial f}{\partial \kappa_{(12)}} \\ \frac{\partial f_2}{\partial \kappa_{11}} = \mu_{12}\frac{\partial f}{\partial \kappa_{11}}, & \frac{\partial f_2}{\partial \kappa_{22}} = \mu_{22}\frac{\partial f}{\partial \kappa_{22}}, & \frac{\partial f_2}{\partial \kappa_{(12)}} = \mu_{23}\frac{\partial f}{\partial \kappa_{(12)}} \\ \frac{\partial f_3}{\partial \kappa_{11}} = \mu_{13}\frac{\partial f}{\partial \kappa_{11}}, & \frac{\partial f_3}{\partial \kappa_{22}} = \mu_{23}\frac{\partial f}{\partial \kappa_{22}}, & \frac{\partial f_3}{\partial \kappa_{(12)}} = \mu_{33}\frac{\partial f}{\partial \kappa_{(12)}} \end{array} \tag{60}$$

Substituting these equations into Equation 55

$$\underline{f}_r = \begin{bmatrix} \mu_{11} & \mu_{12} & \mu_{13} \\ \mu_{12} & \mu_{22} & \mu_{23} \\ \mu_{13} & \mu_{23} & \mu_{33} \end{bmatrix} \begin{Bmatrix} \frac{\partial f}{\partial \kappa_{11}} \\ \frac{\partial f}{\partial \kappa_{22}} \\ \frac{\partial f}{\partial \kappa_{(12)}} \end{Bmatrix} \tag{61}$$

$$= \frac{\left[C^{\alpha\beta\gamma\delta}\right]}{E} \begin{Bmatrix} \frac{\partial f}{\partial \kappa_{11}} \\ \frac{\partial f}{\partial \kappa_{22}} \\ \frac{\partial f}{\partial \kappa_{(12)}} \end{Bmatrix} \tag{62}$$

where $\left[C^{\alpha\beta\gamma\delta}\right]$ is defined in Equation 21. Discretizing Eq. 51 using Eqs. 45 - 50 results in

$$\begin{aligned} DG_M^h \cdot \Delta\underline{\Phi}^h &= \sum_{I=1}^{4}\sum_{J=1}^{4} \delta\underline{\Phi}_I^T \int_{\mathcal{A}} (\underline{B}^I)^T \underline{C}\underline{B}^J d\mathcal{A}^0 \Delta\underline{\Phi}_J \\ &= \sum_{I=1}^{4}\sum_{J=1}^{4} \delta\underline{\Phi}_I^T K_M^{IJ} \Delta\underline{\Phi}_J \end{aligned} \tag{63}$$

where

$$\boxed{K_M^{IJ} \equiv \int_{\mathcal{A}} (\underline{B}^I)^T \underline{C} \underline{B}^J d\mathcal{A}^0} \tag{64}$$

and

$$\underline{B}^I = \begin{bmatrix} \underline{B}^I_m & \underline{0}_{3\times 2} \\ \underline{B}^I_{sm} & \underline{B}^I_{sb} \\ \underline{B}^I_{bm} & \underline{B}^I_{bb} \end{bmatrix} \tag{65}$$

$$\underline{B}^I_m = \begin{bmatrix} N^I_{,1}\underline{\phi}^T_1 \\ N^I_{,2}\underline{\phi}^T_2 \\ N^I_{,1}\underline{\phi}^T_2 + N^I_{,2}\underline{\phi}^T_1 \end{bmatrix} \tag{66}$$

$$\underline{B}^I_s = \begin{bmatrix} \underline{B}^I_{sm} & \underline{B}^I_{sb} \end{bmatrix} = \left[\begin{array}{c|c} N^I_1 \underline{t}^T & \underline{\phi}^I_1 \underline{\bar{\Lambda}}_I \\ N^I_2 \underline{t}^T & \underline{\phi}^I_2 \underline{\bar{\Lambda}}_I \end{array}\right] \tag{67}$$

$$\underline{B}^I_b = \begin{bmatrix} \underline{B}^I_{bm} & \underline{B}^I_{bb} \end{bmatrix} = \left[\begin{array}{c|c} N^I_{,1}\underline{t}^T_{,1} & N^I_{,1}\underline{\phi}^T_{,1}\underline{\bar{\Lambda}}_I \\ N^I_{,2}\underline{t}^T_{,2} & N^I_{,2}\underline{\phi}^T_{,2}\underline{\bar{\Lambda}}_I \\ N^I_{,1}\underline{t}^T_{,2} + N^I_{,2}\underline{t}^T_{,1} & (N^I_{,1}\underline{\phi}^T_{,2} + N^I_{,2}\underline{\phi}^T_{,1})\underline{\bar{\Lambda}}_I \end{array}\right] \tag{68}$$

A.4.4 Geometric Tangent Stiffness Operator

The geometric tangent stiffness operator is obtained by taking the directional derivative of the discrete $\underline{B}$ matrix while holding the stress resultant vector $\underline{R}$ constant

$$\begin{aligned} DG^h_G \cdot \Delta\underline{\Phi}^h &= \int_{\mathcal{A}} \frac{d}{d\epsilon}\left(\underline{B}^h(\underline{\Phi}_\epsilon)\delta\underline{\Phi}\right)\Big|_{\epsilon=0} \cdot \underline{R} d\mathcal{A}^0 \\ &= \int_{\mathcal{A}} \left[\tilde{N}^{\alpha\beta} \left.\frac{d\delta\varepsilon^h_{\alpha\beta}(\epsilon)}{d\epsilon}\right|_{\epsilon=0} + \tilde{Q}^\alpha \left.\frac{d\delta\gamma^h_\alpha(\epsilon)}{d\epsilon}\right|_{\epsilon=0} \right. \\ &\quad \left. + \tilde{M}^{\alpha\beta} \left.\frac{d\delta\kappa^h_{\alpha\beta}(\epsilon)}{d\epsilon}\right|_{\epsilon=0}\right] d\mathcal{A}^0 \end{aligned} \tag{69}$$

After taking the directional derivative of $\delta\varepsilon^h_{\alpha\beta}, \delta\gamma^h_\alpha$ and $\delta\kappa^h_{\alpha\beta}$, we have

$$\left.\frac{d\delta\varepsilon^h_{\alpha\beta}}{d\epsilon}\right|_{\epsilon=0} = \frac{1}{2}\sum_{I=1}^4\sum_{J=1}^4 \delta\underline{\phi}^T_I (N^I_{,\alpha}N^J_{,\beta} + N^I_{,\beta}N^J_{,\alpha})\Delta\underline{\phi}_J \tag{70}$$

$$\begin{aligned} \left.\frac{d\delta\gamma^h_\alpha}{d\epsilon}\right|_{\epsilon=0} &= \sum_{I=1}^4\sum_{J=1}^4 (\delta\underline{\phi}^T_I N^I_{,\alpha} N^J \underline{\bar{\Lambda}}_J \Delta\underline{\bar{T}}_J + \delta\underline{\bar{T}}^T_I N^I N^J_{,\alpha} \underline{\bar{\Lambda}}^T_I \Delta\underline{\phi}_J) \\ &\quad - \sum_{I=1}^4\sum_{J=1}^4 \delta\underline{\bar{T}}^T_I N^I \underline{\phi}_{,\alpha} \cdot \underline{t}_J \delta_{IJ} \underline{I}_2 \Delta\underline{\bar{T}}_J \end{aligned} \tag{71}$$

$$\left.\frac{d\delta\kappa^h_{\alpha\beta}}{d\epsilon}\right|_{\epsilon=0} = \sum_{I=1}^4\sum_{J=1}^4 (\delta\underline{\phi}^T_I N^I_{,\alpha} N^J_{,\beta} \underline{\bar{\Lambda}}_J \Delta\underline{\bar{T}}_J + \delta\underline{\bar{T}}^T_I N^I_{,\beta} N^J_{,\alpha} \underline{\bar{\Lambda}}^T_I \Delta\underline{\phi}_J)$$

$$-\sum_{I=1}^{4}\sum_{J=1}^{4}\delta\underline{\bar{T}}_I^T N^I_{,\beta}(\underline{\phi}_\alpha\cdot\underline{t}_J)\delta_{IJ}\underline{I}_2\Delta\underline{\bar{T}}_J \tag{72}$$

Substituting Eqs. 70 - 72 into Equation 69 and factoring out $\delta\underline{\Phi}^T$ and $\Delta\underline{\Phi}$, the Equation 69 can be expressed as

$$DG^h_G\cdot\Delta\underline{\Phi}^h=\sum_{I=1}^{4}\sum_{J=1}^{4}\delta\underline{\Phi}_I^T\left((\underline{K}_G^{IJ})_m+(\underline{K}_G^{IJ})_s+(\underline{K}_G^{IJ})_b\right)\Delta\underline{\Phi}_J \tag{73}$$

where $(\underline{K}_G^{IJ})_m$, $(\underline{K}_G^{IJ})_s$ and $(\underline{K}_G^{IJ})_b$ are the geometric tangent stiffness matrices for membrane, transverse shear and bending respectively.

$$(\underline{K}_G^{IJ})_m = \int_{\mathcal{A}}\begin{bmatrix}\begin{Bmatrix}N^I_{,1}N^J_{,1}\\ N^I_{,2}N^J_{,2}\\ N^I_{,1}N^J_{,2}+N^I_{,2}N^J_{,1}\end{Bmatrix}\cdot\tilde{\underline{N}}\underline{I}_3 & \underline{0}_{3\times2}\\ \underline{0}_{2\times3} & \underline{0}_{2\times2}\end{bmatrix}d\mathcal{A}^0$$

$$(\underline{K}_G^{IJ})_s = \int_{\mathcal{A}}\begin{bmatrix}\underline{0}_{3\times3} & \tilde{\underline{Q}}\cdot\begin{Bmatrix}N^I_{,1}N^J\\ N^I_{,2}N^J\end{Bmatrix}\bar{\underline{\Lambda}}_J\\ \tilde{\underline{Q}}\cdot\begin{Bmatrix}N^IN^J_{,1}\\ N^IN^J_{,2}\end{Bmatrix}\bar{\underline{\Lambda}}_I^T & -\tilde{\underline{Q}}\cdot\begin{Bmatrix}N^I\underline{\phi}_{,1}\cdot\underline{t}_J\\ N^I\underline{\phi}_{,2}\cdot\underline{t}_J\end{Bmatrix}\delta_{IJ}\underline{I}_2\end{bmatrix}d\mathcal{A}^0$$

$$(\underline{K}_G^{IJ})_b=$$

$$\int_{\mathcal{A}}\begin{bmatrix}\underline{0}_{3\times3} & \tilde{\underline{M}}\cdot\begin{Bmatrix}N^I_{,1}N^J_{,1}\\ N^I_{,2}N^J_{,2}\\ N^I_{,1}N^J_{,2}+N^I_{,2}N^J_{,1}\end{Bmatrix}\bar{\underline{\Lambda}}_J\\ \tilde{\underline{M}}\cdot\begin{Bmatrix}N^I_{,1}N^J_{,1}\\ N^I_{,2}N^J_{,2}\\ N^I_{,1}N^J_{,2}+N^I_{,2}N^J_{,1}\end{Bmatrix}\bar{\underline{\Lambda}}_I & -\tilde{\underline{M}}\cdot\begin{Bmatrix}N^I_{,1}\underline{\phi}_{,1}\cdot\underline{t}_J\\ N^I_{,2}\underline{\phi}_{,2}\cdot\underline{t}_J\\ N^I_{,1}\underline{\phi}_{,2}\cdot\underline{t}_J+N^I_{,2}\underline{\phi}_{,1}\cdot\underline{t}_J\end{Bmatrix}\delta_{IJ}\underline{I}_2\end{bmatrix}d\mathcal{A}^0$$

The full displacement method for the membrane strain field and transverse shear strain field results in poor performance of the finite element approximation due to "membrane locking" and "shear locking" effect respectively. To solve these problems, the mixed interpolation method for membrane locking and assumed strain method for shear locking(Simo, Fox and Rifai(1989)) are adapted to reduce the effects.

A.4.5 Internal Force Vector

Subtracting external forces $G_{ext}(\delta\underline{\phi})$ from the static weak form(Eq. 36), yields the internal force operator as

$$G_{int}(\underline{\Phi},\delta\underline{\Phi})=\delta\underline{\Phi}^T\int_{\mathcal{A}}\left[\underline{B}_m^T\tilde{\underline{N}}+\underline{B}_s^T\tilde{\underline{Q}}+\underline{B}_b^T\tilde{\underline{M}}\right]d\mathcal{A}^0 \tag{74}$$

Then, the internal force vector can be defined as

$$\underline{P} = \int_{\mathcal{A}} \left[\underline{B}_m^T \tilde{\underline{N}} + \underline{B}_s^T \tilde{\underline{Q}} + \underline{B}_b^T \tilde{\underline{M}} \right] d\mathcal{A}^0 \tag{75}$$

While the discretized internal force vector at node I is expressed as

$$\underline{P}^I = \int_{\mathcal{A}} \left[(\underline{B}_m^I)^T \tilde{\underline{N}} + (\underline{B}_s^I)^T \tilde{\underline{Q}} + (\underline{B}_b^I)^T \tilde{\underline{M}} \right] d\mathcal{A}^0 \tag{76}$$