UNIVERSITY OF HONG KONG

DOCTORAL THESIS

Computational Models for Cloth and Origami Simulations

Author:
Qixin LIANG

Supervisor:
Professor Yuguo LI

Co-Supervisor:
Professor K. Y. SZE

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Department of Mechanical Engineering
Faculty of Engineering

Dec 31, 2025

Abstract of thesis entitled

Computational Models for Cloth and Origami Simulations

Submitted by

Qixin LIANG

for the degree of Doctor of Philosophy

at The University of Hong Kong

in May, 2026

Physical simulation tools have found widespread applications in mechanical analysis, industrial design, robotics training, film and game effects, etc. Computational models play a fundamental role in these tools, enabling high-fidelity and high-performance simulations of various physical phenomena. Cloth and origami, with their extreme flexibility and strong geometric nonlinearity, pose challenges in the development of accurate, efficient, and robust computational models. These challenges have drawn broad interest across disciplines. This thesis lies at the intersection of computer graphics and computational mechanics, where it develops a series of computational models for cloth and origami simulations.

In cloth simulation, a hinge-bending model is first considered, where the edge stencil consists of two adjacent 3-node triangular facets sharing a common edge or hinge. The model fits a cylindrical surface over the stencil and expresses the directional curvature perpendicular to the common edge in terms of the hinge angle, which is computed from the altitude vectors connecting the hinge and the opposite nodes. The gradient and Hessian of the hinge angle are derived analytically in concise form. Quantitative and qualitative tests are presented to demonstrate the model's accuracy and robustness over other hinge-bending models.

To improve the efficiency of the hinge-bending model, a directional curvature operator is formulated using a corotational frame and the small-strain-small-curvature assumption. This operator yields a corotational edge-based hinge thin plate/shell model. To improve mesh consistency and convergence under mesh refinement, the methodology is extended to a 6-node triangle-centered stencil consisting of a central triangle and its three neighbors.Based on this stencil, the corotational finite volume method (FVM) hinge-bending model applies the FVM to compute the curvature tensor from the directional curvatures across edges of the central triangle. The corotational smoothed hinge-bending model further transforms these directional curvatures into the curvature tensor. Altogether, six models are derived, all featuring constant bending energy Hessians that benefit implicit solvers. These models strike a balance between accuracy and efficiency, while ensuring robustness.

Sharp creasing artefacts are commonly observed in cloth simulation when a bending model is combined with a constant membrane strain triangle. Here, these artefacts are effectively mitigated by employing a six-node quadratic interpolation scheme over a triangle-centered stencil.

In origami simulations, folding and unfolding of the panels are often assumed to be driven by the energy stored in the creases. The fold angle and its derivatives derived from the hinge-bending model can be directly used for modeling origami structures. To consider curved panels and creases more effectively, a novel computational model in which origami panels are discretized by solid-shell finite elements with nodes on the panel mid-surfaces is proposed. Fold angles varying along a crease are addressed by using the angle between the nodal directors of the elements sharing the same node(s) along the crease. To mitigate locking in the solid-shell elements, the assumed natural strain method is employed. The accuracy and efficacy of the proposed model are validated through quantitative and qualitative tests.

The proposed models enhance the computational toolkit for cloth and origami simulations, providing greater versatility and adaptability.

(500 words)

Computational Models for Cloth and Origami Simulations

by

Qixin LIANG
BEng *HFUT*, MEng *HIT*

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Doctor of Philosophy

at

University of Hong Kong
Dec 31, 2025

Declaration

I, Qixin LIANG, declare that this thesis is submitted in fulfillment of the requirements for the Degree of Doctor of Philosophy, represents my own work except where due acknowledgement is made. I further declared that it has not been previously included in a thesis, dissertation, or report submitted to this University or to any other institution for a degree, diploma or other qualifications.

Signed: Qixin Liang

Date: Dec 31, 2025

Acknowledgements

I would first like to express my deepest gratitude for the opportunity to work with my supervisor, Prof. K. Y. Sze at HKU. He is reliable and professional. His rigorous scholarly attitude will continue to benefit my work and research for a lifetime. From him, I learned what it means to be an independent and responsible researcher. I am also sincerely grateful to Prof. Yuguo Li, who kindly assumed the role of my supervisor after Prof. Sze's retirement.

My heartfelt thanks go as well to my seniors—Dr. Yexin Zhou, Dr. Yucai Hu, Dr. Hongtao Sun, Dr. Zhenghao Wu, Dr. Wenchen Yuan, Dr. Qing Xie, and Dr. Yongqiang Jin. Although they had already graduated when I joined the group, their shared experience and encouragement were invaluable to me. I would also like to thank Ms. Samantha Tsang from our department for her support in maintaining the IT hardware and software facilities. I am further indebted to my collaborators—Prof. K. W. Kwok (CUHK), Dr. Yingqi Li, and Prof. Ke Liu (PKU)—for the opportunity to work together.

Many of my days at HKU were spent in my office at Robert Black College and around the Stanley Ho Sports Centre, places that now hold some of my fondest memories. I am grateful to the University for providing such a supportive environment and abundant learning resources. I also acknowledge the generous funding support from the InnoHK Centre for Transformative Garment Production (TransGP).

Finally, my appreciation goes to my family, whose unwavering love, understanding, and support have accompanied me throughout this entire journey.

Qixin LIANG
The University of Hong Kong
Dec 31, 2025

Contents

List of Figures

List of Tables

List of Abbreviations

AI	**A**rtificial **I**ntelligence
AABB	**A**xis-**A**ligned **B**ounding **B**ox
ADMM	**A**lternating **D**irection **M**ethod of **M**ultipliers algorithm
ANS	**A**ssumed **N**atural **S**train
BDF2	The second-order **B**ackward **D**ifferentiation **F**ormula
BLSS	**Bi**linear **S**olid-**S**hell Element
BQSS	**Bi**quadratic **S**olid-**S**hell Element
BVH	**B**ounding **V**olume **H**ierarchy
CCD	**C**ontinuous **C**ollision **D**etection
CFL	**C**ourant–**F**riedrichs–**L**ewy condition
CST	**C**onstant **S**train **T**riangle
DCD	**D**iscrete **C**ollision **D**etection
DDG	**D**iscrete **D**ifferential **G**eometry
EAS	**E**nhanced **A**ssumed **S**train
EP	Corotational **E**dge-based Hinge Thin **P**late
ES	Corotational **E**dge-based Hinge Thin **S**hell
FEM	**F**inite **E**lement **M**ethod
FVM	**F**inite **V**olume **M**ethod
FP	Corotational **F**VM Hinge Thin **P**late
FS	Corotational **F**VM Hinge Thin **S**hell
FVM	**F**inite **V**olume **M**ethod
GNN	**G**raph **N**eural **N**etwork
GPU	**G**raphics **P**rocessing **U**nit
IP	**I**ncremental **P**otential
IPC	**I**ncremental **P**otential **C**ontact
KL	**K**irchhoff–**L**ove theory
L-BFGS	**L**imited-memory **B**royden–**F**letcher–**G**oldfarb–**S**hanno algorithm
LCP	**L**inear **C**omplementarity **P**roblem

MITC	**M**ixed **I**nterpolation of **T**ensorial **C**omponents
NURBS	**N**on-**U**niform **R**ational **B**-**S**plines
OBB	**O**riented **B**ounding **B**ox
PBD	**P**osition-**B**ased **D**ynamics
PD	**P**rojective **D**ynamics
VLA	**V**ision-**L**anguage-**A**ction
VR	**V**irtual **R**eality
AR	**A**ugmented **R**eality
SI	**S**ystème **I**nternational d'Unités
SM	**S**moothed Hinge **M**embrane Model
SP	Corotational **S**moothed Hinge Thin **P**late
SS	Corotational **S**moothed Hinge Thin **S**hell
StVK	**St**. **V**enant–**K**irchhoff
WAM	**W**orld **A**ction **M**odel

List of Symbols

A	Integration area of a triangle or quadrilateral
$A_{\mathcal{E}}$	Integration area of an edge stencil
$\mathsf{a}_{\alpha\beta}, \mathsf{A}_{\alpha\beta}$	First fundamental form components of the current and initial midsurface
$B(\mathbb{x})$	Contact barrier potential
$D(\mathbb{x})$	Friction potential
D_t	Thickness stiffness
E	Young's modulus
$E(\mathbb{x})$	Incremental potential
$\mathbf{F}$	Deformation gradient
$\mathbf{G}$	Metric tensor in the initial configuration
$G_{\alpha\beta}$	In-plane components of $\mathbf{G}$, $\alpha, \beta = r, s$
G_{tr}, G_{ts}, G_{tt}	Transverse components of $\mathbf{G}$
$\mathbf{g}$	Metric tensor in the current configuration
$g_{\alpha\beta}$	In-plane components of $\mathbf{g}$, $\alpha, \beta = r, s$
g_{tr}, g_{ts}, g_{tt}	Transverse components of $\mathbf{g}$
h	Thickness
$\mathbf{I}$	Third order identity matrix
$\mathbf{I}_{4\times4}, \mathbf{I}_{6\times6}$	Fourth and sixth order identity matrices
J	Jacobian determinant of the coordinate transformation
k_b	Bending stiffness
k_f	Folding stiffness per unit length
l	Length
l_{ij}	Length of edge ij connecting points i and j
L_{fl}	Lagrange interpolation function in the full direction

L_{rd}	Lagrange interpolation function in the reduced direction
L_i	Interpolation function at node i of the crease
$\mathbb{M}$	Mass matrix
N_i	Interpolation function of node i
$\mathbf{n}_{\tilde{x}}, \mathbf{n}_{\tilde{y}}, \mathbf{n}_{\tilde{z}}$	Directions of the $\tilde{x}$-, $\tilde{y}$- and $\tilde{z}$-axes in the current corotational frame
$\mathbf{n}_{\tilde{X}}, \mathbf{n}_{\tilde{Y}}, \mathbf{n}_{\tilde{Z}}$	Directions of the $\tilde{X}$-, $\tilde{Y}$- and $\tilde{Z}$-axes in the initial corotational frame
r, s, t	Curvilinear or parametric coordinates
T_{ijk}	Triangle with nodes i, j and k
$\mathbf{T}_{mb}, \mathbf{T}_s, T_t$	Coordinate transformation matrices for membrane/bending, shear and thickness strain
$\mathbf{U}$	Displacement vector
$\mathbf{U}_o$	Mid-surface displacement vector
$\mathbf{U}_n$	Director displacement vector
$\mathbf{U}_{oi}$	Mid-surface displacement vector of node i
$\mathbf{U}_{ni}$	Director displacement vector of node i
$\mathbf{U}^s$	Collection of nodal displacement vectors in a shell stencil
$\mathbf{U}_c^s$	Collection of nodal displacement vectors in a corotational shell stencil
$\mathbf{U}_n^a, \mathbf{U}_n^b$	Director displacement vectors of elements "a" and "b"
U, V, W	Displacement components in X, Y and Z directions
$\tilde{w}, \tilde{W}$	Transverse displacement in the current and initial corotational frame
$\tilde{w}_i, \tilde{W}_i$	Transverse displacement of node i in the current and initial corotational frame
$\mathbf{X}, \mathbf{x}$	Initial and current position vector
$\mathbf{X}_o, \mathbf{x}_o$	Initial and current position vector of the shell midsurface
$\mathbf{X}_n, \mathbf{x}_n$	Initial and current director vector
$\mathbf{X}_i, \mathbf{x}_i$	Initial and current position vector of node i

$\mathbf{X}^s, \mathbf{x}^s$	Collection of initial and current nodal position vectors in a shell stencil
$\mathbf{X}_c^s, \mathbf{x}_c^s$	Collection of initial and current nodal position vectors of a corotational shell stencil
$\mathbf{X}_{ij}$	Edge vector $\mathbf{X}_{ij} = \mathbf{X}_j - \mathbf{X}_i$
$\mathbf{X}_n^a, \mathbf{X}_n^b$	Initial director vectors of elements "a" and "b"
$\mathbb{x}$	Global position vector
$\tilde{\mathbf{x}}_i, \tilde{\mathbf{X}}_i$	Position vector of node i in the current and initial corotational frame
$\mathbb{x}^t, \mathbb{v}^t$	Global nodal position and velocity vectors at time t
$\hat{\mathbb{x}}$	Predicted nodal position
$\tilde{X}, \tilde{Y}, \tilde{Z}$	Coordinates of the initial corotational frame
$\tilde{x}, \tilde{y}, \tilde{z}$	Coordinates of the current corotational frame
Γ	Boundary of a triangle
Δt	Time step
$\Delta\mathbb{x}$	Incremental displacement
Ω	Mid-surface of a shell
Ψ_{ts}	Total strain energy in a thin shell
Ψ_{ss}	Total strain energy in a shell-like solid
$\Psi_m, \Psi_b, \Psi_s, \Psi_t$	Total membrane, bending, transverse shear and thickness energy of the shell
Ψ_{ss}^e	Total strain energy in a solid-shell element
$\Psi_m^e, \Psi_b^e, \Psi_s^e, \Psi_t^e$	Membrane, bending, transverse shear and thickness energy of the shell element
Ψ_{ts}^s	Total strain energy in a thin shell stencil
Ψ_m^s, Ψ_b^s	Membrane, bending energy of the shell stencil
Ψ_f	Folding energy for a crease
Ψ_b^{model}	Bending energy of a model (EP, ES, FP, FS, SP and SS)
$\Psi_{elastic}$	Elastic energy
Θ, θ	Hinge / fold angles in current and initial configurations
θ_i, Θ_i	Hinge angle across edge i in the current and initial configuration

$\theta_L,\ \theta_R$	Lower and upper angle limit for penalty activation
$\kappa_{\alpha\beta}, K_{\alpha\beta}$	Curvature (second fundamental form) components of the current and initial mid-surface
κ_d, K_d	Directional curvature in the current and initial configuration
$(\kappa_d)_i, (K_d)_i$	Directional curvature across edge i in the current and initial configuration
K_d	Directional curvature in the initial configuration
$(K_d)_i$	Directional curvature across edge i in the initial configuration
$\boldsymbol{\kappa}$	Curvature (Voigt notation) in the current configuration
$\boldsymbol{K}$	Curvature (Voigt notation) in the initial configuration
ν	Poisson's ratio
$\xi,\ \eta, \zeta$	Natural or parametric coordinates
$\varepsilon_{\alpha\beta}$	In-plane Green–Lagrange strain components
$\varepsilon_{m\alpha\beta}$	Membrane strain components
$\varepsilon_{b\alpha\beta}$	Bending strain (curvature change) components
ε_{tt}	Thickness strain in t-axis direction
$\varepsilon_{\tilde{Z}\tilde{Z}}$	Thickness strain in $\tilde{Z}$-axis direction
$\varepsilon^{ANS}_{\xi\xi}, \varepsilon^{ANS}_{\eta\eta}, \varepsilon^{ANS}_{\xi\eta}$	Assumed in-plane strain components
$\varepsilon^{ANS}_{\zeta\zeta}$	Assumed thickness strain
γ_{tr}, γ_{ts}	Shear strain components
$\gamma^{ANS}_{\zeta\xi}, \gamma^{ANS}_{\zeta\eta}$	Assumed transverse shear strain components
$\boldsymbol{\varepsilon}_m$	Membrane strain (Voigt notation)
$\boldsymbol{\varepsilon}_b$	Bending strain (Voigt notation)
$\boldsymbol{\gamma}$	Transverse shear strain (Voigt notation)
ε^H_b	Curvature change of the hinge-bending model
$\boldsymbol{\varepsilon}^{ES}_b, \boldsymbol{\varepsilon}^{FS}_b, \boldsymbol{\varepsilon}^{SS}_b$	Curvature change of the ES, FS and SS models
$\boldsymbol{\varepsilon}^i_m$	Membrane strain at sampling point i
$\boldsymbol{\varepsilon}_{\bar{m}}$	Averaged membrane strain over edge midpoints
$\epsilon_{\mathbb{f}}$	Residual force tolerance in quasi-static solver
$\epsilon_{\Delta\mathbb{x}}$	Incremental displacement limit

$\otimes$	Kronecker product
$\|\ \ \|$	Magnitude (norm) of the included vector
$\|\ \ \|_\infty$	Infinity norm of the included vector

Chapter 1

Introduction

Thin flexible objects are prevalent across engineering, art, and nature. Examples include garments and wearable textiles, deployable space structures, foldable architectural elements, artistic paper folding, and biological surfaces such as leaves and insect wings. Modeling and simulating thin flexible objects have attracted significant attention from different disciplines including computational mechanics and computer graphics. These objects often exhibit pronounced geometric nonlinearity and strong coupling between geometry and mechanics, leading to complex behaviors under external stimuli. Capturing these behaviors necessitates the development of efficient, accurate, and robust numerical simulation techniques. This thesis focuses on the simulation of two representative thin flexible objects: cloth and origami.

Cloth typically exhibits smooth and highly flexible deformations, whereas origami is characterized by discrete panel deformations and sharp folds along the crease. Plates/shells can serve as a conceptual bridge for modeling both cloth and origami. At the macroscopic scale, cloth (fabric) can be physically modeled as plates/shells, while origami structures can be modeled as assemblies of hinged plates/shells. In both cases, computational models are essential to capture the physical behavior as well as enable efficient and robust numerical simulation. As a result, the development of computational models is the focus of this thesis.

This chapter first provides a separate review of cloth simulation and

origami simulation, each with a particular focus on computational models. Then, the physical modeling of shells and shell-like solids is introduced. Finally, the objective and organization of the thesis are outlined.

1.1 Review on Cloth Simulation

Physics-based cloth simulation has a wide range of applications, including fashion design software for rapid garment prototyping [190], virtual try-on systems for online shopping [73], physically realistic clothing dynamics in games and film [85, 169], pretraining simulators for robotic manipulation of flexible objects [112], as well as interactive experiences on Virtual Reality (VR) and Augmented Reality (AR) devices [120, 90]. These simulations rely on physical modeling of cloth tailored to specific application scenarios, capturing mechanical material properties and interactions such as contact and friction with the environment, thereby providing support for downstream applications.

In this section, the essential components and historical development of physics-based cloth simulation are first summarized. Since this thesis focuses primarily on cloth modeling at the macroscopic scale, the discussion then emphasis on computational models for thin plates/shells. Since the application scenarios are more aligned with graphics-oriented tasks, the review of physics-based cloth simulation primarily draws from the computer graphics community. However, the underlying computational models are broadly applicable and have been developed within both the computer graphics and computational mechanics communities.

Beginning with the seminal work by Terzopoulos et al. [185], cloth has been modeled as an elastically deformable surface grounded on differential geometry and elasticity theory. The surface geometry was discretized using a regular grid for the finite difference method, with semi-implicit integration used for time stepping. Collision dynamics were handled via a simple self-repulsive force model. This foundational framework highlights the essential components of a basic cloth simulator: a

numerical time integrator, a contact handling module, and a computational model for expressing the cloth's physical and mechanical properties.

1.1.1 Numerical Time Integrator

Starting with time integration schemes, the semi-implicit method [185, 2] introduced in early simulators was found to introduce excessive numerical damping, which suppresses dynamic behaviors such as natural fluttering or oscillation of folds. High-frequency motions, such as rapid vibrations or fine wrinkles, tend to be overly damped, resulting in simulations that appear overly smooth or viscous and lacking realism. To alleviate the excessive numerical damping, implicit midpoint method [192], backward Euler method [195], hybrid implicit-explicit methods [168, 19], second-order accurate schemes like BDF2 [31], and symplectic integrators [167] were introduced to better preserve the system's total energy. In parallel, explicit time integration with small time steps was widely adopted [20, 18, 193]. However, explicit schemes are constrained by the Courant– Friedrichs–Lewy (CFL) condition [35], and as stiffness and nonlinearity increase, the simulation system may suffer from severe numerical instability.

Implicit schemes offer unconditional numerical stability and support large time steps, making them the predominant choice for cloth simulation. Inspired by the time-stepping of incremental potential (IP) in the variational framework from Ortiz and Stainier [131], Kharevych et al. [86] and Martin et al. [119] reformulated implicit time integration as a nonlinear optimization problem. In this formulation, the local minimizers of the incremental potential yield the forward prediction of the implicit time integration at each time step [83]. This optimization time integrator enables the use of powerful numerical optimization solvers to minimize the energy, leading to more efficient and stable cloth simulations. To further improve efficiency, projective dynamics (PD) [15, 109] introduced an iterative local and global scheme for implicit time stepping. However, the original PD framework is limited to special

cases where the potential energy assumes a quadratic form of the position. Liu et al. [110] extended PD by introducing Sobolev preconditioning, making it applicable to a broader class of hyperelastic constitutive models. They also employed the Sobolev-preconditioner to initialize the Limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm, significantly improving the convergence of the PD solver. In parallel, Narain et al. [125] and Overby et al. [133] demonstrated that PD can be interpreted as a specific Alternating Direction Method of Multipliers (ADMM) algorithm, where the deformation gradient is treated as a local variable during the local solve step. Both L-BFGS and ADMM are techniques for solving nonlinear optimization problems [129]. More recently, Li et al. [99] proposed the incremental potential contact (IPC) method, which formulates elastodynamics with contact as a constrained incremental potential problem incorporating both contact and dissipative potentials. This formulation transforms a constrained optimization problem into an unconstrained one, enabling well-designed solver for intersection- and inversion-free simulation trajectories. Subsequently, IPC has been widely adopted in various scenarios, including cloth and shell simulation [100], rigid body dynamics [50], and higher-order triangles [49].

1.1.2 Contact Handling

Contact involving both collision and friction is a fundamental component of cloth simulation and plays a key role in achieving realistic behavior. Due to the flexible nature of cloth, contact mainly occur as self-collisions and interactions with external objects.

Collision

The collision module typically consists of two stages: collision detection and collision response.

Performing exhaustive collision tests between all primitive pairs is computationally prohibitive. **Collision detection** is commonly divided into a broad phase and a narrow phase.

The broad phase is responsible for quickly eliminating unnecessary triangle-triangle intersection tests, a process known as collision culling. One widely used approach is to construct a bounding volume hierarchy (BVH), which organizes objects into a tree of hierarchical bounding volumes such as boxes or spheres. This structure allows for efficient pruning of non-colliding pairs by traversing the hierarchy. Various types of BVH have been studied, including bounding spheres [76], OBB [62], AABB [6], spherical shells [91] and box-trees [216]. Another approach, spatial hashing, discretizes the simulation space into uniform grid cells and maps objects into these cells via hash tables. This method reduces the search space by limiting collision detections to neighboring cells. Each approach has its own advantages and disadvantages. Spatial hashing is easy to implement and well-suited for GPU execution, but requires rebuilding the hash table after each object update. In contrast, BVH is more involved to implement and less GPU-friendly, yet supports efficient updates by simply adjusting the bounding volumes of affected objects.

The narrow phase focuses on precise contact determination and typically includes discrete collision detection (DCD) and continuous collision detection (CCD). DCD evaluates potential collisions at fixed time steps and is computationally efficient, but it may miss fast-moving or thin features. CCD addresses this limitation by detecting the time of impact under the assumption that primitives follow linear trajectories within each time interval. For triangle meshes, the narrow phase typically requires point–triangle and edge–edge intersection tests, which often reduce to solving cubic equations. However, the numerical stability of cubic equation solvers is critical [197], necessitating robust algorithms for root finding [199, 215].

Collision response aims to resolve detected intersections and prevent interpenetration. The use of repulsive forces or the penalty method is widely adopted due to its simplicity [185, 2, 15, 60, 116, 198]. Several methods have been proposed to untangle colliding regions by minimizing overlap [3] or reducing the contour length of collision areas [194,

213]. Buffet et al. [22] employed implicit field representations to effectively handle interactions in multi-layer cloth. However, these methods typically require highly stiff penalty energies, which may lead to noticeable jittering artifacts and require careful parameter tuning to ensure stability. Alternative formulations have also been proposed to decouple the cloth dynamics and the collision response. Provot [146] introduced the concept of impact zones, which group repeatedly colliding nodes into rigid clusters. Huh et al. [78] further subdivided these zones based on the nodes' original mesh positions. Harmon et al. [69] improved failure handling within impact zones, allowing limited sliding between involved nodes. Bridson et al. [20] proposed a hybrid method combining repulsion impulses, friction, and impact zones. This method was later adopted in an adaptive mesh refinement cloth simulator, Arcsim [127]. IPC [99] reformulates collision response with interior-point methods and differentiable barrier functions, enabling smooth and robust contact handling. It employs spatial hashing for the broad phase and performs CCD in each nonlinear iteration to keep the simulation strictly within the feasible region. IPC guarantees freedom from interpenetration and element inversion while removing the need for parameter tuning.

Friction

In terms of friction modeling, Baraff and Witkin [2] used damped forces to oppose sliding motion tangent to the contact. Bridson et al. [20] implemented Coulomb friction based on explicit time integration. Otaduy et al. [132] were the first to model cloth contact as a sparse linear complementarity problem (LCP), alternating between normal and tangential contact iterations. Without the pyramid approximation of the Coulomb cone, Li et al. [98] used the exact cone and adaptive node refinement to improve accuracy. To enable fully implicit friction modeling, Brown et al. [21] treated friction as a dissipative term within an optimization framework. Ly et al. [115] proposed a PD approach to frictional contact, enforcing non-penetration and Signorini-Coulomb constraints [37] in a semi-implicit manner. IPC introduces a smooth dissipative potential based on barrier functions to model friction robustly and implicitly. Codimensional incremental potential contact (C-IPC) [100] extends IPC to

handle co-dimensional objects, such as sand, rods, cloth, and volumetric solids, under complex contact scenarios.

In this thesis, the cloth simulation models are integrated into the open-source C-IPC codebase which will be used to handle the contact and friction feature to be involved in the simulation.

1.1.3 Towards High-performance and High-fidelity Cloth Simulation

Based on these developments, some researchers have focused on addressing the computational bottlenecks in cloth simulation. An early effort is position-based dynamics (PBD) [124], which deforms objects such as cloth by explicitly and iteratively projecting constraints between particles in a Gauss-Seidel-like fashion, adjusting inter-particle distances. It enables highly parallelizable computations on graphics processing units (GPUs) but overlooks the underlying physical principles. Subsequent work [117] extended PBD constraints to have explicit correspondence with elastic and dissipative energy potentials, aligning it more closely with the local-global solving structure of PD with implicit Euler time stepping. Both PD and PBD friendly support GPU acceleration due to the trivially parallelizable local projection steps. Instead of solving global systems exactly (e.g., via Cholesky factorization), iterative solvers such as Jacobi [198] and Gauss-Seidel [56] bring significant speedups when implemented on GPUs. Lan et al. [94] approximated the logarithmic barrier function in IPC with a progressively strengthened quadratic function, allowing IPC to be embedded within the PD framework. This offers a promising solution to collision handling challenges within PD, thereby enhancing simulation realism. Huang et al. [73, 74] further accelerated the finte element method (FEM)-based IPC simulation framework by parallelizing it on GPUs.

A recent research direction is differentiable simulation [101, 102], which allows backward gradient propagation during forward simulation to optimize material parameters, friction coefficients, and boundary conditions. For example, it can help garment design tools to infer

appropriate initial conditions for a desired draped state. Another data-driven differentiable approach uses cloth simulator-generated datasets to train graph neural networks (GNNs), learning parameters through message passing between graph nodes and edges for fast inference on GPUs [138]. Follow-up learning-based work includes adding physics-based loss terms to the GNN objective for better generalization [63], as well as unsupervised training approaches to directly predict cloth deformation [10, 11]. However, these methods often struggle to guarantee physical accuracy. Feng et al. [48] proposed a simulation-in-the-loop approach that integrates experimental and simulated fabric drape data to enable neural regression of nonlinear, anisotropic bending stiffness coefficients across a wide range of real fabrics. This approach significantly narrows the gap between simulation and reality compared to earlier data-driven parameter estimation [201, 33], offering better material coverage and faster measurement. However, the computational models of cloth discussed above are primarily based on mass-spring systems or edge-based hinge-bending models. Their limitations of these models will be discussed in detail in the next subsection.

As reviewed above, cloth exhibits high flexibility and undergoes large deformations, which significantly influence its visual appearance. This has made cloth simulation a long-standing topic of interest in the computer graphics community. However, the relevance of cloth simulation extends beyond graphics alone. It involves a wide range of technical challenges and research questions that require interdisciplinary collaboration. The review also reveals that, with ongoing advancements in computational hardware, cloth simulation is moving towards high-performance and high-fidelity. Despite this progress, the trade-off between accuracy and efficiency remains a key concern and must be considered for different applications.

A critical observation is that accurate physical modeling of cloth and the development of suitable computational models (spatial discretization) are fundamental to capturing cloth appearance realistically. These models have been mainly developed in both the computational mechanics and computer graphics communities. In this thesis, the class of cloth

that can be modeled as thin plates or shells, which are commonly encountered in daily life, will be disscussed. The essential role of computational models in producing reliable and physically guaranteed simulation results will be highlighted.

1.1.4 Computational Models for Cloth Simulation

Fabrics are thin and flexible. In the related simulation, they are often modeled as thin plates/shells, with their configurations represented by smooth midsurface geometries. For computer simulation, the smooth surface is discretized typically by the positions and/or displacements of a finite number of nodes on the surfaces. This requirement has led to the development of a wide range of computational models for cloth simulation based on discrete geometry. As discussed in the next subsection on continuum plates and shells, when membrane and bending behaviors are uncoupled, their computational models can be designed separately.

The literature on bending models is notably more extensive than that on membrane models. This imbalance can be attributed to two primary reasons. First, membrane stiffness is typically greater than bending stiffness, which means that membrane deformations are often much smaller and less visually significant than bending deformations. Second, modeling extrinsic deformations via discrete curvature introduces significant challenges in balancing accuracy, efficiency, and stability. In this subsection, the reviewed computational models are developed predominantly from the fields of computer graphics and computational mechanics.

Computational bending models

In the computer graphics community, the seminal work by Terzopoulos et al. [185] introduced the second fundamental form, i.e. the curvature, to model cloth and deformable surfaces which are discretized using finite differences on rectangular grids. In pursuit of improved computational efficiency and robustness, subsequent research explored particle-based and mass-spring models [16, 17, 18, 193, 43, 31] as alternatives to the more complex tensorial treatments. In these methods, separate empirical

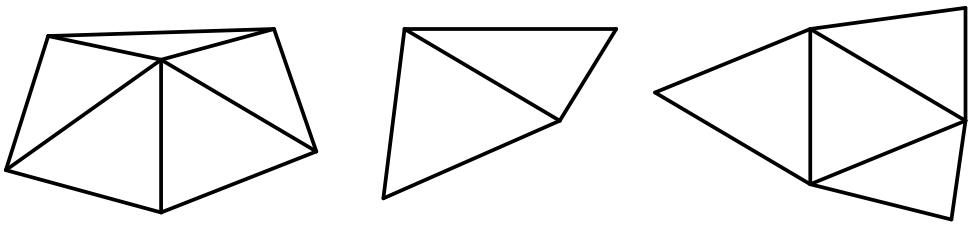


Figure 1.1: Discretization stencils. Left: node-centered stencil; middle: edge-centered stencil; right: triangle-centered stencil.

energy terms were employed to model yarn repulsion, stretching, bending, and trellising deformations. However, such methods often compromised physical fidelity, resulting in material parameters that were mesh-dependent and difficult to transfer across varying mesh topologies. Baraff and Witkin [2] introduced a hinge bending model, akin to the approach of Volino et al. [193], with a focus on cloth simulation for the initially flat configuration. The hinge bending model is computed on an edge stencil, which is formed by two adjacent triangles sharing a common edge (see Figure 1.1 and 1.2). Thus, this bending model is also termed as the edge-based hinge bending model. Based on the edge stencil, Bridson et al. [19] formulated an explicit gradient of the hinge bending energy and integrated it into a mixed explicit-implicit simulation framework. However, their model also only considered the initially flat configuration. In the same conference proceedings [3], Grinspun et al. [66] proposed the discrete shell model, grounded in discrete differential geometry (DDG), to capture the bending energy of thin shells. This approach formulates bending energy of each edge stencil based on the change in mean curvature between the initial and current configurations. This bending energy is similar to the Willmore energy [14], which is popular in the geometry processing community. The gradient and Hessian of the hinge angle (see θ in the left image of Figure 1.2) were initially obtained via automatic differentiation. Automatic differentiation is a technique that computes derivatives by systematically applying the chain rule to a sequence of elementary operations in a program. Later, Tamstorf et al. [183, 182] published the hand-derived formulations and the corresponding bending stiffness. The hinge-bending models of

Bridson et al. and Grinspun et al. are widely adopted in cloth simulators, including Arcsim [127] and C-IPC [100], respectively. Due to its simplicity and robustness, the edge-based hinge bending model [66, 19] is probably the most cited bending model in the graphics community. However, it is limited in its ability to capture complete local curvature behavior. It also suffers significantly from mesh dependency [65] issue, as changes in mesh structure alter the edge stencils used in the bending formulation. Here, the term 'mesh dependency' refers specifically to the sensitivity of the computed bending response to the mesh topology (i.e., the connectivity and orientation of triangle edges) and triangular shape (i.e., aspect ratio and size uniformity), even when the mesh is sufficiently refined. This behavior arises because the edge stencil computes curvature from a single edge and its two adjacent triangles, capturing only the directional curvature perpendicular to that edge. On general surfaces where curvature varies in different directions, the quality of the curvature approximation depends on how well the edge orientations sample the principal curvature directions. Meshes with irregular topology or elongated elements may lead to biased curvature estimates, causing the simulation results to vary with the mesh structure rather than converging consistently to the continuum solution. Moreover, the model still leaves room for improvement in real-time applications. More recently, both the models of Grinspun et al. and Bridson et al. have been shown to deviate from the reference solution introduced by Romero et al. [152]. The reference solution can be taken to be the theoretical solution of the equivalent geometric nonlinear cantilever beam problem, obtained by solving the relevant equilibrium equation numerically. These limitations have spurred ongoing research into improving existing models and developing new formulations for more accurate, efficient and robust cloth and shell simulations.

To enhance the mesh dependency of the edge-based hinge bending model, a triangle-averaged hinge shape operator [61] was described to model the bending strain of a thin shell. The edge-based hinge bending model relies on an edge stencil to compute the discrete mean curvature, while the triangle-averaged hinge bending model computes the sum of

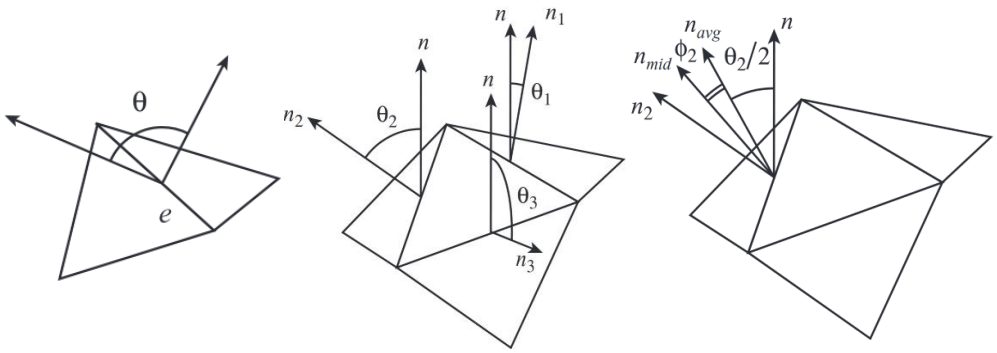


Figure 1.2: Left: edge-based hinge bending model; middle: triangle-averaged hinge bending model; right: midedge shell model. Adapted from [221].

discrete mean curvatures across all edges of the central triangle, averaged by the area of that triangle. The triangle-centered stencil has a central triangle and its three neighbors (see Figure 1.1 and 1.2). The triangle-averaged hinge bending model has been applied to simulate plasticity and fracture [139, 126]. However, the convergence of this model to the ground truth solution remains slow [26, 65]. This issue was effectively addressed by introducing an additional dof which is midedge normal rotation (see ϕ_2 in right image of Figure 1.2) to correct the triangle-averaged hinge shape operator, resulting in what is known as the midedge operator [221, 65]. Just as mesh nodes can be regarded as samples of a continuous surface, midedge normals (see Figure 1.3) defined at the midpoints of triangle edges and pointing toward the smooth surface can be interpreted as samples of the surface normals. This operator has been integrated into libshell [26], which is based on the shear-rigid Koiter shell model, and is also included in the shearable Cosserat shell model [205]. The Cosserat shell model relaxes the KL assumption by introducing independent microrotations, allowing the shell to capture transverse shear deformation. While the midedge operator improves mesh dependency and convergence in thin-shell simulations, the extra midedge dofs introduce a higher computational burden.

To enhance the efficiency of the edge-based hinge bending model, the hinge angle is linearized under the isometry condition, resulting in a constant bending energy Hessian and a linear bending force for thin

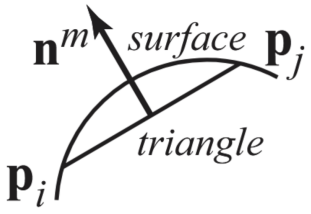


Figure 1.3: A mid-edge normal $\mathbf{n}^m$, defined at the midpoint of a triangle edge and oriented toward the smooth surface. Adapted from [65].

plates, known as the quadratic shell model [8, 9]. This model can be generalized from the cubic shell model [59], which is suitable for initially curved shell configurations. Although both models offer improved efficiency, the mesh dependency issue of the edge-based hinge bending model [66] persists. Wardetzky et al. [203] employed a triangle-centered stencil to construct a discrete Laplacian operator for modeling the pure bending of plates under the isometry assumption. Compared to edge-based stencil models, the triangle-centered approach demonstrates better convergence to the ground truth solutions. More recently, Le et al. [95] proposed a nine-node second-order discrete shell model, demonstrating improved efficiency compared to the discrete shell model [66], thanks to the use of a second-order edge stencil. However, computational modeling with second-order edge stencil requires abandoning the piecewise linear triangle structure.

To enhance the accuracy of the edge-based hinge bending model, Feng et al. [48] presented an improved formulation by treating the common edge in the edge stencil as a torsional spring. This formulation is tested by Romero's cantilever test. In contrast to the DDG shell formulation, this formulation follows a different modeling strategy and therefore does not provide an explicit expression for the bending rigidity. In addition, they employed a neural network to estimate a set of nonlinear and anisotropic material parameters for the model, with the goal of bridging the gap between simulation and visual realism.

To enhance the robustness of the edge-based hinge bending model, Wang et al. [202] and Wu et al. [210] performed eigensystem analyses to refine the Hessian of the bending energy, particularly under extreme

simulation conditions. This line of work was motivated by observations in Tamstorf et al. [182], which noted that the Hessian of the original model can occasionally become non–positive definite, thereby compromising the stability of implicit solvers.

In parallel with the development of DDG shell models, researchers have explored a variety of alternative approaches. For complex knitted fabrics, dedicated representations such as B-spline tubes [82] and crossing rods [145] have been proposed to capture yarn-level fiber behavior in detail. To date, modeling fabric as a homogeneous elastic material with yarn-level fibers still leaves substantial room for improvement. For thick clothing materials like leather, continuum-based thick shell models [29] have been used to account for the constitutive behavior through the material thickness. Etzmuß et al. [46] extract the rotational component from the deformation gradient of each element to map the bending stiffness matrix in the local frame to the global frame. Along similar lines, Thomaszewski et al. [187] adopt subdivision basis functions [32] to enhance simulation accuracy and extract rotation fields from the deformation gradient of each element to handle large rotation. Clyde et al. [33] combined a thin shell model based on subdivision basis functions [32] with data-driven techniques to measure a range of real-world anisotropic fabric parameters. However, a known drawback of subdivision basis functions is their significant computational cost. In addition, a variety of emerging computational models, including the material point method [80], peridynamics [114], and other meshless methods [25], have been explored to simulate cloth behavior. These techniques are widely studied in the computational mechanics community due to their applicability to the simulation and analysis of complex engineering problems.

In the computational mechanics community, the modeling and simulation of plates/shells has been a long-standing topic. Shell finite elements have become increasingly mature and are widely used by engineers in the design and analysis of aerospace vehicles, aircraft, electronic chips, buildings, bridges, among others. Naturally, this has inspired efforts to apply shell finite elements to cloth simulation, and indeed,

many such attempts were made, particularly during the 1990s. Collier et al. [34] developed a large-displacement and small strain analysis method using a four-node anisotropic flat thin shell element to predict the drape coefficient of cotton fabrics. As the interpolated displacement is not first order continuous, the element belongs to the class of incompatible elements. Gan et al. [58] employed a nine-node degenerated shell element. To avoid shear and membrane locking, their formulation incorporated reduced integration and stabilization of spurious zero-energy modes. Kim [89] simulated fabric draping based on the geometrically exact shell theory proposed by Simo et al. [163, 162, 164]. Deng [38] and Eischen et al. [44] extended Kim's work to include buckling, contact, and material nonlinearity. Chen and Govindaraj [24] used a shear-flexible shell element to predict fabric draping behavior, presenting both static results and animated sequences of a square fabric covering a flat square surface. Kang and Yu [84] employed the MITC4 four-node degenerated shell element [4]. However, most of these continuum shell finite element models were limited to simulating low-drapability fabrics. Sze and Liu [174] reported that their attempts to simulate high-drapability fabrics using commercial finite element software either resulted in overly rigid behavior or failed to converge, even when employing arc-length methods. These limitations revealed that conventional plate and shell elements were not particularly successful in capturing the complex behavior of cloth. Consequently, researchers began to explore alternative computational models specifically tailored for cloth simulation.

Chen et al. [27] employed the concept of the finite volume method (FVM) to model the behavior of orthotropic fabrics. The warp and weft curvatures were captured using three nodes along the principal material directions. Membrane tension and compression were modeled by connecting adjacent node pairs along the warp and weft directions, effectively forming line springs. Membrane shear was captured using the edges connecting neighboring nodes in the warp and weft directions. This approach is similar to the particle-based models [17, 43] but benefits from the finite volume framework in which spring constants can be directly computed from experimentally measured stiffness coefficients. Sze and Liu [174, 178] proposed a grid- or particle-based model in which

the curvature along each of the warp and weft directions is computed using a central node and its two adjacent collinear nodes. These nodes are used to define a corotational frame. The corotational formulation subtracts the rigid body displacement from the general displacement such that the difference in the displacements is small and the small displacement assumption can be adopted in calculating the curvature and inplane strain. This enhances both numerical stability and computational efficiency. However, the particle-based methods rely heavily on particle arrangement. Under the small strain/curvature assumption, the method of using transverse displacements in a corotational frame to quantify thin plate bending energy [174, 178] was later extended by Liu and Sze [111]. They constructed a biquadratic displacement interpolation function for curvature evaluation by Eq.(1.30). This significantly relaxed the nodal alignment requirements, allowing it to accommodate nodal distributions consistent with bi-quadratic or 9-node quadrilateral elements. Subsequently, Zhou and Sze [218] generalized this idea to triangle-centered stencils. A quadratic displacement field was interpolated via six nodes of a triangle-centered stencil and used to derive curvature expressions. They proposed corresponding formulations for thin plates and thin shells, validated their method using geometrically nonlinear benchmarks of thin shells [180], and applied their model on the fabric drape simulation. Notably, both [111] and [218] yield constant bending energy Hessians which is a favorable feature for nonlinear solution procedures. Sze and Zhou [176] later integrated the rotation-free triangle model [218] into an explicit time integrator with global adaptive remeshing and a collision module, aiming at more general cloth simulation applications. However, explicit integration suffers from instability and unfortunately does not take advantage of the constant bending energy Hessian property. Additionally, global adaptive remeshing is significantly more computationally expensive than local strategies in Arcsim [127], especially when coupled with small time steps required by explicit schemes. And the collision handling scheme is similar to that of Bridson et al. [20], which still struggles to guarantee penetration-free cloth simulations. Sze and Liu [179] also attempted to simulate drape behavior using a stress-resultant bilinear solid-shell element with assumed

natural transverse shear and thickness strains. However, under high drapability conditions, this method often led to reversed nodal directors, resulting in non-physical solutions. A quasi-engineering strain was introduced as a remedy. Xie et al. [211] extended the bilinear solid-shell elements into a similar cloth simulation pipeline as in [176], and introduced a reversible local adaptive meshing strategy for 4-node quadrilateral and 3-node triangular solid-shell finite elements. Lu and Zheng [113] proposed a NURBS-based isogeometric shell model with only nodal displacements as dofs, also employing explicit time integration and the contact algorithm of [20]. However, the use of high-order basis functions increases computational costs and complicates accurate contact detection. More recently, Shin and Lee [159] extended Lu and Zheng's model by incorporating discontinuous and bubble-type enrichment functions, and demonstrated cloth simulation results that capture tearing behavior. From observing the simulated appearances presented by these computational mechanics researchers, it is fair to conclude that particle and rotation-free plate/shell models [27, 174, 111, 218] yield more visually pleasant results in cloth simulation compared to solid-shell and isogeometric shell models [179, 113, 159]. Here, "rotation-free" refers to shell formulations where nodal rotations are not involved.

For rotation-free plate/shell models, Oñate and his collaborators have conducted a series of studies based on the FVM and triangular meshes. Oñate and Zárate [130] first derived the basic plate triangle model. In this model, using a triangle-centered stencil, the control volume is defined by the central triangle, and its curvature is computed through a superposition of contributions from the triangle edges using the FVM. The displacement field within each triangle is assumed to be linearly interpolated. This approach was later extended to a node-centered stencil (see Figure 1.1) for evaluating the curvature of node-centered control volumes and was further generalized to shell analysis, resulting in the basic shell triangle model [134]. An improved version of basic plate triangle model was subsequently proposed using an assumed strain method with a six-node interpolation scheme [55]. These computational models have been applied in various contexts, such as

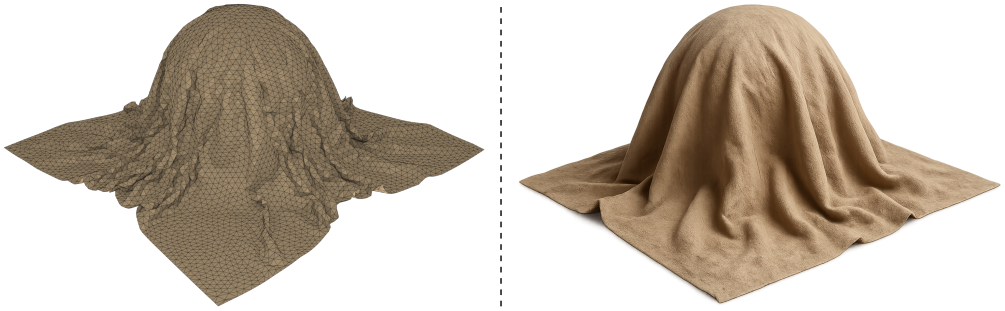


Figure 1.4: Draped cloth on a sphere fixed on the floor. The left image shows the simulated result, while the right one presents the real-world capture. A comparison reveals the presence of non-physical sharp folds in the simulated cloth, also known as sharp creasing artifacts.

meta-forming [54] and fabric drape simulations [53]. Phaal and Calladine [140, 141] developed a model based on an edge stencil, constructing quadratic displacement interpolation functions to describe bending deformation. However, since edge stencils cannot represent a complete quadratic displacement field in two dimensions, the authors introduced the triangle-centered stencil to construct a complete quadratic interpolation. Their analysis, however, was limited to linear plate and linear shell models. Other related approaches [154, 68], although not based on FVM or polynomial interpolation, also utilized triangle-centered stencils. These methods construct element stiffness matrices through coupling side rotations at the edges of the central triangle with the curvature of the central triangle. The side rotation refers to the motion of a flap triangle rotating about the edge of the central triangle. However, these works did not address drape/cloth simulation.

Computational membrane models

For membrane behavior, in particle and mass-spring models [27, 2, 174], stretch and compression are typically modeled by line springs (akin to bar elements), while membrane shear is represented through angular changes between warp and weft directions. On triangular meshes, the discrete shell model [66] measures local stretch via bar elements and models membrane shear through area changes of triangles. However,

this approach is not well-suited for modeling anisotropic materials. Consequently, the most widely used membrane model is the constant strain triangle (CST) element [26, 218, 127, 134]. Based on a triangle-centered stencil, Flores and Oñate [55] introduced a six-node interpolation scheme under the assumed strain approach. Similarly, Sze and Zhou [176] presented a six-node interpolated membrane model to mitigate the non-physical sharp folds in cloth simulations. The non-physical sharp folds are also referred to as sharp creasing artifacts [100], which is illustrated in Figure 1.4 and will be further discussed in Chapter 4. In this thesis, the latter term will be adopted. The six-node interpolation functions in [55] and [176] are both based on a triangle-centered stencil, and are used for interpolating displacements and positions. Quaglino [148] proposed a linear membrane triangle model to address the tendency of linear elements to underestimate intrinsic distances between vertices when the surface undergoes bending. This underestimation often leads to an overprediction of compressive stresses. The model was shown to deliver quantitatively accurate results on geometrically nonlinear shell benchmark tests and qualitatively superior performance in cloth draping scenarios using coarse meshes. Nevertheless, a notable drawback is that the method requires solving a buckling problem for each mesh edge, which poses significant challenges for Newton-type solvers.

From the broader perspective of computational cloth modeling, there is a growing consensus across both the computer graphics and computational mechanics communities that high-performance and high-fidelity cloth simulation demands accurate and efficient computational models. Accordingly, the cloth part of this thesis aims to develop such models through discrete formulations that faithfully replicate smooth physical behavior, achieving a careful balance between accuracy and efficiency while ensuring numerical robustness.

1.2 Review on Origami Simulation

Origami, rooted in paper art, has long fascinated artists. In recent years, however, it has also garnered increasing attention from engineers and

researchers across various disciplines due to its geometrically inspired structures and the resulting rich and multifunctional mechanical properties. These properties include, but are not limited to, negative Poisson's ratio [212], negative bending stiffness [204], bistability [47], and multistability [196]. As a result, origami-based designs have found practical applications in a wide range of fields, including reconfigurable aerospace structures [57], soft robotics [153], mechanical metamaterials [79], biomedical devices [97], electronic skins [5], etc.

Origami structures in three dimensions are typically formed by folding two-dimensional geometric configurations. These geometric forms are closely tied to the crease pattern, which generally categorizes origami structures into two types: straight-crease and curved-crease origami structures. Fundamental differences exist between the two. In curved-crease origami structures, folding along a curved crease inevitably induces bending in the connected or adjacent panels, resulting in a significantly richer deformation space. In contrast, for straight-crease origami structures, the folding path is relatively deterministic once the type of fold (mountain or valley) and the straight creases are specified. This is because each straight crease involves only a single straight line connecting two vertices. Nevertheless, straight-crease origami structures is still capable of producing a wide variety of configurations, including those with complex curvature [41]. This highlights the necessity of simulation tools for the design and analysis of origami structures.

Generally, origami simulations can be divided into two categories: kinematics-based origami simulation and mechanics-based origami simulation.

1.2.1 Computational Models for Kinematics-based Origami Simulation

Early efforts to simulate origami folding can be traced back to the 1970s, with kinematics-based methods being the primary approach [77]. In kinematics-based simulations of straight-crease origami structures, a key

assumption is that the panels remain rigid and undeformed throughout the folding process. This means that the isometry constraint preserves the rigidity of each planar panel. In these models, the structure is typically described using crease folding angles as variables, and the amount of folding is determined by solving kinematic constraint equations. Representative kinematics-based methods include reflection-based [122], trigonometry-based [160, 196], and loop closure-based [181] simulations. For curved-crease origami structures, the isometry constraint is also applied, as seen in approaches such as differential geometry [77, 123] and group orbit tessellation [106]. However, a major limitation of these isometry-based approaches is that their underlying mathematical formulations are difficult to extend to include physical constitutive relations during the deformation process.

1.2.2 Computational Models for Mechanics-base Origami Simulation

In mechanics-based simulations of origami systems, the isometry constraint on panels is relaxed, allowing for both in-plane and bending deformations of the panels. Burgoon et al. [23] manually predefined the crease locations and dense meshes nearly the crease regions. Their simulation was based on the discrete shell model [66], which treats the whole origami structure, i.e. panels cum creases, as a continuous shell structure. Narain et al. [126] introduced adaptive mesh refinement to increase resolution in regions of high curvature, thereby preserving the small-stencil assumption for hinge bending. In DDG shells, the small-stencil assumption allows the discrete geometry to better approximate the smooth geometry, similar to how fine meshing improves accuracy in finite element shell models. Combined with a plastic bending constitutive model, this approach better captures the detailed geometry of creases and wrinkles. Both methods treat the paper as a continuous medium, even at the creases. Other related approaches include the compliant crease or smooth fold models [137, 217, 219], which either use a compliant strip to connect rigid panels or model the crease and panel

as a continuous material with locally reduced stiffness near the crease. These approaches fall outside the scope of this thesis.

This thesis focuses on a class of origami structures modeled as collections of origami panels connected by hinges along crease (fold lines). Mathematically, these origami structures can be locally viewed as a two-dimensional discrete manifold [121]. A manifold is a space that, at each point, locally resembles Euclidean space (e.g., a curve looks like a line, a surface looks like a plane near any point). The geometric features of this manifold are described by a set of (straight- or curved-) creases, where sharp folding occurs. [121]. For this class of structures, Schenk and Guest [157] were the first to employ the bar-hinge model to simulate straight-crease origami. In their formulation, bars constrain in-plane deformation, while hinges capture the folding motion between discrete origami panels. Explicit time integration was used for numerical simulation. Using a similar approach, Dias et al. [39] investigated and analyzed the geometric and mechanical behavior of curved-crease origami. Liu and Paulino [108, 107] derived explicit expressions for the gradient and Hessian of hinge angles, enabling quasi-static simulations of straight-crease origami using the bar-hinge model. They also provide an origami simulation framework, MERLIN2 [107]. Building on this framework, Woodruff et al. [207] extended it to curved-crease origami structures by introducing an edge-based hinge bending model that captures the bending deformation of origami panels. The bar-hinge model remains one of the most widely cited frameworks for origami simulation, largely due to its simplicity and efficiency. Note worthily, its underlying computational model is closely related to cloth simulation models of Bridson et al. [19], thus inherits some of its limitations. For example, in the in-plane bar model, it is difficult to calibrate elasticity parameters consistently across different mesh topologies. In modeling the bending behavior of panels for curved-crease origami, the edge-based hinge bending model is highly mesh-dependent and struggles to converge reliably to the ground-truth of smooth surface deformations.

Hu et al. [72] modeled the membrane deformation of origami panels using 3-node and 4-node three-dimensional membrane elements in

ABAQUS. While the element edges of these elements are straight, the only admissible bending mode is the warping curvature in the 4-node element. In this light, Hu et al. derived a simple analytical expression for the relevant bending energy. These finite element panels were connected via hinges. Their formulation simplifies the gradient of the hinge angle, which, while computationally efficient, results in convergence issues in geometrically nonlinear analyses. Additionally, their model is limited to capturing only the warping bending within individual elements. Filipov et al. [52] modeled origami panels using S4 shell elements in ABAQUS, combined with rotational springs to simulate straight-crease origami structures. Wen et al. [206] modeled panels with S4R shell elements and introduced joint-rotation connectors aligned with fold directions to simulate curved-crease origami structures. In [72], the authors also compared the performance of SC8R and S4R elements in ABAQUS for simulating straight-crease origami structures. They observed that S4R elements required significantly more iterations for convergence. This behavior may be attributed to the nodal rotational dofs of S4R.

Rabinovich et al. [150] proposed a discrete binary characterization for folds between discrete developable surfaces, along with an algorithm that enables simultaneous crease folding and smooth sheet bending. However, the isometry constraints inherent in this model hinder the incorporation of physical constitutive relations for smooth surface deformation. Other ruling-based approaches [87, 165, 166, 184, 158] share similar limitations.

The aforementioned origami structures are typically crafted from thin sheets, often assumed to have negligible thickness. However, in practical engineering applications, panel thickness may need to be considered, giving rise to the concept of thick origami [28]. For related simulations, Kwok [93] employed solid elements to model shell-like solids. This approach often requires fine meshing through the thickness to avoid locking issues.

In summary, despite the diverse and complex nature of origami structures, current computational models remain limited and still offer substantial room for improvement. This thesis aims to enhance the accuracy of bar-hinge model for curved-crease origami structures [107, 207], improve both the efficiency and stability of the model in [72], and propose a new approach for modeling shell-like solid origami panels. Specifically, the solid-shell element is employed to model the origami panels and its nodal directors are used to model folding/unfolding mechanism of the crease.

1.3 Review of Shells and Shell-like Solids

1.3.1 Geometry and Elasticity in Thin Plates/Shells

A thin shell is considered under the Kirchhoff–Love (KL) assumption which states that a straight line of material initially normal to the midsurface of the shell remains straight and normal to the deformed midsurface of the shell. As a result, transverse shear deformation is neglected. Deformation through the thickness is also neglected, so the shell maintains a constant thickness h.

In a thin shell, the positions $\mathbf{X}$ and $\mathbf{x}$ of a point in the initial and current configurations can be expressed as

$$\mathbf{X}(r,s,t) = \mathbf{X}_o(r,s) + t\mathbf{X}_n(r,s), \quad \mathbf{x}(r,s,t) = \mathbf{x}_o(r,s) + t\mathbf{x}_n(r,s), \tag{1.1}$$

where $\mathbf{X}_o$ and $\mathbf{x}_o$ denote the positions of the midsurface in the initial and current configurations, respectively; curvilinear coordinates r and s parametrize the midsurface; $t \in [-h/2, h/2]$ is the coordinate along the thickness direction. Under the KL assumption, the directors $\mathbf{X}_n$ and $\mathbf{x}_n$ are normal to the midsurface, i.e.

$$\mathbf{X}_n = \frac{\mathbf{X}_{o,r} \times \mathbf{X}_{o,s}}{\|\mathbf{X}_{o,r} \times \mathbf{X}_{o,s}\|}, \quad \mathbf{x}_n = \frac{\mathbf{x}_{o,r} \times \mathbf{x}_{o,s}}{\|\mathbf{x}_{o,r} \times \mathbf{x}_{o,s}\|}. \tag{1.2}$$

In the current configuration, the metric tensor

$$\mathbf{g} = \begin{bmatrix} g_{rr} & g_{rs} & g_{rt} \\ g_{sr} & g_{ss} & g_{st} \\ g_{tr} & g_{ts} & g_{tt} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_{,r} \cdot \mathbf{x}_{,r} & \mathbf{x}_{,r} \cdot \mathbf{x}_{,s} & \mathbf{x}_{,r} \cdot \mathbf{x}_{,t} \\ \mathbf{x}_{,s} \cdot \mathbf{x}_{,r} & \mathbf{x}_{,s} \cdot \mathbf{x}_{,s} & \mathbf{x}_{,s} \cdot \mathbf{x}_{,t} \\ \mathbf{x}_{,t} \cdot \mathbf{x}_{,r} & \mathbf{x}_{,t} \cdot \mathbf{x}_{,s} & \mathbf{x}_{,t} \cdot \mathbf{x}_{,t} \end{bmatrix} \tag{1.3}$$

of point $\mathbf{x}(r, s, t)$ captures the local geometric deformation of the body. Under the KL assumption, the metric components hold $g_{tt} = 1$, $g_{rt} = g_{tr} = 0$, and $g_{rs} = g_{sr} = 0$, while the in-plane components $g_{\alpha\beta}$ (with $\alpha, \beta = r, s$) tangentially to the midsurface are given by

$$g_{\alpha\beta} = \mathbf{x}_{,\alpha} \cdot \mathbf{x}_{,\beta} = \mathbf{x}_{o,\alpha} \cdot \mathbf{x}_{o,\beta} + t \left(\mathbf{x}_{o,\alpha} \cdot \mathbf{x}_{n,\beta} + \mathbf{x}_{n,\alpha} \cdot \mathbf{x}_{o,\beta} \right) + t^2 \, \mathbf{x}_{n,\alpha} \cdot \mathbf{x}_{n,\beta}. \tag{1.4}$$

Neglecting the higher-order term in t^2, the metric simplifies to

$$g_{\alpha\beta} = \mathsf{a}_{\alpha\beta} - 2t \, \kappa_{\alpha\beta}, \tag{1.5}$$

where $\mathsf{a}_{\alpha\beta} = \mathbf{x}_{o,\alpha} \cdot \mathbf{x}_{o,\beta}$ is the first fundamental form (metric of the midsurface), and

$$\kappa_{\alpha\beta} = -\frac{1}{2} \left(\mathbf{x}_{o,\alpha} \cdot \mathbf{x}_{n,\beta} + \mathbf{x}_{n,\alpha} \cdot \mathbf{x}_{o,\beta} \right) \tag{1.6}$$

is the curvature (the second fundamental form). Due to the orthogonality between the director vector $\mathbf{x}_n$ and the in-plane tangent vectors $\mathbf{x}_{o,\alpha}$ and $\mathbf{x}_{o,\beta}$, we have

$$\mathbf{x}_n \cdot \mathbf{x}_{o,\alpha} = 0, \quad \mathbf{x}_n \cdot \mathbf{x}_{o,\beta} = 0. \tag{1.7}$$

Taking partial derivatives of the expressions in Eq.(1.7) yields:

$$\begin{aligned} \frac{\partial}{\partial \alpha} (\mathbf{x}_n \cdot \mathbf{x}_{o,\beta}) &= \mathbf{x}_{n,\alpha} \cdot \mathbf{x}_{o,\beta} + \mathbf{x}_n \cdot \mathbf{x}_{o,\beta\alpha} = 0, \\ \frac{\partial}{\partial \beta} (\mathbf{x}_n \cdot \mathbf{x}_{o,\alpha}) &= \mathbf{x}_{n,\beta} \cdot \mathbf{x}_{o,\alpha} + \mathbf{x}_n \cdot \mathbf{x}_{o,\alpha\beta} = 0. \end{aligned} \tag{1.8}$$

Under the assumption of smoothness and symmetry of mixed partial derivatives, it follows that

$$\mathbf{x}_n \cdot \mathbf{x}_{o,\alpha\beta} = \mathbf{x}_n \cdot \mathbf{x}_{o,\beta\alpha} = -\mathbf{x}_{o,\alpha} \cdot \mathbf{x}_{n,\beta} = -\mathbf{x}_{n,\alpha} \cdot \mathbf{x}_{o,\beta}. \tag{1.9}$$

Therefore, the curvature $\kappa_{\alpha\beta}$ can be equivalently written as

$$\begin{aligned}\kappa_{\alpha\beta} &= -\frac{1}{2}\left(\mathbf{x}_{o,\alpha}\cdot\mathbf{x}_{n,\beta}+\mathbf{x}_{n,\alpha}\cdot\mathbf{x}_{o,\beta}\right)\\ &= -\mathbf{x}_{o,\alpha}\cdot\mathbf{x}_{n,\beta} = -\mathbf{x}_{n,\alpha}\cdot\mathbf{x}_{o,\beta}\\ &= \mathbf{x}_{n}\cdot\mathbf{x}_{o,\alpha\beta} = \mathbf{x}_{n}\cdot\mathbf{x}_{o,\beta\alpha}.\end{aligned} \tag{1.10}$$

In the initial configuration, the in-plane components of the metric tensor $\mathbf{G}$ take the form

$$G_{\alpha\beta} = \mathsf{A}_{\alpha\beta} - 2t\,K_{\alpha\beta}, \tag{1.11}$$

where $\mathsf{A}_{\alpha\beta}$ and $K_{\alpha\beta}$ denote the first fundamental form and the curvature evaluated in the initial configuration, respectively.

The in-plane Green–Lagrange strain tensor in curvilinear coordinates is

$$\varepsilon_{\alpha\beta} = \frac{1}{2}\left(g_{\alpha\beta}-G_{\alpha\beta}\right) = \frac{1}{2}\left(\mathsf{a}_{\alpha\beta}-\mathsf{A}_{\alpha\beta}\right) - t\left(\kappa_{\alpha\beta}-K_{\alpha\beta}\right). \tag{1.12}$$

This expression allows the introduction of the membrane strain and curvature change in curvilinear coordinates as

$$\varepsilon_{m\alpha\beta} = \frac{1}{2}\left(\mathsf{a}_{\alpha\beta}-\mathsf{A}_{\alpha\beta}\right), \quad \varepsilon_{b\alpha\beta} = -\left(\kappa_{\alpha\beta}-K_{\alpha\beta}\right). \tag{1.13}$$

The material properties of a shell are typically defined in a local Cartesian frame (e.g., an $\tilde{X}$–$\tilde{Y}$–$\tilde{Z}$ frame), where the $\tilde{X}$–$\tilde{Y}$ plane is tangent to the initial midsurface and the $\tilde{Z}$-axis is parallel to the t-axis. In this setting, the membrane and bending strains in the local Cartesian coordiantes can be expressed as

$$\boldsymbol{\varepsilon}_m = \begin{Bmatrix}\varepsilon_{m\tilde{X}\tilde{X}}\\ \varepsilon_{m\tilde{Y}\tilde{Y}}\\ 2\varepsilon_{m\tilde{X}\tilde{Y}}\end{Bmatrix} = \mathbf{T}_{mb}\begin{Bmatrix}\varepsilon_{mrr}\\ \varepsilon_{mss}\\ 2\varepsilon_{mrs}\end{Bmatrix}, \quad \boldsymbol{\varepsilon}_b = \begin{Bmatrix}\varepsilon_{b\tilde{X}\tilde{X}}\\ \varepsilon_{b\tilde{Y}\tilde{Y}}\\ 2\varepsilon_{b\tilde{X}\tilde{Y}}\end{Bmatrix} = \mathbf{T}_{mb}\begin{Bmatrix}\varepsilon_{brr}\\ \varepsilon_{bss}\\ 2\varepsilon_{brs}\end{Bmatrix}, \tag{1.14}$$

where $\mathbf{T}_{mb}$ is the transformation matrix that maps strains from the curvilinear coordinates to the local Cartesian coordinates. $\mathbf{T}_{mb}$ is stated in the Eq.(C.4).

This thesis primarily focuses on computational models of the thin shell and their comparison with discrete differential geometry (DDG) shells. The main objective of DDG is to develop discrete geometric models that mirror the intrinsic structures of smooth geometry, not merely as numerical approximations, but as mathematically coherent objects that preserve key geometric properties of the continuous theory. DDG shells are often characterized by a coordinate-free formulation of the discrete first and second fundamental forms, and typically adopt the St. Venant–Kirchhoff (StVK) material model for continuous shell model [66, 8, 59, 26]. The StVK model assumes the material is isotropic and homogeneous. For a fair comparasion, the StVK model is adopted for the present computational models. Accordingly, the elastic energy Ψ_{ts} of a thin shell with StVK model can be expressed as

$$\begin{aligned}\Psi_{ts} = \frac{1}{2}\int_{\Omega}\int_{-\frac{h}{2}}^{\frac{h}{2}} \Bigg(&\frac{E\nu}{1-\nu^2}\,\mathrm{Tr}^2([\boldsymbol{\varepsilon}_m + \tilde{Z}\boldsymbol{\varepsilon}_b]) \\ &+ \frac{E}{1+\nu}\,\mathrm{Tr}\left([\boldsymbol{\varepsilon}_m + \tilde{Z}\boldsymbol{\varepsilon}_b]^2\right)\Bigg) d\tilde{Z}\, d\Omega,\end{aligned} \tag{1.15}$$

where E is the Young's modulus, ν is the Poisson's ratio, and Ω is the total area of the midsurface in the initial configuration. $[\boldsymbol{\varepsilon}_m + \tilde{Z}\boldsymbol{\varepsilon}_b]$ denote the in-plane Green-Lagrangian strain in tensor format. Tr returns the trace of a tensor. The continuous shell model in Eq.(1.15) can also be seen as Koiter shell model. In matrix-vector format, Eq.(1.15) can be rewritten as

$$\Psi_{ts} = \frac{1}{2}\int_{\Omega}\int_{-\frac{h}{2}}^{\frac{h}{2}} (\boldsymbol{\varepsilon}_m + \tilde{Z}\boldsymbol{\varepsilon}_b)^T \mathbf{D}_{ps} (\boldsymbol{\varepsilon}_m + \tilde{Z}\boldsymbol{\varepsilon}_b) d\tilde{Z}\, d\Omega, \tag{1.16}$$

where, the plane-stress material matrix $\mathbf{D}_{ps}$ is

$$\mathbf{D}_{ps} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}. \tag{1.17}$$

By integrating through the coordinate $\tilde{Z}$, the energy

$$\Psi_{ts} = \Psi_m + \Psi_b = \frac{h}{2}\int_{\Omega} \boldsymbol{\varepsilon}_m^T \mathbf{D}_{ps} \boldsymbol{\varepsilon}_m d\Omega + \frac{h^3}{24}\int_{\Omega} \boldsymbol{\varepsilon}_b^T \mathbf{D}_{ps} \boldsymbol{\varepsilon}_b d\Omega \tag{1.18}$$

of the thin shell can be decomposed into the membrane energy

$$\Psi_m = \frac{1}{2}\int_\Omega \boldsymbol{\varepsilon}_m^T \mathbf{D}_m \boldsymbol{\varepsilon}_m d\Omega, \tag{1.19}$$

and the bending energy

$$\Psi_b = \frac{1}{2}\int_\Omega \boldsymbol{\varepsilon}_b^T \mathbf{D}_b \boldsymbol{\varepsilon}_b d\Omega = \frac{1}{2}\int_\Omega (\boldsymbol{\kappa} - \boldsymbol{K})^T \mathbf{D}_b (\boldsymbol{\kappa} - \boldsymbol{K}) d\Omega, \tag{1.20}$$

where $\mathbf{D}_m = h\mathbf{D}_{ps}$ and $\mathbf{D}_b = h^3\mathbf{D}_{ps}/12$ are respectively the membrane and bending rigidity matrix. The vector $\boldsymbol{K} = \begin{bmatrix} K_{\tilde{X}\tilde{X}} & K_{\tilde{Y}\tilde{Y}} & 2K_{\tilde{X}\tilde{Y}} \end{bmatrix}^T$ represents the curvature in the initial configuration, while $\boldsymbol{\kappa}$ denotes its counterpart in the current configuration. The bending deformation of the thin shell can be measured by the curvature change of the midsurface from the initial configuration to the current configuration. For a thin plate, which has an initial flat state, the $\boldsymbol{K}$ vanishes.

For a thin plate deformed with small displacement, the midsurface of the thin plate is assumed to lie in the X–Y plane of the global Cartesian frame, and the displacement field is given by (U, V, W). The curvature can then be expressed in terms of this displacement field. A point on the midsurface of the plate in the initial and current configuration can be expressed as

$$\begin{aligned} \mathbf{X}_o &= (X_o, Y_o, Z_o) = (X, Y, 0), \\ \mathbf{x}_o &= (x_o, y_o, z_o) = (X + U, Y + V, W). \end{aligned} \tag{1.21}$$

By recalling Eq.(1.6), the curvature can be expressed as

$$\boldsymbol{\kappa} = \begin{Bmatrix} \kappa_{XX} \\ \kappa_{YY} \\ 2\kappa_{XY} \end{Bmatrix} = \begin{Bmatrix} \mathbf{x}_n \cdot \frac{\partial^2 \mathbf{x}_o}{\partial X^2} \\ \mathbf{x}_n \cdot \frac{\partial^2 \mathbf{x}_o}{\partial Y^2} \\ 2\mathbf{x}_n \cdot \frac{\partial^2 \mathbf{x}_o}{\partial X \partial Y} \end{Bmatrix}, \tag{1.22}$$

where

$$\mathbf{x}_n = \left(\frac{\partial \mathbf{x}_o}{\partial X} \times \frac{\partial \mathbf{x}_o}{\partial Y}\right) / \left\| \frac{\partial \mathbf{x}_o}{\partial X} \times \frac{\partial \mathbf{x}_o}{\partial Y} \right\|. \tag{1.23}$$

The derivatives of the $\mathbf{x}_o$ are

$$\frac{\partial \mathbf{x}_o}{\partial X} = (1 + \frac{\partial U}{\partial X}, \frac{\partial V}{\partial X}, \frac{\partial W}{\partial X}), \quad \frac{\partial \mathbf{x}_o}{\partial Y} = (\frac{\partial U}{\partial Y}, 1 + \frac{\partial V}{\partial Y}, \frac{\partial W}{\partial Y}), \tag{1.24}$$

and

$$\begin{aligned}\frac{\partial^2 \mathbf{x}_o}{\partial X^2} &= (\frac{\partial^2 U}{\partial X^2}, \frac{\partial^2 V}{\partial X^2}, \frac{\partial^2 W}{\partial X^2}),\\ \frac{\partial^2 \mathbf{x}_o}{\partial Y^2} &= (\frac{\partial^2 U}{\partial Y^2}, \frac{\partial^2 V}{\partial Y^2}, \frac{\partial^2 W}{\partial Y^2}),\\ \frac{\partial^2 \mathbf{x}_o}{\partial X \partial Y} &= (\frac{\partial^2 U}{\partial X \partial Y}, \frac{\partial^2 V}{\partial X \partial Y}, \frac{\partial^2 W}{\partial X \partial Y}).\end{aligned} \tag{1.25}$$

The cross product between the tangents of the deformed midsurface with the Eq.(1.24) is

$$\begin{aligned}\frac{\partial \mathbf{x}_o}{\partial X} \times \frac{\partial \mathbf{x}_o}{\partial Y} = \Bigg(&-\frac{\partial W}{\partial X} + \frac{\partial V}{\partial X}\frac{\partial W}{\partial Y} - \frac{\partial W}{\partial X}\frac{\partial V}{\partial Y},\\ &-\frac{\partial W}{\partial Y} + \frac{\partial W}{\partial X}\frac{\partial U}{\partial Y} - \frac{\partial U}{\partial X}\frac{\partial W}{\partial Y},\\ &1 + \frac{\partial U}{\partial X} + \frac{\partial V}{\partial Y} + \frac{\partial U}{\partial X}\frac{\partial V}{\partial Y} - \frac{\partial V}{\partial X}\frac{\partial U}{\partial Y}\Bigg).\end{aligned} \tag{1.26}$$

Assuming that the in-plane deformation of the plate is nearly inextensible, the in-plane displacements U and V and their derivatives are close to zero. Although the transverse displacement W is also small, it is typically larger than the in-plane displacements. The partial derivatives $\partial W/\partial X$ and $\partial W/\partial Y$ represent the rotations about the Y- and X-axes, respectively. Under these assumptions, the norm of the cross product between the tangents of the deformed midsurface becomes

$$\left\| \frac{\partial \mathbf{x}_o}{\partial X} \times \frac{\partial \mathbf{x}_o}{\partial Y} \right\| \simeq \sqrt{1 + \left(-\frac{\partial W}{\partial X}\right)^2 + \left(-\frac{\partial W}{\partial Y}\right)^2} \simeq 1, \tag{1.27}$$

where $\partial W/\partial X^2$ and $\partial W/\partial Y^2$ are second-order small quantities. Therefore, the unit normal vector $\mathbf{x}_n$ defined in Eq. (1.23) can be approximated as

$$\mathbf{x}_n \simeq \left(-\frac{\partial W}{\partial X}, -\frac{\partial W}{\partial Y}, 1\right). \tag{1.28}$$

Substituting Eqs.(1.28) and (1.25) into Eq.(1.22), and neglecting third-order and higher-order terms, the curvature vector $\boldsymbol{\kappa}$ can be expressed purely in terms of the second derivatives of the transverse displacement

W as

$$\boldsymbol{\kappa} = \begin{Bmatrix} \kappa_{XX} \\ \kappa_{YY} \\ 2\kappa_{XY} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial^2 W}{\partial X^2} \\ \frac{\partial^2 W}{\partial Y^2} \\ 2\frac{\partial^2 W}{\partial X \partial Y} \end{Bmatrix}. \tag{1.29}$$

Curvature describes the degree of bending of a curve or surface in the vicinity of a given point. The method for measuring curvature discussed above can be extended to more general surfaces. At any point on a surface, a local Cartesian frame $(\tilde{X}, \tilde{Y}, \tilde{Z})$ can be defined such that the $\tilde{X}$–$\tilde{Y}$ plane is tangential to the surface. Let the displacement field relative to this local frame be denoted by $(\tilde{U}, \tilde{V}, \tilde{W})$. Then, the curvature at that point can be defined as

$$\begin{Bmatrix} \kappa_{\tilde{X}\tilde{X}} \\ \kappa_{\tilde{Y}\tilde{Y}} \\ 2\kappa_{\tilde{X}\tilde{Y}} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial^2 \tilde{W}}{\partial \tilde{X}^2} \\ \frac{\partial^2 \tilde{W}}{\partial \tilde{Y}^2} \\ 2\frac{\partial^2 \tilde{W}}{\partial \tilde{X} \partial \tilde{Y}} \end{Bmatrix}. \tag{1.30}$$

This approach to measuring curvature will be adopted in Chapter 3.

1.3.2 Geometry and Elasticity in Shell-like Solid

Compared to thin shells discussed in Section 1.3.1, shell-like solids require additional consideration of the stiffness and energy contributions of the transverse shear and through-thickness strains. They arise in the components of the metric tensors $\mathbf{g}$ involving the thickness direction, i.e.

$$g_{tr} = \mathbf{x}_{,t} \cdot \mathbf{x}_{,r},\ g_{ts} = \mathbf{x}_{,t} \cdot \mathbf{x}_{,s},\ g_{tt} = \mathbf{x}_{,t} \cdot \mathbf{x}_{,t}. \tag{1.31}$$

By invoking $\mathbf{x} = \mathbf{x}_o + t\mathbf{x}_n$ in Eq.(1.1), the components in Eq.(1.31) can be expressed as

$$\begin{aligned} g_{tr} &= \mathbf{x}_n \cdot \mathbf{x}_{o,r} + t\, \mathbf{x}_n \cdot \mathbf{x}_{n,r}, \\ g_{ts} &= \mathbf{x}_n \cdot \mathbf{x}_{o,s} + t\, \mathbf{x}_n \cdot \mathbf{x}_{n,s}, \\ g_{tt} &= \mathbf{x}_n \cdot \mathbf{x}_n. \end{aligned} \tag{1.32}$$

By neglecting the linear t terms in the transverse shear components and invoking initial metric tensor components G_{tr} and G_{ts}, one gets

$$\begin{aligned}\gamma_{tr} &= 2\varepsilon_{tr} = g_{tr} - G_{tr} = \mathbf{x}_n \cdot \mathbf{x}_{o,r} - \mathbf{X}_n \cdot \mathbf{X}_{o,r}, \\ \gamma_{ts} &= 2\varepsilon_{ts} = g_{ts} - G_{tt} = \mathbf{x}_n \cdot \mathbf{x}_{o,s} - \mathbf{X}_n \cdot \mathbf{X}_{o,s},\end{aligned} \tag{1.33}$$

where $(\varepsilon_{tr}, \varepsilon_{ts})$ are the Green–Lagrange transverse shear strain tensor components in curvilinear coordinates and $(\gamma_{tr}, \gamma_{ts})$ are their Voigt counterparts. Similarly, the Green–Lagrange thickness strain in curvilinear coordinates is

$$\varepsilon_{tt} = \frac{1}{2}\left(g_{tt} - G_{tt}\right) = \frac{1}{2}\left(\mathbf{x}_n \cdot \mathbf{x}_n - \mathbf{X}_n \cdot \mathbf{X}_n\right). \tag{1.34}$$

It should be noted that, compared to the KL thin shell model, the shell-like solid model relaxes the KL assumption, meaning that the current director $\mathbf{x}_n$ is no longer necessarily perpendicular to the midsurface. The current midsurface position $\mathbf{x}_o$ and current director $\mathbf{x}_n$ are given by

$$\mathbf{x}_o = \mathbf{X}_o + \mathbf{U}_o, \; \mathbf{x}_n = \mathbf{X}_n + \mathbf{U}_n, \tag{1.35}$$

where $\mathbf{U}_o$ and $\mathbf{U}_n$ represent the midsurface displacement and the director displacement, respectively.

As the perpendicularity of $\mathbf{X}_n$ to $\mathbf{X}_{o,r}$ and $\mathbf{X}_{o,s}$ is assumed, the transverse shear strain and thickness strain in the local Cartesian coordinates $\tilde{X}, \tilde{Y}, \tilde{Z}$ are

$$\boldsymbol{\gamma} = \begin{Bmatrix}\gamma_{\tilde{Z}\tilde{X}} \\ \gamma_{\tilde{Z}\tilde{Y}}\end{Bmatrix} = \mathbf{T}_s \begin{Bmatrix}\gamma_{tr} \\ \gamma_{ts}\end{Bmatrix}, \quad \varepsilon_{\tilde{Z}\tilde{Z}} = T_t \varepsilon_{tt}, \tag{1.36}$$

where $\mathbf{T}_s$ and T_t are defined in the Eq.(C.6) and Eq.(C.8), respectively.

By enforcing the plane-stress condition on the shell-like solid and assuming that the bending, membrane, transverse shear, and thickness energies are uncoupled, the membrane and bending energy contributions become identical to those in the thin shell. The transverse shear

energy of the shell-like solid is

$$\Psi_s = \frac{1}{2}\int_\Omega \int_{-\frac{h}{2}}^{\frac{h}{2}} \boldsymbol{\gamma}^T \mathbf{D}_\gamma \boldsymbol{\gamma} d\tilde{Z}\, d\Omega = \frac{1}{2}\int_\Omega \boldsymbol{\gamma}^T \mathbf{D}_s \boldsymbol{\gamma}\, d\Omega, \tag{1.37}$$

and the thickness energy of the shell-like solid is

$$\Psi_t = \frac{1}{2}\int_\Omega \int_{-\frac{h}{2}}^{\frac{h}{2}} D_n \varepsilon_{\tilde{Z}\tilde{Z}}^2 d\tilde{Z}\, d\Omega = \frac{1}{2}\int_\Omega D_t \varepsilon_{\tilde{Z}\tilde{Z}}^2\, d\Omega. \tag{1.38}$$

Here, $\mathbf{D}_s = h\mathbf{D}_\gamma$ is the shear rigidity matrix, and $D_t = hD_n$ is the thickness rigidity.

$$\mathbf{D}_\gamma = \frac{5}{6}G\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ and } D_n = E \tag{1.39}$$

describe the isotropic material behavior. By recalling the membrane energy Ψ_m in Eq.(1.19) and bending energy Ψ_b in Eq.(1.20), the total energy Ψ_{ss} of the shell-like solid can be expressed as

$$\begin{aligned} \Psi_{ss} &= \Psi_m + \Psi_b + \Psi_s + \Psi_t \\ &= \frac{1}{2}\int_\Omega (\boldsymbol{\varepsilon}_m^T \mathbf{D}_m \boldsymbol{\varepsilon}_m + \boldsymbol{\varepsilon}_b^T \mathbf{D}_b \boldsymbol{\varepsilon}_b + \boldsymbol{\gamma}^T \mathbf{D}_s \boldsymbol{\gamma} + D_t \varepsilon_{\tilde{Z}\tilde{Z}}^2)\, d\Omega. \end{aligned} \tag{1.40}$$

The shell-like solid can be discretized and simulated by the solid-shell elements. The relative contributions of these four energy terms depend on the geometry and loading conditions. For cloth simulation, the material is extremely thin (thickness-to-length ratio on the order of 10^{-3} to 10^{-4}), so the Kirchhoff-Love (KL) shell assumption is often appropriate; under this assumption, transverse shear and thickness strains are kinematically constrained to zero. The dominant terms are therefore Ψ_m and Ψ_b. At typical cloth simulation scales (length $\gg$ thickness), gravity-driven draping generates significant in-plane deformation, so membrane energy dominates over bending energy. For example, in the cloth-on-sphere qualitative test of Section 3.4.2 (corotational smoothed hinge plate model, 1 m $\times$ 1 m cloth), the energy partition gives $\Psi_m \approx 93.5\%$, $\Psi_b \approx 6.5\%$. By contrast, in the geometrically nonlinear cantilever benchmark of Section 3.4.1, the bending-dominated loading yields $\Psi_b > 99.96\%$, $\Psi_m < 0.04\%$. For origami modelled with solid-shell elements, all four

energy terms in Eq. (1.40) are present in the formulation. To validate the solid-shell element itself, the geometrically nonlinear cantilever benchmark of Section 5.3.1 is also employed: the bending-dominated deformation yields $\Psi_b \approx 99.98\%$, $\Psi_m \approx 0.004\%$, $\Psi_s \approx 0.008\%$, $\Psi_t \approx 0.003\%$. In the curved-crease folding simulation of Section 5.3.2, the energy partition shifts with the fold angle, from $\Psi_b \approx 98.5\%$, $\Psi_m \approx 0.03\%$, $\Psi_s \approx 0.5\%$, $\Psi_c \approx 1.0\%$ at $W \approx -0.09\,\mathrm{m}$ to $\Psi_b \approx 80.7\%$, $\Psi_m \approx 18.6\%$, $\Psi_s \approx 0.3\%$, $\Psi_c \approx 0.4\%$ at the maximum prescribed transverse displacement ($W = -3.5\,\mathrm{m}$), reflecting the onset of non-isometric panel stretching as the crease fold angle increases.

1.3.3 Lockings and Their Remedies for Solid-shell Elements

Solid-shell elements have been extensively studied in the computational mechanics community since the late 1980s [1]. Unlike conventional finite shell elements like degenerated-shell elements, solid-shell elements do not have nodal rotational degrees of freedom. They were initially developed with the goal of enabling three-dimensional solid elements to handle plate and shell analyses. They can also avoid the complexity of updating the nodal rotations in geometric nonlinear analyses [171]. It was also shown in numerical tests that they can tolerate larger prescribed load or displacement increments [135, 88] without divergence in the nonlinear solution procedure than the shell elements with rotation dofs. Compared to the discrete differential geometry/rotation-free thin shells [66, 65, 134, 26, 103], 3D constitutive models can be directly adopted in the solid-shell elements [12, 71].

This thesis aims to utilize solid-shell elements to accurately capture the behavior of origami structures whose creases and/or panels become curved during the folding process. To achieve this, various locking issues, including membrane, shear, trapezoidal, thickness, and volumetric locking [170] should be addressed. Locking is a numerical issue where a structure becomes excessively stiff with respect to its real or physical flexibility. The numerical issue might be caused by poor geometric discretizations, unsuitable interpolation functions, excessive constraints, or

oversampling of strain fields.

Shear and membrane locking are common issues in both solid-shell and degenerated-shell elements, causing more rigid behavier when the shell is thinner. Shear locking is caused by the coupling between the relative in-plane translation of the top and bottom shell surfaces and the transverse translation of the midsurface in the transverse shear strain. Membrane locking is caused by the coupling between the relative in-plane translation of the top and bottom surfaces and the in-plane translation of the midsurface in the membrane strain. These locking effects are linked to parasitic shear and membrane strains. The parasitic strains do not contribute to actual deformation, yet they still affect the element stiffness. For example, in bending-dominated problems, these parasitic strains may introduce artificial stiffness to the bending deformation. Possible remedies including the penalty scaling method [172], reduced integration [220], hybrid-stress [143, 177], hybrid-strain [7], the assumed natural strain (ANS) method [118] and the enhanced assumed strain (EAS) method [161]. The penalty scaling method weakly enforces certain constraints by introducing a scaled penalty term, helping to avoid excessive stiffness. Reduced integration lowers the number of integration points in order to reduce over-constraint and alleviate locking. The hybrid-stress method assumes independent stress fields in the variational formulation, improving flexibility and reducing locking. The hybrid-strain method, on the other hand, assumes independent strain fields to enhance strain approximation. The ANS method alleviates shear and membrane locking by sampling strains at locations where they are more accurately represented and interpolating them back to the element domain. Lastly, the EAS method enriches the strain field with additional assumed modes independent of the displacement field, thereby enhancing the element's ability to represent deformation and effectively alleviating locking.

Trapezoidal locking occurs when low-order solid shell elements are applied to simulate curved shell structures; its name derives from the trapezoidal shape of the element cross-section (see Figure 1.5). The thickness strain is the parasitic strain responsible for this locking, and it is typically corrected using the ANS method by interpolating the thickness

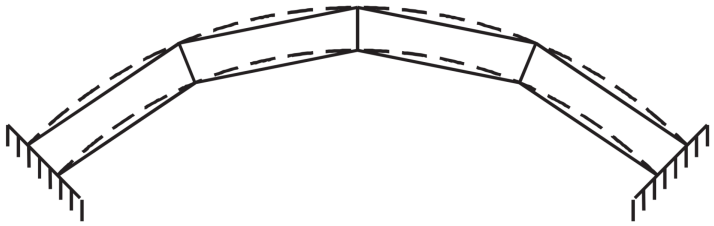


Figure 1.5: Trapezoidal cross-sections are commonly observed in low-order solid-shell elements (e.g., bilinear quadrilaterals and linear triangles) used for curved shell modeling. Adapted from [170].

strain sample at nodes [175].

Thickness locking occurs in elements in which the thickness strain does not vary along the thickness direction, and is caused by the coupling between the thickness stress and in-plane strains, induced by Poisson's ratio in the constitutive matrix. When an element is subjected to pure bending, it incorrectly predicts a plane strain state instead of the actual plane stress state. This locking defect can be resolved when Poisson's ratio is assumed to be zero. Other successful mitigation methods include the plane stress enforcement [135, 88], hybrid-stress [175], and the EAS method [70].

Volumetric locking (also known as dilatational locking or incompressibility locking) is caused by the near-incompressibility of the material, such as in rubber-like materials and metals undergoing plastic flow. For incompressible, homogeneous, isotropic materials (where Poisson's ratio equals 0.5), the condition of zero dilatational strain is numerically enforced by using a large material bulk modulus. The methods to mitigate dilatational locking include reduced integration [40], mixed/hybrid formulations [142], and the EAS method [161]. Furthermore, when thickness locking is eliminated by enforcing a plane stress condition, volumetric locking does not exist as the enforced condition allows free expansion/contraction in the thickness direction, thereby always accommodating the material's incompressibility.

For shell-like solids under flexural loadings, the plane-stress condition occurs naturally, which helps to alleviate thickness and volumetric locking. In the thesis, both bilinear and biquadratic interpolation

schemes are adopted in solid-shell elements. The ANS method is employed to alleviate membrane, shear, and trapezoidal lockings. ANS mitigates excessive constraints by sampling and interpolating the potentially parasitic strain components at judiciously selected points and, mostly, boundary points. In bilinear solid-shell elements which are preliminary flat, membrane locking is rarely encountered and thus can be ignored.

1.4 The Objective and Organization of the Thesis

This thesis is motivated by supporting flexible choices across different application scenarios in cloth and origami simulations, with contributions centered on the development of a series of computational models. Each model is evaluated through both quantitative and qualitative tests. To clarify these contributions, the thesis is organized into six chapters:

Chapter 1 introduces the background and objectives of the thesis. It reviews existing works on cloth and origami simulations, with a particular emphasis on computational models, and outlines the scope and structure of the thesis.

Chapter 2 considers the hinge model based on the edge stencil. A hinge-bending model for cloth simulation is devised by fitting a cylindrical surface to an edge stencil and the discrete curvature of the surface is expressed in terms of the hinge angle which is obtained from the altitude vectors of two adjacent triangle facets. The gradient and Hessian of the hinge angle are derived analytically and expressed in concise forms. A hinge-folding model is also devised for the folding and unfolding simulation of straight- and curved-crease origami structures. The elastic energy involved in folding and unfolding the hinge is stored in the hinge. The efficacy of the models are illustrated by applying them to commonly studied examples.

Chapter 3 presents six thin plate/shell models, derived from three distinct types of curvature operators formulated using the corotational

frame, for simulating both initially flat and curved triangular meshes. Each curvature operator corresponds to the discrete curvature in both the plate and shell models. The corotational edge-based hinge model uses an edge stencil to compute directional curvature. On the other hand, the corotational FVM hinge model utilizes a triangle-centered stencil and applies the FVM to superposition directional curvatures across edges of the central triangle to yield the generalized curvature. The corotational smoothed hinge model also employs a triangle-centered stencil but transforms directional curvatures into a generalized curvature based on a quadratic surface fit. All models assume small strain and small curvature, leading to constant bending energy Hessians, which benefit implicit integrators. Through quantitative benchmarks and qualitative elastodynamic simulations with large time steps, the accuracy, efficiency, and stability of these models are demonstrated. These models give the potential for fast cloth simulation and beyond. This chapter is primarily based on the work reported in "Qixin Liang, Corotational hinge-based thin plates/shells, Computer Graphics Forum (Proc. Eurographics 2025), May 2025".

Chapter 4 presents a smoothed hinge membrane model for cloth simulation. This model founds on a triangle-centered stencil with quadratic interpolated positions and samples membrane strains at the midpoints of the central triangle. This model mitigates sharp creasing artifacts commonly observed in cloth simulations using traditional CST-based membrane models. Both quantitative and qualitative results demonstrate its effectiveness. This model enriches the family of computational models for realistic cloth simulation. This chapter is primarily based on the work reported in "Qixin Liang, Smoothed-hinge model for cloth simulation, ACM SIGGRAPH/Eurographics Symposium on Computer Animation, August 2024". Additional experiments and discussions have been included to extend the original publication.

Chapter 5 presents a novel computational model for modeling and simulating origami structures. In this framework, bilinear and biquadratic solid-shell elements are employed to model the origami panels while

crease folding is considered through the angle between the director vectors of the adjacent panels. The director vector is the vector normal to the midsurface before displacement/deformation. The ANS method is employed to mitigate locking issues in the solid-shell element. The computational origami simulations featuring both straight- and curved-creases are conducted to validate the effectiveness of this novel model. The accuracy and efficacy of the model are demonstrated through quantitative and qualitative analyses. This approach provides an effective framework for the design and analysis of origami structures. This chapter is primarily based on the work reported in "Qixin Liang, Modeling and simulating origami structures using bilinear solid-shell element, Proc. SIGGRAPH Asia 2025 Technical Communications, December 2025". Additional experiments and discussions have been included to extend the original publication.

Chapter 6 summarizes the thesis and provides recommendations for future research directions.

Chapter 2

Hinge Models for Cloth and Origami Simulations

2.1 Introduction

This chapter discusses hinge models based on edge stencils extracted from piecewise linear triangular meshes. Each edge stencil consists of two adjacent triangular facets sharing a common edge. The simplicity of hinge models has led to their widespread use across various domains, including early applications in cloth simulation [2, 19, 66, 193], and later in engineering contexts such as origami simulation [107, 108, 157]. These models are increasingly adopted for simulating various thin flexible structures [75]. Extensive research has been conducted to enhance their accuracy, efficiency, and robustness for different application scenarios. This chapter presents the following developments:

- **Hinge-bending model:** The directional curvature in the hinge bending model for cloth simulation is computed by fitting an edge stencil to a cylindrical surface. Moreover, a concise analytical formulation for the gradient and Hessian of the hinge angle is derived using the altitude vectors of the triangular facets. This approach avoids the complexities associated with normal vector to the triangular facet.
- **Hinge-folding model:** Without fitting the cylindrical surface, the hinge along the common edge of two (flat) facets is the natural

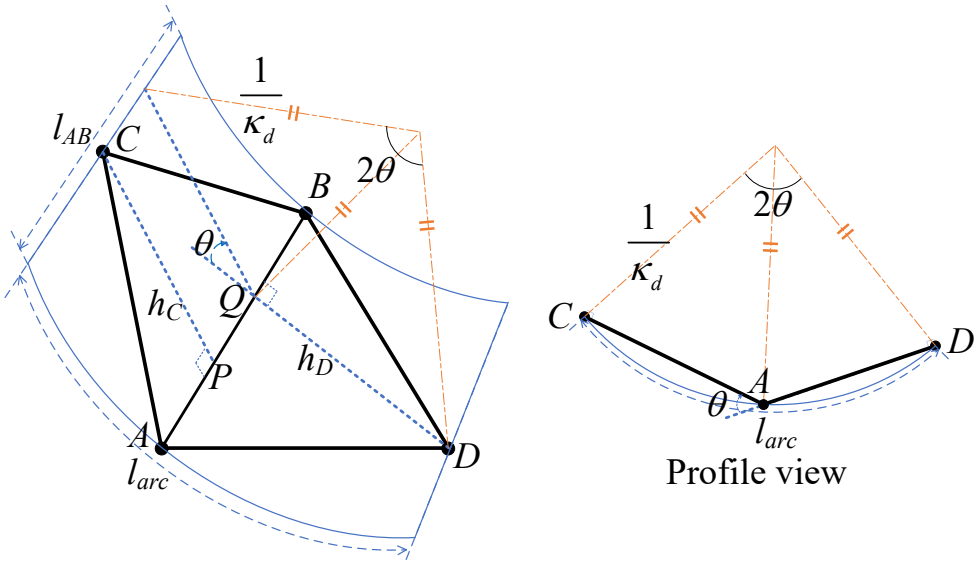


Figure 2.1: A schematic diagram of an edge stencil fitted onto a cylindrical surface. The left-hand figure shows an edge stencil formed by two adjacent triangles, T_{ABC} and T_{BAD}, with their common edge AB aligned with the generatrix of a cylindrical surface. The directional curvature orthogonal to the common edge of the surface is κ_d, l_{AB} and l_{arc} are the lengths of the generatrix and the circular arc, respectively; h_C and h_D are triangle heights. The angle subtended by the arc at the axis of the cylindrical surface is 2θ, whereas the hinge angle is θ. The right-hand figure shows the view along the direction of edge AB. In the derivation of the bending model, the hinge angle is assumed to be small to ensure the validity of geometric approximations.

choice for the folding and unfolding simulation of origami structures. The afore-derived gradient and Hessian of the hinge angle does not assume the isometry of the facets. They improve the convergence of the finite element based simulation presented in [72].

- The accuracy of the hinge-bending model with respect to other models in the open literature will be demonstrated through a linear plate bending and geometrically nonlinear cantilever tests. The hinge-bending model also improves the accuracy of the widely used origami simulation tool, MERLIN2 [107, 207], which models bending deformation with the panel using a hinge-based approach.

2.2 Hinge Model

2.2.1 Hinge-bending Model

The hinge-bending model is built upon an edge stencil comprising two triangular facets with a common edge. In this stencil, both triangular facets are flat, and the common edge is straight. Consequently, there is no curvature along the direction of the common edge and but only in the direction orthogonal to it. Accordingly, this discrete geometric structure (edge stencil) can serve as an approximation of a smooth geometry (a cylindrical surface). The cylindrical surface is featured by a circular arc of radius $1/\kappa_d$ and arc length l_{arc}, as well as a generatrix of length l_{AB} (see Figure 2.1). The cylindrical surface has zero curvature along the generatrix direction and a nonzero directional curvature κ_d in the orthogonal direction. In hinge-bending models, the values 0 and κ_d are taken to be the principal curvatures of the curvature tensor. Assuming the hinge angle is small, the directional curvature κ_d of the cylindrical surface can be approximated from the geometric relation:

$$l_{arc} l_{AB} = 2\theta \cdot \frac{1}{\kappa_d} \cdot l_{AB} \simeq (h_C + h_D) l_{AB}, \tag{2.1}$$

which leads to

$$\kappa_d \simeq \frac{2\theta}{h_C + h_D}. \tag{2.2}$$

Here, θ is the hinge angle in the current configuration. Under the small strain assumption (quasi-isometric assumption), the edge lengths and heights remain nearly unchanged between initial and current configurations. The heights in the initial configuration can be computed as $h_C = \|\mathbf{X}_{PC}\|$ and $h_D = \|\mathbf{X}_{QD}\|$, where $\mathbf{X}$ denoted the initial position vector, corresponding to the heights of triangles T_{ABC} and T_{BAD} (see Figure 2.1), respectively.

The bending energy associated with an edge stencil, expressed in terms of the directional (principal) curvatures, is

$$\Psi_b^s = \frac{1}{2} A_{\mathcal{E}} k_b (\varepsilon_b^H)^2, \tag{2.3}$$

where $k_b = Eh^3/(12(1-\nu^2))$ is the bending rigidity. The curvature change

$$\varepsilon_b^H = \kappa_d - K_d \tag{2.4}$$

is the difference of directional curvature in the current and initial configurations. Superscript "H" denotes the hinge-bending model. K_d is the counterpart of κ_d, i.e.

$$K_d \simeq \frac{2\Theta}{h_C + h_D}, \tag{2.5}$$

where Θ is the hinge angle in the initial configuration. The integration area of the edge stencil is given by

$$A_{\mathcal{E}} = \frac{(h_C + h_D) l_{AB}}{2}, \tag{2.6}$$

where $l_{AB} = \|\mathbf{X}_{AB}\|$. The integration area approximates the area of the smooth geometry enclosed by the geodesics AC, AD, BC, and BD on the cylindrical surface. Substituting Eqs.(2.2) and (2.4)-(2.6) into Eq.(2.3) with the expression of integration area, the bending energy becomes

$$\Psi_b^s = k_b \frac{l_{AB}}{h_C + h_D} (\theta - \Theta)^2. \tag{2.7}$$

For a general shell mid-surface discretized into a triangular mesh, the total bending energy Ψ_b can be obtained by summing the bending energy contributions from the interior edge stencils, which are counted by all edges excluding boundary edges, i.e.

$$\Psi_b = \sum \Psi_b^s. \tag{2.8}$$

2.2.2 Hinge-folding Model

In simulating the folding and unfolding of origami structures, the practice of fitting a curved surface over each edge stencil is not necessary. The folding and unfolding are often considered by the elastic energy of the crease, i.e.

$$\Psi_f^s = \frac{1}{2} k_f l_{AB} (\theta - \Theta)^2, \tag{2.9}$$

where k_f is the folding stiffness per unit length of the crease.

2.2.3 Gradient and Hessian of the Bending/Folding Energy

The bending energy and folding energy designated by respectively subscripts "b" and "f" in Eq.(2.7) and Eq.(2.9) can be expressed in the following format:

$$\Psi_J^s = \frac{1}{2} D_J (\theta - \Theta)^2, \tag{2.10}$$

where $D_b = 2k_b l_{AB}/(h_C + h_D)$ and $D_f = k_f l_{AB}$.

The gradient and Hessian of the bending/folding energy are essential for implicit simulation. The gradient of the energy with respect to the current nodal positions is

$$\frac{\partial \Psi_J^s}{\partial \mathbf{x}^s} = D_J (\theta - \Theta) \frac{\partial \theta}{\partial \mathbf{x}^s}, \tag{2.11}$$

and the corresponding Hessian takes the form

$$\frac{\partial^2 \Psi_J^s}{\partial \mathbf{x}^s \partial (\mathbf{x}^s)^T} = D_J \frac{\partial \theta}{\partial \mathbf{x}^s} \frac{\partial \theta}{\partial (\mathbf{x}^s)^T} + D_J (\theta - \Theta) \frac{\partial^2 \theta}{\partial \mathbf{x}^s \partial (\mathbf{x}^s)^T}. \tag{2.12}$$

Here, $\mathbf{x}^s = \begin{bmatrix} \mathbf{x}_A^T & \mathbf{x}_B^T & \mathbf{x}_C^T & \mathbf{x}_D^T \end{bmatrix}^T$ collects the current position vectors of nodes A, B, C, and D.

Hence, one needs to derive the gradient and Hessian of the hinge angle θ for those of Ψ_J^s.

2.2.4 Gradient and Hessian of the Hinge Angle

Before deriving the gradient and Hessian of the hinge angle θ, the angle can be first computed from

$$pq \cos\theta = -\mathbf{p} \cdot \mathbf{q}, \tag{2.13}$$

where $p = \|\mathbf{p}\|$ and $q = \|\mathbf{q}\|$. Moreover, $\mathbf{p} = \mathbf{x}_{PC}$ and $\mathbf{q} = \mathbf{x}_{QD}$ (see Figure 2.2) are respectively the altitude vectors of T_{ABC} and T_{BAD}, and

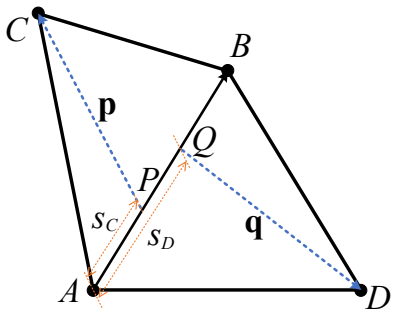


Figure 2.2: Schematic illustration of $\mathbf{p}$, $\mathbf{q}$, s_C, and s_D.

they can be expressed as

$$\mathbf{p} = \mathbf{x}_{AC} - s_C\mathbf{x}_{AB},\ \mathbf{q} = \mathbf{x}_{AD} - s_D\mathbf{x}_{AB}. \tag{2.14}$$

Here the non-dimensional coordinates s_C and s_D (see Figure 2.2) along the unit vector $\mathbf{s} = \mathbf{x}_{AB}/\|\mathbf{x}_{AB}\|$ are

$$s_C = \frac{\mathbf{x}_{AC}\cdot\mathbf{x}_{AB}}{\|\mathbf{x}_{AB}\|^2}, s_D = \frac{\mathbf{x}_{AD}\cdot\mathbf{x}_{AB}}{\|\mathbf{x}_{AB}\|^2}. \tag{2.15}$$

They are the ratios of the projections of edges $\mathbf{x}_{AC}$ and $\mathbf{x}_{AD}$ onto edge $\mathbf{x}_{AB}$, relative to the length of $\mathbf{x}_{AB}$.

The following standard relations for any vectors $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$, which will be used, are listed

$$\frac{\partial}{\partial\mathbf{c}}(\mathbf{a}\cdot\mathbf{b}) = \frac{\partial\mathbf{a}^T}{\partial\mathbf{c}}\mathbf{b} + \frac{\partial\mathbf{b}^T}{\partial\mathbf{c}}\mathbf{a} \tag{2.14a}$$

$$\frac{\partial|\mathbf{a}|}{\partial\mathbf{c}} = \frac{\partial\mathbf{a}^T}{\partial\mathbf{c}}\frac{\mathbf{a}}{|\mathbf{a}|} \tag{2.14b}$$

$$\mathbf{a}\times\mathbf{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}\mathbf{b} = [\mathbf{a}]_\times\mathbf{b} \tag{2.14c}$$

$$\frac{\partial}{\partial\mathbf{c}^T}(\mathbf{a}\times\mathbf{b}) = [\mathbf{a}]_\times\frac{\partial\mathbf{b}}{\partial\mathbf{c}^T} - [\mathbf{b}]_\times\frac{\partial\mathbf{a}}{\partial\mathbf{c}^T} \tag{2.14d}$$

$$\mathbf{a}\times(\mathbf{b}\times\mathbf{c}) = (\mathbf{a}\cdot\mathbf{c})\mathbf{b} - (\mathbf{a}\cdot\mathbf{b})\mathbf{c} \tag{2.14e}$$

$$\mathbf{a}(\mathbf{a}\times\mathbf{b})^T - (\mathbf{a}\times\mathbf{b})\mathbf{a}^T = (\mathbf{a}\cdot\mathbf{a})[\mathbf{b}]_\times - (\mathbf{a}\cdot\mathbf{b})[\mathbf{a}]_\times, \tag{2.14f}$$

where the skew symmetric matrix $[\mathbf{a}]_\times$ for vector $\mathbf{a}$ is self-defined.

By differentiating Eq.(2.13) with respect to $\mathbf{x}^e$ and applying Eq.(2.14b), the gradient of the fold angle is computed as

$$\frac{\partial \theta}{\partial \mathbf{x}^s} = -\frac{\partial \mathbf{p}^T}{\partial \mathbf{x}^s}\frac{(\mathbf{p}\cdot\mathbf{q})\mathbf{p} - p^2\mathbf{q}}{p^3 q \sin\theta} - \frac{\partial \mathbf{q}^T}{\partial \mathbf{x}^s}\frac{(\mathbf{p}\cdot\mathbf{q})\mathbf{q} - q^2\mathbf{p}}{pq^3 \sin\theta}. \tag{2.17}$$

Here, $\partial \mathbf{p}^T/\partial \mathbf{x}^s$ and $\partial \mathbf{q}^T/\partial \mathbf{x}^s$ are the gradients of the triangle altitude vectors $\mathbf{p}$ and $\mathbf{q}$, respectively. To compute these gradients, it is necessary to invoke the gradients of the non-dimensional variables s_C and s_D

$$\begin{gathered}
\frac{\partial s_C}{\partial \mathbf{x}_A} = -\frac{\mathbf{p} + (1-s_C)\,\mathbf{x}_{AB}}{\mathbf{x}_{AB}^2}, \frac{\partial s_C}{\partial \mathbf{x}_B} = \frac{\mathbf{p} - s_C\mathbf{x}_{AB}}{\mathbf{x}_{AB}^2}, \\
\frac{\partial s_C}{\partial \mathbf{x}_C} = \frac{\mathbf{x}_{AB}}{\mathbf{x}_{AB}^2}, \frac{\partial s_C}{\partial \mathbf{x}_D} = \mathbf{0}, \\
\frac{\partial s_D}{\partial \mathbf{x}_A} = -\frac{\mathbf{q} + (1-s_D)\,\mathbf{x}_{AB}}{\mathbf{x}_{AB}^2}, \frac{\partial s_D}{\partial \mathbf{x}_B} = \frac{\mathbf{q} - s_D\mathbf{x}_{AB}}{\mathbf{x}_{AB}^2}, \\
\frac{\partial s_D}{\partial \mathbf{x}_C} = \mathbf{0}, \frac{\partial s_D}{\partial \mathbf{x}_D} = \frac{\mathbf{x}_{AB}}{\mathbf{x}_{AB}^2},
\end{gathered} \tag{2.18}$$

and the gradient of the unit vector $\mathbf{s}$

$$\frac{\partial \mathbf{s}}{\partial \mathbf{x}_A^T} = \frac{-1}{\|\mathbf{x}_{AB}\|}\left(\mathbf{I} - \mathbf{s}\mathbf{s}^T\right), \frac{\partial \mathbf{s}}{\partial \mathbf{x}_B^T} = \frac{1}{\|\mathbf{x}_{AB}\|}\left(\mathbf{I} - \mathbf{s}\mathbf{s}^T\right), \frac{\partial \mathbf{s}}{\partial \mathbf{x}_C^T} = \mathbf{0}, \frac{\partial \mathbf{s}}{\partial \mathbf{x}_D^T} = \mathbf{0} \tag{2.19}$$

in which $\mathbf{I}$ is the third order identity matrix and $\mathbf{0}$ is either a zero matrix or vector. Consequently, the components of the gradients of the triangle height vectors $\mathbf{p}$ and $\mathbf{q}$ can be expressed as

$$\begin{gathered}
\frac{\partial \mathbf{p}^T}{\partial \mathbf{x}_A} = -(1-s_C)\left(\mathbf{I} - \mathbf{s}\mathbf{s}^T\right) + \frac{\mathbf{p}\mathbf{s}^T}{\|\mathbf{x}_{AB}\|}, \frac{\partial \mathbf{p}^T}{\partial \mathbf{x}_B} = -s_C\left(\mathbf{I} - \mathbf{s}\mathbf{s}^T\right) - \frac{\mathbf{p}\mathbf{s}^T}{\|\mathbf{x}_{AB}\|}, \\
\frac{\partial \mathbf{p}^T}{\partial \mathbf{x}_C} = \mathbf{I} - \mathbf{s}\mathbf{s}^T, \frac{\partial \mathbf{p}^T}{\partial \mathbf{x}_D} = \mathbf{0},
\end{gathered} \tag{2.20}$$

and

$$\frac{\partial \mathbf{q}^T}{\partial \mathbf{x}_A} = -(1-s_D)\left(\mathbf{I}-\mathbf{s}\mathbf{s}^T\right) + \frac{\mathbf{q}\mathbf{s}^T}{\|\mathbf{x}_{AB}\|}, \frac{\partial \mathbf{q}^T}{\partial \mathbf{x}_B} = -s_D\left(\mathbf{I}-\mathbf{s}\mathbf{s}^T\right) - \frac{\mathbf{q}\mathbf{s}^T}{\|\mathbf{x}_{AB}\|},$$
$$\frac{\partial \mathbf{q}^T}{\partial \mathbf{x}_C} = \mathbf{0}, \frac{\partial \mathbf{q}^T}{\partial \mathbf{x}_D} = \mathbf{I}-\mathbf{s}\mathbf{s}^T. \tag{2.21}$$

By invoking Eq.(2.14e) and the geometric relation

$$\mathbf{p}\times\mathbf{q} = (pqsin\theta)\mathbf{s}, \tag{2.22}$$

Eq.(2.17) becomes

$$\frac{\partial \theta}{\partial \mathbf{x}^s} = \frac{\partial \mathbf{p}^T}{\partial \mathbf{x}^s}\frac{\mathbf{s}\times\mathbf{p}}{p^2} + \frac{\partial \mathbf{q}^T}{\partial \mathbf{x}^s}\frac{\mathbf{q}\times\mathbf{s}}{q^2}. \tag{2.23}$$

From the above relationships, along with Eqs.(2.20) and (2.21), the relevant components of the gradient of the fold angle can be computed as

$$\frac{\partial \theta}{\partial \mathbf{x}_C} = \frac{\mathbf{s}\times\mathbf{p}}{p^2}, \frac{\partial \theta}{\partial \mathbf{x}_D} = \frac{\mathbf{q}\times\mathbf{s}}{q^2}$$
$$\frac{\partial \theta}{\partial \mathbf{x}_A} = -(1-s_C)\frac{\partial \theta}{\partial \mathbf{x}_C} - (1-s_D)\frac{\partial \theta}{\partial \mathbf{x}_D}, \tag{2.22a-d}$$
$$\frac{\partial \theta}{\partial \mathbf{x}_B} = -s_C\frac{\partial \theta}{\partial \mathbf{x}_C} - s_D\frac{\partial \theta}{\partial \mathbf{x}_D}.$$

Before deriving the Hessian of the fold angle, the notation $\boldsymbol{\theta}_I = \partial\theta/\partial\mathbf{x}_I$ and $\boldsymbol{\theta}_{IJ} = \partial\boldsymbol{\theta}_I^T/\partial\mathbf{x}_J$ is introduced. Furthermore, the following expressions

$$\mathbf{p}\boldsymbol{\theta}_C^T - \boldsymbol{\theta}_C\mathbf{p}^T = [\mathbf{s}]_\times,\ -p^2\mathbf{s}\boldsymbol{\theta}_C^T + p^2\boldsymbol{\theta}_C\mathbf{s}^T = [\mathbf{p}]_\times,$$
$$\mathbf{q}\boldsymbol{\theta}_D^T - \boldsymbol{\theta}_D\mathbf{q}^T = [\mathbf{s}]_\times,\ -q^2\mathbf{s}\boldsymbol{\theta}_D^T + q^2\boldsymbol{\theta}_D\mathbf{s}^T = [\mathbf{q}]_\times. \tag{2.23a-d}$$

are obtained by taking $(\mathbf{a},\mathbf{b})$ in Eq.(2.14f) to be $(\mathbf{p},\mathbf{s})$, $(\mathbf{s},\mathbf{p})$, $(\mathbf{q},\mathbf{s})$ and $(\mathbf{s},\mathbf{q})$, and then simplifying them with orthogonality relations

$$\mathbf{s}\perp\boldsymbol{\theta}_C,\ \boldsymbol{\theta}_C\perp\mathbf{p},\ \mathbf{s}\perp\boldsymbol{\theta}_D,\ \boldsymbol{\theta}_D\perp\mathbf{q}. \tag{2.24}$$

By invoking Eq.(2.14d) and Eq.(2.24), the derivatives of Eq.(2.22a, b) with respect to $\mathbf{x}_J$ are

$$\begin{aligned} p^2\boldsymbol{\theta}_{CJ} &= [\mathbf{p}]_\times \frac{\partial \mathbf{s}}{\partial \mathbf{x}_J^T} - \left([\mathbf{s}]_\times + 2\boldsymbol{\theta}_C \mathbf{p}^T\right) \frac{\partial \mathbf{p}}{\partial \mathbf{x}_J^T} \\ &= p^2 \left(\boldsymbol{\theta}_C \mathbf{s}^T - \mathbf{s}\boldsymbol{\theta}_C^T\right) \frac{\partial \mathbf{s}}{\partial \mathbf{x}_J^T} - \left(\boldsymbol{\theta}_C \mathbf{p}^T + \mathbf{p}\boldsymbol{\theta}_C^T\right) \frac{\partial \mathbf{p}}{\partial \mathbf{x}_J^T}, \\ q^2\boldsymbol{\theta}_{DJ} &= [\mathbf{q}]_\times \frac{\partial \mathbf{s}}{\partial \mathbf{x}_J^T} - \left([\mathbf{s}]_\times + 2\boldsymbol{\theta}_D \mathbf{q}^T\right) \frac{\partial \mathbf{q}}{\partial \mathbf{x}_J^T} \\ &= q^2 \left(\boldsymbol{\theta}_D \mathbf{s}^T - \mathbf{s}\boldsymbol{\theta}_D^T\right) \frac{\partial \mathbf{s}}{\partial \mathbf{x}_J^T} - \left(\boldsymbol{\theta}_D \mathbf{q}^T + \mathbf{q}\boldsymbol{\theta}_D^T\right) \frac{\partial \mathbf{q}}{\partial \mathbf{x}_J^T}. \end{aligned} \tag{2.25}$$

For different J, the following results can be obtained by recalling Eq.(2.19-2.21) and (2.24):

$$\begin{aligned} \boldsymbol{\theta}_{CC} &= -\frac{1}{p^2}\mathbf{M}_{Cp},\ \boldsymbol{\theta}_{CA} = \frac{1-s_C}{p^2}\mathbf{M}_{Cp} + \frac{1}{|\mathbf{x}_{AB}|}\mathbf{s}\boldsymbol{\theta}_C^T, \\ \boldsymbol{\theta}_{CB} &= \frac{s_C}{p^2}\mathbf{M}_{Cp} - \frac{1}{|\mathbf{x}_{AB}|}\mathbf{s}\boldsymbol{\theta}_C^T,\ \boldsymbol{\theta}_{CD} = \mathbf{0},\ \boldsymbol{\theta}_{DD} = \frac{-1}{q^2}\mathbf{M}_{Dq}, \\ \boldsymbol{\theta}_{DA} &= \frac{1-s_D}{q^2}\mathbf{M}_{Dq} + \frac{1}{|\mathbf{x}_{AB}|}\mathbf{s}\boldsymbol{\theta}_D^T,\ \boldsymbol{\theta}_{DB} = \frac{s_D}{q^2}\mathbf{M}_{Dq} - \frac{1}{|\mathbf{x}_{AB}|}\mathbf{s}\boldsymbol{\theta}_D^T, \end{aligned} \tag{2.26a-g}$$

where

$$\mathbf{M}_{Ja} = \boldsymbol{\theta}_J \mathbf{a}^T + \mathbf{a}\left(\boldsymbol{\theta}_J\right)^T \tag{2.27}$$

for J = C, D and a = p, q, s. Further to Eq.(2.25), derivatives of Eq.(2.22c, d) with respect to $\mathbf{x}_J$ are

$$\begin{aligned} \boldsymbol{\theta}_{AJ} &= -\left(1-s_C\right)\boldsymbol{\theta}_{CJ} - \left(1-s_D\right)\boldsymbol{\theta}_{DJ} + \boldsymbol{\theta}_C \frac{\partial s_C}{\partial \mathbf{x}_J^T} + \boldsymbol{\theta}_D \frac{\partial s_D}{\partial \mathbf{x}_J^T}, \\ \boldsymbol{\theta}_{BJ} &= -s_C\boldsymbol{\theta}_{CJ} - s_D\boldsymbol{\theta}_{DJ} - \boldsymbol{\theta}_C \frac{\partial s_C}{\partial \mathbf{x}_J^T} - \boldsymbol{\theta}_D \frac{\partial s_D}{\partial \mathbf{x}_J^T}. \end{aligned} \tag{2.28}$$

For different J, the following results can be obtained by invoking Eq.(2.18) and Eq.(2.23a-d):

$$\begin{aligned}
\boldsymbol{\theta}_{AA} &= -\frac{(1-s_C)^2}{p^2}\mathbf{M}_{Cp} - \frac{(1-s_D)^2}{q^2}\mathbf{M}_{Dq} - \frac{1-s_C}{\|\mathbf{x}_{AB}\|}\mathbf{M}_{Cs} \\
&\quad -\frac{1-s_D}{\|\mathbf{x}_{AB}\|}\mathbf{M}_{Ds} - \frac{1}{\|\mathbf{x}_{AB}\|^2}\left(\boldsymbol{\theta}_C\mathbf{p}^T + \boldsymbol{\theta}_D\mathbf{q}^T\right), \\
\boldsymbol{\theta}_{BB} &= -\frac{s_C^2}{p^2}\mathbf{M}_{Cp} - \frac{s_D^2}{q^2}\mathbf{M}_{Dq} + \frac{s_C}{\|\mathbf{x}_{AB}\|}\mathbf{M}_{Cs} + \frac{s_D}{\|\mathbf{x}_{AB}\|}\mathbf{M}_{Ds} \\
&\quad -\frac{1}{\|\mathbf{x}_{AB}\|^2}\left(\boldsymbol{\theta}_C\mathbf{p}^T + \boldsymbol{\theta}_D\mathbf{q}^T\right) \\
\boldsymbol{\theta}_{AB} &= -\frac{s_C(1-s_C)}{p^2}\mathbf{M}_{Cp} - \frac{s_D(1-s_D)}{q^2}\mathbf{M}_{Dq} + \frac{1}{\|\mathbf{x}_{AB}\|}\mathbf{s}\left(\boldsymbol{\theta}_C^T + \boldsymbol{\theta}_D^T\right) \\
&\quad -\frac{s_C}{\|\mathbf{x}_{AB}\|}\mathbf{M}_{Cs} - \frac{s_D}{\|\mathbf{x}_{AB}\|}\mathbf{M}_{Ds} + \frac{1}{\|\mathbf{x}_{AB}\|^2}\left(\boldsymbol{\theta}_C\mathbf{p}^T + \boldsymbol{\theta}_D\mathbf{q}^T\right).
\end{aligned} \tag{2.29a-c}$$

There are ten $\boldsymbol{\theta}_{IJ}$s in Eq.(2.26a-g) and (2.29a-c). The remaining six $\boldsymbol{\theta}_{IJ}$s can be obtained by symmetry, i.e. $\boldsymbol{\theta}_{JI} = \boldsymbol{\theta}_{IJ}^T$. By taking the difference Eq.(2.23a) and (2.23c), the term $(\boldsymbol{\theta}_C\mathbf{p}^T - \boldsymbol{\theta}_D\mathbf{q}^T)$ in $\boldsymbol{\theta}_{AA}$ and $\boldsymbol{\theta}_{BB}$ is symmetric. Thus, the symmetry of $\boldsymbol{\theta}_{BB}$ and $\boldsymbol{\theta}_{CC}$ follow.

More derivation details can be found in Appendix A.

2.3 Numerical Tests

This section starts with four quantitative tests evaluating the accuracy and efficiency of the present formulation, followed by two qualitative tests demonstrating its robustness. All quantities are expressed in SI.

2.3.1 Quantitative Tests

Linear plate bending benchmark test

Here, a linear small-deflection plate bending problem [103] is employed to examine the convergence behaviour of the hinge-bending model under mesh refinement (see Figure 2.3). A square plate with edge length $a = 8$ is simply-supported along its entire boundary and subjected to a

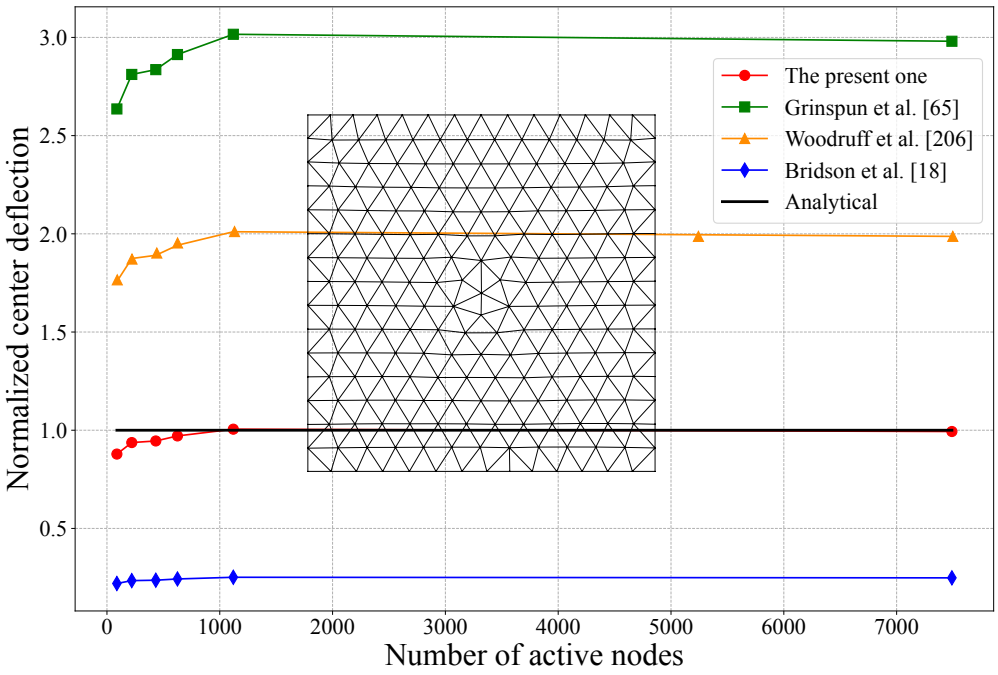


Figure 2.3: A comparative study on the convergence behavior of a simply-supported square plate under a small uniform transverse load. The vertical axis shows the computed maximum deflection normalized by the linear analytical solution. As the number of nodes increases, the present hinge-bending model converges to the analytical solution, while other models deviate noticeably.

uniform transverse force of magnitude $B = 9.81$ per unit area. The problem parameters include $E = 2 \times 10^{11}$, $\nu = 0.3$, and $h = 0.01$. The analytical solution for the maximum deflection is given by $0.048744Ba^4(1 - \nu^2)/(Eh^3)$ [188]. Figure 2.3 shows that the hinge-bending model converges to the analytical solution as the number of nodes increases.

It is worth noting that the hinge-bending energy of Grinspun et al. [66] is three times higher than in the present formulation. In contrast, the bending model proposed by Bridson et al. [19] yields an energy that is one quarter of the present one. Their model uses $\sin\theta$, which is close to θ in the small-curvature regime to evaluate the bending energy. On the other hand, the bending energy in the model of Woodruff et al. [207] is twice that of the present one. The comparison of different hinge-bending energy formulations with the present one can be found in Appendix G.

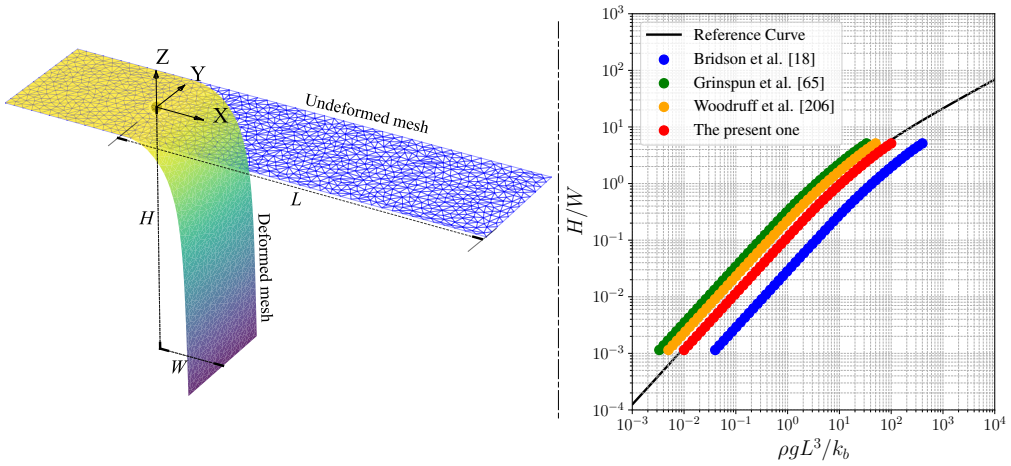


Figure 2.4: Geometric nonlinear cantilever test. The left-hand side figure shows a cantilever strip clamped at one end. The initially flat strip deforms under its own weight into an equilibrium configuration with an aspect ratio H/W. In this subfigure, the bending rigidity is $k_b = 7.1442 \times 10^{-2}$, corresponding to the rightmost data point in the right-hand figure. The horizontal axis of the right-hand figure represents the dimensionless gravito-bending parameter, $\rho g L^3/k_b$. A total of 50 fabrics with identical geometry and mesh but varying bending rigidity are considered. From left to right, the bending rigidity decreases. The reference curve is extracted from Romero et al. [152].

Geometric nonlinear cantilever test

In this test, a one-end clamped strip of width 0.03, overall length 0.12 and free-hanging length (L) 0.09 is considered. The mesh employed by Feng et al. [48] with 800 active nodes, $X > 0$, and 287 inactive/clamped nodes, $X \leq 0$, are adopted. Following [48], the strip area density (mass per unit area) ρ and the acceleration due to gravity g are taken to be 0.1 and 9.8, respectively. On the other hand, the bending stiffness k_b ranges from 7.1442×10^{-6} to 7.1442×10^{-2}. The reference solution is obtained by solving the equilibrium equation of the geometric nonlinear cantilever beam numerically [152]. The relation between the aspect ratio H/W of the deformed strip at equilibrium and the dimensionless gravito-bending parameter $\rho g L^3/k_b$ is plotted in Figure 2.4. In the dimensionless parameter, ρ is the product of the material density and the strip thickness. Simulated results of the present model are very close to the reference solution. In contrast, the methods proposed by Grinspun

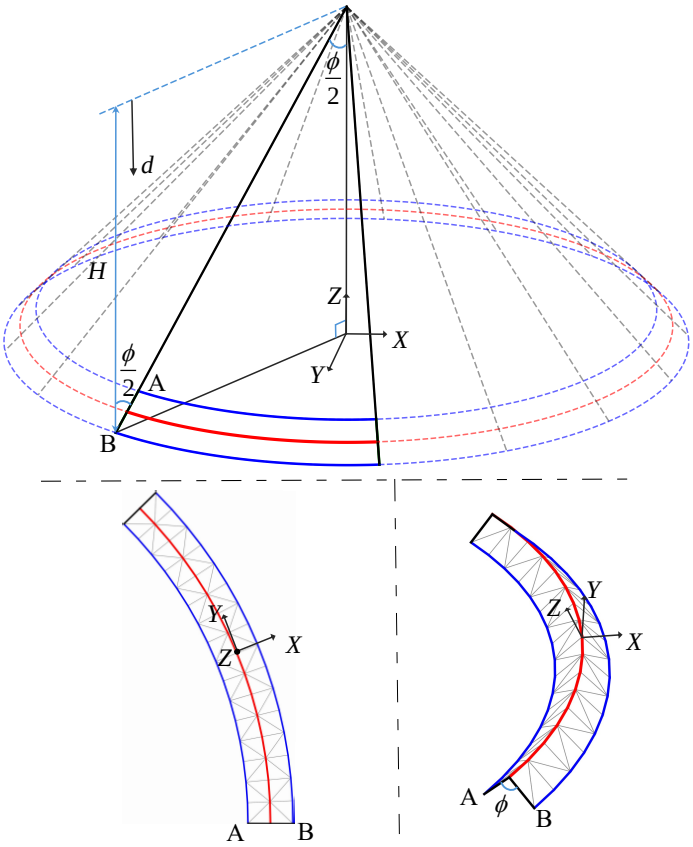


Figure 2.5: The bottom left image shows a 45° annular sector with radii 100 ± 5 and the middle arc in red. The annular sector is extracted from the cone in the top image. The bottom right image illustrates the folded state of the annular sector, which is a curved-crease origami structure with a fold angle of ϕ.

et al. [66], Bridson et al. [19], and Woodruff et al. [207] exhibit noticeable deviations from the reference. Noteworthily, Romero et al. [152] conducted tests on one-end clamped strips made of different materials. The measured results agree well with the afore-used reference solution.

Folding the annular sector into two cone surfaces

Following [207], an annular sector is cut from a cone, flattened into a planar sheet, and then folded along the curved crease to form a curved-crease structure (see Figure 2.5). Folding the initially flat sheet along a curved crease involves bending the developable cone surfaces during the motion. Under the isometric assumption, the theoretical bending energy

of the annular region on the right cone surface can be computed as

$$\Psi_{\text{annular}} = \frac{1}{2}\int_{\Omega} k_b \kappa_p^2 \, d\Omega, \tag{2.24}$$

where $k_b = Eh^3/12(1-\nu^2)$ is the bending stiffness of an isotropic shell, Ω is the area of the annular sector, and the principal curvature κ_p is

$$\kappa_p = \frac{1}{\tan(\phi/2)\sqrt{1+\tan^2(\phi/2)d}}, \tag{2.25}$$

where d is the distance from the apex of the cone measured along its height, and $\phi/2$ is half of the cone's apex angle (see Figure 2.5). Following [207], when the fold angle of the curved-crease structure is also $\phi = 90^\circ$, the resulting folded shape becomes an exact segment of a cone.

In the simulation, the present hinge-bending model is incorporated into MERLIN2 [207, 107], which is a popular open-sourced bar-hinge code. Woodruff et al. [207] previously evaluated the convergence behavior of their model under mesh refinement. Based on their study, dividing the fold line into 50 segments leads to close to convergent predictions. The grey lines in the bottom images of Figure 2.5 indicate the distribution of bending hinges. The geometric and material parameters follow those in [207]. The geometry consists of a 45° annular sector with radii 0.1 ± 0.005 and the middle arc shown in red (see bottom left image in Figure 2.5). The material properties are $E = 4 \times 10^9$, $\nu = 0$, and $h = 0.1$. To satisfy the condition for the fold angle equal to 90°, the following boundary conditions are applied: the displacements in the X-, Y-, and Z-displacements are fixed at point A; the Y- and Z-displacements are fixed at point B; and the Z-displacements of the inner and outer rings are fixed. A vertical displacement of magnitude $0.005/\sqrt{2}$ is applied to the middle arc in the Z-direction.

The theoretical solution for the stored energy in the resulting right cone is 6.39×10^{-6}. The present hinge-bending model yields a result of 6.12×10^{-6} which is far much more accurate than 12×10^{-6} predicted by Woodruff et al.'s model.

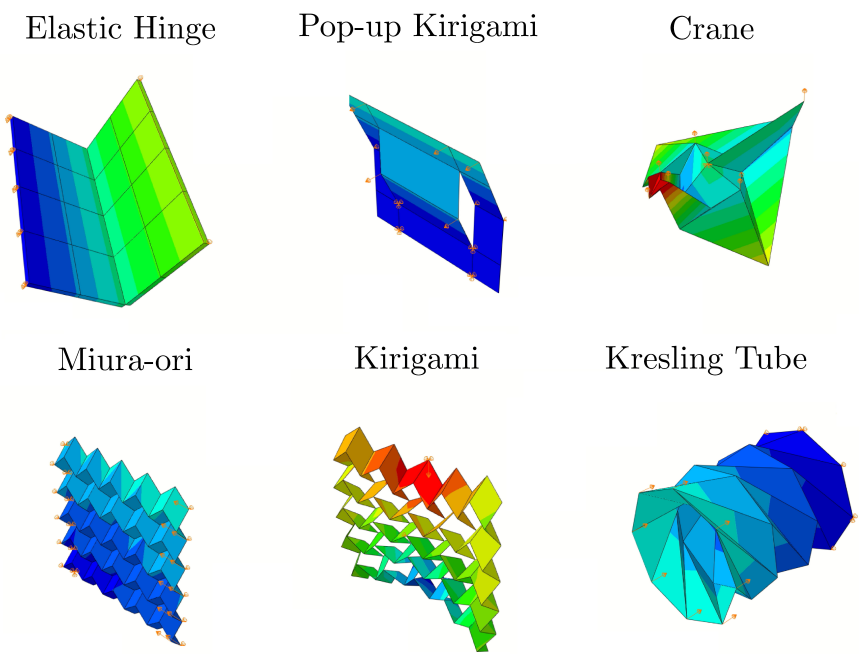


Figure 2.6: The schematic geometries and boundary conditions of six different straight-crease origami structures from [72].

Table 2.1: Comparison of the number of load increments between Hu et al. [72] and the present hinge-folding model. In the "Cases" column, entries without parentheses denote origami panels modeled using M3D4/M3D3 elements together with their developed corotational Q4 bending element.

Cases	**Hu et al.**	**Ours**
Elastic Hinge (SC8R)	31	31
Elastic Hinge (SC8R side)	**44**	**31**
Elastic Hinge (S4R)	674	674
Elastic Hinge	22	22
Pop-up Kirigami using rest angle loading	12	12
Pop-up Kirigami using displacement loading	**18**	**12**
Crane using rest angle loading	11	11
Crane using displacement loading	8	8
Compressing Miura-ori	12	12
Compressing the derived Kirigami	17	17
Pinching Miura-ori	9	9
Pinching the derived Kirigami	**9**	**8**
Kresling tube under compression	11	11

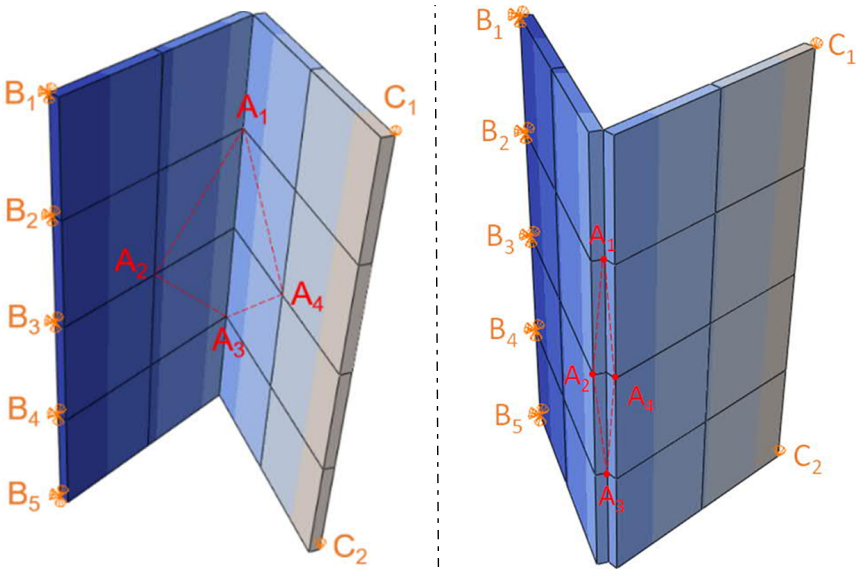


Figure 2.7: The origami panels are modeled using SC8R elements, where A_1, A_2, A_3, and A_4 are the four nodes of an edge stencil. The difference between the left and right images lies in the stencil position; the configuration shown on the right is referred to as "SC8R side" in Table 2.1.

Origami simulation in ABAQUS

In this test, the hinge-folding model is implemented in an ABAQUS UEL subroutine, and the origami simulations attempted by Hu et al. [72] are rerun to demonstrate the efficiency of the present hinge-folding model. Simulation details follow [72]: the material adopts paperboard properties $E = 3 \times 10^9$, $\nu = 0.3$, $h = 0.27 \times 10^{-3}$, and a fold stiffness per unit crease length of $k_f = 0.1$. The solver employs the Newton–Raphson method per increment with simultaneous convergence criteria of 0.5% force tolerance and 1% displacement tolerance; the automatic time incrementer is used (initial step = 1, minimum step $= 10^{-5}$); if a step fails to converge within 16 iterations the increment is reduced to one quarter and retried, with up to five reductions before aborting; if the last two converged steps both required ≤ 5 iterations, the increment is increased by 50%. Hu et al. test six straight-crease origami structures (representative snapshots are shown in Figure 2.6). In Table 2.1, their "Elastic Hinge" study compares origami panel discretizations using Abaqus S4R (4-node reduced-integration conventional shell element), SC8R (8-node

reduced-integration continuum-shell element), and their in-house corotational Q4 bending element combined with M3D4. They report that M3D4 combined with the corotational Q4 bending element yields the best computational efficiency, and the other five origami structures were simulated using M3D4/M3D3 (4-node/3-node 3D membrane elements) with corotational Q4 bending element. In these six origami structures, the creases are modeled with the edge-stenciled hinge-folding model. "Elastic Hinge (SC8R side)" denotes the SC8R-based elastic-hinge model with the hinge arranged in the alternative "side" configuration (see Figure 2.7). In Table 2.1, "rest angle loading" and "displacement loading" represent two different loading schemes.

In most cases, the present hinge-folding model yields predictions consistent with those of Hu et al. [72]. However, notable differences arise in the number of load increments required, as summarized in Table 2.1, particularly in scenarios such as the SC8R plate with a side elastic hinge, the pop-up Kirigami under displacement control, and the pinching of derived Kirigami structures. In these cases, the presented hinge-folding model produces comparable results while requiring fewer increments, highlighting its improved computational efficiency without compromising accuracy.

This improvement stems from the fact that Hu et al. simplify the gradient of the hinge angle by treating the scalar weights s_C and s_D as constants. This assumption is equivalent to enforcing isometry along the fold line. However, each origami panel undergoes small but non-zero membrane deformation, which violates the strict isometry assumption and negatively affects the force equilibrium in the iterative solver.

A more extreme example of this limitation arises when each small triangle in a cloth mesh is treated as an origami panel connected by folding hinges. Due to the soft nature of cloth materials, these panels are more prone to membrane deformation, making the assumption of constant s_C and s_D even more problematic. The detailed setup for this cloth simulation is presented in the next subsection.

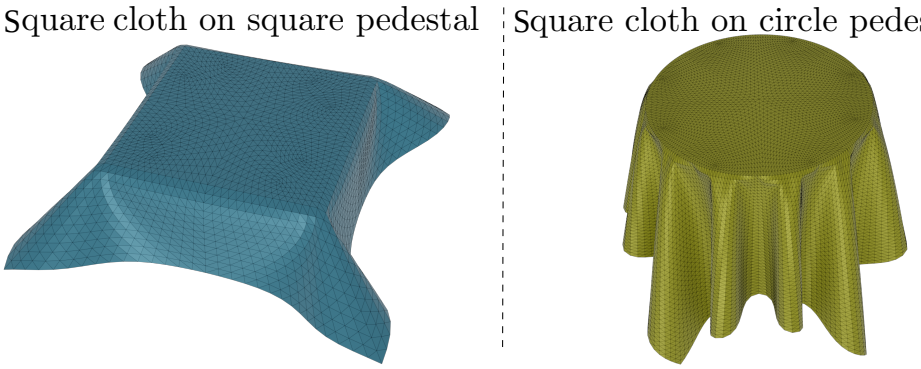


Figure 2.8: Cloth drape simulations. Left: Draped pattern of a 0.2×0.2 cloth over a 0.1×0.1 pedestal predicted by using 1,440 active nodes. Right: Draped pattern of a 0.5×0.5 over a $\phi 0.2$ pedestal predicted by using 10,560 active nodes. Self-contact occurs due to the longer free-hanging length. The hinge angle gradient used in the present model leads to convergent results in both simulations whereas that in Hu et al. [72]'s model does not.

2.3.2 Qualitative Tests

This subsection first highlights the limitation of simplifying the gradient of the hinge angle, as done by Hu et al. [72], by applying it to cloth drape simulations. It then considers a cloth benchmark [100] to further demonstrate the robustness of the present model. In these simulations, the material properties of a 100% cotton fabric [100] are used: area density $\rho = 0.1503$, thickness $h = 3.18 \times 10^{-4}$, Young's modulus $E = 8.21 \times 10^5$, and Poisson's ratio $\nu = 0.243$. The gravitational acceleration is set to 9.81, and a friction coefficient of 0.4 is employed where applicable. For elastodynamic simulations, the time step is 0.04.

Cloth drape simulation

The drape test is a critical evaluation in the textile industry [84, 48]. In this test, two specific scenarios are examined to demonstrate the robustness of the presented hinge angle gradient over the one used by Hu et al. [72]'s model: (1) draping a 0.2×0.2 square cloth (1,440 active nodes) over a 0.1×0.1 square pedestal (shown on the left-hand side of Figure 2.8), and (2) draping a 0.5×0.5 cloth (10,560 active nodes) over a $\phi 0.2$ pedestal (see the right-hand side of Figure 2.8). Self-contact occurs

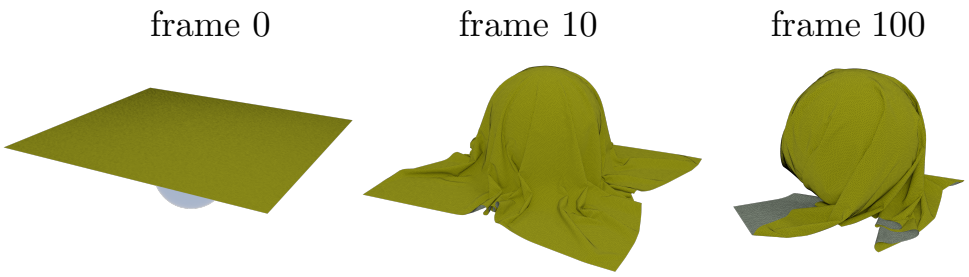


Figure 2.9: Simulation of a square cloth with 85,451 nodes dropped onto a rotating sphere on a floor. As the sphere rotates, friction pulls the cloth inward, forming intricate wrinkles and folds. This test demonstrates the robustness of the present hinge-bending model under complex conditions involving tight wrinkling, friction, and contact.

in the second but not the first scenario due to the difference in the free-hanging length. Convergence can be obtained by using the present hinge angle gradient, but not the one employed in Hu et al. [72]'s model.

Cloth on rotating sphere

This test assesses the robustness of the present formulation under complicated conditions involving tight wrinkling, friction, and complex contact scenarios, based on the setup from previous studies [20, 100]. In this test, a 1×1 square cloth with 85,451 nodes drops onto a $\phi 0.2$ sphere rotating on a flat floor (see Figure 2.9). The rotation axis is vertical to the floor, with an angular speed of π. As the sphere rotates, friction pulls the cloth inward, generating intricate patterns of wrinkles and folds.

2.4 Conclusion

In this chapter, a hinge-bending model that approximates discrete curvature by fitting an edge stencil onto a cylindrical surface is first introduced. Through a series of quantitative tests, the present hinge-bending model demonstrates higher accuracy than existing hinge-bending models [19, 66, 207]. A detailed comparison of these hinge-bending energy formulations is provided in Appendix G, showing that they differ from the present formulation only by a constant factor. For applications built upon these earlier formulations, the present hinge-bending model can

readily serve as an improved replacement by adjusting the constant factor for higher accuracy. Furthermore, the hinge angle and its derivatives are directly used as a hinge-folding model that connects non-coplanar origami panels. By leveraging both the gradient and Hessian of the hinge angle, the hinge-folding model bridges seamlessly with finite element-based origami panels, leading to enhanced efficiency and robustness compared to existing finite element-based origami simulations [72].

It should be noted that although the hinge-bending model converges under mesh refinement in the linear plate bending benchmark with equilateral triangle meshes, the edge-stenciled hinge model exhibits noticeable mesh dependency issue (as defined in Section 1.1.4) when applied to general plate or shell deformations with arbitrary mesh structures [65, 221]. This issue motivates the development of triangle-centered stencil models in Chapter 3. Nevertheless, due to its simplicity and robustness, the present hinge-bending formulation remains a practical choice for qualitative shell simulations. For practical applications, equilateral triangular meshes are recommended for the edge-stenciled hinge-bending model, as the convergence study (Figure 2.3) shows that the model converges to the analytical solution only on equilateral meshes, while regular (non-equilateral structured) meshes already exhibit noticeable deviation (see also Figure 3.8). It is worth noting that, based on our experience, mesh irregularity does not affect the stability of the solver — the simulations remain robust even on irregular meshes, though the accuracy of the bending response degrades.

Chapter 3

Corotational Hinge-based Thin Plates/Shell Models

3.1 Introduction

Modeling and simulation of thin plates and shells have long been critical topics in both the computational mechanics and computer graphics communities. In computational mechanics, the primary focus has traditionally been on achieving high accuracy. In computer graphics, efficiency has been a dominant concern. In recent years, there has been a growing convergence between these two fields, with increasing interest in developing computational models that are both efficient and accurate [26, 155, 218]. Beyond efficiency and accuracy, robustness is also an essential requirement [19, 92, 100]. In this context, robustness refers to the ability to stably reach the user-specified total simulation time in complex scenarios involving strong geometric nonlinearities, large time steps, and complex collisions/frictions.

Inspired from both discrete differential geometry shells [8, 59, 64, 65, 205, 221] and rotation-free shells [134, 218]. Section 1.1.4 provides a review of discrete differential geometry shells and rotation-free shells. This chapter presents a suite of bending models for thin plates and shells that aim to strike a balance between efficiency and accuracy, while also ensuring robustness. Specifically:

- Three types of corotational hinge-based discrete curvature operators are developed for thin plate/shell simulation, namely the

edge-based, FVM, and smoothed variants.

- All presented six thin plate/shell models are characterized by constant bending energy Hessians and incorporate carefully designed boundary conditions for accurate simulation.

Built upon the linear or 3-node triangle mesh, these models are straightforward to integrate into existing finite element frameworks and thin shell (cloth) simulators. This chapter is based on the results originally reported in [103], with additional analysis and discussion included here.

3.2 Geometric Discretization

3.2.1 Kinematics

The smooth midsurface of KL shell is discretized by the linear triangular mesh which is piecewise planar. Two types of shell stencil are considered. The corotational edge-based hinge bending model is based on an edge stencil, consisting of two adjacent triangles and their common edge (see the middle image in Figure 1.1). The corotational FVM/ smoothed hinge bending model is based on a triangle-centered stencil, which includes one central triangle and its three neighboring triangles (see the right image in Figure 1.1). For a given point $\mathbf{x}$ on a stencil, the position vector in the current configuration is

$$\mathbf{x} = \mathbf{X} + \mathbf{U}, \tag{3.1}$$

where $\mathbf{X}$ is the position vector in the initial configuration, and $\mathbf{U}$ is the displacement vector.

3.2.2 Terminologies and Remarks

Terminologies. In this chapter, superscript "s" denotes a vector that collects the nodal vectors in a stencil. For instance, the vector for an edge stencil is

$$\mathbf{x}^s = \begin{bmatrix} \mathbf{x}_1^T & \mathbf{x}_2^T & \mathbf{x}_3^T & \mathbf{x}_4^T \end{bmatrix}^T,$$

while the vector for a triangle-centered stencil is

$$\mathbf{x}^s = \begin{bmatrix} \mathbf{x}_1^T & \mathbf{x}_2^T & \mathbf{x}_3^T & \mathbf{x}_4^T & \mathbf{x}_5^T & \mathbf{x}_6^T \end{bmatrix}^T.$$

The quantity defined in the corotational frame is equipped with a tilde "~". $\tilde{X} - \tilde{Y} - \tilde{Z}$ and $\tilde{x} - \tilde{y} - \tilde{z}$ are the initial and current corotational frames and the corresponding position vectors are $\tilde{\mathbf{X}} = [\tilde{X}, \tilde{Y}, \tilde{Z}]^T$ and $\tilde{\mathbf{x}} = [\tilde{x}, \tilde{y}, \tilde{z}]^T$, respectively. The initial and current corotational frames are the local Cartesian frames attached to the initial and current stencils, respectively. These frames are used to quantify the nodal projected distances of these stencils to discretize the curvatures. As usual, the quantities defined in the corotational frame can be transformed from those in the global Cartesian frame. The coordinate transformation is illustrated in Appendix D. The subscript "c" denotes the projection onto either the $\tilde{X}$–$\tilde{Y}$ plane of the initial corotational frame or the $\tilde{x}$–$\tilde{y}$ plane of the current corotational frame. Graphical illustration of the projection can be found in the Figure 3.1, Figure 3.2 and Figure 3.5.

The abbreviations "EP", "ES", "FP", "FS", "SP" and "SS" are used to distinguish between different bending formulations, when necessary. In the first letter of the abbreviation, "E", "F" and "S" refer to the edge-based, FVM and smoothed hinges, respectively. In the second letter of the abbreviation, "P" and "S" refer to plates and shells, respectively.

Remarks. To facilitate understanding of the derivation process, several key points are first mentioned. The kinematics of a shell stencil deformed from its initial configuration to the current configuration can be described by

$$\mathbf{x}^s = \mathbf{X}^s + \mathbf{U}^s.$$

In the initial configuration, a shell stencil with $\tilde{\mathbf{X}}_c^s$ obtained by projecting $\tilde{\mathbf{X}}^s$ along $\tilde{Z}$ onto the $\tilde{X} - \tilde{Y}$ plane is referred to as the initial corotational shell stencil. Using $\tilde{\mathbf{X}}^s - \tilde{\mathbf{X}}_c^s$ which approximates the projection of $\tilde{\mathbf{X}}^s$ along $\tilde{Z}$ and introducing discrete curvature operators constructed from $\tilde{\mathbf{X}}_c^s$, the initial curvature can be discretized. The discrete curvature operator approximates smooth curvature on meshes by quantifying local geometric bending. Both $\tilde{\mathbf{x}}_c^s$ and $\tilde{\mathbf{X}}_c^s$ can be transformed by $\mathbf{x}_c^s$ and $\mathbf{X}_c^s$,

respectively (see Appendix D). Under the small strain and small curvature assumption, the relative positions of the projected positions $\tilde{\mathbf{x}}_c^s$ in the current corotational frame and $\tilde{\mathbf{X}}_c^s$ in the initial corotational frame are approximately the same. Another view is that the corotational shell stencil deforms from the initial configuration to the current configuration, i.e.

$$\mathbf{x}_c^s = \mathbf{X}_c^s + \mathbf{U}_c^s,$$

which approximately undergoes rigid-body motion. So, the current discrete curvature can be defined using the discrete curvature operators of the initial curvature. Alternatively, if the discrete curvature operators are derived in the initial corotational frame, they remain constant during the simulation. The discrete curvature operator is a numerical formulation defined on a discretized surface to approximate the continuous curvature operator on a smooth surface for geometric processing or simulation [203]. Recalling the shell theory in Section 1.3.1, the bending deformation in Eq.(1.13) is quantified by the change in curvature between the initial and current configurations. Then, the bending energy of a stencil is formulated. Further details are provided in the subsequent subsections to get the constant bending energy Hessians. The source code (`https://github.com/liangqx-hku/libThinPlateShells`) is also made available for free access.

3.2.3 Corotational Edge-based Hinge Bending Model

In an edge stencil (see Figure 3.1), the curvature along the edge 2-3 is zero. In the current configuration, the discrete directional curvature κ_d across the edge in Eq.(2.2) of Chapter 2 is

$$\kappa_d = \frac{2\theta}{h_1 + h_4}, \tag{3.2}$$

where $h_1 = \|\mathbf{X}_{1P}\|$ and $h_4 = \|\mathbf{X}_{4Q}\|$ are the heights of the triangles T_{123} and T_{432}. The two heights are approximated by those in the initial configuration under the small strain assumption. Under the small curvature

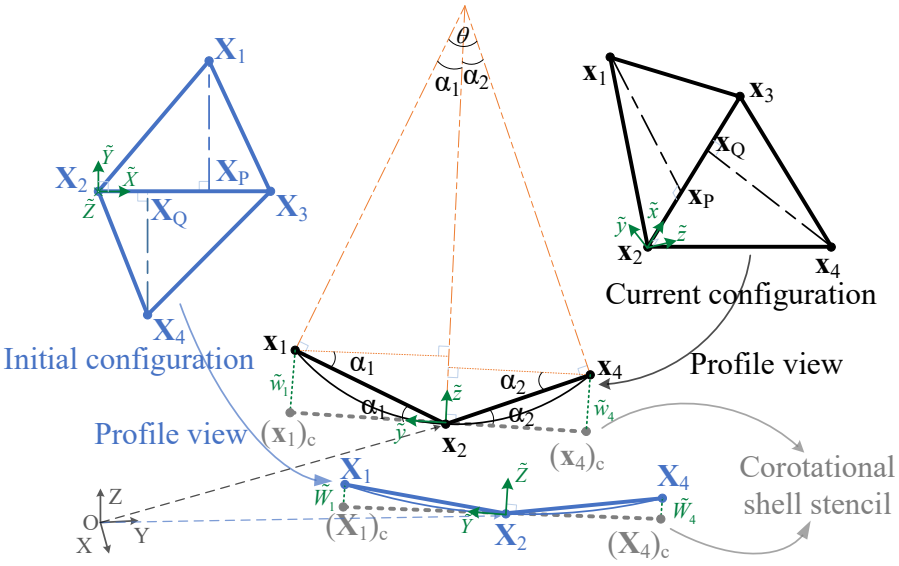


Figure 3.1: An edge stencil deforms from its initial configuration (blue) to the current configuration (black). The corotational frame (green) is employed to capture nodal projected distances relative to the corotational shell stencil (grey dashed line). $X-Y-Z$ is the global Cartesian frame. In the current configuration, the $\tilde{x}-\tilde{y}-\tilde{z}$ frame is the current corotational frame, where the $\tilde{x}$-axis aligns with the edge 23, and the $\tilde{z}$-axis direction is approximately along the bisector of the hinge angle θ. The $\tilde{y}$-axis is determined by the right-hand rule. The corotational shell stencil is the projection of the shell stencil onto the $\tilde{x}-\tilde{y}$ plane. α_1 and α_4 describe the bend angles of two adjacent triangles relative to the corotational shell stencil. The point $(\mathbf{x}_i)_c$ is the projection of $\mathbf{x}_i$ onto the $\tilde{x}-\tilde{y}$ plane, while the transverse displacement $\tilde{w}_i$ corresponds to the projection of the relative difference between $\mathbf{x}_i$ and $(\mathbf{x}_i)_c$ along the $\tilde{z}$-axis. Variables related to the shell stencil in the initial configuration can be extended from those in the current configuration.

assumption, the hinge angle in the current configuration can be approximated by

$$\theta = \alpha_1 + \alpha_2 \simeq \sin\alpha_1 + \sin\alpha_2 \simeq \frac{\mathbf{n}_{\tilde{z}}^T(\tilde{\mathbf{x}}_1 - \tilde{\mathbf{x}}_P)}{h_1} + \frac{\mathbf{n}_{\tilde{z}}^T(\tilde{\mathbf{x}}_4 - \tilde{\mathbf{x}}_Q)}{h_4}. \tag{3.3}$$

Here, the perpendicular feet $\tilde{\mathbf{x}}_P$ and $\tilde{\mathbf{x}}_Q$ are given by

$$\tilde{\mathbf{x}}_P = \frac{\|\tilde{\mathbf{x}}_{P3}\|}{\|\tilde{\mathbf{x}}_{23}\|}\tilde{\mathbf{x}}_2 + \frac{\|\tilde{\mathbf{x}}_{P2}\|}{\|\tilde{\mathbf{x}}_{23}\|}\tilde{\mathbf{x}}_3,\ \tilde{\mathbf{x}}_Q = \frac{\|\tilde{\mathbf{x}}_{Q3}\|}{\|\tilde{\mathbf{x}}_{23}\|}\tilde{\mathbf{x}}_2 + \frac{\|\tilde{\mathbf{x}}_{Q2}\|}{\|\tilde{\mathbf{x}}_{23}\|}\tilde{\mathbf{x}}_3. \tag{3.4}$$

Under the small strain assumption, these perpendicular feet in Eq.(3.4) can be approximated as

$$\tilde{\mathbf{x}}_P = \frac{\|\tilde{\mathbf{X}}_{P3}\|}{\|\tilde{\mathbf{X}}_{23}\|}\tilde{\mathbf{x}}_2 + \frac{\|\tilde{\mathbf{X}}_{P2}\|}{\|\tilde{\mathbf{X}}_{23}\|}\tilde{\mathbf{x}}_3,\ \tilde{\mathbf{x}}_Q = \frac{\|\tilde{\mathbf{X}}_{Q3}\|}{\|\tilde{\mathbf{X}}_{23}\|}\tilde{\mathbf{x}}_2 + \frac{\|\tilde{\mathbf{X}}_{Q2}\|}{\|\tilde{\mathbf{X}}_{23}\|}\tilde{\mathbf{x}}_3, \tag{3.5}$$

and $\mathbf{n}_{\tilde{z}} = \mathbf{n}_c/\|\mathbf{n}_c\|$ is the direction of the $\tilde{z}$-axis, where $\mathbf{n}_c$ is computed by

$$\mathbf{n}_c = \frac{\mathbf{x}_{P1}}{\|\mathbf{x}_{P1}\|} + \frac{\mathbf{x}_{Q4}}{\|\mathbf{x}_{Q4}\|}. \tag{3.6}$$

In case that T_{123} and T_{432} are coplanar, $\mathbf{n}_{\tilde{z}}$ is taken to be the normal of triangle T_{123} (T_{432}). It can be seen that the direction $\mathbf{n}_{\tilde{z}}$, which bisects the supplementary angle $(\pi - \theta)$ of the hinge angle, approximates the normal direction of the smooth shell midsurface in the current configuration.

By substituting Eq.(3.3) and Eq.(3.5) into Eq.(3.2), the discrete directional curvature in the current configuration is discretized as

$$\kappa_d = \mathbf{L}_d \mathbb{m}^T \tilde{\mathbf{x}}^s = \mathbf{L}_d \mathbf{d}^s. \tag{3.7}$$

Here, $\mathbf{L}_d$ is the corotational edge-based hinge curvature operator, i.e.

$$\mathbf{L}_d = \frac{2}{(h_1+h_4)}\left[\frac{1}{h_1} \quad -\left(\frac{\|\tilde{\mathbf{X}}_{P3}\|}{\|\tilde{\mathbf{X}}_{23}\|h_1} + \frac{\|\tilde{\mathbf{X}}_{Q3}\|}{\|\tilde{\mathbf{X}}_{23}\|h_4}\right) \quad -\left(\frac{\|\tilde{\mathbf{X}}_{P2}\|}{\|\tilde{\mathbf{X}}_{23}\|h_1} + \frac{\|\tilde{\mathbf{X}}_{Q2}\|}{\|\tilde{\mathbf{X}}_{23}\|h_4}\right) \quad \frac{1}{h_4}\right], \tag{3.8}$$

$\mathbb{m} = \mathbf{I}_{4\times4} \otimes \mathbf{n}_{\tilde{z}}$, where $\otimes$ is the Kronecker product, and $\mathbf{I}_{4\times4}$ is the fourth order identity matrix. The transverse displacement vector of the edge stencil with respect to the $\tilde{x} - \tilde{y} - \tilde{z}$ frame is

$$\begin{aligned}\mathbf{d}^s &= \mathbb{m}^T \tilde{\mathbf{x}}^s \\ &= \begin{bmatrix}\mathbf{n}_{\tilde{z}}^T\tilde{\mathbf{x}}_1 & \mathbf{n}_{\tilde{z}}^T\tilde{\mathbf{x}}_2 & \mathbf{n}_{\tilde{z}}^T\tilde{\mathbf{x}}_3 & \mathbf{n}_{\tilde{z}}^T\tilde{\mathbf{x}}_4\end{bmatrix}^T \\ &= \begin{bmatrix}\tilde{w}_1 & \tilde{w}_2 & \tilde{w}_3 & \tilde{w}_4\end{bmatrix}^T,\end{aligned} \tag{3.9}$$

which quantifies the projected distances of the nodes away from the projected nodes in the corotational shell stencil (see Figure 3.1).

To facilitate the computation of curvature derivatives, the transverse

displacement vector in Eq.(3.9) is expressed in global Cartesian coordinates, i.e.

$$\mathbf{d}^s = \begin{bmatrix} \tilde{w}_1 & \tilde{w}_2 & \tilde{w}_3 & \tilde{w}_4 \end{bmatrix}^T = \mathbb{m}^T(\mathbf{x}^s - \mathbf{x}_c^s), \tag{3.10}$$

leading the discrete directional curvature in Eq.(3.8) to

$$\kappa_d = \mathbf{L}_d \mathbb{m}^T(\mathbf{x}^s - \mathbf{x}_c^s). \tag{3.11}$$

In the geometric view, the points in $\mathbf{x}_c^s$ lie in the $\tilde{x} - \tilde{y}$ plane of the current corotational frame. Since the matrix $\mathbb{m}$ spans the normal direction(s) of this frame, it follows directly that

$$\mathbb{m}^T \mathbf{x}_c^s = \mathbf{0},$$

which implies

$$\mathbf{L}_d \mathbb{m}^T \mathbf{x}_c^s = \mathbf{0}. \tag{3.12}$$

This result can also be obtained by rewriting the left-hand side of Eq.(3.12) as

$$\mathbf{L}_d \mathbb{m}^T \mathbf{x}_c^s = \mathbf{n}_{\tilde{z}}^T (\mathbf{L}_d \otimes \mathbf{I}) \mathbf{x}_c^s, \tag{3.13}$$

where

$$(\mathbf{L}_d \otimes \mathbf{I}) \mathbf{x}_c^s = \mathbf{0}. \tag{3.14}$$

This relation can be interpreted using [8, 9, 203], where the operator $\mathbf{L}_d$ corresponds to a discrete Laplacian constructed using the cotangent formula [144] (see Appendix F). When applied to a set of points lying on a common plane, the operator yields zero due to the vanishing curvature of the planar surface.

Thus, the Eq.(3.8) of the discrete directional curvature in the current configuration becomes

$$\kappa_d = \mathbf{L}_d \mathbb{m}^T \mathbf{x}^s. \tag{3.15}$$

Similarly, the directional curvature in the initial configuration can be expressed as

$$K_d = \mathbf{L}_d \mathbf{D}^s = \mathbf{L}_d \mathbb{N}^T \mathbf{X}^s, \tag{3.16}$$

where $\mathbb{N} = \mathbf{I}_{4\times 4} \otimes \mathbf{n}_{\tilde{Z}}$, and $\mathbf{D}^s = \mathbb{N}^T \mathbf{X}^s$ quantifies the projected distances

of the edge stencil away from the corotational shell stencil in the initial configuration.

It should be mentioned that these discrete directional curvatures can also be rewritten as

$$\kappa_d = \mathbf{n}_{\tilde{z}}^T(\mathbf{L}_d \otimes \mathbf{I})\mathbf{x}^s,\ K_d = \mathbf{n}_{\tilde{Z}}^T(\mathbf{L}_d \otimes \mathbf{I})\mathbf{X}^s, \tag{3.17}$$

where $\mathbf{I}$ is the third order identity matrix. By comparing the discrete directional curvatures in Eq.(3.17) with the smooth curvature in Eq.(1.10) and (1.30) of Chapter 1, the smooth counterparts of Eq.(3.17) should be $\kappa_{\tilde{Y}\tilde{Y}}$ and $K_{\tilde{Y}\tilde{Y}}$. Therefore, $\mathbf{L}_d$ is referred to as the discrete curvature operator. This implies that if a discrete expression of the curvature operator is obtained in the initial corotational frame, the curvature discretizations will be effectively resolved.

The curvature derivation in the corotational edge-based hinge bending model can be extended to the corotational FVM/smoothed hinge bending model. The distinction lies in how curvature is discretized, using the edge stencil in the former and the triangle-centered stencil in the latter. For the corotational FVM/smoothed hinge bending model, the direction of the $\tilde{z}$-axis in the current corotational frame is exactly given by the normal of the central triangle T_{123}, that is, $\mathbf{n}_c$ in Eq. (3.6) should be replaced by

$$\mathbf{n}_c = \mathbf{x}_{12} \times \mathbf{x}_{13}. \tag{3.18}$$

Corotational edge-based hinge thin plate

The bending energy of the corotational edge-based hinge thin plate can be expressed as

$$\Psi_b^{EP} = \frac{1}{2} A_{\mathcal{E}} k_b \kappa_d^2, \tag{3.19}$$

where $A_{\mathcal{E}}$ is the total area of the edge stencil in the initial configuration. The bending rigidity is $k_b = Eh^3/[12(1-\nu^2)]$. By applying Eq.(3.11), the bending energy in Eq.(3.19) can be discretized as

$$\Psi_b^{EP} = \frac{A_{\mathcal{E}}}{2} k_b \kappa_d^2 = \frac{A_{\mathcal{E}}}{2} k_b (\mathbf{x}^s - \mathbf{x}_c^s)^T \mathbb{m} \mathbf{L}_d^T \mathbf{L}_d \mathbb{m}^T (\mathbf{x}^s - \mathbf{x}_c^s), \tag{3.20}$$

which can be regrouped as

$$\Psi_b^{EP} = \frac{A_{\mathcal{E}}}{2} k_b (\mathbf{x}^s - \mathbf{x}_c^s)^T (\mathbf{L}_d^T \mathbf{L}_d \otimes \mathbf{I}) \mathbb{m}\mathbb{m}^T (\mathbf{x}^s - \mathbf{x}_c^s). \tag{3.21}$$

For any point on the shell stencil under the small strain/curvature assumption, the relation

$$\mathbf{n}_{\tilde{z}} \mathbf{n}_{\tilde{z}}^T (\mathbf{x} - \mathbf{x}_c) = \mathbf{n}_{\tilde{z}} \tilde{w} \simeq \mathbf{x} - \mathbf{x}_c \tag{3.22}$$

holds. Consequently, Eq.(3.21) becomes

$$\begin{aligned} \Psi_b^{EP} &= \frac{A_{\mathcal{E}}}{2} k_b (\mathbf{x}^s - \mathbf{x}_c^s)^T (\mathbf{L}_d^T \mathbf{L}_d \otimes \mathbf{I}) (\mathbf{x}^s - \mathbf{x}_c^s) \\ &= \frac{A_{\mathcal{E}}}{2} k_b (\mathbf{x}^s - \mathbf{x}_c^s)^T (\mathbf{L}_d \otimes \mathbf{I})^T (\mathbf{L}_d \otimes \mathbf{I}) (\mathbf{x}^s - \mathbf{x}_c^s). \end{aligned} \tag{3.23}$$

By invoking Eq.(3.14), the bending energy is simplified to

$$\Psi_b^{EP} = \frac{A_{\mathcal{E}}}{2} k_b (\mathbf{x}^s)^T (\mathbf{L}_d^T \mathbf{L}_d \otimes \mathbf{I}) \mathbf{x}^s. \tag{3.24}$$

This shows that the bending energy is quadratic with respect to the nodal positions. As a result, the bending energy Hessian is constant, i.e.

$$\frac{\partial^2 \Psi_b^{EP}}{\partial \mathbf{x}^s \partial (\mathbf{x}^s)^T} = k_b A_{\mathcal{E}} \mathbf{L}_d^T \mathbf{L}_d \otimes \mathbf{I}, \tag{3.25}$$

and the gradient of the bending energy is linear with respect to the nodal positions, i.e.

$$\frac{\partial \Psi_b^{EP}}{\partial \mathbf{x}^s} = \left(\frac{\partial^2 \Psi_b^{EP}}{\partial \mathbf{x}^s \partial (\mathbf{x}^s)^T} \right) \mathbf{x}^s. \tag{3.26}$$

It is important to note that quantifying the discrete directional curvature operator in the global Cartesian frame leads to a expression similar to that of the quadratic shell model [8]. However, this model overestimates the bending energy by a factor of three, as numerical results will be discussed in detail in Section 3.4.1. The fundamental difference and accuracy discrepancy have been detailed in Appendix F.

Corotational edge-based hinge thin shell

The bending energy of the corotational edge-based hinge thin shell is

$$\Psi_b^{ES} = \frac{1}{2} A_{\mathcal{E}} k_b (\varepsilon_b^{ES})^2, \tag{3.27}$$

where the curvature change is

$$\varepsilon_b^{ES} = \kappa_d - K_d = \mathbf{L}_d \mathbb{m}^T \mathbf{x}^s - \mathbf{L}_d \mathbb{N}^T \mathbf{X}^s. \tag{3.28}$$

The gradient of the bending energy is

$$\frac{\partial \Psi_b^{ES}}{\partial \mathbf{x}^s} = A_{\mathcal{E}} k_b \frac{\partial \varepsilon_b^{ES}}{\partial \mathbf{x}^s} \varepsilon_b^{ES}, \tag{3.29}$$

where the gradient of the curvature change is

$$\frac{\partial \varepsilon_b^{ES}}{\partial \mathbf{x}^s} = (\mathbf{x}^s)^T \frac{\partial \mathbb{m}}{\partial \mathbf{x}^s} \mathbf{L}_d^T + \mathbb{m} \mathbf{L}_d^T. \tag{3.30}$$

Here, $\partial \mathbb{m} / \partial \mathbf{x}^s = \mathbf{I}_{4\times 4} \otimes (\partial \mathbf{n}_{\tilde{z}} / \partial \mathbf{x}^s)$, where $\partial \mathbf{n}_{\tilde{z}} / \partial \mathbf{x}^s$ is detailed in Appendix E.

Before deriving the bending energy Hessian, substituting Eq.(3.28) into Eq.(3.27) yields

$$\Psi_b^{ES} = \Psi_{flat}^{ES} + \Psi_{curved}^{ES}, \tag{3.31}$$

which decomposes into a flat part

$$\Psi_{flat}^{ES} = \frac{A_{\mathcal{E}}}{2} k_b (\mathbf{x}^s)^T \mathbb{m} (\mathbf{L}_d^T \mathbf{L}_d \otimes \mathbf{I}) \mathbb{m}^T \mathbf{x}^s \tag{3.32}$$

and a curved part

$$\Psi_{curved}^{ES} = \frac{A_{\mathcal{E}}}{2} k_b (-2(\mathbf{x}^s)^T (\mathbf{L}_d^T \mathbf{L}_d \otimes \mathbf{I}) \mathbb{m} \mathbb{N}^T + (\mathbf{X}^s)^T (\mathbf{L}_d^T \mathbf{L}_d \otimes \mathbf{I}) \mathbb{N} \mathbb{N}^T) \mathbf{X}^s. \tag{3.33}$$

The flat part in Eq.(3.32) is identical to the bending energy of the corotational edge-based hinge thin plate in Eq.(3.20), and the curvature is small. Consequently, the simplified bending energy Hessian from the thin plate model in Eq.(3.25) can be employed in the thin shell model. A

similar approach can be found in the cubic shell paper [59].

Boundary conditions.

In thin shell simulations, the most commonly used boundary conditions are the clamped boundary condition, the free boundary condition, and the simply-supported boundary condition. The simply supported boundary condition can be effectively achieved by combining the free boundary condition with fixed boundary nodes, so the clamped and free boundary conditions will be discussed. In the boundary edge stencil $MNLL'$ (see Figure 3.7), the node L' is a virtual node that is symmetrically positioned with respect to node L across the midpoint P of the boundary edge MN.

The Clamped Boundary Condition. For a boundary edge stencil with a clamped boundary condition, a symmetric virtual transverse displacement is applied, corresponding to the virtual node L'

$$\tilde{w}_{L'} = \tilde{w}_L, \tag{3.34}$$

which ensures the preservation of the normal direction perpendicular to the boundary edge stencil. When this condition is applied to the boundary edge stencil, the corotational edge-based hinge curvature operator $\mathbf{L}_d$ is modified to

$$\mathbf{L}_d^{'} = \frac{1}{h_1}\begin{bmatrix} \frac{2}{h_1} & -\frac{\|\tilde{\mathbf{x}}_{P3}\|+\|\tilde{\mathbf{x}}_{Q3}\|}{\|\tilde{\mathbf{x}}_{23}\|h_1} & -\frac{\|\tilde{\mathbf{x}}_{P2}\|+\|\tilde{\mathbf{x}}_{Q2}\|}{\|\tilde{\mathbf{x}}_{23}\|h_1} & 0 \end{bmatrix} \tag{3.35}$$

This zero-slope condition is also applicable in symmetric structural finite element analysis.

The Free Boundary Condition. For a free boundary edge, there is no bending energy in the boundary edge stencil, resulting in zero entries for both the gradient and Hessian related to this boundary (zero-curvature condition).

In conclusion, the modification of the curvature operator according to the virtual nodes enables the boundary conditions be prescribed at the stencil level. Boundary conditions are crucial in engineering shell

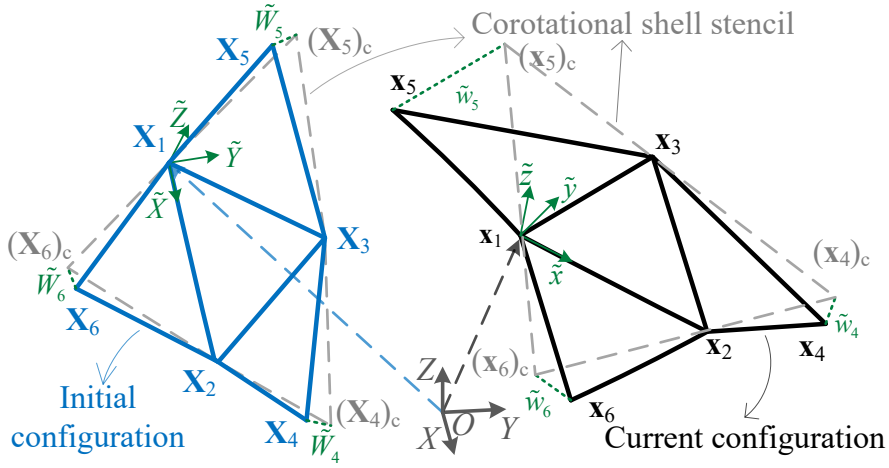


Figure 3.2: A triangle-centered stencil for modeling the corotational FVM hinge bending model deforms from its initial configuration (blue) to the current configuration (black). $X-Y-Z$ is the global Cartesian frame. In the initial configuration, the $\tilde{X}-\tilde{Y}-\tilde{Z}$ frame (green) is the initial corotational frame, where the $\tilde{X}$-axis aligns with the edge 12, and the $\tilde{Z}$-axis direction is along the normal direction of the central triangle. The $\tilde{Y}$-axis is determined by the right-hand rule. The corotational shell stencil (grey dashed line) is the projection of the stencil onto the $\tilde{X}-\tilde{Y}$ plane. The point $(\mathbf{X}_i)_c$ represents the projection of $\mathbf{X}_i$ onto the $\tilde{X}-\tilde{Y}$ plane, while $\tilde{W}_i$ corresponds to the projection of the relative difference between $\mathbf{X}_i$ and $(\mathbf{X}_i)_c$ along the $\tilde{Z}$-axis. Variables related to the shell stencil in the current configuration can be extended from those in the initial configuration.

simulations to achieve accurate results. However, the impact of the free boundary condition on visual effects in computer animation is often negligible.

3.2.4 Corotational FVM Hinge Bending Model

The curvature of the central triangle can be evaluated using the finite volume method (FVM) within the corotational frame. In the initial configuration, the vector of the curvature components [134] is

$$\boldsymbol{K} = \frac{1}{A}\int_A \begin{pmatrix} \frac{\partial^2 \tilde{W}}{\partial \tilde{X}^2} \\ \frac{\partial^2 \tilde{W}}{\partial \tilde{Y}^2} \\ 2\frac{\partial^2 \tilde{W}}{\partial \tilde{X}\partial \tilde{Y}} \end{pmatrix} dA, \tag{3.36}$$

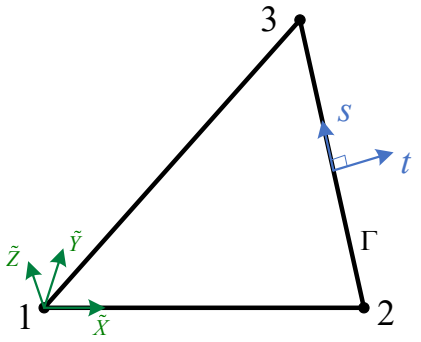


Figure 3.3: A schematic diagram of the s-t frame, which lies in the $\tilde{X}$-$\tilde{Y}$ plane.

which is constant over the central triangle (control cell). The bending deformation is described by Eq.(1.30) in Chapter 1. Here, A is the area of the central triangle in the initial configuration. By applying the Stokes's theorem, Eq.(3.36) can be rewritten as

$$\boldsymbol{K} = \frac{1}{A}\int_{\Gamma} \mathbf{T}\begin{pmatrix}\frac{\partial \tilde{W}}{\partial \tilde{X}} \\ \frac{\partial \tilde{W}}{\partial \tilde{Y}}\end{pmatrix} d\Gamma, \tag{3.37}$$

where Γ is the boundary of the central triangle in the initial configuration, and $\mathbf{T}$ is defined as

$$\begin{bmatrix} m_s & 0 \\ 0 & m_t \\ m_t & m_s \end{bmatrix}, \tag{3.38}$$

with the normalized normal $\tilde{\mathbf{m}} = \begin{bmatrix} m_s & m_t \end{bmatrix}^T$ outward to the boundary Γ surrounding the central triangle in the $\tilde{X} - \tilde{Y}$ plane of the initial corotational frame. The s-axis is tangential to the boundary Γ and the t-axis is perpendicular to the s-axis (see Figure 3.3). The gradient of the transverse displacement $\tilde{W}$ can be transformed from the directional gradient using

$$\begin{pmatrix}\frac{\partial \tilde{W}}{\partial \tilde{X}} \\ \frac{\partial \tilde{W}}{\partial \tilde{Y}}\end{pmatrix} = \begin{bmatrix} m_s & -m_t \\ m_t & m_s \end{bmatrix} \begin{pmatrix}\frac{\partial \tilde{W}}{\partial t} \\ \frac{\partial \tilde{W}}{\partial s}\end{pmatrix}, \tag{3.39}$$

where $\partial\tilde{W}/\partial t$ and $\partial\tilde{W}/\partial s$ are the rotations about the s-axis and t-axis, respectively.

Given that there is no curvature along each edge of the central triangle (see Figure 3.2), i.e., $\partial\tilde{W}/\partial t \neq 0$ and $\partial\tilde{W}/\partial s = 0$ in Eq.(3.39), the curvature in Eq.(3.37) can be discritized as

$$\boldsymbol{K} = \sum_{i=1}^{3} \begin{pmatrix} (m_s)_i^2 \\ (m_t)_i^2 \\ 2(m_s)_i(m_t)_i \end{pmatrix} (K_d)i, \tag{3.40}$$

where $(K_d)_i = (\partial\tilde{W}/\partial t)_i l_i/A$ is the directional curvature with l_i being the edge length in the initial configuration. It can be seen that the FVM approach obtains the vector of curvature components from the directional curvatures of the central triangle. To discretize the curvature and consider the adjacent triangles' contribution to the common edge, several discretized schemes can be found in [134, 61].

In this chapter, the Eq.(3.2) is first extended to express the directional curvatures, $\boldsymbol{K}$ can then be rewritten as

$$\boldsymbol{K} = \mathbf{R} \begin{pmatrix} (K_d)_1 \\ (K_d)_2 \\ (K_d)_3 \end{pmatrix} \simeq \mathbf{R}\boldsymbol{K}_p = \mathbf{R} \begin{pmatrix} \frac{2\Theta_1}{h_1+h_4} \\ \frac{2\Theta_2}{h_2+h_5} \\ \frac{2\Theta_3}{h_3+h_6} \end{pmatrix}, \tag{3.41}$$

where vector $\boldsymbol{K}_p$ collects the discrete directional curvatures in the initial configuration, Θ_i $(i = 1,2,3)$ is the hinge angle in initial configuration, h_i $(i = 1,2,\cdots,6)$ is the triangle height in the initial configuration (see Figure 3.4), and the transform matrix $\mathbf{R}$ is

$$\mathbf{R} = \begin{bmatrix} (m_s)_1^2 & (m_s)_2^2 & (m_s)_3^2 \\ (m_t)_1^2 & (m_t)_2^2 & (m_t)_3^2 \\ 2(m_s)_1(m_t)_1 & 2(m_s)_2(m_t)_2 & 2(m_s)_3(m_t)_3 \end{bmatrix}. \tag{3.42}$$

Finally, under the small strain/curvature assumption, each discrete directional curvature in Eq.(3.41) is approximated using the corotational edge-based hinge curvature operator in Eq.(3.8).

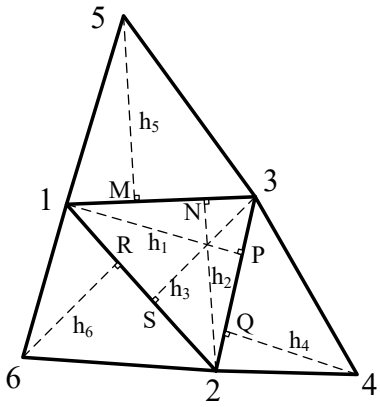


Figure 3.4: Definition of geometric quantities. The heights of the triangles are denoted as h_i where i ranges from 1 to 6, and the perpendicular feet are $M, N, P, Q, R,$ and$S$.

Corotational FVM hinge thin plate

The bending energy of the corotational FVM hinge thin plate is

$$\Psi_b^{FP} = \frac{A}{2}\boldsymbol{\kappa}_p^T \mathbf{D}_b^p \boldsymbol{\kappa}_p. \tag{3.43}$$

Here $\mathbf{D}_b^p = \mathbf{R}^T \mathbf{D}_b \mathbf{R}$, and the vector of discrete directional curvatures in the current configuration is

$$\boldsymbol{\kappa}_p = \mathbf{L}_p \mathbb{m}^T \mathbf{x}^s, \tag{3.44}$$

where $\mathbb{m} = \mathbf{I}_{6\times 6} \otimes \mathbf{n}_{\tilde{z}}$ with $\mathbf{n}_{\tilde{z}}$ normalized as described $\mathbf{n}_c$ in Eq.(3.18). $\mathbf{I}_{6\times 6}$ is a sixth order identity matrix. The curvature operator $\mathbf{L}_p$ is

$$\mathbf{L}_p = -\mathbf{H}\begin{bmatrix} -\frac{1}{h_1} & \left(\frac{\|\tilde{\mathbf{X}}_{M3}\|}{ah_5} + \frac{\|\tilde{\mathbf{X}}_{N3}\|}{ah_2}\right) & \left(\frac{\|\tilde{\mathbf{X}}_{R2}\|}{bh_6} + \frac{\|\tilde{\mathbf{X}}_{S2}\|}{bh_3}\right) \\ \left(\frac{\|\tilde{\mathbf{X}}_{Q3}\|}{ch_4} + \frac{\|\tilde{\mathbf{X}}_{P3}\|}{ch_1}\right) & -\frac{1}{h_2} & \left(\frac{\|\tilde{\mathbf{X}}_{R1}\|}{bh_6} + \frac{\|\tilde{\mathbf{X}}_{S1}\|}{bh_3}\right) \\ \left(\frac{\|\tilde{\mathbf{X}}_{Q2}\|}{ch_4} + \frac{\|\tilde{\mathbf{X}}_{P2}\|}{ch_1}\right) & \left(\frac{\|\tilde{\mathbf{X}}_{M1}\|}{ah_5} + \frac{\|\tilde{\mathbf{X}}_{N1}\|}{ah_2}\right) & -\frac{1}{h_3} \\ -\frac{1}{h_4} & 0 & 0 \\ 0 & -\frac{1}{h_5} & 0 \\ 0 & 0 & -\frac{1}{h_6} \end{bmatrix}^T, \tag{3.45}$$

with $a = \|\tilde{\mathbf{X}}_{31}\|$, $b = \|\tilde{\mathbf{X}}_{12}\|$, $c = \|\tilde{\mathbf{X}}_{23}\|$, and

$$\mathbf{H} = \begin{bmatrix} \frac{2}{h_1+h_4} & 0 & 0 \\ 0 & \frac{2}{h_2+h_5} & 0 \\ 0 & 0 & \frac{2}{h_3+h_6} \end{bmatrix}. \tag{3.46}$$

Consequently, the vector of curvature components in Eq.(3.40) can be discretized by the corotational FVM hinge curvature operator $\mathbf{RL}_p$.

In deriving the derivatives of the bending energy of corotational FVM hinge thin plate, the constant bending energy Hessian can also be obtained similar to the corotational edge-based hinge thin plate in Eq.(3.25), i.e.

$$\frac{\partial^2 \Psi_b^{FP}}{\partial \mathbf{x}^s \partial (\mathbf{x}^s)^T} = A(\mathbf{L}_p^T \mathbf{D}_b^p \mathbf{L}_p \otimes \mathbf{I}). \tag{3.47}$$

The operation in Eq.(3.26) can similarly be employed to compute the gradient of the corotational FVM hinge thin plate model.

Corotational FVM hinge thin shell

The bending energy of the corotational FVM hinge thin shell is

$$\Psi_b^{FS} = \frac{A}{2} (\boldsymbol{\varepsilon}_b^{FS})^T \mathbf{D}_b \boldsymbol{\varepsilon}_b^{FS}, \tag{3.48}$$

where the curvature change vector $\boldsymbol{\varepsilon}_b^{FS}$ is

$$\boldsymbol{\varepsilon}_b^{FS} = \mathbf{R}\boldsymbol{\kappa}_p - \mathbf{R}\boldsymbol{K}_p = \mathbf{RL}_p \mathbb{m}^T \mathbf{x}^s - \mathbf{RL}_p \mathbb{N}^T \mathbf{X}^s \tag{3.49}$$

with

$$\mathbb{N} = \mathbf{I}_{6\times 6} \otimes \mathbf{n}_{\tilde{Z}}. \tag{3.50}$$

The gradient of the bending energy of corotational FVM hinge thin shell is generalized from the corotational edge-based hinge thin shell, i.e.

$$\frac{\partial \Psi_b^{FS}}{\partial \mathbf{x}^s} = A \frac{\partial (\boldsymbol{\varepsilon}_b^{FS})^T}{\partial \mathbf{x}^s} \mathbf{D}_b \boldsymbol{\varepsilon}_b^{FS}, \tag{3.51}$$

where the gradient of the curvature change vector is

$$\frac{\partial \boldsymbol{\varepsilon}_b^{FS}}{\partial \mathbf{x}^s} = (\mathbf{x}^s)^T \frac{\partial \mathbb{m}}{\partial \mathbf{x}^s} \mathbf{L}_p^T \mathbf{R}^T + \mathbb{m} \mathbf{L}_p^T \mathbf{R}^T. \tag{3.52}$$

The bending energy Hessian remains the same as in the corotational FVM hinge thin plate model, for reasons similar to those discussed in the corotational edge-based hinge thin shell model in Section 3.2.3.

Boundary conditions.

For boundary triangle-centered stencils in which at least one triangle is missing, the boundary conditions from the corotational edge-based hinge bending model can be applied to the boundary triangle stencil (see Figure 3.7). It is crucial to set the mixed second derivative in curvature vector to zero for the boundary shell stencil with free edge boundary condition to ensure accuracy in passing the "engineering shell obstacle benchmark" tests, shown in Section 3.4.1.

3.2.5 Corotational Smoothed Hinge Bending Model

In the initial configuration, the vector of curvature components in the smoothed hinge bending model is

$$\boldsymbol{K} = \begin{pmatrix} \frac{\partial^2 \tilde{W}}{\partial \tilde{X}^2} \\ \frac{\partial^2 \tilde{W}}{\partial \tilde{Y}^2} \\ 2\frac{\partial^2 \tilde{W}}{\partial \tilde{X} \partial \tilde{Y}} \end{pmatrix}, \tag{3.53}$$

which can be discretized as

$$\boldsymbol{K} = \mathbf{L}_s \mathbf{D}^s, \tag{3.54}$$

where

$$\mathbf{D}^s = \begin{bmatrix} \tilde{W}_1 & \tilde{W}_2 & \tilde{W}_3 & \tilde{W}_4 & \tilde{W}_5 & \tilde{W}_6 \end{bmatrix}^T \tag{3.55}$$

is the vector of transverse displacements in the initial corotational frame. In the smoothed hinge model, a quadratic fitting surface is used to smooth the triangle-centered stencil (see Figure 3.5), and $\mathbf{L}_s$ is the corotational

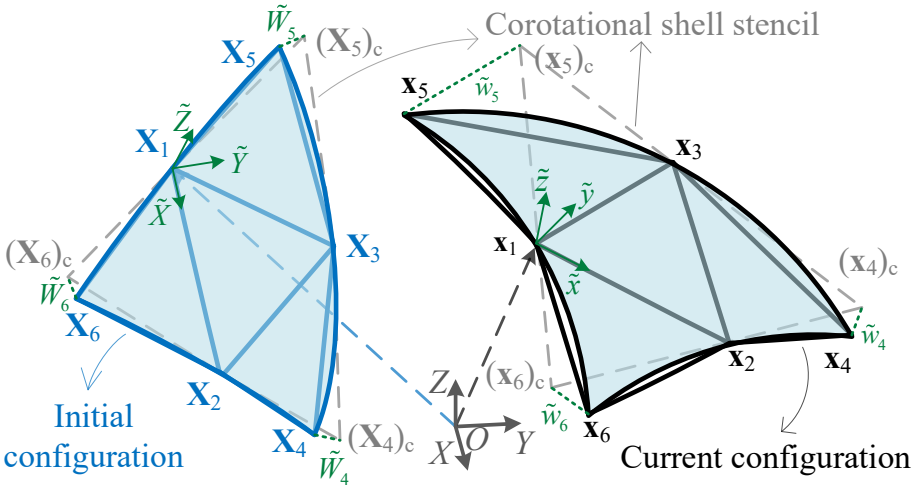


Figure 3.5: A triangle-centered stencil, fitted to a quadratic surface (light blue), models the corotational smoothed hinge bending model as it deforms from its initial configuration (blue) to the current configuration (black). $X-Y-Z$ is the global Cartesian frame, and $\tilde{X}-\tilde{Y}-\tilde{Z}$ (green) is the initial corotational frame, which captures nodal projected distances relative to the corotational shell stencil (grey dashed line). $(\mathbf{X}_i)_c$ is the projection of $\mathbf{X}_i$ onto the $\tilde{X}-\tilde{Y}$ plane, while $\tilde{W}_i$ is the relative difference between $\mathbf{X}_i$ and $(\mathbf{X}_i)_c$ projected along the $\tilde{Z}$-axis. Variables of the shell stencil in the current configuration can be extended from the initial configuration.

smoothed hinge curvature operator. The $\mathbf{L}_s$ will be derived later. The transverse displacements in Eq.(3.55) on the shell midsurface can be interpolated as

$$\tilde{W} = c_1 + c_2\tilde{X} + c_3\tilde{Y} + c_4\tilde{X}^2/2 + c_5\tilde{Y}^2/2 + c_6\tilde{X}\tilde{Y}/2, \tag{3.56}$$

where c_i $(i = 1, \cdots, 6)$ are linear combinations of the components in $\mathbf{D}_e$. The first three c_is correspond to the rigid-body modes, while the last three c_is lead to curvature components of the quadratic surface. By introducing Eq.(3.56) into Eq.(3.53), the vector of curvature components in Eq.(3.54) can also be written as

$$\boldsymbol{K} = \mathbf{L}_s\mathbf{D}^s = \begin{pmatrix} c_4 \\ c_5 \\ c_6 \end{pmatrix}. \tag{3.57}$$

The three components of $\mathbf{K}$ can be expressed by converting the three directional curvature constituent parts in the vector of directional curvatures

$$\boldsymbol{K}_p = \mathbf{L}_p \mathbf{D}^s, \tag{3.58}$$

which can be obtained by extending Eq.(3.7), Eq.(3.15) and Eq.(3.44) to the initial configuration. The $\mathbf{D}^s$ in Eq.(3.55) is interpolated by the quadratic function in Eq.(3.56), and the $\mathbf{L}_p$ is a discrete second order operator, which eliminates the linear terms. Consequently, $\mathbf{K}_p$ in Eq.(3.58) can be reformulated as

$$\mathbf{K}_p = \mathbf{L}_p \mathbf{D}^s = \mathbf{L}_p \mathbf{C}_p \begin{pmatrix} c_4 \\ c_5 \\ c_6 \end{pmatrix}, \tag{3.59}$$

where

$$\mathbf{C}_p = \begin{bmatrix} \tilde{X}_1^2/2 & \tilde{Y}_1^2/2 & \tilde{X}_1\tilde{Y}_1/2 \\ \vdots & \vdots & \vdots \\ \tilde{X}_6^2/2 & \tilde{Y}_6^2/2 & \tilde{X}_6\tilde{Y}_6/2 \end{bmatrix}.$$

Substituting Eq.(3.57) into Eq.(3.59), one can obtain

$$\mathbf{K}_p = \mathbf{L}_p \mathbf{C}_p \mathbf{L}_s \mathbf{D}^s. \tag{3.60}$$

Then, the corotational smoothed hinge curvature operator can be explicitly expressed as

$$\mathbf{L}_s = (\mathbf{L}_p \mathbf{C}_p)^{-1} \mathbf{L}_p. \tag{3.61}$$

This operator can be used to express the curvature vector of the smoothed hinge bending model in the initial configuration, i.e.

$$\boldsymbol{K} = \mathbf{L}_s \mathbb{N}^T \mathbf{X}^s, \tag{3.62}$$

where $\mathbb{N}$ has been defined in Eq.(3.49).

It is worth noting that the constant curvature of a quadratic surface can also be computed using the interpolation method, which involves solving a sixth-order linear system for each curvature operator [218]. However, in practice, the singularity arises in the configuration shown in Figure, where the proposed method remains effective. Another explicit

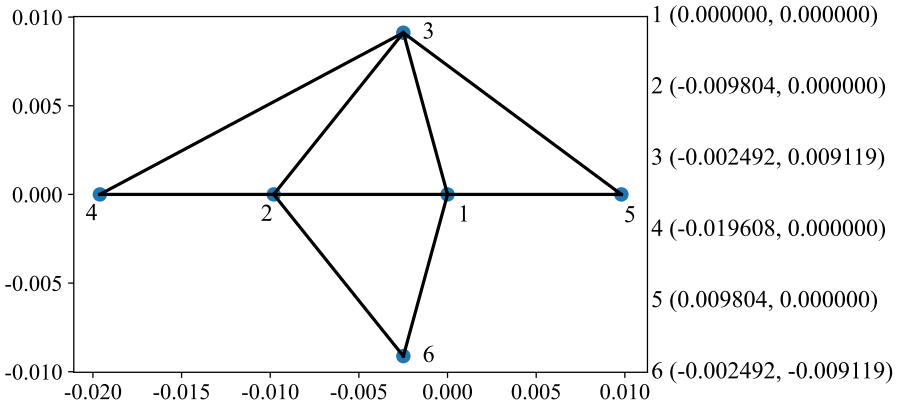


Figure 3.6: A stencil configuration exhibiting singularity arises when using the interpolation method [218], whereas the present corotational smoothed-hinge model does not.

formulation is presented in a follow-up note by Reisman et al. [151], which continues the work of Grinspun et al. [65] and highlights certain drawbacks of the curvature operator fitted on the quadratic surface. One such drawback arises in near-conic degenerate configurations, where the six nodes of a triangle-centered stencil lie approximately on a conic [151], leading to numerical instabilities. Thanks to the small strain/curvature assumption with the corotational approach, the bending energy Hessian matrix can be assembled once using the initial geometric data. A well-designed mesh initialization can effectively avoid numerical instabilities. Nevertheless, numerous numerical exercises indicate that the present model remains stable even without any special initialization. More detailed results can be found in the Section 3.4. Additionally, a concise treatment of boundary conditions is provided for the corotational smoothed hinge curvature operator, which will be elaborated in Section 3.2.5.

Corotational smoothed hinge thin plate

The bending energy of the corotational smoothed hinge thin plate is

$$\Psi_b^{SP} = \frac{A}{2}\boldsymbol{\kappa}^T \mathbf{D}_b \boldsymbol{\kappa}. \tag{3.63}$$

Its Hessian

$$\frac{\partial \Psi_b^{SP}}{\partial \mathbf{x}^s} = A(\mathbf{L}_s^T \mathbf{D}_b \mathbf{L}_s \otimes \mathbf{I}) \tag{3.64}$$

is also constant, and the gradient, which is linear about $\mathbf{x}^e$, can also be computed by the approach in Eq.(3.26).

Corotational smoothed hinge thin shell

The bending energy of the corotational smoothed hinge thin shell is

$$\Psi_b^{SS} = \frac{A}{2}(\boldsymbol{\varepsilon}_b^{SS})^T \mathbf{D}_b \boldsymbol{\varepsilon}_b^{SS}, \tag{3.65}$$

where the curvature change vector $\boldsymbol{\varepsilon}_b^{SS}$ is

$$\boldsymbol{\varepsilon}_b^{SS} = \boldsymbol{\kappa} - \boldsymbol{K} = \mathbf{L}_s \mathbb{m}^T \mathbf{x}^s - \mathbf{L}_s \mathbb{N}^T \mathbf{X}^s. \tag{3.66}$$

The gradient of the bending energy of corotational smoothed hinge thin shell is

$$\frac{\partial \Psi_b^{SS}}{\partial \mathbf{x}^s} = A \frac{\partial (\boldsymbol{\varepsilon}_b^{SS})^T}{\partial \mathbf{x}^s} \mathbf{D}_b \boldsymbol{\varepsilon}_b^{SS}, \tag{3.67}$$

where the gradient of the curvature change vector is

$$\frac{\partial (\boldsymbol{\varepsilon}_b^{SS})^T}{\partial \mathbf{x}^s} = (\mathbf{x}^s)^T \frac{\partial \mathbb{m}}{\partial \mathbf{x}^s} \mathbf{L}_s^T + \mathbb{m} \mathbf{L}_s^T. \tag{3.68}$$

The bending energy Hessian is the same as that of the corotational smoothed hinge thin plate model in Eq.(3.64). The rationale behind this similarity is consistent with the explanation provided for the corotational edge-based hinge thin shell model in Section 3.2.3.

Boundary conditions.

Within the context of the boundary edge MN depicted in Figure 3.7, the mesh topology informs that the nodal label $L' = L + 3$. Consequently, the curvature in the boundary shell stencil can be expressed as

$$\boldsymbol{\kappa} = \sum \mathbf{L}_{sj} \tilde{w}_j + \mathbf{L}_{s(L+3)} \tilde{w}_{L+3}, \text{ for } j = N, M, L, N+3, M+3. \tag{3.69}$$

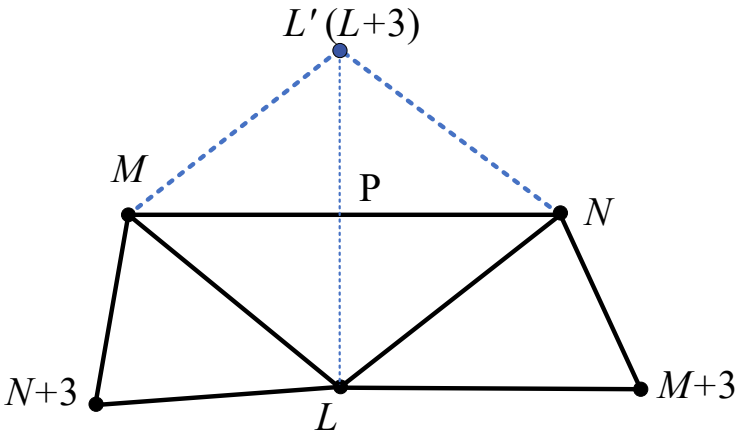


Figure 3.7: The artificial node L' (shown in blue) and node L are symmetric with respect to the midpoint P of the boundary edge NM.

Here, $\mathbf{L}_{sj}$ is the j column of the curvature operator matrix $\mathbf{L}_s$.

The Clamped Boundary Condition. To integrate the zero-slope condition, using Eq.(3.34), the modified curvature with a clamped edge becomes

$$\begin{aligned}\boldsymbol{\kappa}_{clamp} =&(\mathbf{L}_{sL} + \mathbf{L}_{s(L+3)})\tilde{w}_L + \mathbf{L}_{sM}\tilde{w}_M \\ &+ \mathbf{L}_{sN}\tilde{w}_N + \mathbf{L}_{s(N+3)}\tilde{w}_{N+3} + \mathbf{L}_{s(M+3)}\tilde{w}_{M+3}.\end{aligned} \tag{3.70}$$

The Free Boundary Condition. For the zero-curvature condition, the virtual transverse displacement of the virtual node should satisfy

$$\tilde{w}_{L'} = 2\tilde{w}_P - \tilde{w}_L, \tag{3.71}$$

where $\tilde{w}_P = (\tilde{w}_M + \tilde{w}_N)/2$ is the transverse displacement of the middle point P of the boundary edge NM. The modified curvature with a free edge is

$$\begin{aligned}\boldsymbol{\kappa}_{free} =&(\mathbf{L}_{sN} + \mathbf{L}_{s(L+3)})\tilde{w}_N + (\mathbf{L}_{sM} + \mathbf{L}_{s(L+3)})\tilde{w}_M \\ &+ (\mathbf{L}_{sL} - \mathbf{L}_{s(L+3)})\tilde{w}_L + \mathbf{L}_{s(N+3)}\tilde{w}_{N+3} + \mathbf{L}_{s(M+3)}\tilde{w}_{M+3}.\end{aligned} \tag{3.72}$$

If a boundary triangle includes one more boundary edge, the operations outlined in Eq.(3.70) and Eq.(3.72) can be superimposed.

It should be noted that the free boundary condition of the corotational edge-based hinge bending model is defined on the edge stencil, with the total structural energy accumulated over edges; consequently, the energy at the boundary is zero. In contrast, the corotational smoothed hinge bending model accumulates energy over triangles, and the energy of a boundary triangle must account for the influence of its neighboring triangles.

3.3 Implementations

Dynamics Simulation. The incremental potential [83] for elastodynamic simulations can be expressed as

$$E(\mathbb{x}) = \frac{1}{2}(\mathbb{x} - \hat{\mathbb{x}})^T \mathbb{M}(\mathbb{x} - \hat{\mathbb{x}}) + \Delta t^2 \Psi_{elastic} + B(\mathbb{x}) + D(\mathbb{x}), \tag{3.73}$$

where $\mathbb{M} \in R^{3n \times 3n}$ is the mass matrix, $\mathbb{x} \in R^{3n}$ is the global position vector, n is the number of the nodes in the mesh, and Δt is the time step. The predicted global position vector

$$\hat{\mathbb{x}} = \mathbb{x}^t + \Delta t \mathbb{v}^t + \Delta t^2 \mathbb{M}^{-1} \mathbb{f}_{ext}$$

is computed from the implicit Euler integration. Here, $\mathbb{f}_{ext} \in R^{3n}$ is the global external force vector, $\mathbb{x}^t \in R^{3n}$ and $\mathbb{v}^t \in R^{3n}$ are the global position and velocity vector at time t, respectively. $\Psi_{elastic}$ refers to elastic potential, which contains the elastic shell energy Ψ_{shell}. $B(\mathbb{x})$ and $D(\mathbb{x})$ are the contact barrier potential and the friction potential [99, 100], respectively. In Eq.(3.73), both elastic and contact interactions are incorporated. The global position vector at time $t+1$ is updated by minimizing the total potential in Eq.(3.73), i.e.

$$\mathbb{x}^{t+1} = \text{argmin}_x \, E(\mathbb{x}), \tag{3.74}$$

where the solution $\mathbb{x}^{t+1}$ is obtained iteratively using a Newton-type solver along with a continuous collision detection filter, ensuring intersection-free trajectories. This chapter builds upon the C-IPC codebase [100] for the dynamics simulation framework.

Linear and Quasi-static Simulations. The quasi-static simulation is used to evaluate the accuracy and efficiency of different formulations in Section 3.4.1, in which the contact is not involved, and equilibria are determined by setting the gradient of the total potential to zero, i.e.

$$\frac{\partial \Psi_{shell}}{\partial \mathbb{x}} + \mathbb{f}_{ext} = \mathbf{0}. \tag{3.75}$$

Since the results of ABAQUS will be as the reference solutions in quantitative benchmarks, a standard Newton-Raphson method [36] is employed under one load step to solve the geometrically nonlinear problem. The convergence criteria is $\|\partial \Psi_{\text{shell}}/\partial \mathbb{x} + \mathbb{f}_{\text{ext}}\| < \epsilon_{\mathbb{f}}$, where $\epsilon_{\mathbb{f}}$ is the residual force tolerance. At each Newton iteration,

$$\Delta \mathbb{x} = (\frac{\partial^2 \Psi_{\text{shell}}}{\partial \mathbb{x} \partial \mathbb{x}^T})^{-1} (\frac{\partial \Psi_{\text{shell}}}{\partial \mathbb{x}} + \mathbb{f}_{\text{ext}})$$

is the incremental displacement. When the maximum nodal displacement increment in a single iteration becomes too large, the entire displacement update is proportionally scaled down to prevent the structure from undergoing an unreasonably large deformation in one step. Specifically, if the infinity norm $\|\Delta \mathbb{x}\|_\infty$ exceeds the incremental displacement limit $\epsilon_{\Delta \mathbb{x}}$, the incremental displacement will be scaled by $\epsilon_{\Delta \mathbb{x}}/\|\Delta \mathbb{x}\|_\infty$. When the solution is far from equilibrium, this scaling method can effectively reduce the incremental displacement for geometrically nonlinear problems, including bending-dominated problems. A similar strategy can be referred to [174]. For the linear plate bending benchmark in Section 3.4.1, the solution can be obtained with a linear system solver.

3.4 Numerical Tests

Quantitative numerical experiments are compared with reference solutions, which include the analytical solution for the linear plate bending benchmark test and results from ABAQUS for geometrically nonlinear benchmark tests. The present models are also compared against several state-of-the-art formulations from the discrete differential geometry shell

library, libshell [26], which includes three types of formulations: MidedgeTan, MidedgeSin, and MidedgeAve. These three formulations are all based on a triangle-centered stencil, differing only in the discrete approximation of curvature. The first two introduce additional degrees of freedom associated with mid-edge normal rotations, which is illustrated in Figure 1.2. In addition, comparisons are made with the quadratic and cubic shell models [8, 59], all of which employ constant bending energy Hessians. These comparisons demonstrate that the proposed models achieve competitive accuracy and efficiency. Moreover, qualitative numerical experiments further illustrate the robustness of the models in elastodynamic simulations. All formulations are implemented on the codebase of libshell [26, 30] for fairly quantitative comparison, and all of the present formulations are integrated into the C-IPC [100] for qualitative experiments. All quantities are expressed in the International System of Units (SI).

3.4.1 Quantitative Tests

Linear plate bending benchmark test

The linear plate bending benchmark is investigated using three different mesh structures (see Figure 3.8) to evaluate how the present models and existing models depend on the mesh structure and to observe their convergent behaviour under mesh refinement. Since comparable studies do not provide the detailed implementation of clamped boundary conditions, the simply supported boundary condition is applied on the entire boundary of the square plate, which is subjected to a uniform load perpendicular to its plane, to ensure a fair comparison. The square plate has an edge length of $a = 8$, with a uniform load of $B = 9.81$ acting on the body. The material properties are defined by $E = 2 \times 10^{11}$, $\nu = 0.3$, and $h = 0.01$. Using the double-sine series, the converged solution for the maximum deflection, i.e., $0.048744Ba^4(1-\nu^2)/(Eh^3)$ [188], is taken as the reference solution.

The present EP/ES model yields results in close agreement with the reference solution only on the equilateral triangular mesh (see left image

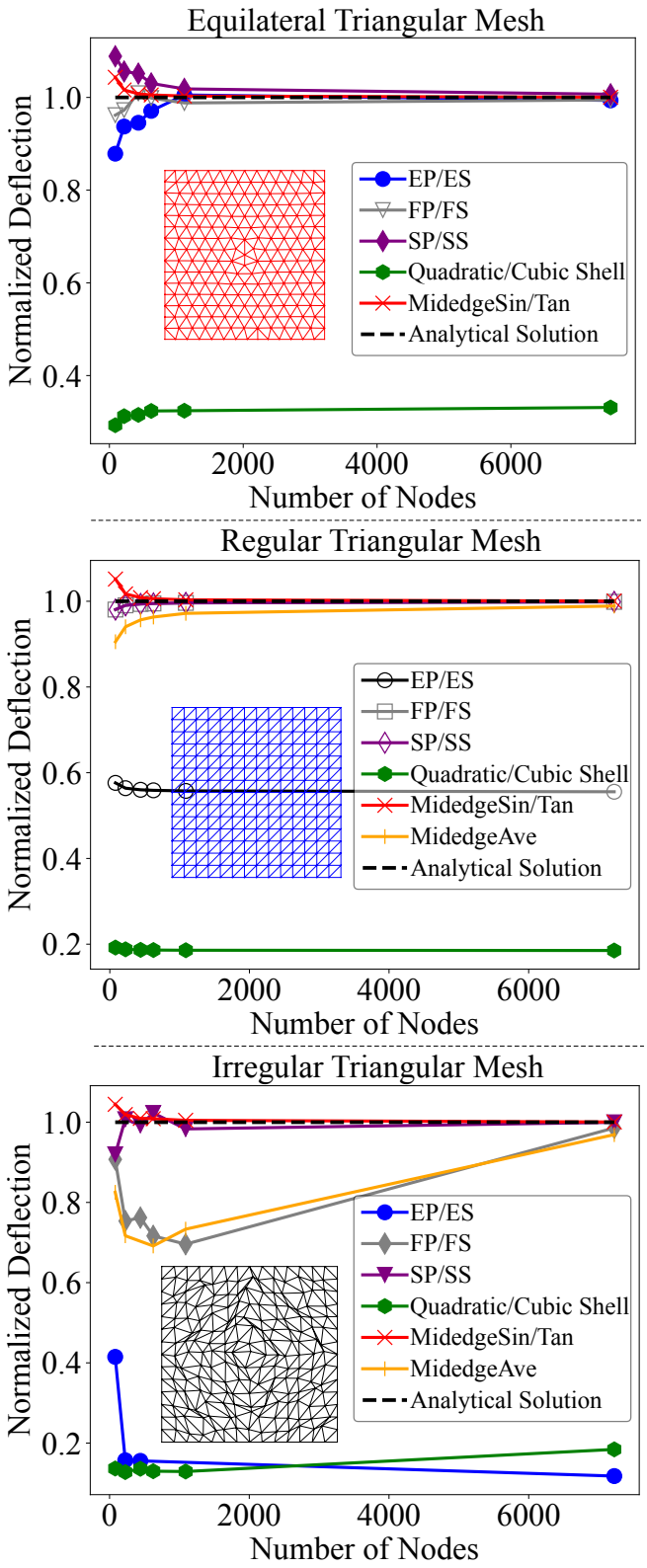


Figure 3.8: *Linear plate bending benchmark test.* Convergence and mesh dependency analysis of a simply supported linear plate under uniform load across three different mesh structures. The vertical axis represents the computed deflection normalized by the reference solution. EP/ES yield identical predictions, as do FP/FS and SP/SS. MidedgeSin and MidedgeTan also produce the same predictions, so a single marker is used for each group. MidedgeAve results in "NaN" values on the equilateral triangular mesh, and therefore its corresponding line is omitted.

in Figure 2.3). However, the quadratic/cubic shell model does not perform well across all tests, even on the equilateral triangular mesh. The primary reason is that the bending energy in the quadratic/cubic shell model is three times higher than the EP/ES model (see Appendix F). The MidedgeAve model also shows difficulties on the equilateral triangular mesh, where a singularity issue arises. On the regular triangular mesh (see middle image in Figure 2.3), the present FP/FS and SP/SS models perform slightly better than the MidedgeSin and MidedgeTan formulations. On the irregular mesh (see right image in Figure 2.3), MidedgeSin and MidedgeTan exhibit better mesh independence than other models. Among the present models, SP and SS outperform FP and FS, which have comparable performance to the MidedgeAve formulation.

The data from the last column in Table 3.1 highlights that the FP, FS, SP, and SS models are nearly twice as fast as the MidedgeSin and MidedgeTan formulations in one linear system solve on the equilateral triangular mesh. Additionally, the EP/ES and quadratic/cubic shell models based on the edge stencil demonstrate exceptional speed. The numerical performance differences across various mesh tessellations primarily arise from the curvature operators used in these formulations.

Geometrically nonlinear benchmark tests

This subsection aims to verify the expected accuracy and efficiency of the present models in geometrically nonlinear tests (see Table 3.1). The tests are derived from engineering obstacles [180]. The reference solutions are obtained using the S4R shell element in ABAQUS with a sufficiently high mesh density. The underlined geometry of S4R is quadrilateral, so the tested mesh is generated by splitting each quadrilateral into two triangles. For each simulation, the residual force tolerance $\epsilon_{\mathbb{f}}$ is 0.001, and the incremental displacement limit is $\epsilon_{\Delta\mathbb{x}} = 0.1$.

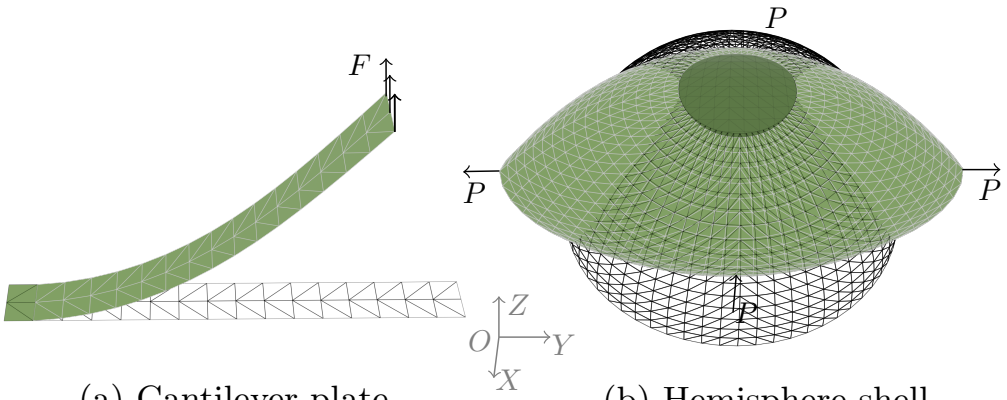
Cantilever Plate Subjected to End Shear Force. In this test, a cantilever plate of dimensions 10×1 is subjected to 4 units of end shear force, applied as concentrated loads of equal magnitude $F = 4/3$ at the three free-end nodes, as illustrated in Figure 3.9. The concentrated loads are along the Z-axis. The geometry is discretized into 51 nodes, with two

Model	Cantilever		Hemisphere Shell			Linear Plate Bending
	W_{tip}	Iterations (Time)	U_{min}	V_{max}	Iterations (Time)	Time (7459 nodes)
EP	5.388	61 (0.293s)				1.758s
ES	5.387	61 (0.343s)	-4.072	2.799	56 (27.334s)	1.778s
FP	6.056	68 (0.412s)				3.713s
FS	6.072	68 (0.464s)	-5.752	3.403	84 (44.153s)	3.788s
SP	6.055	67 (0.535s)				3.620s
SS	6.072	67 (0.595s)	-5.923	3.534	87 (45.035s)	3.759s
Quadratic Shell [8]	2.510	29 (0.145s)				1.757s
Cubic Shell [59]	2.510	29 (0.153s)	-3.193	2.414	51 (17.784s)	1.761s
MidedgeTan [26]	5.405	77 (2.545s)	-5.831	3.422	93 (173.531s)	6.963s
MidedgeSin [26]	5.418	77 (2.596s)	-5.886	3.451	94 (180.139s)	6.979s
MidedgeAve [26]	5.380	76 (2.213s)	-5.564	3.331	93 (105.898s)	NaN
ABAQUS S4R	*6.012*	*106*	*-5.902*	*3.406*	*112*	

Table 3.1: Displacement, Newton iteration and time data for the Cantilever under End Shear Force, Hemispherical Shell under Alternating Radial Forces, and Linear Plate Bending examples. "NaN" represents numerical issues encountered by the model. The results of *ABAQUS S4R* act as reference solutions. W_{tip} represents the displacement along the positive Z-direction of the midpoint on the right-hand side of the cantilever plate. U_{min} is the maximum displacement along the negative X-direction and V_{max} is the maximum displacement along the positive Y-direction.

adjacent rows at the clamped end to enforce the hard constraints. The material parameters are $E = 1.2 \times 10^6$, $\nu = 0.1$, and $h = 0.1$. As summarized in Table 3.1, the FP, FS, SP, and SS models demonstrate superior accuracy compared to others. The MidedgeTan and MidedgeSin formulations, which offer the second-highest accuracy, are nearly five times slower than the FP, FS, SP, and SS models, thanks to their constant bending energy Hessians. The EP and ES models rank third in accuracy, outperforming the MidedgeAve model. While the quadratic and cubic shell models are fast, they produce smaller deflections due to the overestimation of bending energy. If we scale down the bending energy of the Quadratic and Cubic Shell models, they can predict deflections comparable to those of the EP model. Among the present models, EP, FP and SP, specifically designed for initially flat shell configurations, perform slightly better in terms of accuracy and speed compared to ES, FS and SS, which can handle both initially flat and curved shell configurations.

Hemispherical Shell Subjected to Alternating Radial Forces. To test the performance of the initially curved shell models, a hemispherical shell



(a) Cantilever plate (b) Hemisphere shell

Figure 3.9: *Geometrically nonlinear benchmarks test.* (a) Cantilever plate (51 nodes) subjected to an end shear force F and (b) hemisphere shell with a 18° cut (1088 nodes) subjected to alternating radial forces P are tested to evaluate the accuracy and efficiency of different bending formulations in geometrically nonlinear analysis [180]. The green structure with a white wireframe represents the deformed configuration, while the black wireframe illustrates the undeformed configuration for comparison. $X - Y - Z$ is the global Cartesian frame.

with radius $R = 10$ and an 18° circular cutout at the pole is considered. The shell is subjected to alternating radial point forces of $P = 200$ at 90° intervals (see Figure 3.9). Two point forces along the X-axis induce compression, while two along the Y-axis induce tension. To minimize boundary condition effects across different formulations, the entire shell structure is analyzed instead of only a quarter section. Boundary conditions are applied as follows: for nodes lying in the $Y - Z$ plane, the X-direction dofs are fixed; for nodes in the $X - Z$ plane, the Y-direction dofs are fixed. Additionally, for nodes on the top circular cut that lie in the $Y - Z$ plane, the Z-direction dofs are constrained to ensure equivalence with the benchmark case provided in [180]. The shell geometry is discretized into 1088 nodes. The material parameters are $E = 6.825 \times 10^7$, $\nu = 0.3$, and $h = 0.04$. As shown in Table 3.1, the ES model outperforms the cubic shell models in terms of accuracy. However, the ES model deviates from the reference solution more than the others. While the MidedgeTan and MidedgeSin models provide more accurate results overall, the FS and SS models deliver competitive accuracy with nearly four times the computational speed of the MidedgeTan and MidedgeSin models.

3.4.2 Qualitative Tests

In this subsection, the material properties of a 100% cotton fabric [100] are used: area density $\rho = 0.1503$, thickness $h = 3.18 \times 10^{-4}$, Young's modulus $E = 8.21 \times 10^5$, and Poisson's ratio $\nu = 0.243$. The gravitational acceleration is set to 9.81, and a friction coefficient of 0.4 is employed where applicable. For elastodynamic simulations, the time step is 0.04.

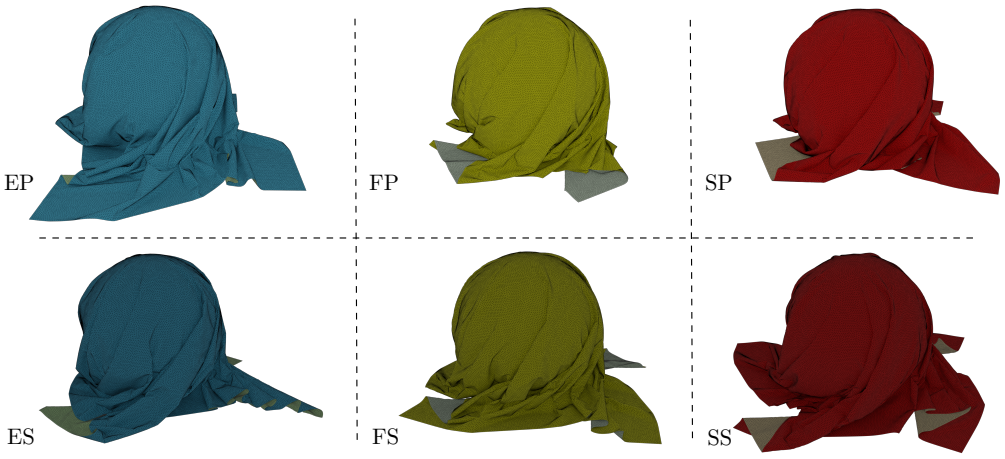


Figure 3.10: *Cloth on rotating sphere.* The images (the 100th frame) show the cloth's response to being dropped onto a sphere rotating on a flat floor. The cloth discretized by 85,451 nodes is pulled inward by friction, generating a complex structure of wrinkles and folds. The top row (lighter) illustrates the cloth behaviour for corotational edge-based hinge thin plate (EP), corotational FVM hinge thin plate (FP), and corotational smoothed hinge thin plate (SP) models, while the bottom row displays results for corotational edge-based hinge thin shell (ES), corotational FVM hinge thin shell (FS), and corotational smoothed hinge thin shell (SS) models, highlighting the robustness of each formulation in the cloth benchmark.

Cloth on Rotating Sphere. This test evaluates the robustness of the present formulations (EP, ES, FP, FS, SP and SS) under extreme stress-test conditions, such as tight wrinkling, friction, and contact processing. In this test, a 1×1 square cloth with 85,451 nodes drops onto a $\phi 0.2$ sphere rotating a fixed horizonal floor about its vertical axis at an angular speed π. As the sphere rotates, friction draws the cloth inward, creating a complex structure of wrinkles and folds that capture fine details of the cloth's behaviour (see Figure 3.10).

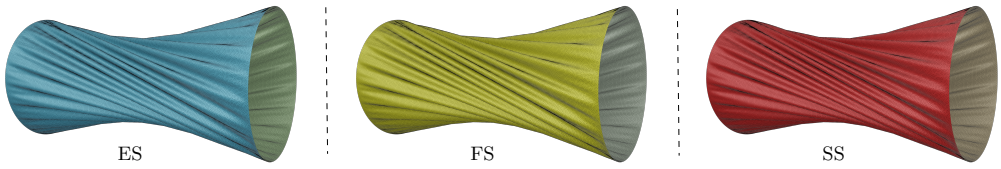


Figure 3.11: *Twisted cylinder.* The the simulation (the 10th frame) shows a cotton cylinder with width 1 and $\phi 0.5$ (87,600 nodes). The cylinder is twisted at $\pi/5$ while the ends are drawn together at 5×10^{-3}. Contact barrier is triggered at a threshold 1×10^{-3}, and gravity is excluded to prevent sagging, resulting in pronounced wrinkling and folding, demonstrating the robustness of the present models with initially curved geometry. From left to right, the simulated frames are respectively generated by the corotational edge-based hinge thin shell (ES), corotational FVM hinge thin shell (FS), and corotational smoothed hinge thin shell (SS) models.

Twisted Cylinder. In this test, a cotton cylinder (width 1 and $\phi 0.5$) modelled by 87,600 nodes is simulated. The IPC [99] contact force is activated at a threshold distance of 1×10^{-3}. The cylinder is twisted at a rate of $\pi/5$ while the two sides are simultaneously brought together at a rate of 5×10^{-3}. Gravity is excluded from the simulation to prevent sagging. As illustrated in Figure 3.11, global wrinkling and folding effects emerge as the cylinder is deformed, showcasing the ability of the present models (ES, FS and SS) to handle initially curved geometry robustly. It's worthy to mention that the SS model generates 20 wrinkles, but ES and FS both give 19 waves.

3.5 Conclusion

In this chapter, the present edge-stenciled models (EP and ES) are more accurate compared to the quadratic and cubic shell models. The formulations of quadratic thin plate/shell (QTP/QTS), a variation of the present EP/ES that quantifies the curvature operator in the global Cartesian frame, are provided in Appendix F. By introducing the formulations of QTP/QTS, the accuracy discrepancy between the present EP/ES models with the quadratic/cubic Shells [8, 59] is clarified in Appendix F. Like

the quadratic and cubic shell models, the EP and ES models are computationally efficient. However, they share the same limitations common to all edge-based hinge bending models in Chapter 2, as also discussed by Grinspun et al. [65]. The present triangle-centered models (FP, FS, SP and SS) partially address the mesh dependency issue (as defined in Section 1.1.4). Among these models, FP and FS exhibit slightly stronger mesh dependency than SP and SS, which benefit from the smoothing effect of quadratic interpolation functions. These differences are demonstrated in the linear plate bending benchmark (Figure 3.8), where three representative mesh structures are tested, and SP and SS show more consistent convergence behavior across these mesh configurations. For practical applications, the triangle-centered stencil models (FP/FS/SP/SS) are more tolerant of mesh irregularity than the edge-stenciled models (EP/ES). When mesh quality cannot be tightly controlled, SP or SS are preferred due to the smoothing effect of quadratic interpolation. Nevertheless, the present models are constrained by small-strain and small-curvature assumptions, and a failure arises when the hinge angle between a flap triangle and the central triangle exceeds 90°, resulting in an underestimation of the bending energy. This underestimation originates from the fact that once the transverse displacement of the flat nodes relative to the plane of the central triangle surpasses 90°, it begins to decrease, whereas the hinge angle continues to increase. Adaptive mesh refinement [49, 67, 127, 176] can be employed to reduce the hinge angle. Despite these assumptions, the use of a corotational approach to handle large displacements and rotations offers significant advantages, allowing the bending energy Hessian to remain constant in Newton-type implicit solvers. In the quasi-static simulations, a basic Newton solver is employed to evaluate the quantitative performance of different bending formulations in comparison to the shell element provided in ABAQUS. Therefore, it is recommended to integrate these formulations into a more robust and well-designed solver to fully exploit their efficiency in practical applications.

Owing to their simplicity, accuracy, efficiency, and generality, the present bending models are expected to serve as valuable additions to existing thin plate and shell libraries for simulating thin flexible objects,

and the implementation has been released as open-source.

Chapter 4

Smoothed-Hinge Membrane Model for Mitigating Sharp Creasing Artifact in Cloth Simulation

4.1 Introduction

Sharp creasing artifacts predicted by the numerical model can significantly reduce the fidelity of cloth simulation [100]. Although "non-physical sharp folds" and "sharp creasing artifacts" differ in terminology, they refer to the same visual phenomenon. For consistency, the sharp creasing artifacts is adopted in this thesis.

Strain limiting method for mitigating sharp creasing artifacts

For cloth simulations based on the mass–spring model, each spring undergoes either tension or compression, corresponding to tensile and compressive strains. The strain limiting method restricts these strains within certain bounds [147, 20, 19]. For cloth simulations based on the CST membrane element, strain limits are typically applied to the principal values of the membrane strain tensor [189, 186, 200]. In the Arcsim

simulator [127], strain limiting is formulated as a set of inequality constraints for each triangular membrane element and solved using an augmented Lagrangian method. In this simulator, the stretching and compression are enforced bilaterally. Considering that real fabrics resist tension but not compression, and exhibit weak stiffness under small strains followed by rapid stiffening beyond a material-specific threshold. Jin et al. [81] proposed constraining excessive stretching while leaving compressive strain unrestricted. This unilateral constraint is only activated when the stretching strain reaches its limit, reducing the risk of over-constraining the system compared to bilateral constraint. Implemented using an interior point method [129], this approach effectively controls overstretching and artificially stiffened bending behavior in mass-spring cloth simulations.

More recently, Li et al. [100] demonstrated that sharp creasing artifacts appeared in a cloth draping case. The artifacts were showed using their C-IPC simulator with real-world material parameters from [33, 136], and the Arcsim simulator [127] with material parameters from [201]. Both C-IPC and Arcsim use the same underlying computational models: the CST membrane model and the edge-based hinge-bending model [19, 66]. To mitigate sharp creasing artifacts, Li et al. proposed a constitutive strain limiting potential that integrates strain limiting constraints into the overall energy potential, transforming the constrained optimization problem into an unconstrained one solved using the interior point method. However, the constitutive potential requires expensive backtracking line search filtering and numerical eigendecomposition in the nonlinear solving procedure. To avoid these costs, Huang et al. [74] introduced a cubic inexact strain limiting potential with analytic eigensystems, eliminating the need for filtering while maintaining efficiency. Li et al. also observed that when the strain limit is set extremely low, sharp creasing artifacts remain difficult to suppress. They recommend further investigation into the underlying computational model of membrane components as a promising direction for future improvement.

Computational models for mitigating sharp creasing artifacts

English and Bridson [45] modeled cloth as a developable surface

by assigning element dofs to the midedge nodes of triangle edges rather than the vertices, and enforced zero membrane deformation by constraining the distances between these midedge nodes using Lagrange multipliers. However, this element requires coupling with a ghost triangle for collision handling. Quaglino [149] introduced a linear membrane triangle and a discrete linear strain triangle. They both show promising potential for cloth simulation. However, the formulations of these two triangles require solving a buckling problem for each mesh edge to enforce isometry constraint, a process that is computationally intensive and incompatible with implicit solvers. More recently, Sze and Zhou [176] observed that when the CST membrane element is combined with their rotation-free triangle, non-physical sharp folds may appear in fabric draping simulations. This issue was effectively alleviated by enhancing the 3-node displacement interpolation in the CST with a 6-node interpolation that involves all nodes in the rotation-free triangle. Previously, Flores and Oñate [55] also proposed a six-node displacement interpolation for deriving the membrane element in their rotation-free triangle element, and applied it successfully in both meta-forming [55] and fabric drape [53] simulations.

In this chapter, sharp creasing artifacts will be revisited in the context of CST membrane element combined with the hinge-bending model in Chapter2, as well as with the EP, FP, and SP models in Chapter 3. Similar artifacts are also observed when CST is combined with the ES, FS, and SS models; only their thin plate formulations (EP, FP and SP) will be discussed. Furthermore, a six-node interpolated position with an averaged strain scheme for the membrane model will be introduced as a potential solution for mitigating the sharp creasing artifacts. Both quantitative and qualitative numerical tests will be presented to show the efficacy of the present computational model. This chapter is based on the results originally reported in [105], with additional analysis and discussion included here.

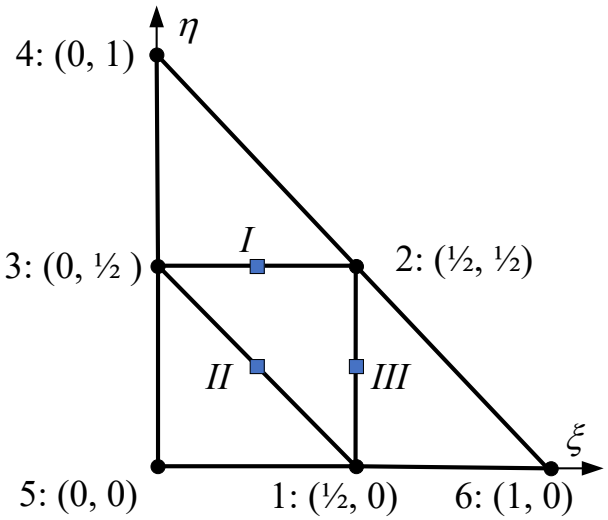


Figure 4.1: A triangle-centered stencil in the nature coordinates (ξ, η). I, II and III in blue are the edge midpoints of the central triangle T_{123}.

4.1.1 Smoothed-hinge Membrane Model

Based on the triangle-centered stencil in the nature coordinates (see Figure 4.1), the standard quadratic interpolated functions are

$$\begin{aligned} N_1 = 4\zeta\xi,\ N_2 = 4\xi\eta,\ N_3 = 4\eta\zeta, \\ N_4 = \eta(2\eta - 1),\ N_5 = \zeta(2\zeta - 1),\ N_6 = \xi(2\xi - 1), \end{aligned} \tag{4.1}$$

where the nature coordinates $\xi, \eta \in [0, 1]$ and $\zeta = 1 - \xi - \eta$. Thus, the interpolated initial and current position vectors are

$$\mathbf{X} = \sum_{i=1}^{6} N_i \mathbf{X}_i \text{ and } \mathbf{x} = \sum_{i=1}^{6} N_i \mathbf{x}_i. \tag{4.2}$$

In cloth simulation, sewing patterns are typically initialized as flat geometries, allowing the undeformed mesh placed in a plane parallel to the $X - Y$ plane of the global Cartesian coordinate frame X-Y-Z. Under this initialization, each triangle can directly adopt the $X - Y - Z$ frame

to define its material properties. The membrane strain is expressed as

$$\boldsymbol{\varepsilon}_m = \begin{Bmatrix} \varepsilon_{mXX} \\ \varepsilon_{mYY} \\ 2\varepsilon_{mXY} \end{Bmatrix} = \begin{Bmatrix} \frac{1}{2}\left(\mathbf{F}_X^T\mathbf{F}_X - 1\right) \\ \frac{1}{2}\left(\mathbf{F}_Y^T\mathbf{F}_Y - 1\right) \\ \mathbf{F}_X^T\mathbf{F}_Y \end{Bmatrix}. \tag{4.3}$$

Here, $\mathbf{F}_X$ and $\mathbf{F}_Y$ denote the column vectors of the 3×2 deformation gradient

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_X & \mathbf{F}_Y \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathbf{x}}{\partial X} & \frac{\partial \mathbf{x}}{\partial Y} \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathbf{x}}{\partial \xi} & \frac{\partial \mathbf{x}}{\partial \eta} \end{bmatrix} \begin{bmatrix} \partial X/\partial \xi & \partial X/\partial \eta \\ \partial Y/\partial \xi & \partial Y/\partial \eta \end{bmatrix}^{-1}. \tag{4.4}$$

In a triangle-centered stencil, the interpolated function in Eq.(4.1) ensures that the gradient at the edge midpoint of central triangle depends only on its four surrounding nodes, providing a unique gradient at the edge midpoint for the two triangles sharing the same edge [55]. More details can be checked in Appendix I. To consider the contributions from the neighboring triangles, the averaged membrane strain at the three edge mid-points is taken to be the membrane strain in the central triangle, i.e.

$$\boldsymbol{\varepsilon}_{\bar{m}} = \frac{1}{3}\sum_{i=I}^{III} \boldsymbol{\varepsilon}_m^i, \tag{4.5}$$

where I, II and III are the edge midpoints, see Figure 4.1. Consequently, the membrane energy in the central triangle is

$$\Psi_m^s = \frac{1}{2} A \boldsymbol{\varepsilon}_{\bar{m}}^T \mathbf{D}_m \boldsymbol{\varepsilon}_{\bar{m}}, \tag{4.6}$$

where A is the area of the central triangle.

The gradient of the membrane energy is

$$\frac{\partial \Psi_m^s}{\partial \mathbf{x}^e} = A \frac{\partial \boldsymbol{\varepsilon}_{\bar{m}}^T}{\partial \mathbf{x}^e} \mathbf{D}_m \boldsymbol{\varepsilon}_{\bar{m}}, \tag{4.7}$$

where $\mathbf{x}^s = \begin{bmatrix} \mathbf{x}_1^T & \mathbf{x}_2^T & \mathbf{x}_3^T & \mathbf{x}_4^T & \mathbf{x}_5^T & \mathbf{x}_6^T \end{bmatrix}^T$ collects the current nodal positions of triangle-centered stencil. The gradient of $\boldsymbol{\varepsilon}_{\bar{m}}$ can be computed

by

$$\frac{\partial \boldsymbol{\varepsilon}_{\bar{m}}}{\partial (\mathbf{x}^s)^T} = \frac{1}{3}\sum_{i=I}^{III} \frac{\partial \boldsymbol{\varepsilon}_m}{\partial (\mathbf{x}^s)^T}|_i, \tag{4.8}$$

where the gradient of $\boldsymbol{\varepsilon}_m$ is

$$\frac{\partial \boldsymbol{\varepsilon}_m}{\partial (\mathbf{x}^s)^T} = \begin{bmatrix} \mathbf{F}_X^T \frac{\partial \mathbf{F}_X}{\partial (\mathbf{x}^s)^T} \\ \mathbf{F}_Y^T \frac{\partial \mathbf{F}_Y}{\partial (\mathbf{x}^s)^T} \\ \mathbf{F}_Y^T \frac{\partial \mathbf{F}_X}{\partial (\mathbf{x}^s)^T} + \mathbf{F}_X^T \frac{\partial \mathbf{F}_Y}{\partial (\mathbf{x}^s)^T} \end{bmatrix} \tag{4.9}$$

with the gradients of deformation gradient components $\partial \mathbf{F}_X / \partial (\mathbf{x}^s)^T = \mathbf{B}_{\mathrm{F}_X}^T \otimes \mathbf{I}$ and $\partial \mathbf{F}_Y / \partial (\mathbf{x}^s)^T = \mathbf{B}_{\mathrm{F}_Y}^T \otimes \mathbf{I}$. Here, $\mathbf{B}_{\mathrm{F}_X}$ and $\mathbf{B}_{\mathrm{F}_Y}$ are the columns of

$$\begin{bmatrix} \mathbf{B}_{\mathrm{F}_X} & \mathbf{B}_{\mathrm{F}_Y} \end{bmatrix} = \begin{bmatrix} \frac{\partial N_1}{\partial X} & \frac{\partial N_1}{\partial Y} \\ \vdots & \vdots \\ \frac{\partial N_6}{\partial X} & \frac{\partial N_6}{\partial Y} \end{bmatrix} = \begin{bmatrix} \frac{\partial N_1}{\partial \xi} & \frac{\partial N_1}{\partial \eta} \\ \vdots & \vdots \\ \frac{\partial N_6}{\partial \xi} & \frac{\partial N_6}{\partial \eta} \end{bmatrix} \begin{bmatrix} \partial X/\partial \xi & \partial X/\partial \eta \\ \partial Y/\partial \xi & \partial Y/\partial \eta \end{bmatrix}^{-1}, \tag{4.10}$$

where $X = \sum_{i=1}^{6} N_i X_i$ and $Y = \sum_{i=1}^{6} N_i Y_i$ are the interpolated coordinates of interpolated initial position $\mathbf{X}$ in Eq.(4.3).

The Hessian of the membrane energy is

$$\frac{\partial^2 \Psi_m^s}{\partial \mathbf{x}^s \partial (\mathbf{x}^s)^T} = \frac{\partial \boldsymbol{\varepsilon}_m^T}{\partial \mathbf{x}^s} \mathbf{D}_m^T \frac{\partial \boldsymbol{\varepsilon}_m}{\partial (\mathbf{x}^s)^T} + \boldsymbol{\varepsilon}_m^T \mathbf{D}_m^T vec\left(\frac{\partial^2 \boldsymbol{\varepsilon}_m}{\partial \mathbf{x}^s \partial (\mathbf{x}^s)^T}\right), \tag{4.11}$$

where the vectorized Hessian of membrane strain is

$$vec\left(\frac{\partial^2 \boldsymbol{\varepsilon}_{\bar{m}}}{\partial \mathbf{x}^s \partial (\mathbf{x}^s)^T}\right) = \frac{1}{3}\sum_{i=I}^{III} vec\left(\frac{\partial^2 \boldsymbol{\varepsilon}_m}{\partial \mathbf{x}^s \partial (\mathbf{x}^s)^T}\right)|_i \tag{4.12}$$

with

$$vec\left(\frac{\partial^2 \boldsymbol{\varepsilon}_m}{\partial \mathbf{x}^s \partial (\mathbf{x}^s)^T}\right) = \left\{ \begin{array}{c} \frac{\partial \mathbf{F}_X^T}{\partial \mathbf{x}^s} \frac{\partial \mathbf{F}_X}{\partial (\mathbf{x}^s)^T} \\ \frac{\partial \mathbf{F}_Y^T}{\partial \mathbf{x}^s} \frac{\partial \mathbf{F}_Y}{(\mathbf{x}^s)^T} \\ \frac{\partial \mathbf{F}_Y^T}{\partial \mathbf{x}^s} \frac{\partial \mathbf{F}_X}{\partial (\mathbf{x}^s)^T} + \frac{\partial \mathbf{F}_X^T}{\partial \mathbf{x}^s} \frac{\partial \mathbf{F}_Y}{\partial (\mathbf{x}^s)^T} \end{array} \right\}. \tag{4.13}$$

It should be mentioned that the boundary triangles are conveniently adopted by the CST elements.

Node	X	Y	Z
1	-0.26272011	-0.08827895	0.00000000
2	-0.25765348	-0.06650603	0.00000000
3	-0.23353052	-0.09670550	0.00000000
4	-0.22901440	-0.06075549	0.00000000
5	-0.25724816	-0.11300337	0.00000000
6	-0.28291058	-0.06896079	0.00000000

Table 4.1: Coordinates of a triangle-centered stencil for the simple drape test in Section 4.2.1.

4.2 Numerical Tests

This section illustrates the efficacy of the smoothed-hinge membrane model in mitigating sharp creasing artifacts. All cases utilize the cotton material parameters from [100] and are implemented in the C-IPC codebase [100] with elastodynamic simulation. SI units are employed. The material properties of a 100% cotton fabric [100] are used: area density $\rho = 0.1503$, thickness $h = 3.18 \times 10^{-4}$, Young's modulus $E = 8.21 \times 10^5$, and Poisson's ratio $\nu = 0.243$. The gravitational acceleration is set to 9.81, and a friction coefficient of 0.4 is employed where applicable. For elastodynamic simulations, the time step is 0.04.

4.2.1 Quantitative Tests

A simple drape test

In this case, a triangle-centered stencil subjected to a downward uniform load is considered. The initial nodal coordinates of the stencil given in Table 4.1 are arbitrarily chosen for testing purposes. Nodes 1 to 5 are on the Z = 0 horizontal plane and only node 6 is free. To produce noticeable deformation, the magnitude of the load is set to 500 times of the weight. Four bending models, including the hinge-bending model, EP, FP, and SP, are combined with either the CST or the smoothed-hinge membrane model for comparative evaluation. As shown in Figure 4.2 and Figure 4.3, for all bending models, the maximum transverse displacement of node 6 is markedly smaller when coupled with the smoothed-hinge

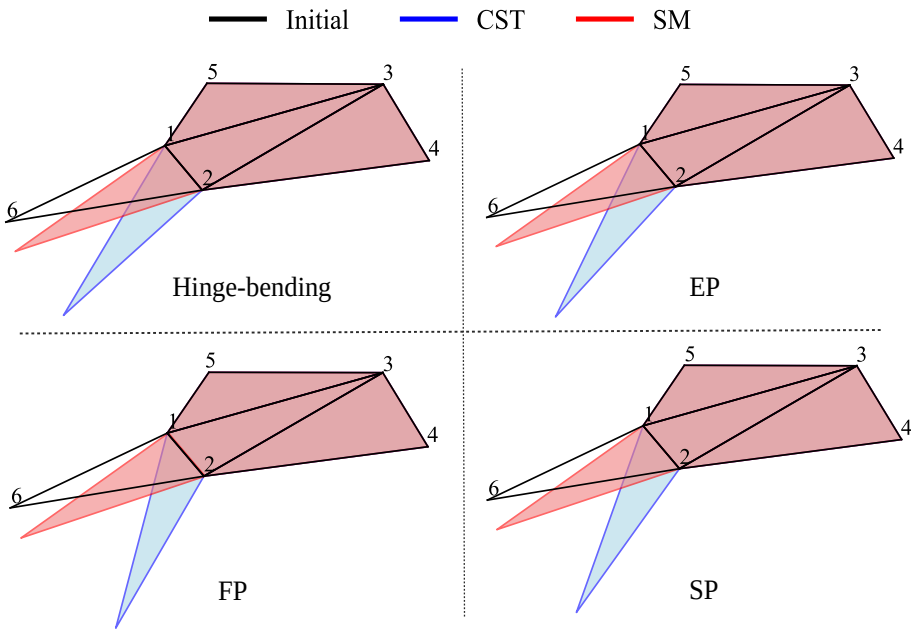


Figure 4.2: Comparison of four bending models (hinge-bending, EP, FP, and SP) combined with either the CST or smoothed hinge membrane (SM) model. In each subfigure, the black outline indicates the initially flat configuration, the red outline shows the deformation with the SM model, and the blue outline shows the deformation with the CST model. In all cases, the transverse displacement of node 6 is smaller when using the SM model is employed.

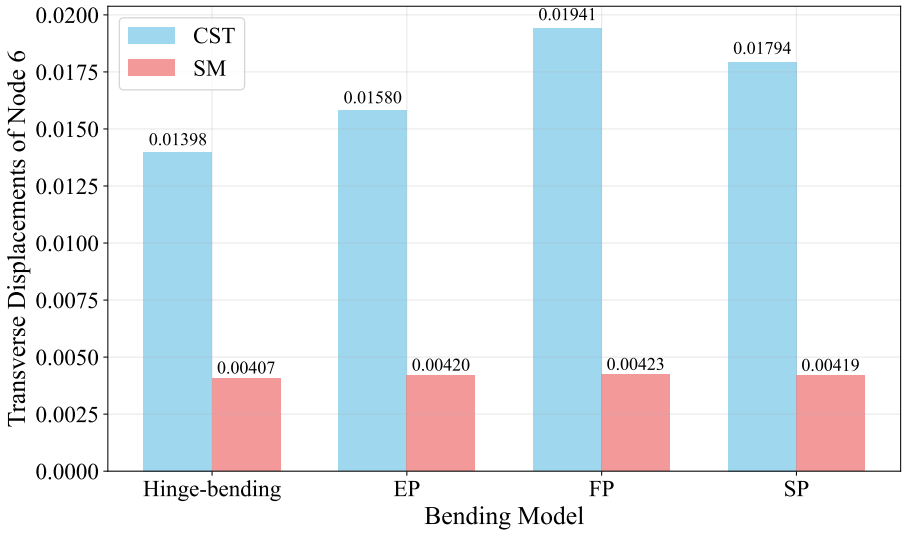


Figure 4.3: Transverse displacements of node 6, see Figure 4.2, predicted by different combinations of bending models (hinge-bending, EP, FP and SP) and membrane models (CST and SM).

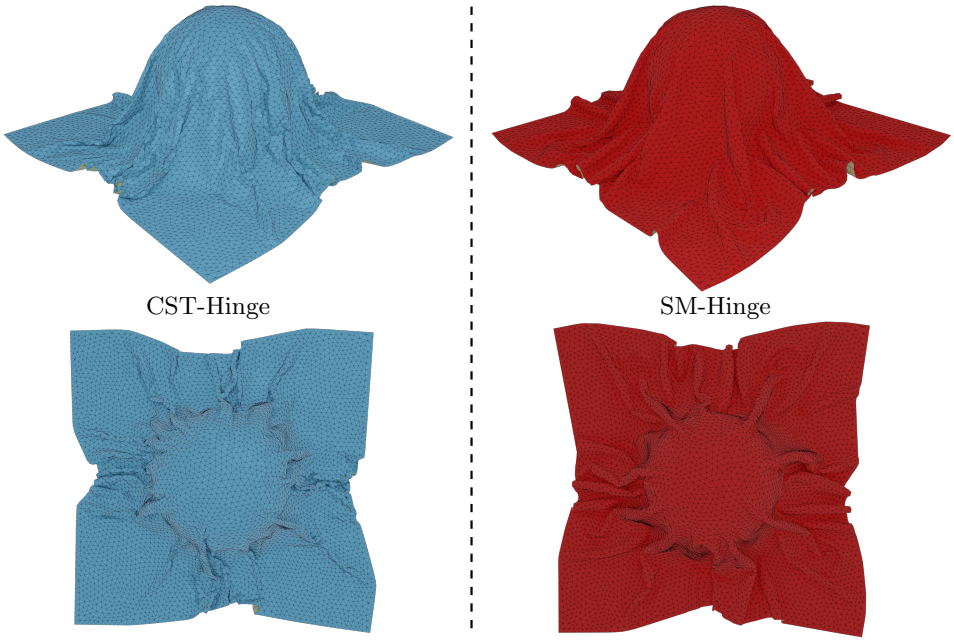


Figure 4.4: Draped cloth on a sphere fixed on the floor predicted by using the hinge-bending model combined with either the CST element or the smoothed hinge membrane (SM) model. The latter (right figure) effectively mitigates sharp creasing artifacts.

membrane model than with the CST membrane model. In the triangle-centered stencil, only triangle T_{126} can rotate about edge 12, while the other triangles are constrained. Under this condition, when T_{126} rotates, the CST membrane element produces no membrane energy in the central triangle T_{123}. In contrast, the smoothed-hinge membrane model does generate membrane energy in T_{123} as its interpolation involves six nodes across four triangles. Compared with the CST membrane element, this additional membrane energy in the smoothed-hinge model acts as an extra penalty that helps suppress excessive bending deformation between adjacent triangles. This observation provides a plausible explanation for the effectiveness of the smoothed-hinge membrane model in mitigating sharp creasing artifacts.

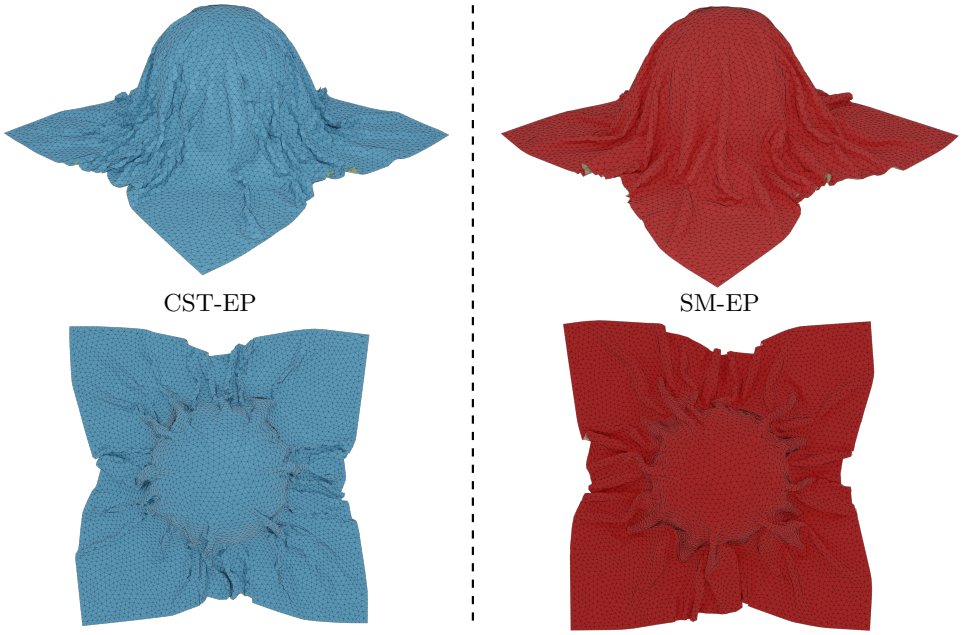


Figure 4.5: Draped cloth on a sphere fixed on the floor predicted by using the EP bending model combined with either the CST element or the smoothed hinge membrane (SM) model. The latter (right figure) effectively mitigates sharp creasing artifacts.

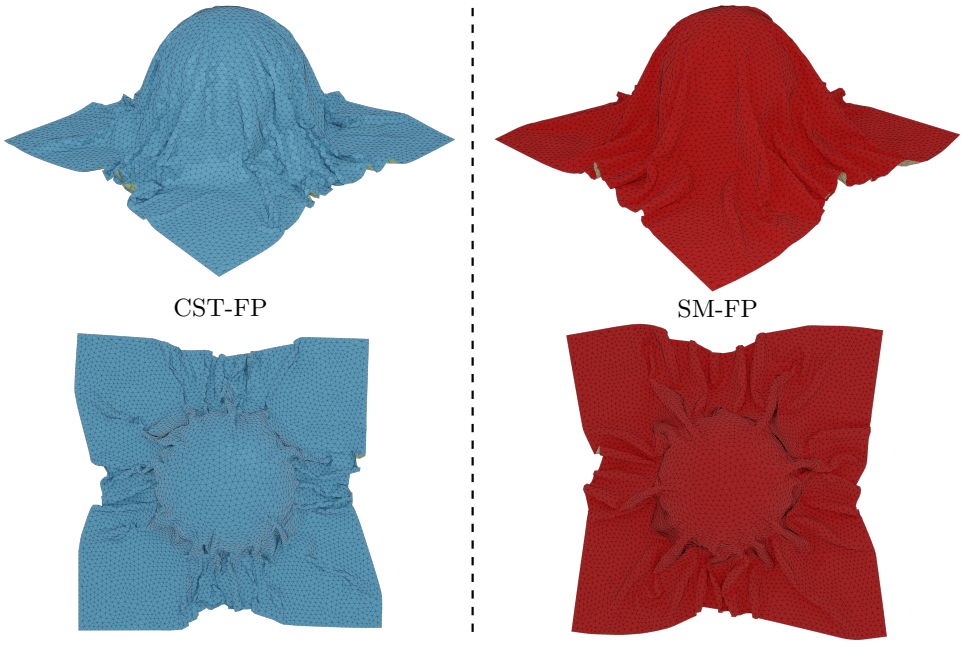


Figure 4.6: Draped cloth on a sphere fixed on the floor predicted by using the FP bending model combined with either the CST element or the smoothed hinge membrane (SM) model. The latter (right figure) effectively mitigates sharp creasing artifacts.

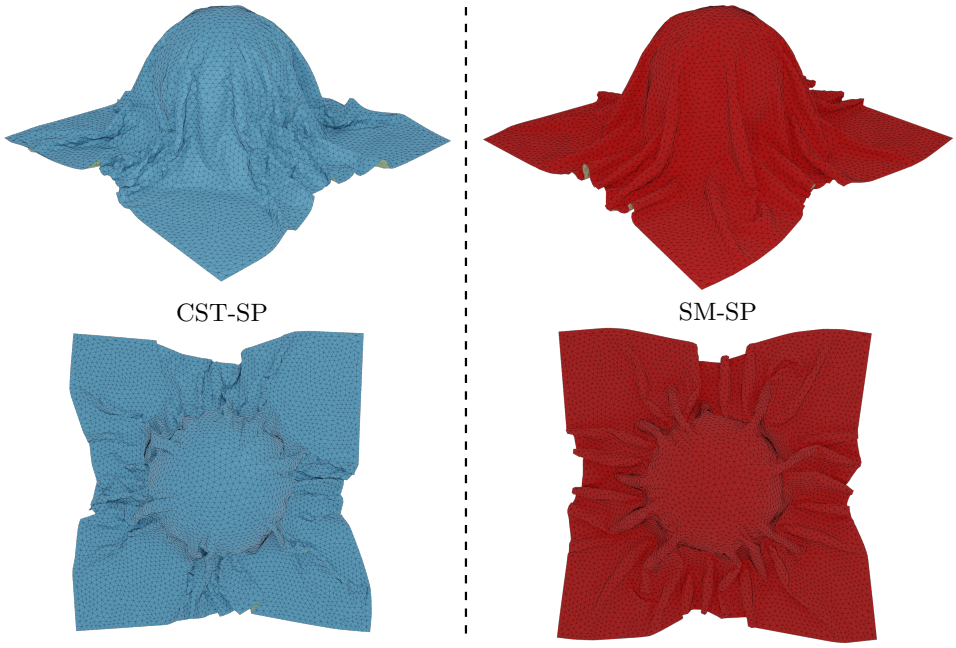


Figure 4.7: Draped cloth on a sphere fixed on the floor predicted by using the SP bending model combined with either the CST element or the smoothed hinge membrane (SM) model. The latter (right figure) effectively mitigates sharp creasing artifacts.

4.2.2 Qualitative Tests

Cloth draped on a fixed sphere on the floor

This test revisits the sharp creasing artifact reported by Li et al. [100], following the original setup: a 1×1 square cotton cloth draping over a $\phi 0.3$ sphere fixed on the floor. In their study, Li et al. used a computational model which combines the CST membrane element and the hinge-bending model proposed by Grinspun et al. [66]. When using a 3-node triangular mesh with 8,282 nodes, noticeable sharp creasing artifacts were observed. Reducing the resolution to 1,939 nodes resulted in a loss of wrinkling detail and more pronounced artifacts. Increasing the nodal population to 85,451 effectively alleviated the artifacts; however, this led to a significantly higher computational cost. Here, the mesh with 8,282 nodes is used. As shown in Figures 4.4, 4.5, 4.6, 4.7 and 4.8, noticeable sharp creasing artifacts appear when CST is combined with the hinge-bending, EP, FP, and SP models. In contrast, these artifacts are effectively mitigated when the bending models are coupled with

the smoothed-hinge membrane model. The reason for this improvement is explained in Section 4.2.1. This case further demonstrates that the smoothed-hinge membrane model provides an effective solution for suppressing sharp creasing artifacts in cloth simulation.

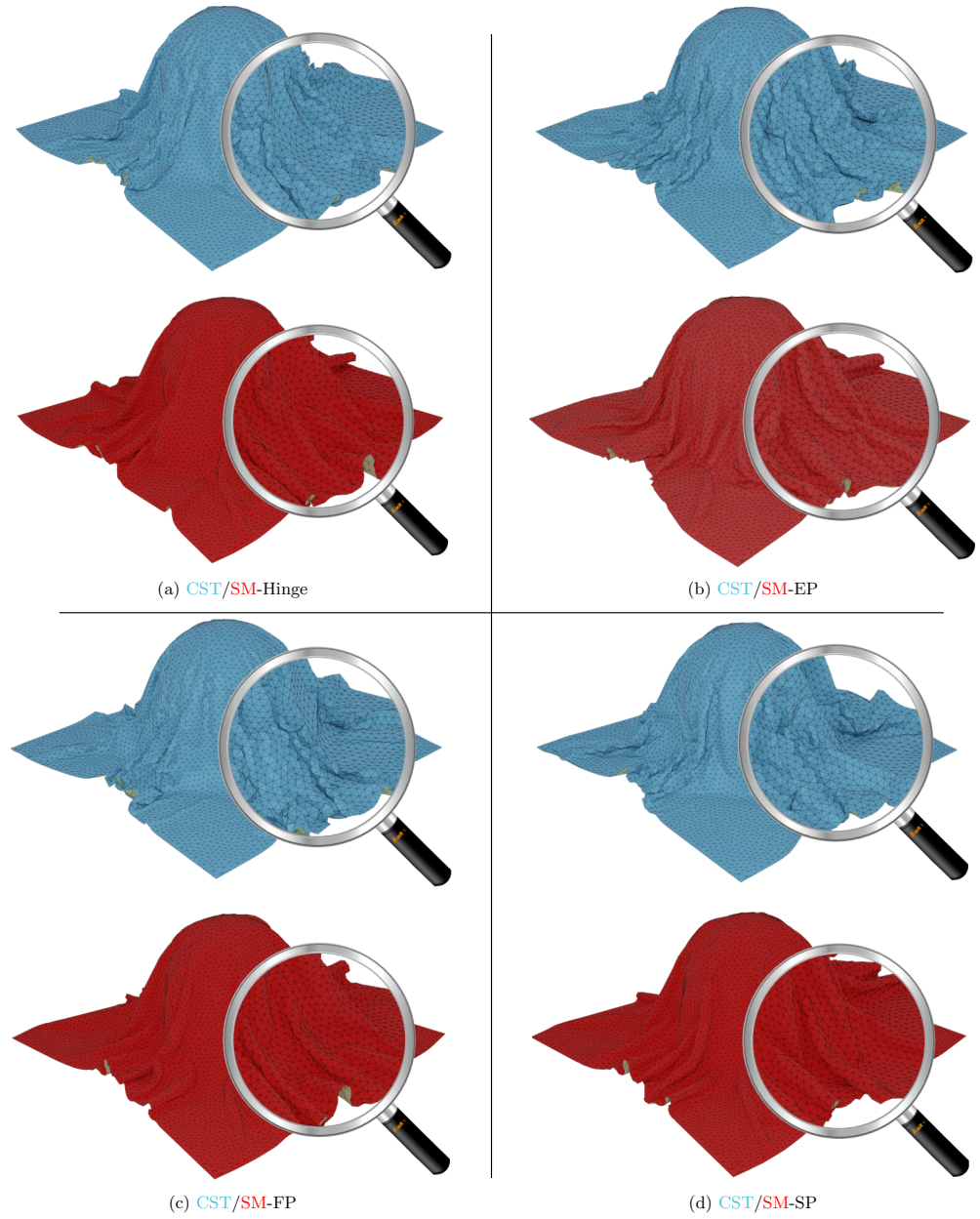
Garment simulation

As stitching patterns always begin as flat pieces as illustrated in Figure 4.9(a), the smoothed-hinge membrane model is well-suited for garment simulation. Only the results using the SP bending model are shown here, as visual differences among various bending models become difficult to discern when dense wrinkling occurs, as illustrated in Figures 4.4, 4.5, 4.6 and 4.7. After assembling the garment patterns, a snapshot in Figure 4.9(b) is taken from a dancing sequence vividly capturing the dynamic folds and intricate wrinkle details.

4.3 Conclusion

In this chapter, the formulation of the smoothed-hinge membrane model is first presented. This membrane model is then combined with the hinge-bending, EP, FP, and SP bending models introduced in previous chapters. Using the example of cloth draping over a fixed sphere on the floor, the model effectively demonstrates its ability to qualitatively mitigate sharp creasing artifacts in complex scenarios. Additionally, a simple drape test with a single triangle-centered stencil provides a quantitative explanation for suppressing artifacts. Finally, the model is applied to garment simulation, showcasing its ability in garment design. Overall, the proposed model enriches the library of high-fidelity computational tools for cloth simulation.

However, several limitations remain. The current formulation is specifically designed for initially flat configurations, which are typical in cloth simulation. Extending the model to more general shell simulations involving initially curved geometries is an important direction for future work. Moreover, addressing locking phenomena under initially curved geometries needs further investigation. Additionally, exploring



(a) CST/SM-Hinge (b) CST/SM-EP

(c) CST/SM-FP (d) CST/SM-SP

Figure 4.8: Zoom-in views illustrate that the smoothed-hinge membrane (SM) model effectively mitigates sharp creasing artifacts compared with the CST membrane element.

(a) Staged garment patterns **(b)** Back view of a character

Figure 4.9: (a) Garment patterns are staged on a mannequin. (b) After stitching, the garment details are vividly simulated in a dancing sequence.

alternative strain interpolation schemes may also offer further improvements in mitigating sharp creasing artifacts. When comparing the same drape problem simulated with CST and SM, the non-physical sharp folds largely disappear with SM, and the remaining wrinkles match the daily perception of draped cloth more closely. However, the qualitative tests suggest that the SM model may introduce additional membrane stiffness compared to the CST element, resulting in fewer wrinkles overall. A real-world experimental solution would provide a more definitive benchmark; however, since draped cloth exhibits multiple stable equilibrium configurations, establishing a reproducible and unambiguous ground-truth configuration for comparison remains an open challenge.

Chapter 5

Modeling and Simulating Origami Structures using Solid-Shell Elements

5.1 Introduction

Origami, rooted in the ancient art of paper folding, has evolved into a multidisciplinary field of science and engineering. As origami structures have transitioned from rigid panels to deformable panels exhibiting complex multi-physics responses, there is a growing need for simulation techniques that can accurately predict both their geometric and physical behaviors.

Origami simulation methods can be broadly classified into kinematics-based and mechanics-based ones. Kinematics-based methods [181] assume the panels are rigid, enabling folding angles only as variables to describe deformation. These methods are computationally efficient but cannot consider panel deformations. Mechanics-based methods relax the rigidity assumption, allowing membrane and bending deformations of the panel. Similar to the mass–spring model in cloth simulation [2, 20], the widely used bar-hinge model [157, 108] suffers from mesh-dependent issues in which material parameters vary with mesh density and are not easily transferable across different mesh topologies. Discrete shell models [66, 23] improve membrane accuracy of the origami panels but still

exhibit mesh-dependent bending behavior. Other discrete and ruling-based models [150, 165] impose strict isometry constraints, making them difficult to consider physical constitutive laws. In [72], the warping energy of the geometric nonlinear three-dimensional membrane element is derived using a corotational approach.

In parallel, solid-shell elements were originally developed as three-dimensional continuum-based elements for plate and shell analysis. Like the aforementioned discrete shell models, these elements do not require nodal rotational degrees of freedom and are capable of handling larger load or displacement increments compared to conventional finite shell elements [170].

Motivated by these developments, this chapter presents a novel computational framework for origami simulation that employs quadrilateral solid-shell elements. The main developments are as follows:

- Bilinear and biquadratic geometric nonlinear quadrilateral solid-shell elements, extended from [171], are developed. The various locking phenomena in the elements are effectively alleviated by the assumed natural strain (ANS) method.
- A crease model is proposed based on the director vectors of the solid-shell elements.

This chapter is based on the results originally reported in [104], with additional analysis and discussion included here.

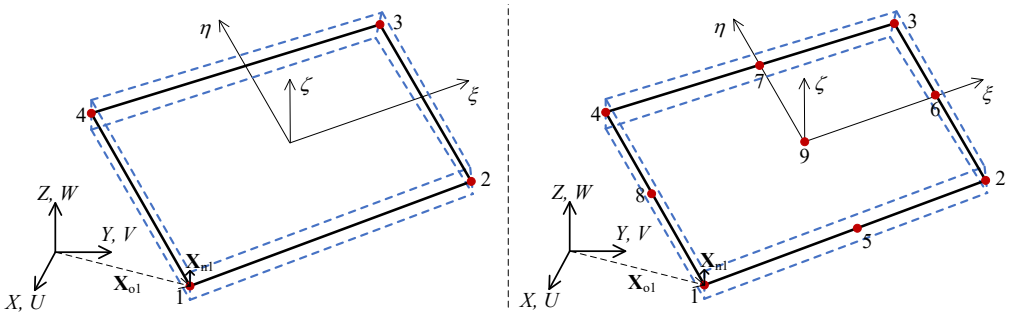


Figure 5.1: The bilinear (4-node) and biquadratic (9-node) solid-shell elements. Nodes are located on the mid-surface.

5.2 Computational Models for Origami Simulation

5.2.1 Solid-Shell Element

Consider a quadrilateral solid-shell element with nodes on its mid-surface (see Figure 5.1), the interpolated initial position vector $\mathbf{X}$ and displacement vector $\mathbf{U}$ can be written as:

$$\begin{aligned}\mathbf{X} &= \mathbf{X}_o + \zeta \mathbf{X}_n = \sum_i N_i(\xi,\eta)\,\mathbf{X}_{oi} + \zeta \sum_i N_i(\xi,\eta)\,\mathbf{X}_{ni},\\ \mathbf{U} &= \mathbf{U}_o + \zeta \mathbf{U}_n = \sum_i N_i(\xi,\eta)\,\mathbf{U}_{oi} + \zeta \sum_i N_i(\xi,\eta)\,\mathbf{U}_{ni},\end{aligned} \tag{5.1}$$

where the summation is over the element nodes; N_i is the interpolation function of the i-th node; $\mathbf{X}_o = [X_o, Y_o, Z_o]^T$ and $\mathbf{X}_n = [X_n, Y_n, Z_n]^T$ are the initial mid-surface position and director vectors, respectively, $\mathbf{U}_o = [U_o, V_o, W_o]^T$ and $\mathbf{U}_n = [U_n, V_n, W_n]^T$ are the mid-surface and director displacement vectors, respectively. For the bilinear element,

$$\begin{aligned}N_1(\xi,\eta) &= \frac{1}{4}(1-\xi)(1-\eta), N_2(\xi,\eta) = \frac{1}{4}(1+\xi)(1-\eta),\\ N_3(\xi,\eta) &= \frac{1}{4}(1+\xi)(1+\eta), N_4(\xi,\eta) = \frac{1}{4}(1-\xi)(1+\eta).\end{aligned} \tag{5.2}$$

For the bi-qudratic element,

$$
\begin{aligned}
N_1(\xi,\eta) &= \frac{1}{4}(\xi-\xi^2)(\eta-\eta^2), N_2(\xi,\eta) = -\frac{1}{4}(\xi+\xi^2)(\eta-\eta^2),\\
N_3(\xi,\eta) &= \frac{1}{4}(\xi+\xi^2)(\eta+\eta^2), N_4(\xi,\eta) = -\frac{1}{4}(\xi-\xi^2)(\eta+\eta^2),\\
N_5(\xi,\eta) &= -\frac{1}{2}(1-\xi^2)(\eta-\eta^2), N_6(\xi,\eta) = \frac{1}{2}(\xi+\xi^2)(1-\eta^2),\\
N_7(\xi,\eta) &= \frac{1}{2}(1-\xi^2)(\eta+\eta^2), N_8(\xi,\eta) = -\frac{1}{2}(\xi-\xi^2)(1-\eta^2),\\
N_9(\xi,\eta) &= (1-\xi^2)(1-\eta^2).
\end{aligned}
\tag{5.3}
$$

In general, the initial nodal director $\mathbf{X}_{ni}$ is specified to be the vector perpendicular to the actual initial mid-surface which may be different from the interpolated initial mid-surface. Apparently, there are six dofs per node in the mid-suface from $\mathbf{U}_{oi}$ and $\mathbf{U}_{ni}$. The natural or parametric coordinates (ξ,η,ζ) are bounded by -1 and 1. The membrane, bending, transverse shear and thickness strain components with respect to the natural coordinates are

$$
\begin{aligned}
\varepsilon_{m\alpha\beta} &= \frac{1}{2}\big(\mathbf{X}_{o,\alpha}^T\mathbf{U}_{o,\beta} + \mathbf{X}_{o,\beta}^T\mathbf{U}_{o,\alpha} + \mathbf{U}_{o,\alpha}^T\mathbf{U}_{o,\beta}\big),\\
\varepsilon_{b\alpha\beta} &= \frac{1}{2}\big(\mathbf{X}_{n,\alpha}^T\mathbf{U}_{o,\beta} + \mathbf{X}_{o,\alpha}^T\mathbf{U}_{n,\beta} + \mathbf{X}_{n,\beta}^T\mathbf{U}_{o,\alpha} + \mathbf{X}_{o,\beta}^T\mathbf{U}_{n,\alpha}\\
&\qquad + \mathbf{U}_{n,\alpha}^T\mathbf{U}_{o,\beta} + \mathbf{U}_{o,\alpha}^T\mathbf{U}_{n,\beta}\big),\\
\gamma_{\zeta\alpha} &= \mathbf{X}_n^T\mathbf{U}_{o,\alpha} + \mathbf{X}_{o,\alpha}^T\mathbf{U}_n + \mathbf{U}_n^T\mathbf{U}_{o,\alpha}, \varepsilon_{\zeta\zeta} = \mathbf{X}_n^T\mathbf{U}_n + \frac{1}{2}\mathbf{U}_n^T\mathbf{U}_n,
\end{aligned}
\tag{5.4}
$$

where $\alpha,\beta = \xi,\eta$. They are often termed as natural strains. These strain components are derived from Eqs.(1.8), (1.24), and (1.25) in Chapter 1; the derivation details are provided in Appendix B.

With respect to the local Cartesian coordinates $(\tilde{X},\tilde{Y},\tilde{Z})$ with the $\tilde{X}$-$\tilde{Y}$-plane tangential to the initial mid-surface, the strain transformation

relations can be expressed as:

$$\boldsymbol{\varepsilon}_m = \begin{Bmatrix} \varepsilon_{m\tilde{X}\tilde{X}} \\ \varepsilon_{m\tilde{Y}\tilde{Y}} \\ 2\varepsilon_{m\tilde{X}\tilde{Y}} \end{Bmatrix} = \mathbf{T}_{mb} \begin{Bmatrix} \varepsilon_{m\xi\xi} \\ \varepsilon_{m\eta\eta} \\ 2\varepsilon_{m\xi\eta} \end{Bmatrix}, \quad \boldsymbol{\varepsilon}_b = \begin{Bmatrix} \varepsilon_{b\tilde{X}\tilde{X}} \\ \varepsilon_{b\tilde{Y}\tilde{Y}} \\ 2\varepsilon_{b\tilde{X}\tilde{Y}} \end{Bmatrix} = \mathbf{T}_{mb} \begin{Bmatrix} \varepsilon_{b\xi\xi} \\ \varepsilon_{b\eta\eta} \\ 2\varepsilon_{b\xi\eta} \end{Bmatrix},$$
$$\boldsymbol{\gamma} = \begin{Bmatrix} \gamma_{zx} \\ \gamma_{zy} \end{Bmatrix} = \mathbf{T}_s \begin{Bmatrix} \gamma_{\zeta\xi} \\ \gamma_{\zeta\eta} \end{Bmatrix}, \quad \varepsilon_{zz} = T_t \varepsilon_{\zeta\zeta} \tag{5.5}$$

in which $\mathbf{T}_{mb}$, $\mathbf{T}_s$ and T_t are derived in Appendix C. The above transformation relations are exact when ζ and $\tilde{Z}$ are parallel. Following Eq.(1.40), the element strain energy can be written as

$$\Psi_{ss}^e = \frac{1}{2} \int_{-1}^{1} \int_{-1}^{1} (\boldsymbol{\varepsilon}_m^T \mathbf{D}_m \boldsymbol{\varepsilon}_m + \boldsymbol{\varepsilon}_b^T \mathbf{D}_b \boldsymbol{\varepsilon}_b + \boldsymbol{\gamma}^T \mathbf{D}_s \boldsymbol{\gamma} + D_t \varepsilon_{zz}^2)\, J_o d\xi d\eta, \tag{5.6}$$

where the Jacobain determinant

$$J = \begin{bmatrix} \frac{\partial \mathbf{X}}{\partial \xi} & \frac{\partial \mathbf{X}}{\partial \eta} & \frac{\partial \mathbf{X}}{\partial \zeta} \end{bmatrix} \tag{5.7}$$

for global Cartesian coordinates (X, Y, Z) and (ξ, η, ζ) is approximated by $J_o = J|_{\zeta=0}$. The stiffness matrices $\mathbf{D}_m$, $\mathbf{D}_b$, $\mathbf{D}_s$ and D_t are defined in Eq.(1.20) and Eq.(1.38). Full numerical integration over the ξ–η plane is performed using 2×2 and 3×3 Gaussian quadrature schemes for the bilinear and biquadratic elements, respectively.

It is well-known that solid-shell elements are prone to membrane, shear and thickness lockings due to the excessive enforcements of the zero membrane, transverse shear and thickness strains in thin shells. While membrane locking can be neglected in lower-order elements such as the present bilinear one, it is alleviated in the biquadratic element through the ANS method [173]. For both bilinear and biquadratic elements, shear and trapezoidal locking can be mitigated by ANS in which the relevant natural strain components are interpolated at judiciously

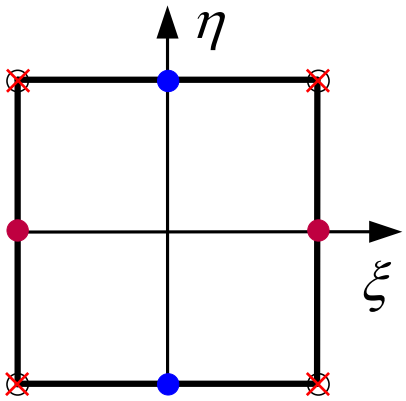


Figure 5.2: The sampling points of the bilinear solid-shell element. ○ denotes the node, × denotes the sampling point for $\varepsilon_{\zeta\zeta}^{ANS}$, • and • denote the sampling points for $\gamma_{\zeta\xi}^{ANS}$ and $\gamma_{\zeta\eta}^{ANS}$, respectively.

selected sampling points. The sampling scheme for the bilinear element [42, 171] is

$$\begin{aligned}
\gamma_{\zeta\xi}^{ANS} &= \frac{(1-\eta)}{2}\,\gamma_{\zeta\xi}\Big|_{\xi=0,\,\eta=-1} + \frac{(1+\eta)}{2}\,\gamma_{\zeta\xi}\Big|_{\xi=0,\,\eta=+1}, \\
\gamma_{\zeta\eta}^{ANS} &= \frac{(1-\xi)}{2}\,\gamma_{\zeta\eta}\Big|_{\xi=-1,\,\eta=0} + \frac{(1+\xi)}{2}\,\gamma_{\zeta\eta}\Big|_{\xi=+1,\,\eta=0}, \\
\varepsilon_{\zeta\zeta}^{ANS} &= \sum_{i=1}^{4} N_i(\xi,\eta)\,(\varepsilon_{\zeta\zeta})_i.
\end{aligned} \tag{5.8}$$

The sampling scheme of the biquadratic element follows the quadratic case of the generic assumed natural strain formulation proposed in [173] in which assumed natural strain and stabilized Lobatto-Lagrange C^0 plate/shell elements of order ≥ 2 were presented. The assumed in-plane strain components are

$$\begin{aligned}
\varepsilon_{\xi\xi}^{ANS} &= \sum_{m=1}^{2}\sum_{n=1}^{3} L_{rd}^{(m)}(\xi) L_{fl}^{(n)}(\eta)\varepsilon_{\xi\xi}|_{\xi=r^{(m)},\eta=s^{(m)}}, \\
\varepsilon_{\eta\eta}^{ANS} &= \sum_{n=1}^{3}\sum_{m=1}^{2} L_{fl}^{(n)}(\xi) L_{rd}^{(m)}(\eta)\varepsilon_{\eta\eta}|_{\xi=r^{(n)},\eta=s^{(m)}}, \\
\varepsilon_{\xi\eta}^{ANS} &= \sum_{m=1}^{2}\sum_{n=1}^{2} L_{rd}^{(m)}(\xi) L_{rd}^{(n)}(\eta)\varepsilon_{\xi\eta}|_{\xi=r^{(m)},\eta=s^{(m)}},
\end{aligned} \tag{5.9}$$

where the subscript rd and fl distinguish the reduced and full Lagrange interpolation functions, respectively, defined along a single natural coordinate direction. The assumed transverse strain components are

$$\begin{aligned}\gamma_{\zeta\xi}^{ANS} &= \sum_{m=1}^{2}\sum_{n=1}^{3} L_{rd}^{(m)}(\xi)L_{fl}^{(n)}(\eta)\gamma_{\zeta\xi}|_{\xi=r^{(m)},\eta=s^{(m)}},\\ \gamma_{\zeta\eta}^{ANS} &= \sum_{n=1}^{3}\sum_{m=1}^{2} L_{fl}^{(n)}(\xi)L_{rd}^{(m)}(\eta)\gamma_{\zeta\eta}|_{\xi=r^{(n)},\eta=s^{(m)}},\\ \varepsilon_{\zeta\zeta}^{ANS} &= \sum_{i=1}^{9} N_i(\xi,\eta)\,(\varepsilon_{\zeta\zeta})_i.\end{aligned} \tag{5.10}$$

Here, N_i and $(\varepsilon_{\zeta\zeta})_i$ denote the interpolation function of and natural thickness strain at the i-th element node, respectively. The superscripts (m) and (n) label the sample coordinates. In addition, the Lagrangian interpolation functions are

$$L_{rd}^{(i)}(\tau) = \begin{cases}\frac{1-\sqrt{3}\tau}{2}, & i=1\\ \frac{1+\sqrt{3}\tau}{2}, & i=2\end{cases}, L_{fl}^{(j)}(\tau) = \begin{cases}\frac{1}{2}\tau(\tau-1), & j=1\\ 1-\tau^2, & j=2\\ \frac{1}{2}\tau(\tau+1), & j=3\end{cases}, \tag{5.11}$$

and the sampling coordinates are

$$\tau^{(i)} = \begin{cases}-\frac{1}{\sqrt{3}}, & i=1\\ \frac{1}{\sqrt{3}}, & i=2\end{cases}, \tau^{(j)} = \begin{cases}-1, & j=1\\ 0, & j=2\\ 1, & j=3\end{cases}. \tag{5.12}$$

Here, $\tau = r, s$ and $i, j = m, n$.

The assumed natural strains in Eq. (5.8) for the bilinear solid-shell element, and in Eqs. (5.9) and (5.10) for the biquadratic solid-shell element, are respectively substituted into Eq. (5.5) to replace the natural strains, thereby yielding the element strain energy Ψ_{ss}^{e} in Eq. (5.6). In the nonlinear solution procedure, gradient and Hessian of Ψ_{ss}^{e} with respect to the element vector of nodal dofs are required. They are presented in Appendix H.

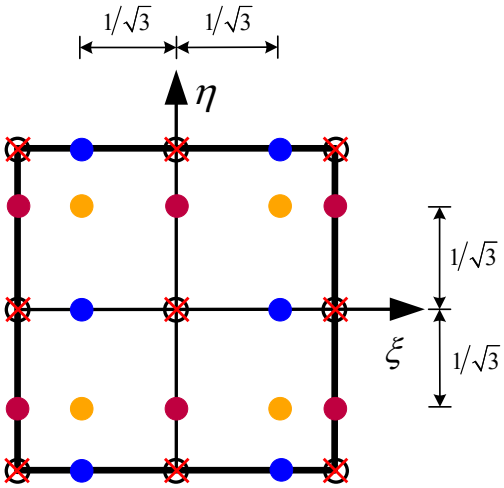


Figure 5.3: The sampling points of the biquadratic solid-shell element. ∘ denotes the node, × denotes the sampling point for $\varepsilon^{ANS}_{\zeta\zeta}$, • denotes the sampling points for $\varepsilon^{ANS}_{\xi\xi}$ and $\gamma^{ANS}_{\zeta\xi}$. • denotes the sampling points for $\varepsilon^{ANS}_{\eta\eta}$ and $\gamma^{ANS}_{\zeta\eta}$. • denotes the sampling points for $\varepsilon^{ANS}_{\zeta\zeta}$.

5.2.2 Crease Model

The origami structure is formed by folding a flat sheet of material. Thus, the initial mid-surface position vector $\mathbf{X}_o$ and director vector $\mathbf{X}_n$ along the common crease between elements "a" and "b" (refer to Figure 5.4) meet the conditions $\mathbf{X}^T_{o,s}\mathbf{X}_n = 0$ and $\mathbf{X}^a_n = \mathbf{X}^b_n = \mathbf{X}_n$ where $s \in [-1, 1]$ is the non-dimensional coordinate along the crease. For the two elements, the transverse shear strains along the crease are

$$\gamma^a_{\zeta s} = (\mathbf{X}_o + \mathbf{U}_o)^T_{,s}(\mathbf{X}^a_n + \mathbf{U}^a_n), \gamma^b_{\zeta s} = (\mathbf{X}_o + \mathbf{U}_o)^T_{,s}(\mathbf{X}^b_n + \mathbf{U}^b_n). \tag{5.13}$$

which are valid only when the orthogonality condition $\mathbf{X}^T_{o,s}\mathbf{X}_n = 0$ is assumed. The penalty enforcement of the zero shear strain implies that the components of displaced directors, i.e. $\mathbf{X}^a_n + \mathbf{U}^a_n$ and $\mathbf{X}^b_n + \mathbf{U}^b_n$ along the running direction of the displaced creased $(\mathbf{X}_o + \mathbf{U}_o)_{,s}$, vanish approximately. Let

$$\mathbf{p} = \mathbf{X}^a_n + \mathbf{U}^a_n, \mathbf{q} = \mathbf{X}^b_n + \mathbf{U}^b_n. \tag{5.14}$$

The penalty enforcements of zero thickness and transverse shear strain imply that angle between $\mathbf{p}$ and $\mathbf{q}$ can be used approximately as the fold angle θ of the crease (see Figure 5.4). When the assumed natural strain

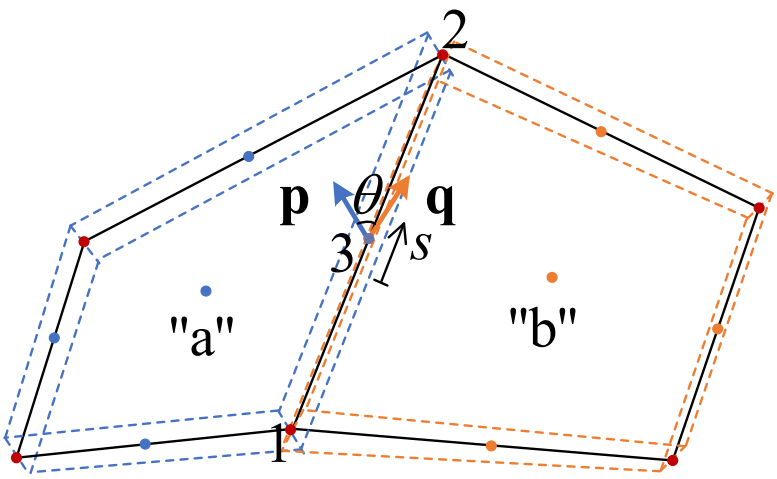


Figure 5.4: Solid-shell elements at two sides of a crease. The fold angle θ is the angle between the directors $\mathbf{p}$ and $\mathbf{q}$ of elements "a" and "b".

method is used to sample the natural transverse shear strain, the approximation is expected to be most accurate at the relevant strain sampling point.

In terms of the non-dimensional coordinate s along the crease, the interpolated $\mathbf{X}_o$, $\mathbf{U}_o$, $\mathbf{X}_n^a$, $\mathbf{X}_n^b$, $\mathbf{U}_n^a$, and $\mathbf{U}_n^b$ can be expressed as

$$(\mathbf{X}_o, \mathbf{U}_o, \mathbf{X}_n^a, \mathbf{X}_n^b, \mathbf{U}_n^a, \mathbf{U}_n^b) = \sum_i L_i(\mathbf{X}_{oi}, \mathbf{U}_{oi}, \mathbf{X}_{ni}^a, \mathbf{X}_{ni}^b, \mathbf{U}_{ni}^a, \mathbf{U}_{ni}^b), \tag{5.15}$$

where the interpolation functions L_is for the bilinear element are

$$L_1 = \frac{1-s}{2}, \quad L_2 = \frac{1+s}{2}, \tag{5.16}$$

and for the biquadratic element are

$$L_1 = \frac{1}{2}s(s-1), \quad L_2 = \frac{1}{2}s(s+1), \quad L_3 = 1-s^2. \tag{5.17}$$

Although $\mathbf{X}_o$ and $\mathbf{U}_o$ are common to both elements, $\mathbf{p}$ and $\mathbf{q}$ can vary independently. The folding at the crease can be characterized by the fold angle

$$\theta = \begin{cases} \cos^{-1}\frac{\mathbf{p}\cdot\mathbf{q}}{pq} & \text{for } (\mathbf{p}\times\mathbf{q})\cdot(\mathbf{x}_{o2}-\mathbf{x}_{o1}) \geq 0 \\ -\cos^{-1}\frac{\mathbf{p}\cdot\mathbf{q}}{pq} & \text{otherwise} \end{cases} \tag{5.18}$$

where $p = \|\mathbf{p}\|$ and $q = \|\mathbf{q}\|$. In the case that the two elements are flat, θ ranges from $[-\pi, \pi]$ with $\theta = \pm\pi$ indicating that the elements are fully folded.

To prevent the physically inadmissible self-intersecting configuration of the fully folded Miura-ori case in Section 5.3.1, the following folding energy

$$\Psi_f^e = l \begin{cases} \frac{1}{2}k_f(\theta_0 - \theta_L)^2 + k_f(\theta_0 - \theta_L)(\theta_L - \theta) \\ \quad - \frac{4k_f(\theta_L + \pi)^2}{\pi^2} \ln\left|\cos\left(\frac{\pi(\theta_L - \theta)}{2(\theta_L + \pi)}\right)\right| & -\pi < \theta < \theta_L \\ \frac{1}{2}k_f(\theta - \theta_0)^2 & \theta_L \le \theta \le \theta_R \\ \frac{1}{2}k_f(\theta_R - \theta_0)^2 + k_f(\theta_R - \theta_0)(\theta - \theta_R) \\ \quad - \frac{4k_f(\pi - \theta_R)^2}{\pi^2} \ln\left|\cos\left(\frac{\pi(\theta - \theta_R)}{2(\pi - \theta_R)}\right)\right| & \theta_R < \theta < \pi \end{cases} \tag{5.19}$$

is adopted. Here, k_f is the fold stiffness per unit length of the crease [108], and $l = \int_{-1}^{+1} |\mathbf{X}_{,s}| ds$ is the crease length, which is evaluated using one-point Gaussian integration for the bilinear element and two-point Gaussian integration for the biquadratic element. The integration points echoes the ANS sampling points. Within the interval $[\theta_L, \theta_R]$, the energy exhibits a standard quadratic behavior centered at the rest angle θ_0. Outside this range, the logarithmic term grows rapidly to impose strong penalty on configurations approaching $\theta = \pm\pi$, thereby effectively preventing self-intersection (see Figure 5.5).

The nodal dofs vector associated with the linearly interpolated crease is

$$\mathbf{d}_c = \begin{bmatrix} (\mathbf{U}_{n1}^a)^T & (\mathbf{U}_{n1}^b)^T & (\mathbf{U}_{n2}^a)^T & (\mathbf{U}_{n2}^b)^T \end{bmatrix}^T, \tag{5.20}$$

whereas for the quadratically interpolated crease is

$$\mathbf{d}_c = \begin{bmatrix} (\mathbf{U}_{n1}^a)^T & (\mathbf{U}_{n1}^b)^T & (\mathbf{U}_{n2}^a)^T & (\mathbf{U}_{n2}^b)^T & (\mathbf{U}_{n3}^a)^T & (\mathbf{U}_{n3}^b)^T \end{bmatrix}^T. \tag{5.21}$$

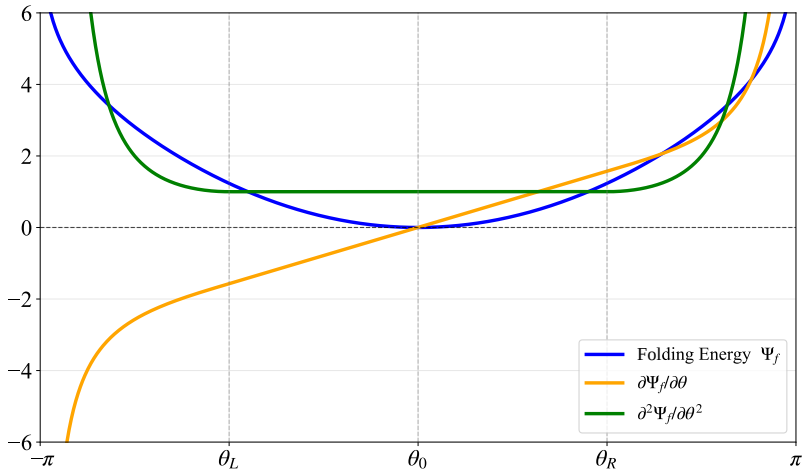


Figure 5.5: Nonlinear behavior of the folding energy. The limit angles θ_L and θ_R are tunable.

From the chain rule and the interpolation in Eq.(5.13),

$$\frac{\partial}{\partial \mathbf{U}^a_{ni}} = \frac{\partial (\mathbf{U}^a_n)^T}{\partial \mathbf{U}^a_{ni}} \frac{\partial}{\partial \mathbf{U}^a_n} = L_i \frac{\partial}{\partial \mathbf{U}^a_n}, \quad \frac{\partial}{\partial \mathbf{U}^b_{ni}} = \frac{\partial (\mathbf{U}^b_n)^T}{\partial \mathbf{U}^b_{ni}} \frac{\partial}{\partial \mathbf{U}^b_n} = L_i \frac{\partial}{\partial \mathbf{U}^b_n}, \tag{5.22}$$

the gradient of the fold angle with respect to $\mathbf{U}^a_n$ and $\mathbf{U}^b_n$ are:

$$\frac{\partial \theta}{\partial \mathbf{U}^a_n} = \frac{\hat{\mathbf{p}} \times \hat{\mathbf{r}}}{p}, \frac{\partial \theta}{\partial \mathbf{U}^b_n} = \frac{\hat{\mathbf{r}} \times \hat{\mathbf{q}}}{q}, \tag{5.23}$$

where $\hat{\mathbf{p}} = \mathbf{p}/p$, $\hat{\mathbf{q}} = \mathbf{q}/p$ and $\hat{\mathbf{r}} = \hat{\mathbf{p}} \times \hat{\mathbf{q}}/\sin\theta$. The derivation of the fold-angle gradient follows the same procedure as that of the edge-stenciled hinge model in Section 2.2.4. The distinction between the two arises from the definitions of $\mathbf{p}$ and $\mathbf{q}$ in 2.14 and 5.14. Similar to Eq.(2.17), one can get

$$p^3 q \sin\theta \frac{\partial \theta}{\partial \mathbf{U}^a_n} = (\mathbf{p} \cdot \mathbf{q})\mathbf{p} - p^2 \mathbf{q}. \tag{5.24}$$

By taking the derivative of both sides of Eq.(5.24) with respect to $(\mathbf{U}^a_n)^T$, the left-hand side of Eq.(5.24) becomes

$$\frac{\partial \left(p^3 q \sin\theta \frac{\partial \theta}{\partial \mathbf{U}^a_n} \right)}{\partial (\mathbf{U}^a_n)^T} = p^3 q \sin\theta \frac{\partial^2 \theta}{\partial (\mathbf{U}^a_n)^T \partial \mathbf{U}^a_n} + \frac{\partial \theta}{\partial \mathbf{U}^a_n} \frac{\partial \left(p^3 q \sin\theta \right)}{\partial (\mathbf{U}^a_n)^T} \tag{5.25}$$

with

$$\begin{aligned}\frac{\partial\left(p^3 q\sin\theta\right)}{\partial\left(\mathbf{U}_n^a\right)^T} &= \frac{\partial\left(p^3\right)}{\partial\left(\mathbf{U}_n^a\right)^T}\cdot q\sin\theta + p^3\frac{\partial q}{\partial\left(\mathbf{U}_n^a\right)^T}\cdot\sin\theta + p^3 q\frac{\partial(\sin\theta)}{\partial\left(\mathbf{U}_n^a\right)^T}, \\ &= 3p^2\mathbf{p}^T q\sin\theta + p^3 q\cos\theta\cdot\frac{\partial\theta}{\partial\left(\mathbf{U}_n^a\right)^T}\end{aligned} \tag{5.26}$$

and the right-hand side in Eq. (5.24) becomes

$$\frac{\partial[(\mathbf{p}\cdot\mathbf{q})\mathbf{p}]}{\partial\left(\mathbf{U}_n^a\right)^T} - \frac{\partial\left(p^2\mathbf{q}\right)}{\partial\left(\mathbf{U}_n^a\right)^T} = (\mathbf{p}\cdot\mathbf{q})\mathbf{I} + \mathbf{p}\mathbf{q}^T - 2\mathbf{q}\mathbf{p}^T. \tag{5.27}$$

Substituting Eqs.(5.25)-(5.27) into Eq. (5.24), one can obtain

$$\frac{\partial^2\theta}{\partial\left(\mathbf{u}_n^a\right)^T\partial\mathbf{u}_n^a} = \frac{\left[\mathbf{I} - 3\hat{\mathbf{p}}\hat{\mathbf{p}}^T - (\hat{\mathbf{p}}\times\hat{\boldsymbol{r}})(\hat{\mathbf{p}}\times\hat{\boldsymbol{r}})^T\right]\cos\theta + \hat{\mathbf{p}}\hat{\mathbf{q}}^T + \hat{\mathbf{q}}\hat{\mathbf{p}}^T}{p^2\sin\theta}. \tag{5.28}$$

By taking the derivative of both sides of Eq.(5.24) with respect to $(\mathbf{U}_n^b)^T$, one can obtain

$$\frac{\partial^2\theta}{\partial(\mathbf{U}_n^b)^T\partial\mathbf{U}_n^a} = \frac{\hat{\mathbf{p}}\hat{\mathbf{p}}^T + \hat{\mathbf{q}}\hat{\mathbf{q}}^T - \mathbf{I} - \left[\hat{\mathbf{p}}\hat{\mathbf{q}}^T + (\hat{\mathbf{p}}\times\hat{\boldsymbol{r}})(\hat{\boldsymbol{r}}\times\hat{\mathbf{q}})^T\right]\cos\theta}{pq\sin\theta}. \tag{5.29}$$

Similarly, one can obtain

$$\frac{\partial^2\theta}{\partial(\mathbf{U}_n^b)^T\partial\mathbf{U}_n^b} = \frac{\left[\mathbf{I} - 3\hat{\mathbf{q}}\hat{\mathbf{q}}^T - (\hat{\boldsymbol{r}}\times\hat{\mathbf{q}})(\hat{\boldsymbol{r}}\times\hat{\mathbf{q}})^T\right]\cos\theta + \hat{\mathbf{p}}\hat{\mathbf{q}}^T + \hat{\mathbf{q}}\hat{\mathbf{p}}^T}{q^2\sin\theta}. \tag{5.30}$$

In addition,

$$\frac{\partial^2\theta}{\partial(\mathbf{U}_n^a)^T\partial\mathbf{U}_n^b} = \left(\frac{\partial^2\theta}{\partial(\mathbf{U}_n^b)^T\partial\mathbf{U}_n^a}\right)^T. \tag{5.31}$$

Furthermore, the gradient of the crease energy is

$$\frac{\partial\Psi_f^e}{\partial\mathbf{d}_c} = \frac{\partial\Psi_f^e}{\partial\theta}\frac{\partial\theta}{\partial\mathbf{d}_c}, \tag{5.32}$$

and the Hessian of the crease energy is

$$\frac{\partial^2\Psi_f^e}{\partial\mathbf{d}_c^T\partial\mathbf{d}_c} = \frac{\partial^2\Psi_f^e}{\partial^2\theta}\frac{\partial\theta}{\partial\mathbf{d}_c}\frac{\partial\theta}{\partial\mathbf{d}_c^T} + \frac{\partial\Psi_f^e}{\partial\theta}\frac{\partial^2\theta}{\partial\mathbf{d}_c^T\partial\mathbf{d}_c}. \tag{5.33}$$

Here, the gradient of the crease energy with respect to the fold angle θ is

$$\frac{\partial \Psi_f^e}{\partial \theta} = l \begin{cases} k_f(\theta_L - \theta_0) + k_f \frac{2(\theta_L+\pi)}{\pi} \tan\left(\frac{\pi(\theta-\theta_L)}{2(\theta_L+\pi)}\right) & -\pi < \theta < \theta_L \\ k_f(\theta - \theta_0) & \theta_L \leq \theta \leq \theta_R \\ k_f(\theta_R - \theta_0) + k_f \frac{2(\pi-\theta_R)}{\pi} \tan\left(\frac{\pi(\theta-\theta_R)}{2(\pi-\theta_R)}\right) & \theta_R < \theta < \pi \end{cases}, \tag{5.34}$$

and the Hessian of the crease energy with respect to the fold angle θ is

$$\frac{\partial^2 \Psi_f^e}{\partial \theta^2} = l \begin{cases} k_f \sec^2\left(\frac{\pi(\theta-\theta_L)}{2(\theta_L+\pi)}\right) & -\pi < \theta < \theta_L \\ k_f & \theta_L \leq \theta \leq \theta_R \\ k_f \sec^2\left(\frac{\pi(\theta-\theta_R)}{2(\pi-\theta_R)}\right) & \theta_R < \theta < \pi \end{cases}. \tag{5.35}$$

For the bilinear solid-shell element, the graident and Hessian of θ with respect to $\mathbf{d}_c$ is

$$\frac{\partial \theta}{\partial \mathbf{d}_c} = \begin{bmatrix} N_1 \partial\theta/\partial \boldsymbol{U}_n^a \\ N_2 \partial\theta/\partial \boldsymbol{U}_n^a \\ N_1 \partial\theta/\partial \boldsymbol{U}_n^b \\ N_2 \partial\theta/\partial \boldsymbol{U}_n^b \end{bmatrix}, \frac{\partial^2 \theta}{\partial \mathbf{d}_c^T \partial \mathbf{d}_c} = \begin{bmatrix} N_1 \partial/\partial \boldsymbol{U}_n^a \\ N_2 \partial/\partial \boldsymbol{U}_n^a \\ N_1 \partial/\partial \boldsymbol{U}_n^b \\ N_2 \partial/\partial \boldsymbol{U}_n^b \end{bmatrix} \begin{bmatrix} N_1 \partial\theta/\partial \boldsymbol{U}_n^a \\ N_2 \partial\theta/\partial \boldsymbol{U}_n^a \\ N_1 \partial\theta/\partial \boldsymbol{U}_n^b \\ N_2 \partial\theta/\partial \boldsymbol{U}_n^b \end{bmatrix}^T. \tag{5.36}$$

For the biquadratic solid-shell element, the graident and Hessian of θ with respect to $\mathbf{d}_c$ is

$$\frac{\partial \theta}{\partial \mathbf{d}_c} = \begin{bmatrix} N_1 \partial\theta/\partial \boldsymbol{U}_n^a \\ N_2 \partial\theta/\partial \boldsymbol{U}_n^a \\ N_3 \partial\theta/\partial \boldsymbol{U}_n^a \\ N_1 \partial\theta/\partial \boldsymbol{U}_n^b \\ N_2 \partial\theta/\partial \boldsymbol{U}_n^b \\ N_3 \partial\theta/\partial \boldsymbol{U}_n^b \end{bmatrix}, \frac{\partial^2 \theta}{\partial \mathbf{d}_c^T \partial \mathbf{d}_c} = \begin{bmatrix} N_1 \partial/\partial \boldsymbol{U}_n^a \\ N_2 \partial/\partial \boldsymbol{U}_n^a \\ N_3 \partial/\partial \boldsymbol{U}_n^a \\ N_1 \partial/\partial \boldsymbol{U}_n^b \\ N_2 \partial/\partial \boldsymbol{U}_n^b \\ N_3 \partial/\partial \boldsymbol{U}_n^b \end{bmatrix} \begin{bmatrix} N_1 \partial\theta/\partial \boldsymbol{U}_n^a \\ N_2 \partial\theta/\partial \boldsymbol{U}_n^a \\ N_3 \partial\theta/\partial \boldsymbol{U}_n^a \\ N_1 \partial\theta/\partial \boldsymbol{U}_n^b \\ N_2 \partial\theta/\partial \boldsymbol{U}_n^b \\ N_3 \partial\theta/\partial \boldsymbol{U}_n^b \end{bmatrix}^T. \tag{5.37}$$

5.3 Numerical Tests

The presented computational models can be implemented in most, if not all, finite element programs. This chapter uses a damped Newton solver

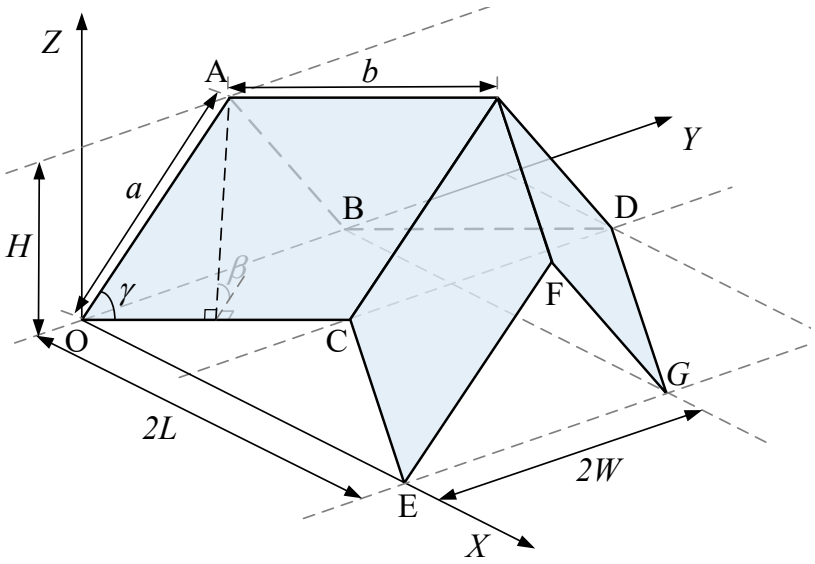


Figure 5.6: A Miura-ori unit cell.

with the adaptive increment method [209]. In the tests below, SI units are used in expressing all material and geometric parameters. The quantitative validation includes two benchmark tests with known analytical solutions. Subsequently, a series of qualitative tests are provided to illustrate the applicability to more complex scenarios. "BLSS" and "BQSS" denote the bilinear and biquadratic solid-shell elements, respectively.

5.3.1 Quantitative Tests

Compressing the Miura-ori unit cell

The geometry and folding kinematics of the Miura-ori unit cell play a crucial role in the design and mechanical analysis of metamaterials. A Miura-ori unit cell formed by 4 panels is illustrated in Figure 5.6. During the folding process, the panels are rigid and the analytical solution is given by [156], i.e.

$$H = a \sin\beta \sin\gamma,$$
$$L = 2b \frac{\cos\gamma \tan\beta}{\sqrt{1 + \cos^2\gamma \tan^2\beta}}, \quad W = 2a\sqrt{1 - \sin^2\gamma \sin^2\beta}. \tag{5.38}$$

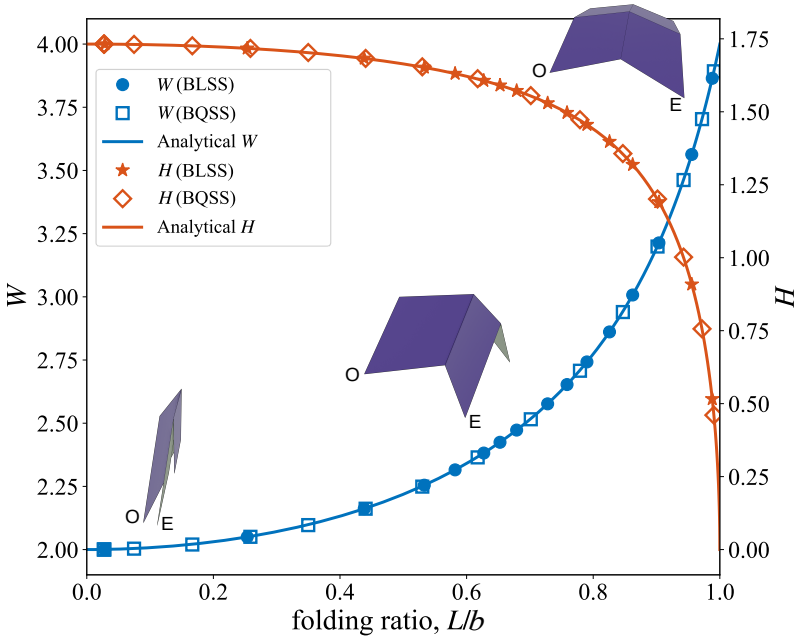


Figure 5.7: The semi-width W and the height H plotted against the folding ratio L/b, see Figure 5.6. The predictions yielded by BLSS and BQSS are graphically indistinguishable from the analytical solutions.

For the Miura-ori unit cell in Figure 5.6, the parameters $a = 2$, $b = 2$, and $\gamma = 60°$. The current configuration of the unit cell can be characterized by the dihedral angle β between the parallelogram panels and the $X - Y$ plane. To model the Miura-ori unit cell as a "rigid origami" system—consisting of rigid panels connected by compliant hinges—the penalty approach in [108] is employed by setting the bending stiffness of the panels to be 10^5 times greater than the folding stiffness. The material parameters used are $E = 12 \times 10^9$, $\nu = 0.3$, $h = 0.01$, and $k_f = 0.01$. To avoid buckling, β in the initial configuration is set to be $15°$, making the Miura-ori unit cell nearly flat. The boundary conditions are imposed as follows: $\mathbf{U}_o = \mathbf{0}$ at node O; $U_o = 0$ along OAB; $W_o = 0$ along OCE and BDG; $U_o = -3.44$ is prescribed along EFG so as to further compress the unit cell. Figure 5.7 plots the semi-width W and hieght H against the folding ratio L/b, the results yielded by both BLSS and BQSS are graphically indistinguishable from the analytical solution.

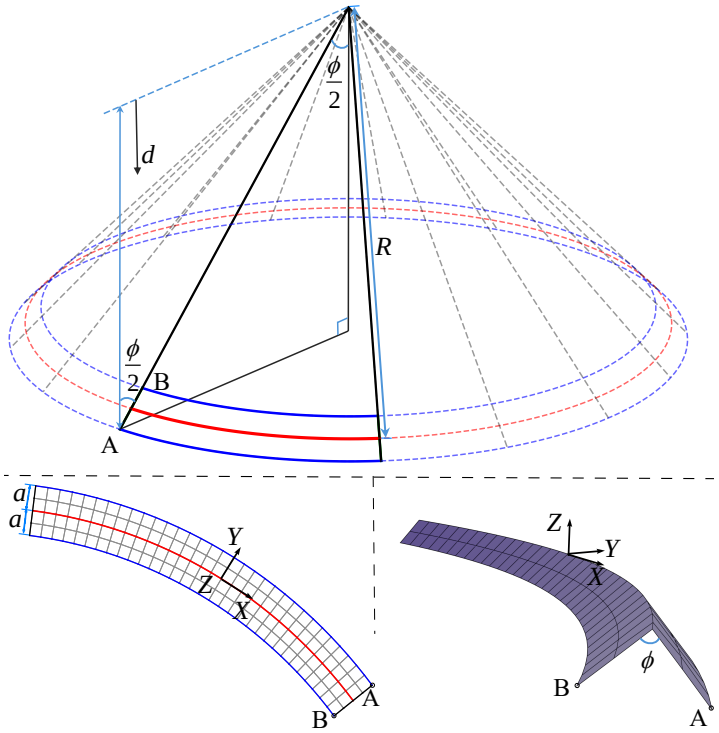


Figure 5.8: The annular sector extracted from the cone with apex angle $\phi = 90^\circ$ is meshed into 32×4 elements. Length of the red arc is $\pi R/4$. The annular sector was folded along the red arc, leading to curved-crease origami structure with the fold angle equal to the apex angle.

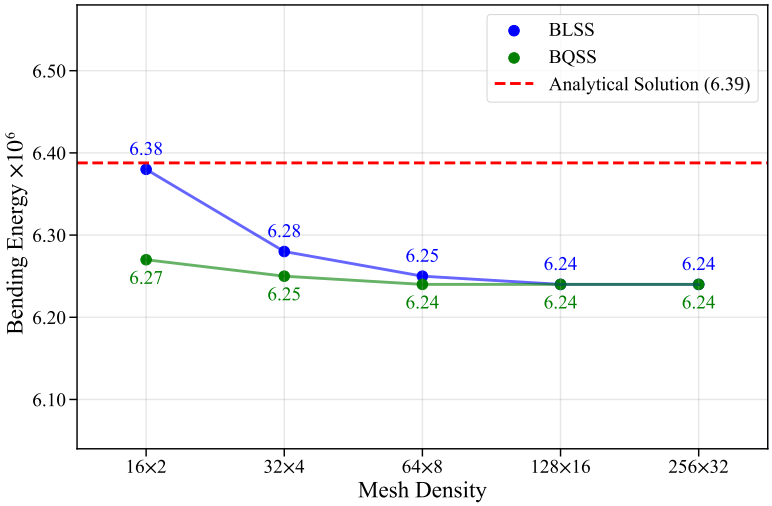


Figure 5.9: The bending energy in the folded annular sector versus the mesh density. Both BLSS and BQSS converge to 6.24×10^{-6}.

Folding an annular sector into two conic surfaces

Following [208, 207], an annular sector of inner and outer radii equal to $R \pm a$ is cut from a cone with apex angle ϕ (see Figure 5.8). It is then flattened and fold along the red curved crease with radius R into a curved-crease structure. The bending energy of the annular region on the right cone surface is

$$\Psi_b = \frac{1}{2} \int_\Omega k_b \kappa_p^2 \, d\Omega, \tag{5.39}$$

where $k_b = Eh^3/12(1-\nu^2)$ is the bending rigidity of an isotropic shell, Ω is the area of the annular region, and

$$\kappa_p = \frac{1}{\tan(\phi/2)\sqrt{1+\tan^2(\phi/2)d}}, \tag{5.40}$$

is the only non-zero principal curvature of the surface. In the expression, d is the distance from the apex of the cone measured along its height (see Figure 5.8). Following [207], when the fold angle of the curved-crease structure is also $\phi = 90^\circ$, the resulting folded shape becomes an exact segment of a cone.

In the numerical simulation, the geometric and material parameters are adopted from [207]. These include: $R = 0.1$, $a = 0.005$, the length of the crease is $\pi R/4$, $E = 4 \times 10^9$, $\nu = 0$, and $h = 0.1$ and the folding stiffness $k_f = 0.1$ which is typically scaled relative to the bending stiffness of the shell [96, 51]. The boundary conditions are imposed as follows: $\mathbf{U}_o = \mathbf{0}$ at node A; $V_o = W_o = 0$ at node B; $W_o = 0$ is also prescribed for all nodes along the inner and outer arcs; $W_o = a/\sqrt{2}$ is applied to the crease to induce a folding angle of 90°. It should be noted that when the annular sector is discretized using BQSS element, the mid-side nodes of the elements along the crease are unconstrained to avoid over-constraining each element during the deformation process.

Figure 5.9 shows that the bending energies predicted by BLSS and BQSS elements rapidly converge to the same value as the mesh is refined. The BQSS element exhibits a faster convergence rate compared to the BLSS element. The converged value is slight lower than the analytical solution, as also observed in [208, 207] which attribute this discrepancy to

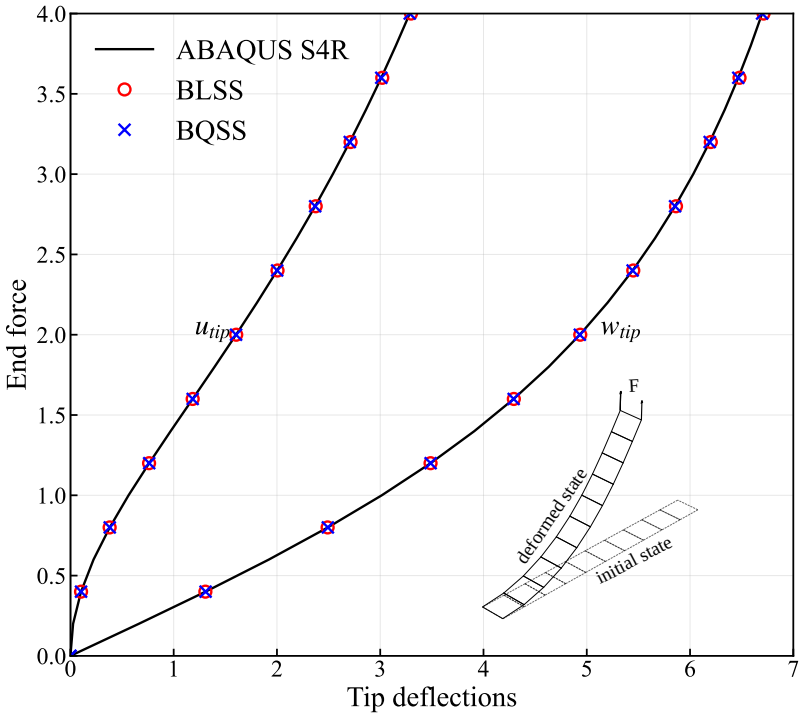


Figure 5.10: Comparison of the predicted end deflections using the present solid-shell elements with the reference solutions obtained from ABAQUS S4R element.

the boundary conditions, potentially preventing the structure from fully achieving the curvature of a pure cone. The derivation of the analytical solution is detailed in Appendix J. The converged result shows a discrepancy of about 2.4% compared with the analytical solution. Building on this, the accuracy of the solid-shell elements is examined, indicating that enhancing the fidelity of the crease modeling could be a promising approach to further reduce the error. This test illustrates the potential of the present method for simulating and analyzing the curved-crease origami structures.

Accuracy assessment of the solid-shell elements. A cantilever beam with dimensions 10 (length), 1 (width) and 0.1 (thickness) is considered. The material properties are defined by a Young's modulus of 1.2×10^9 and a Poisson's ratio of 0. The beam is discretized into 10 quadrilateral solid-shell elements. For comparison, a reference solution is obtained using the ABAQUS S4R shell element with a sufficiently fine mesh of

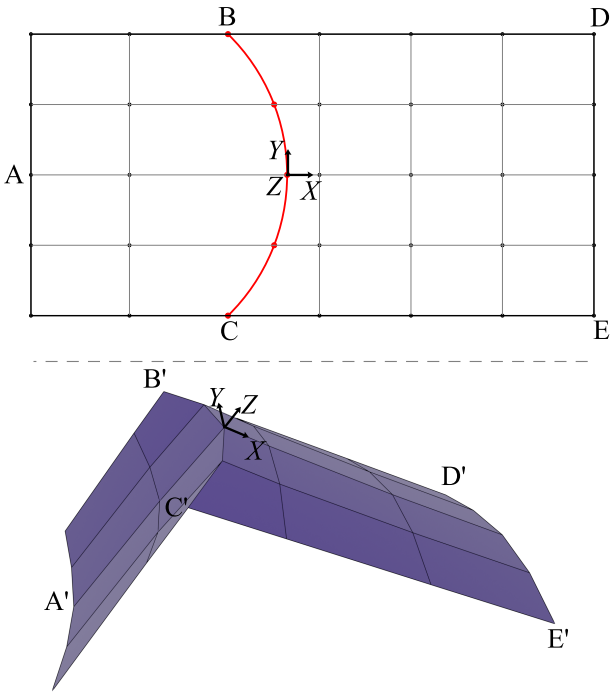


Figure 5.11: Folding a plate outlined by the grey rectangular mesh along the red curved line induces significant bending deformation as illustrated by the deformed configuration in purple. In the case of BLSS, the red curve bcomes multilinear. A, B, C, D and E are displaced to A′, B′, C′, D′ and E′

40×4 elements to ensure convergence. Figure 5.10 shows the predicted vertical and horizontal tip deflections under a shear force ranging from 0 to 4,000 (2,000 applied at each tip node) in 10 increments. The results demonstrate a good agreement between the present elements and the reference solution.

5.3.2 Qualitative Tests

In this subsection, three qualitative tests will be illustrated. In these tests, the parameters of the paper plate include $E = 3 \times 10^9$, $\nu = 0.3$, $h = 0.1 \times 10^{-3}$ and $k_f = 0.1$ which are taken from [72].

Folding a plate with a curved crease

Following [191], a 0.1×0.05 paper plate is folded along the red curved crease in Figure 5.11. The prescribed boundary conditions include: $W_o =$

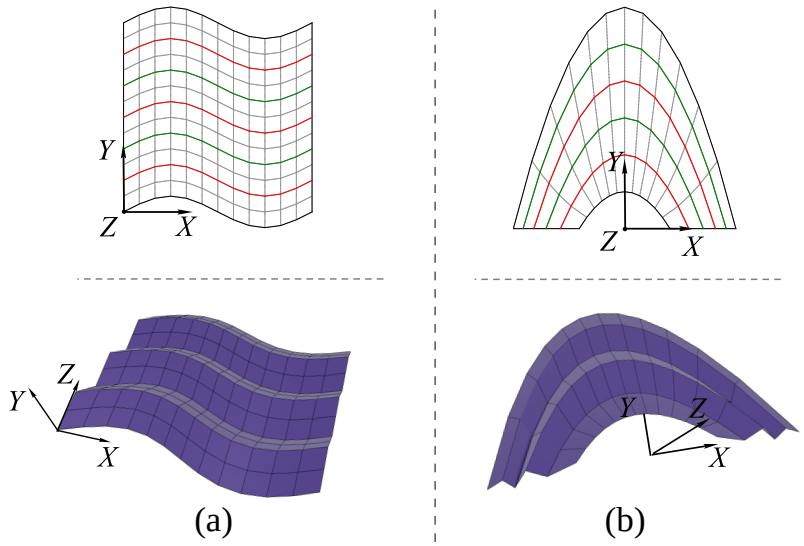


Figure 5.12: (a) The upper figure shows the undeformed mesh of a Miura-ori structure formed by sinusoidal creases. (b) The upper figure shows the undeformed mesh of a canopy structure formed by parabolic creases . In the meshes, red and green lines also indicate mountain and valley creases, respectively. The deformed structures are portrayed in purple.

0 along BD and CE; $U_o = 0$ at nodes D and E; the $V_o = 0$ at node E is constrained to prevent rigid body motion. the $W_o = -0.35$ is prescribed at node A to induce folding motion. Folding-induced bending is clearly observed, which highlights a key feature of curved-crease origami.

Origami structures with multiple curved creases

In Figure 5.12a shows a Miura-ori structure formed by folding multiple sinusoidalcreases whilst Figure 5.12b shows a canopy structure formed by folding multiple parabolic creases. The initially flat geometry of the Miura-ori structure in the upper portion of Figure 5.12a consists of two baseline segments at $X = 0$ and $X = 6$, along with seven parabolic creases defined by $Y = 0.5\sin(\pi X/3) + i$, where $i = 0, 1, 2, \ldots, 6$. The initial geometry of the canopy structure in the upper portion of Figure 5.12b consists of one baseline at $Y = 0$ and six downward-opening parabolic creases given by $Y_i = -\frac{2}{3}(X-3)^2 + i$, where $i = 1, 2, \ldots, 6$. These geometric configurations are outlined by black lines whilst red and green lines are mountain and valley creases, respectively. The gray

lines together with other lines show the meshes employed. In other words, 144 and 60 elements are employed in the Miura-ori structure and canopy, respectively. To realize the folded configurations, vertical displacements of $W_o = 0.5$ and $W_o = 0.25$ are prescribed to the mountain creases of the Miura-ori and the canopy structures, respectively, while $W_o = 0$ is prescribed along the valley creases of both structures.

Buckled annular

In this test, the buckled state of a annulus [39] is considered. Figure 5.13 shows the flat undeformed annulus. The diameters of the inner, red central and outer circles are 0.18, 0.2 and 0.22, respectively. Among them, the red central circle is a crease. The buckled state is attained by going through an intermediate state for which the following conditions are prescribed: $W_o = 0$ along the inner and outer circles and $W_o = 0.005$ along the crease. After attaining the intermediate state, $U_o = 0$ along EF and GH, $V_o = 0$ along AMB and CND and $W_o = 0.04$ at nodes M and N are prescribed to deform the annulus. Total elastic energies in the intermediate state and the last state are 109.15×10^{-3} and 0.44×10^{-3}. As a result, the present models are able to vividly reproduce the buckled state observed in the physical experiment in [39] (see Figure 5.13).

5.4 Conclusion

In this chapter, solid-shell elements are introduced to model origami structures, where the naturally embedded director vectors of the elements are utilized to model creases. The fold angle in the crease is approximated computing in terms of the director vectors of the elements. Origami structures with straight and curved creases are analyzed and the predictions are in good agreements with the analytical solutions. These structures may serve as useful tests for validating other origami simulation tools. Quantitative tests are also presented to demonstrate the effectiveness of the present computational model, highlighting its potential for the design and analysis of origami structures.

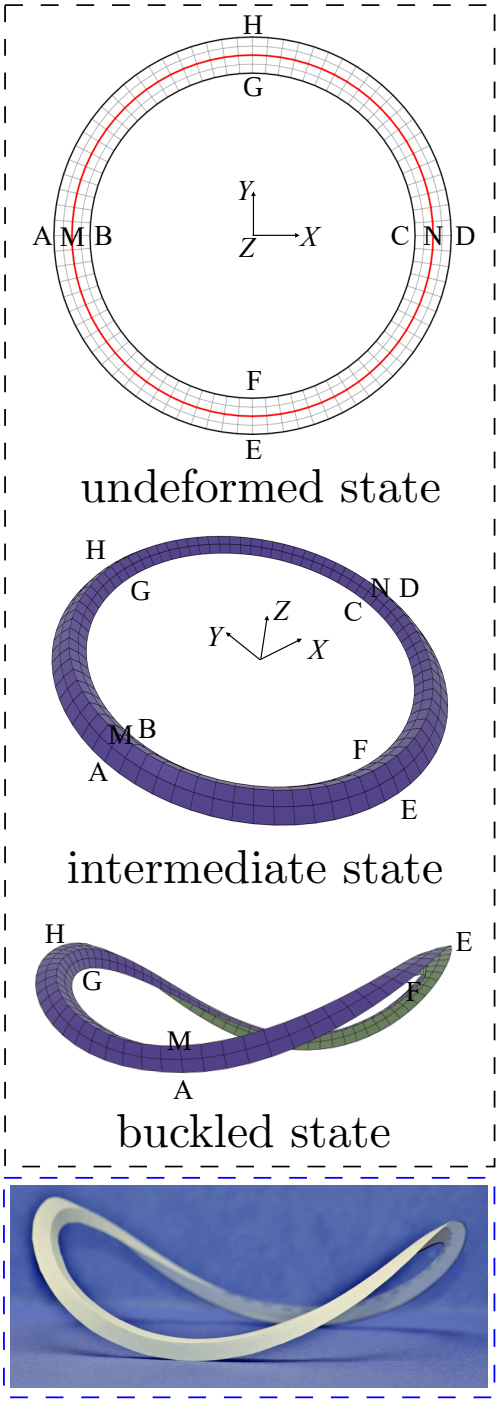


Figure 5.13: Buckle an annulus. The three states within the black dashed box are obtained from simulation while the photo on the bottom is extracted from [204].

The present computational models can be naturally extended to triangular solid-shell elements while continuing to employ director vectors to model creases, thereby enabling the simulation of origami structures with curved creases [106], which remain challenging for existing methods. It can also be extended to higher-order (> 2) elements and crease representations. Scaling such extensions to large multi-panel assemblies also raises an implementation challenge: the current scheme assigns the same number of degrees of freedom to every node based on the most complex junction in the mesh, forcing all other nodes to carry redundant unused entries; a variable-length storage scheme in which each node stores only the degrees of freedom it actually needs would be required to avoid this overhead. Furthermore, the incorporation of more comprehensive contact algorithms would facilitate the simulation of multi-body interactions in complex folded assemblies.

Chapter 6

Conclusions and Future Works

6.1 Conclusions

In conclusion, the contributions of this thesis include: (1) a hinge-bending model and a hinge-folding model; (2) six corotational hinge-based bending models; (3) a smoothed hinge membrane model for mitigating sharp creasing artifacts; and (4) bilinear and biquadratic solid-shell elements incorporating crease models formed by midsurface directors. Except for the hinge-folding and crease models, the remaining computational models are different spatial discretizations of continuum plates/shells.

In cloth simulations, a 4-node hinge-bending model based on an edge stencil is introduced. It obtains the curvature of the cylindrical surface fitted to the hinge and nodes. It is evaluated through commonly used quantitative tests for hinge-bending models [66, 19, 207], showing that it is more accurate than the existing models. To improve the efficiency of the hinge-bending model, the small-strain-small-curvature assumption is adopted to quantify the curvature of the cylindrical surface in a corotational frame. Based on this assumption, a corotational edge-based hinge curvature operator is formulated, which in turn yields the corresponding thin plate and shell models (EP and ES). Compared to the quadratic shell model [8] and cubic shell model [59], the present EP and ES achieve improved accuracy. The aforementioned models, based on

edge stencils, are referred to as edge-stenciled models. The edge stencil is probably the most basic geometric structure for discretely approximating smooth geometry [221]. Nevertheless, the edge-stenciled models have limited capacity to reproduce the curvature of general surfaces, resulting in the strong mesh dependency issue [65]. To address this issue, the triangle-centered stencil is adopted and the derivation methodology used for the EP and ES models is extended. Based on this stencil, the corotational FVM and smoothed hinge-based curvature operators are formulated, and their corresponding bending models (FP, FS, SP, and SS) are derived. Together, these six corotational hinge-based bending models (EP/ES, FP/FS, and SP/SS) possess the same feature: their bending energy Hessians are constant, which significantly benefits implicit simulation. In particular, the FP/FS and SP/SS models demonstrate accuracy competitive with the state-of-the-art DDG shell models [26], while offering roughly four times higher computational efficiency in shell benchmark tests. When membrane strain is discretized using CST and coupled with the present bending models, sharp creasing artifacts often arise. To mitigate the artifacts, a smoothed hinge membrane model is introduced. The model uses the 6 nodes in the triangle-centered stencil to interpolate the displaced position from which the membrane strain is derived. The membrane strain in the central triangle is taken to be the average of those at its edge mid- points. When the membrane model is coupled with the present bending models, the sharp creasing artifacts can be effectively mitigated. A four-node interpolation scheme, analogous the aforementioned six-node scheme, is also introduced to further elucidate the notion of a "smoothed hinge membrane". All proposed models have been integrated into the C-IPC solver [100], and they remain stable under large time steps in challenging cloth simulation scenarios involving complex contact and friction.

In origami simulation, the hinge angle and its derivatives from the hinge-bending model are adopted for a dedicated hinge-folding model. This model integrates seamlessly with finite element-based origami panels, offering improved efficiency and robustness compared to existing finite element approach implemented using ABAQUS UEL [72].

In addition, the bilinear and biquadratic solid-shell elements are presented for modeling origami structures, where the crease models are naturally modeled by director vectors. This approach eliminates the need for additional connector elements at creases. Both elements are implemented in an adaptive implicit Newton solver. The present models are benchmarked by both straight- and curved-crease origami structures against analytical solutions. Further qualitative tests demonstrate the efficacy of the proposed models.

Notably, the proposed computational models are modular and can be easily integrated into other physical simulation frameworks across diverse application domains. In garment CAD design, the models can be incorporated into pattern design software to improve drape prediction, potentially reducing the number of physical prototypes. In visual effects, the constant bending energy Hessians reduce per-iteration cost in Newton-type solvers, benefiting production efficiency. In robotics, the cloth models can be integrated into high-fidelity simulators for training robotic manipulation policies on deformable objects (e.g., fabric folding), while the origami solid-shell models can assist in the design of foldable robotic structures and metamaterials, where accurate crease behavior prediction under actuation is critical.

Despite these contributions, the current work has several limitations. First, the corotational hinge-based models rely on the small-strain-small-curvature assumption, which leads to reduced accuracy when finite strains are involved. Second, the smoothed hinge membrane model is formulated specifically for initially flat configurations, limiting its direct applicability to initially curved shells; moreover, the qualitative tests suggest that it may introduce additional membrane stiffness compared to the CST element. Third, all simulations in this thesis are performed on CPU-based solvers, and the computational cost scales with mesh resolution. For deployment in interactive applications such as virtual try-on or real-time physics engines, integration with GPU-accelerated solvers and/or machine-learning-based surrogate models would be necessary.

Fourth, the solid-shell origami models are validated under elastic material behavior with moderate in-panel strains; while the elements naturally accommodate the large displacements and rotations inherent to origami folding, the response under large panel strains and inelastic material behavior (e.g., plasticity, viscoelasticity) in creases remains to be explored. Finally, the presented models are validated against analytical and numerical solutions. Since the contributions are at the discretization level, these benchmarks suffice to isolate discretization accuracy without interference from constitutive modeling or material calibration errors. Experimental validation of cloth draping simulations remains nontrivial, as complex draping configurations typically admit multiple stable equilibria with different wrinkle patterns, making a direct quantitative comparison between simulation and experiment difficult.

6.2 Future Works

In cloth simulations, the proposed models can be integrated into the GPU-based simulators [128]. With the constant bending energy Hessian be further exploited, the computational efficiency of the models developed in this thesis can be further enhanced. Moreover, the analytically derived gradients and Hessians make the proposed models directly compatible with differentiable simulation frameworks, enabling gradient-based optimization and learning. Furthermore, the current framework treats external forces as prescribed nodal force vectors and does not explicitly model aerodynamic drag from the relative motion between cloth and the surrounding air. A natural extension is to couple the present cloth model with a fluid solver to capture these aerodynamic effects. For more complex fluid environments such as wet cloth, the fabric can be modelled in yarn level, with the fluid-solid interaction decomposed into a pressure gradient force governing buoyancy and a drag force arising from liquid-solid friction.

The corotational hinge-based bending models developed in this work can also be extended to quadrilateral meshes by constructing discrete

curvatures specifically for quadrilaterals. Alternatively, discrete curvature can also be constructed using node-based stencils. It remains worthwhile to explore other strain sampling techniques based on triangle-centered stencils, other than a quadratic interpolation function for interpolating displaced positions. In addition, modeling inelastic creases caused by plastic deformation has the potential to enhance the visual realism and physical fidelity of cloth simulations.

In origami simulations, accurately predicting the geometric and mechanical behaviors of the crease is essential for improving the simulation quality. One possible extension is to discretize creases using additional elements that can stretch, twist, and bend. Such an extension may provide a more complete representation of creases.

Beyond cloth and origami, incorporating additional physical fields such as electromagnetic, thermal, or chemical effects into the deformation fields may significantly extend their applicability across various environments. Designing reduced-order model and/or data-driven surrogate models for specific simulation scenarios can significantly reduce inference time on hardware with limited computational resources. Another promising direction is to integrate the cloth/origami simulation models into sim-to-real training systems for robotic manipulation of deformable materials. In addition, these models can serve as benchmarks for evaluating the deformable object manipulation capabilities of vision-language-action (VLA) models [13] and world action models (WAMs) [214], where the fidelity of the underlying physics is critical for assessing model performance. More broadly, while AI-based methods offer computational speed, they struggle with physical guarantees; accurate computational models can benefit high-fidelity data engines for physical AI, offering physically accurate simulations that go beyond what current generative AI methods alone can provide in scenarios where physical correctness is required. Furthermore, although the computational models in this thesis are motivated by cloth and origami, they are broadly applicable to the simulation and analysis of other thin and flexible structures.

Appendix A

Detailed Derivation of Gradient and Hessian of Hinge Angle

The following standard relations for any vectors $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$, which will be used, are listed

$$\frac{\partial}{\partial \mathbf{c}}(\mathbf{a}\cdot\mathbf{b}) = \frac{\partial \mathbf{a}^T}{\partial \mathbf{c}}\mathbf{b} + \frac{\partial \mathbf{b}^T}{\partial \mathbf{c}}\mathbf{a} \tag{A.1a}$$

$$\frac{\partial |\mathbf{a}|}{\partial \mathbf{c}} = \frac{\partial \mathbf{a}^T}{\partial \mathbf{c}}\frac{\mathbf{a}}{|\mathbf{a}|} \tag{A.1b}$$

$$\mathbf{a}\times\mathbf{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}\mathbf{b} = [\mathbf{a}]_\times \mathbf{b} \tag{A.1c}$$

$$\frac{\partial}{\partial \mathbf{c}^T}(\mathbf{a}\times\mathbf{b}) = [\mathbf{a}]_\times \frac{\partial \mathbf{b}}{\partial \mathbf{c}^T} - [\mathbf{b}]_\times \frac{\partial \mathbf{a}}{\partial \mathbf{c}^T} \tag{A.1d}$$

$$\mathbf{a}\times(\mathbf{b}\times\mathbf{c}) = (\mathbf{a}\cdot\mathbf{c})\mathbf{b} - (\mathbf{a}\cdot\mathbf{b})\mathbf{c} \tag{A.1e}$$

$$\mathbf{a}(\mathbf{a}\times\mathbf{b})^T - (\mathbf{a}\times\mathbf{b})\mathbf{a}^T = (\mathbf{a}\cdot\mathbf{a})[\mathbf{b}]_\times - (\mathbf{a}\cdot\mathbf{b})[\mathbf{a}]_\times, \tag{A.1f}$$

$$\mathbf{a}\mathbf{a}^T\mathbf{b} = (\mathbf{a}\cdot\mathbf{b})\mathbf{a}, \tag{A.1g}$$

$$(\mathbf{a}\mathbf{b}^T)\mathbf{c} = \mathbf{a}(\mathbf{b}^T\mathbf{c}), \tag{A.1h}$$

where the skew symmetric matrix $[\mathbf{a}]_\times$ for vector $\mathbf{a}$ is self-defined.

A.1 Gradient of the Hinge Angle

By rewriting Eq.(2.11) in Chapter 2 as

$$\cos\theta = -\frac{\mathbf{p}\cdot\mathbf{q}}{pq} \tag{A.2}$$

and differentiating Eq.(A.2) with respect to $\mathbf{x}^e$, one can obtain

$$-\sin\theta\frac{\partial\theta}{\partial\mathbf{x}^s} = -\frac{\partial\left(\frac{\mathbf{p}\cdot\mathbf{q}}{pq}\right)}{\partial\mathbf{x}^s}, \tag{A.3}$$

where the right-hand side can be further derived as

$$\frac{\partial\left(\frac{\mathbf{p}\cdot\mathbf{q}}{pq}\right)}{\partial\mathbf{x}^s} = \frac{\frac{\partial(\mathbf{p}\cdot\mathbf{q})}{\partial\mathbf{x}^s}pq - \mathbf{p}\cdot\mathbf{q}\frac{\partial(pq)}{\partial\mathbf{x}^s}}{(pq)^2}. \tag{A.4}$$

By invoking Eqs.(A.1a) and (A.1b), one can obtain

$$\frac{\partial\left(\mathbf{p}\cdot\mathbf{q}\right)}{\partial\mathbf{x}^s} = \frac{\partial\mathbf{p}^T}{\partial\mathbf{x}^s}\mathbf{q} + \frac{\partial\mathbf{q}^T}{\partial\mathbf{x}^s}\mathbf{p}, \tag{A.5}$$

and

$$\frac{\partial\left(pq\right)}{\partial\mathbf{x}^s} = p\frac{\partial q}{\partial\mathbf{x}^s} + q\frac{\partial p}{\partial\mathbf{x}^s} = p\frac{\partial\mathbf{q}^T}{\partial\mathbf{x}^s}\frac{\mathbf{q}}{q} + q\frac{\partial\mathbf{p}^T}{\partial\mathbf{x}^s}\frac{\mathbf{p}}{p}. \tag{A.6}$$

By substituting Eqs.(A.5) and (A.6) into Eq.(A.4), one can obtain

$$\frac{\partial\theta}{\partial\mathbf{x}^s} = \frac{\left(\frac{\partial\mathbf{p}^T}{\partial\mathbf{x}^s}\mathbf{q} + \frac{\partial\mathbf{q}^T}{\partial\mathbf{x}^s}\mathbf{p}\right)(pq) - (\mathbf{p}\cdot\mathbf{q})\left(p\frac{\partial\mathbf{q}^T}{\partial\mathbf{x}^s}\frac{\mathbf{q}}{q} + q\frac{\partial\mathbf{p}^T}{\partial\mathbf{x}^s}\frac{\mathbf{p}}{p}\right)}{p^2q^2\sin\theta}, \tag{A.7}$$

which can be regrouped as

$$\frac{\partial\theta}{\partial\mathbf{x}^s} = \frac{\partial\mathbf{p}^T}{\partial\mathbf{x}^s}\frac{p^2\mathbf{q} - (\mathbf{p}\cdot\mathbf{q})\mathbf{p}}{p^3q\sin\theta} + \frac{\partial\mathbf{q}^T}{\partial\mathbf{x}^s}\frac{q^2\mathbf{p} - (\mathbf{p}\cdot\mathbf{q})\mathbf{q}}{pq^3\sin\theta}. \tag{A.8}$$

Here, $\partial\mathbf{p}^T/\partial\mathbf{x}^s$ and $\partial\mathbf{q}^T/\partial\mathbf{x}^s$ are the gradients of the triangle altitude vectors $\mathbf{p}$ and $\mathbf{q}$, respectively. To compute these gradients, it is necessary to invoke the gradients of the non-dimensional variables s_C and s_D.

By differentiating s_C in Eq.(2.13) of Chapter 2 with respect to $\mathbf{x}^e$, one can obtain

$$\frac{\partial s_C}{\partial \mathbf{x}_A} = \frac{\partial \left(\frac{\mathbf{x}_{AC} \cdot \mathbf{x}_{AB}}{\|\mathbf{x}_{AB}\|^2} \right)}{\partial \mathbf{x}_A} = \frac{\frac{\partial (\mathbf{x}_{AC} \cdot \mathbf{x}_{AB})}{\partial \mathbf{x}_A} \|\mathbf{x}_{AB}\|^2 - (\mathbf{x}_{AC} \cdot \mathbf{x}_{AB}) \frac{\partial \|\mathbf{x}_{AB}\|^2}{\partial \mathbf{x}_A}}{\|\mathbf{x}_{AB}\|^4}. \tag{A.9}$$

By invoking Eqs.(A.1a) and (A.1b), one can obtain

$$\frac{\partial (\mathbf{x}_{AC} \cdot \mathbf{x}_{AB})}{\partial \mathbf{x}_A} = \frac{\partial \mathbf{x}_{AC}^T}{\partial \mathbf{x}_A} \mathbf{x}_{AB} + \frac{\partial \mathbf{x}_{AB}^T}{\partial \mathbf{x}_A} \mathbf{x}_{AC} = -\mathbf{x}_{AB} - \mathbf{x}_{AC}, \tag{A.10}$$

and

$$\frac{\partial \left(\|\mathbf{x}_{AB}\|^2 \right)}{\partial \mathbf{x}^s} = 2\|\mathbf{x}_{AB}\| \frac{\partial \mathbf{x}_{AB}^T}{\partial \mathbf{x}_A} \frac{\mathbf{x}_{AB}}{\|\mathbf{x}_{AB}\|} = -2\mathbf{x}_{AB}. \tag{A.11}$$

By substituting Eqs.(A.10) and (A.11) into Eq.(A.9), one can obtain

$$\begin{aligned} \frac{\partial s_C}{\partial \mathbf{x}_A} &= \frac{(-\mathbf{x}_{AB} - \mathbf{x}_{AC})\|\mathbf{x}_{AB}\|^2 + 2\mathbf{x}_{AB}(\mathbf{x}_{AC} \cdot \mathbf{x}_{AB})}{\|\mathbf{x}_{AB}\|^4} \\ &= \frac{(-\mathbf{x}_{AB} - \mathbf{x}_{AC}) + 2\mathbf{x}_{AB} \frac{(\mathbf{x}_{AC} \cdot \mathbf{x}_{AB})}{\|\mathbf{x}_{AB}\|^2}}{\|\mathbf{x}_{AB}\|^2}. \end{aligned} \tag{A.12}$$

By invoking $\mathbf{p} = \mathbf{x}_{AC} - s_C \mathbf{x}_{AB}$ in Eq.(2.12) and $s_C = (\mathbf{x}_{AC} \cdot \mathbf{x}_{AB}) / \|\mathbf{x}_{AB}\|^2$ in Eq.(2.13) of Chapter 2, the Eq.(A.12) can be rewritten as

$$\begin{aligned} \frac{\partial s_C}{\partial \mathbf{x}_A} &= \frac{(-\mathbf{x}_{AB} - \mathbf{x}_{AC}) + 2\mathbf{x}_{AB} s_C}{\|\mathbf{x}_{AB}\|^2} \\ &= \frac{-\mathbf{x}_{AC} + s_C \mathbf{x}_{AB} - \mathbf{x}_{AB} + s_C \mathbf{x}_{AB}}{\|\mathbf{x}_{AB}\|^2} \\ &= -\frac{\mathbf{p} + (1 - s_C)\mathbf{x}_{AB}}{\|\mathbf{x}_{AB}\|^2}. \end{aligned} \tag{A.13}$$

Similarly, other gradients of s_C and s_D can be derived as

$$\begin{aligned} \frac{\partial s_C}{\partial \mathbf{x}_B} = \frac{\mathbf{p} - s_C \mathbf{x}_{AB}}{\mathbf{x}_{AB}^2}, \frac{\partial s_C}{\partial \mathbf{x}_C} = \frac{\mathbf{x}_{AB}}{\mathbf{x}_{AB}^2}, \frac{\partial s_C}{\partial \mathbf{x}_D} = \mathbf{0}, \\ \frac{\partial s_D}{\partial \mathbf{x}_A} = -\frac{\mathbf{q} + (1 - s_D)\mathbf{x}_{AB}}{\mathbf{x}_{AB}^2}, \frac{\partial s_D}{\partial \mathbf{x}_B} = \frac{\mathbf{q} - s_D \mathbf{x}_{AB}}{\mathbf{x}_{AB}^2}, \\ \frac{\partial s_D}{\partial \mathbf{x}_C} = \mathbf{0}, \frac{\partial s_D}{\partial \mathbf{x}_D} = \frac{\mathbf{x}_{AB}}{\mathbf{x}_{AB}^2}. \end{aligned} \tag{A.14}$$

Consequently, one can get

$$\begin{aligned}\frac{\partial \mathbf{p}^T}{\partial \mathbf{x}_A} &= \frac{\partial (\mathbf{x}_{AC} - s_C \mathbf{x}_{AB})^T}{\partial \mathbf{x}_A} \\ &= -\mathbf{I} - \frac{\partial s_C}{\partial \mathbf{x}_A}\mathbf{x}_{AB}^T + s_C \mathbf{I} \\ &= -(1-s_C)\mathbf{I} + \frac{\mathbf{p} + (1-s_C)\mathbf{x}_{AB}}{\|\mathbf{x}_{AB}\|^2}\mathbf{x}_{AB}^T \\ &= -(1-s_C)\left(\mathbf{I} - \mathbf{s}\mathbf{s}^T\right) + \frac{\mathbf{p}\mathbf{s}^T}{\|\mathbf{x}_{AB}\|},\end{aligned} \tag{A.15}$$

which uses the Eq.(A.13) and the unit vector $\mathbf{s} = \mathbf{x}_{AB}/\|\mathbf{x}_{AB}\|$. Similarly, the other components of the gradients of the triangle altitude vectors $\mathbf{p}$ and $\mathbf{q}$ can be expressed as

$$\frac{\partial \mathbf{p}^T}{\partial \mathbf{x}_B} = -s_C\left(\mathbf{I} - \mathbf{s}\mathbf{s}^T\right) - \frac{\mathbf{p}\mathbf{s}^T}{|\mathbf{x}_{AB}|}, \frac{\partial \mathbf{p}^T}{\partial \mathbf{x}_C} = \mathbf{I} - \mathbf{s}\mathbf{s}^T, \frac{\partial \mathbf{p}^T}{\partial \mathbf{x}_D} = \mathbf{0}, \tag{A.16}$$

and

$$\begin{aligned}\frac{\partial \mathbf{q}^T}{\partial \mathbf{x}_A} = -(1-s_D)\left(\mathbf{I} - \mathbf{s}\mathbf{s}^T\right) + \frac{\mathbf{q}\mathbf{s}^T}{\|\mathbf{x}_{AB}\|}, \frac{\partial \mathbf{q}^T}{\partial \mathbf{x}_B} = -s_D\left(\mathbf{I} - \mathbf{s}\mathbf{s}^T\right) - \frac{\mathbf{q}\mathbf{s}^T}{\|\mathbf{x}_{AB}\|}, \\ \frac{\partial \mathbf{q}^T}{\partial \mathbf{x}_C} = \mathbf{0}, \frac{\partial \mathbf{q}^T}{\partial \mathbf{x}_D} = \mathbf{I} - \mathbf{s}\mathbf{s}^T.\end{aligned} \tag{A.17}$$

By invoking the geometric relation

$$\mathbf{p} \times \mathbf{q} = (pq\sin\theta)\mathbf{s} \tag{A.18}$$

and Eq.(A.1e), one can get

$$\begin{aligned}p^2\mathbf{q} - (\mathbf{p}\cdot\mathbf{q})\mathbf{p} = \mathbf{p}\times(\mathbf{q}\times\mathbf{p}) = \mathbf{s}\times\mathbf{p}(pq\sin\theta), \\ q^2\mathbf{p} - (\mathbf{p}\cdot\mathbf{q})\mathbf{q} = \mathbf{q}\times(\mathbf{p}\times\mathbf{q}) = \mathbf{q}\times\mathbf{s}(pq\sin\theta).\end{aligned} \tag{A.19}$$

By introducing Eq.(A.19), Eq.(A.8) becomes

$$\frac{\partial \theta}{\partial \mathbf{x}^s} = \frac{\partial \mathbf{p}^T}{\partial \mathbf{x}^s}\frac{\mathbf{s}\times\mathbf{p}}{p^2} + \frac{\partial \mathbf{q}^T}{\partial \mathbf{x}^s}\frac{\mathbf{q}\times\mathbf{s}}{q^2}. \tag{A.20}$$

By substituting $\partial \mathbf{p}^T / \partial \mathbf{x}_C = \mathbf{I} - \mathbf{s}\mathbf{s}^T$ in Eq.(A.16) and $\partial \mathbf{q}^T / \partial \mathbf{x}_C = \mathbf{0}$ in Eq.(A.17) into Eq.(A.20), one can get the component

$$\frac{\partial \theta}{\partial \mathbf{x}_C} = \frac{\partial \mathbf{p}^T}{\partial \mathbf{x}_C} \frac{\mathbf{s} \times \mathbf{p}}{p^2} = (\mathbf{I} - \mathbf{s}\mathbf{s}^T) \frac{\mathbf{s} \times \mathbf{p}}{p^2}. \tag{A.21}$$

By introducing the orthogonality relations

$$(\mathbf{s} \times \mathbf{p}) \perp \mathbf{s}, (\mathbf{s} \times \mathbf{p}) \perp \mathbf{p}, \mathbf{s} \perp \mathbf{p} \tag{A.22}$$

and the Eq.(A.1g), one can obtain

$$\mathbf{s}\mathbf{s}^T \frac{(\mathbf{s} \times \mathbf{p})}{p^2} = (\mathbf{s} \cdot \frac{(\mathbf{s} \times \mathbf{p})}{p^2})\mathbf{s} = \mathbf{0} \tag{A.23}$$

and, thus, the Eq.(A.21) becomes

$$\frac{\partial \theta}{\partial \mathbf{x}_C} = \frac{\mathbf{s} \times \mathbf{p}}{p^2}. \tag{A.24}$$

Similarly, by introducing the orthogonality relations,

$$(\mathbf{q} \times \mathbf{s}) \perp \mathbf{s}, (\mathbf{q} \times \mathbf{s}) \perp \mathbf{p}, \mathbf{s} \perp \mathbf{q}, \tag{A.25}$$

one can obtain

$$\frac{\partial \theta}{\partial \mathbf{x}_D} = \frac{\mathbf{q} \times \mathbf{s}}{q^2}. \tag{A.26}$$

By substituting $\partial \mathbf{p}^T / \partial \mathbf{x}_A$ in Eq.(A.15) and $\partial \mathbf{q}^T / \partial \mathbf{x}_A$ in Eq.(A.17) into Eq.(A.20), one can get the component

$$\begin{aligned} \frac{\partial \theta}{\partial \mathbf{x}_A} = &- \left((1 - s_C) \left(\mathbf{I} - \mathbf{s}\mathbf{s}^T \right) - \frac{\mathbf{p}\mathbf{s}^T}{\|\mathbf{x}_{AB}\|} \right) \frac{(\mathbf{s} \times \mathbf{p})}{p^2} \\ &- \left((1 - s_D) \left(\mathbf{I} - \mathbf{s}\mathbf{s}^T \right) - \frac{\mathbf{q}\mathbf{s}^T}{\|\mathbf{x}_{AB}\|} \right) \frac{(\mathbf{q} \times \mathbf{s})}{q^2}. \end{aligned} \tag{A.27}$$

In terms of Eqs.(A.1h), (A.22) and (A.25), one can get

$$\begin{aligned}\frac{\mathbf{p}\mathbf{s}^T}{\|\mathbf{x}_{AB}\|}\frac{(\mathbf{s}\times\mathbf{p})}{p^2} &= \frac{\mathbf{p}(\mathbf{s}^T(\mathbf{s}\times\mathbf{p}))}{\|\mathbf{x}_{AB}\|p^2} = \mathbf{0},\\ \frac{\mathbf{q}\mathbf{s}^T}{\|\mathbf{x}_{AB}\|}\frac{(\mathbf{q}\times\mathbf{s})}{q^2} &= \frac{\mathbf{q}(\mathbf{s}^T(\mathbf{q}\times\mathbf{s}))}{\|\mathbf{x}_{AB}\|q^2} = \mathbf{0}.\end{aligned} \tag{A.28}$$

Thus, the Eq.(A.27) becomes

$$\frac{\partial\theta}{\partial\mathbf{x}_A} = -(1-s_C)\frac{\mathbf{s}\times\mathbf{p}}{p^2} - (1-s_D)\frac{\mathbf{q}\times\mathbf{s}}{q^2} \tag{A.29}$$

Similarly,

$$\frac{\partial\theta}{\partial\mathbf{x}_B} = -s_C\frac{\mathbf{s}\times\mathbf{p}}{p^2} - s_D\frac{\mathbf{q}\times\mathbf{s}}{q^2}. \tag{A.30}$$

Before deriving the Hessian of the fold angle, the gradient of the unit vector $\mathbf{s}$

$$\frac{\partial\mathbf{s}}{\partial\mathbf{x}_A^T} = \frac{-1}{|\mathbf{x}_{AB}|}\left(\mathbf{I}-\mathbf{s}\mathbf{s}^T\right), \frac{\partial\mathbf{s}}{\partial\mathbf{x}_B^T} = \frac{1}{|\mathbf{x}_{AB}|}\left(\mathbf{I}-\mathbf{s}\mathbf{s}^T\right), \frac{\partial\mathbf{s}}{\partial\mathbf{x}_C^T} = \mathbf{0}, \frac{\partial\mathbf{s}}{\partial\mathbf{x}_D^T} = \mathbf{0} \tag{A.31}$$

is also derived.

A.2 Hessian of the Hinge Angle

The gradient of $\partial\theta/\partial\mathbf{x}_C$ with respect to $\mathbf{x}_C$ is $\partial^2\theta/(\partial\mathbf{x}_C\partial\mathbf{x}_C^T)$, specifically,

$$\frac{\partial^2\theta}{\partial\mathbf{x}_C\partial\mathbf{x}_C^T} = \frac{\partial\left(\frac{\partial\theta}{\partial\mathbf{x}_C}\right)}{\partial\mathbf{x}_C^T} = \frac{\frac{\partial(\mathbf{s}\times\mathbf{p})}{\partial\mathbf{x}_C^T}p^2 - (\mathbf{s}\times\mathbf{p})\frac{\partial p^2}{\partial\mathbf{x}_C^T}}{p^4}. \tag{A.32}$$

By invoking Eqs.(A.1b) and (A.1d), as well as $\partial\mathbf{p}^T/\partial\mathbf{x}_C$ in Eq.(A.16) and $\partial\mathbf{q}^T/\partial\mathbf{x}_C$ in Eq.(A.31), one can get

$$\frac{\partial(\mathbf{s}\times\mathbf{p})}{\partial\mathbf{x}_C^T} = [\mathbf{s}]_\times\frac{\partial\mathbf{p}}{\partial\mathbf{x}_C^T} - [\mathbf{p}]_\times\frac{\partial\mathbf{s}}{\partial\mathbf{x}_C^T} = [\mathbf{s}]_\times, \tag{A.33}$$

and

$$\frac{\partial p^2}{\partial\mathbf{x}_C^T} = 2p\left(\frac{\partial p}{\partial\mathbf{x}_C}\right)^T = 2\mathbf{p}^T\frac{\partial\mathbf{p}}{\partial\mathbf{x}_C^T} = 2\mathbf{p}^T. \tag{A.34}$$

By substituting Eqs.(A.33) and (A.34) into Eq.(A.32), as well as invoking Eq.(A.1f), one can get

$$\begin{aligned}\frac{\partial^2\theta}{\partial\mathbf{x}_C\partial\mathbf{x}_C^T} &= \frac{[\mathbf{s}]_\times p^2 + 2(\mathbf{p}\times\mathbf{s})\mathbf{p}^T}{p^4} \\ &= \frac{\mathbf{p}^T(\mathbf{p}\times\mathbf{s}) - (\mathbf{p}\times\mathbf{s})\mathbf{p}^T + 2(\mathbf{p}\times\mathbf{s})\mathbf{p}^T}{p^4} \\ &= \frac{\mathbf{p}^T(\mathbf{p}\times\mathbf{s}) + (\mathbf{p}\times\mathbf{s})\mathbf{p}^T}{p^4}.\end{aligned} \tag{A.35}$$

Similarly, one can obtain

$$\frac{\partial^2\theta}{\partial\mathbf{x}_D\partial\mathbf{x}_D^T} = \frac{\mathbf{q}^T(\mathbf{s}\times\mathbf{q}) + (\mathbf{s}\times\mathbf{q})\mathbf{q}^T}{q^4}. \tag{A.36}$$

By invoking Eqs.(A.14), (A.24), and (A.35), the gradient of $\partial\theta/\partial\mathbf{x}_A$ with respect to $\mathbf{x}_C$ is $\partial^2\theta/(\partial\mathbf{x}_A\partial\mathbf{x}_C^T)$, i.e.

$$\begin{aligned}\frac{\partial^2\theta}{\partial\mathbf{x}_A\partial\mathbf{x}_C^T} &= \frac{\partial\left(\frac{\partial\theta}{\partial\mathbf{x}_A}\right)}{\partial\mathbf{x}_C^T} \\ &= -(1-s_C)\frac{\partial^2\theta}{\partial\mathbf{x}_C\partial\mathbf{x}_C^T} + \frac{\partial\theta}{\partial\mathbf{x}_C}\frac{\partial s_C}{\partial\mathbf{x}_C^T} \\ &= -(1-s_C)\frac{\mathbf{p}^T(\mathbf{p}\times\mathbf{s}) + (\mathbf{p}\times\mathbf{s})\mathbf{p}^T}{p^4} + \frac{(\mathbf{s}\times\mathbf{p})}{p^2}\frac{\mathbf{s}^T}{\|\mathbf{x}_{AB}\|}.\end{aligned} \tag{A.37}$$

Similarly, one can obtain

$$\begin{aligned}\frac{\partial^2\theta}{\partial\mathbf{x}_D\partial\mathbf{x}_C^T} = \mathbf{0},\ \frac{\partial^2\theta}{\partial\mathbf{x}_B\partial\mathbf{x}_C^T} &= -s_C\frac{\mathbf{p}^T(\mathbf{p}\times\mathbf{s}) + (\mathbf{p}\times\mathbf{s})\mathbf{p}^T}{p^4} - \frac{(\mathbf{s}\times\mathbf{p})}{p^2}\frac{\mathbf{s}^T}{\|\mathbf{x}_{AB}\|}, \\ \frac{\partial^2\theta}{\partial\mathbf{x}_A\partial\mathbf{x}_D^T} &= -(1-s_D)\frac{\mathbf{q}^T(\mathbf{s}\times\mathbf{q}) + (\mathbf{s}\times\mathbf{q})\mathbf{q}^T}{p^4} + \frac{(\mathbf{q}\times\mathbf{s})}{q^2}\frac{\mathbf{s}^T}{\|\mathbf{x}_{AB}\|}, \\ \frac{\partial^2\theta}{\partial\mathbf{x}_B\partial\mathbf{x}_D^T} &= s_D\frac{\mathbf{q}^T(\mathbf{s}\times\mathbf{q}) + (\mathbf{s}\times\mathbf{q})\mathbf{q}^T}{p^4} - \frac{(\mathbf{q}\times\mathbf{s})}{q^2}\frac{\mathbf{s}^T}{\|\mathbf{x}_{AB}\|}.\end{aligned} \tag{A.38}$$

By invoking Eqs.(A.13), (A.14), (A.24), (A.26), (A.37), and (A.38), one can obtain

$$
\begin{aligned}
\frac{\partial^2\theta}{\partial \mathbf{x}_A \partial \mathbf{x}_A^T} &= -(1-s_C)\left(\frac{\partial^2\theta}{\partial \mathbf{x}_A \partial \mathbf{x}_C^T}\right)^T + \frac{\partial\theta}{\partial \mathbf{x}_C}\frac{\partial s_C}{\partial \mathbf{x}_A^T} \\
&\quad - (1-s_D)\left(\frac{\partial^2\theta}{\partial \mathbf{x}_A \partial \mathbf{x}_D^T}\right)^T + \frac{\partial\theta}{\partial \mathbf{x}_D}\frac{\partial s_D}{\partial \mathbf{x}_A^T}, \\
&= (1-s_C)^2\frac{\mathbf{p}^T(\mathbf{p}\times\mathbf{s}) + (\mathbf{p}\times\mathbf{s})\mathbf{p}^T}{p^4} - (1-s_C)\frac{(\mathbf{s}\times\mathbf{p})}{p^2}\frac{\mathbf{s}^T}{\|\mathbf{x}_{AB}\|} \\
&\quad - \frac{(\mathbf{s}\times\mathbf{p})}{p^2}\left(\frac{\mathbf{p}+(1-s_C)\mathbf{x}_{AB}}{\|\mathbf{x}_{AB}\|^2}\right)^T, \\
&\quad (1-s_D)^2\frac{\mathbf{q}^T(\mathbf{s}\times\mathbf{q}) + (\mathbf{s}\times\mathbf{q})\mathbf{q}^T}{q^4} - (1-s_D)\frac{(\mathbf{q}\times\mathbf{s})}{q^2}\frac{\mathbf{s}^T}{\|\mathbf{x}_{AB}\|}, \\
&\quad - \frac{(\mathbf{q}\times\mathbf{s})}{q^2}\left(\frac{\mathbf{q}+(1-s_D)\mathbf{x}_{AB}}{\|\mathbf{x}_{AB}\|^2}\right)^T, \\
&= (1-s_C)^2\frac{\mathbf{p}^T(\mathbf{p}\times\mathbf{s}) + (\mathbf{p}\times\mathbf{s})\mathbf{p}^T}{p^4} \\
&\quad + (1-s_D)^2\frac{\mathbf{q}^T(\mathbf{s}\times\mathbf{q}) + (\mathbf{s}\times\mathbf{q})\mathbf{q}^T}{q^4} \\
&\quad - \frac{(1-s_C)}{\|\mathbf{x}_{AB}\|}\left(\mathbf{s}\frac{(\mathbf{s}\times\mathbf{p})^T}{p^2} + \frac{(\mathbf{s}\times\mathbf{p})}{p^2}\mathbf{s}^T\right) \\
&\quad - \frac{(1-s_D)}{\|\mathbf{x}_{AB}\|}\left(\mathbf{s}\frac{(\mathbf{q}\times\mathbf{s})^T}{q^2} + \frac{(\mathbf{q}\times\mathbf{s})}{q^2}\mathbf{s}^T\right) \\
&\quad - \frac{1}{\|\mathbf{x}_{AB}\|^2}\left(\frac{(\mathbf{s}\times\mathbf{p})}{p^2}\mathbf{p}^T + \frac{(\mathbf{q}\times\mathbf{s})}{q^2}\mathbf{q}^T\right).
\end{aligned}
\tag{A.39}
$$

Similarly, one can get

$$
\begin{aligned}
\frac{\partial^2\theta}{\partial \mathbf{x}_B \partial \mathbf{x}_B^T} &= s_C^2\frac{\mathbf{p}^T(\mathbf{p}\times\mathbf{s}) + (\mathbf{p}\times\mathbf{s})\mathbf{p}^T}{p^4} + s_D^2\frac{\mathbf{q}^T(\mathbf{s}\times\mathbf{q}) + (\mathbf{s}\times\mathbf{q})\mathbf{q}^T}{q^4} \\
&\quad + \frac{s_C}{\|\mathbf{x}_{AB}\|}\left(\mathbf{s}\frac{(\mathbf{s}\times\mathbf{p})^T}{p^2} + \frac{(\mathbf{s}\times\mathbf{p})}{p^2}\mathbf{s}^T\right) \\
&\quad + \frac{s_D}{\|\mathbf{x}_{AB}\|}\left(\mathbf{s}\frac{(\mathbf{q}\times\mathbf{s})^T}{q^2} + \frac{(\mathbf{q}\times\mathbf{s})}{q^2}\mathbf{s}^T\right) \\
&\quad - \frac{1}{\|\mathbf{x}_{AB}\|^2}\left(\frac{(\mathbf{s}\times\mathbf{p})}{p^2}\mathbf{p}^T + \frac{(\mathbf{q}\times\mathbf{s})}{q^2}\mathbf{q}^T\right)
\end{aligned}
\tag{A.40}
$$

and

$$\begin{aligned}\frac{\partial^2\theta}{\partial\mathbf{x}_B\partial\mathbf{x}_A^T} &= s_C(1-s_C)\frac{\mathbf{p}^T(\mathbf{p}\times\mathbf{s})+(\mathbf{p}\times\mathbf{s})\mathbf{p}^T}{p^4}\\ &+ s_D(1-s_D)\frac{\mathbf{q}^T(\mathbf{s}\times\mathbf{q})+(\mathbf{s}\times\mathbf{q})\mathbf{q}^T}{q^4}\\ &- \frac{s_C}{\|\mathbf{x}_{AB}\|}\left(\mathbf{s}\frac{(\mathbf{s}\times\mathbf{p})^T}{p^2}+\frac{(\mathbf{s}\times\mathbf{p})}{p^2}\mathbf{s}^T\right)\\ &- \frac{s_D}{\|\mathbf{x}_{AB}\|}\left(\mathbf{s}\frac{(\mathbf{q}\times\mathbf{s})^T}{q^2}+\frac{(\mathbf{q}\times\mathbf{s})}{q^2}\mathbf{s}^T\right)\\ &+ \frac{1}{\|\mathbf{x}_{AB}\|^2}\left(\frac{(\mathbf{s}\times\mathbf{p})}{p^2}\mathbf{p}^T+\frac{(\mathbf{q}\times\mathbf{s})}{q^2}\mathbf{q}^T\right)\\ &+ \frac{1}{\|\mathbf{x}_{AB}\|}\left(\frac{(\mathbf{s}\times\mathbf{p})}{p^2}+\frac{(\mathbf{q}\times\mathbf{s})}{q^2}\right)\mathbf{s}^T.\end{aligned} \tag{A.41}$$

Finally, all the components in Eqs.(A.24), (A.26), (A.29), (A.30), (A.35), (A.36)-(A.41) can be used to assembly the gradient $\partial\theta/\partial\mathbf{x}^e$ and Hessian $\partial^2\theta/(\partial\mathbf{x}^e\partial(\mathbf{x}^e)^T)$, i.e.

$$\frac{\partial\theta}{\partial\mathbf{x}^e}=\begin{bmatrix}\frac{\partial\theta}{\partial\mathbf{x}_A}\\ \frac{\partial\theta}{\partial\mathbf{x}_B}\\ \frac{\partial\theta}{\partial\mathbf{x}_C}\\ \frac{\partial\theta}{\partial\mathbf{x}_D}\end{bmatrix},\frac{\partial^2\theta}{\partial\mathbf{x}^e\partial(\mathbf{x}^e)^T}=\begin{bmatrix}\frac{\partial^2\theta}{\partial\mathbf{x}_A\partial\mathbf{x}_A^T} & \frac{\partial^2\theta}{\partial\mathbf{x}_A\partial\mathbf{x}_B^T} & \frac{\partial^2\theta}{\partial\mathbf{x}_A\partial\mathbf{x}_C^T} & \frac{\partial^2\theta}{\partial\mathbf{x}_A\partial\mathbf{x}_D^T}\\ \frac{\partial^2\theta}{\partial\mathbf{x}_B\partial\mathbf{x}_A^T} & \frac{\partial^2\theta}{\partial\mathbf{x}_B\partial\mathbf{x}_B^T} & \frac{\partial^2\theta}{\partial\mathbf{x}_B\partial\mathbf{x}_C^T} & \frac{\partial^2\theta}{\partial\mathbf{x}_B\partial\mathbf{x}_D^T}\\ \frac{\partial^2\theta}{\partial\mathbf{x}_C\partial\mathbf{x}_A^T} & \frac{\partial^2\theta}{\partial\mathbf{x}_C\partial\mathbf{x}_B^T} & \frac{\partial^2\theta}{\partial\mathbf{x}_C\partial\mathbf{x}_C^T} & \frac{\partial^2\theta}{\partial\mathbf{x}_C\partial\mathbf{x}_D^T}\\ \frac{\partial^2\theta}{\partial\mathbf{x}_D\partial\mathbf{x}_A^T} & \frac{\partial^2\theta}{\partial\mathbf{x}_D\partial\mathbf{x}_B^T} & \frac{\partial^2\theta}{\partial\mathbf{x}_D\partial\mathbf{x}_C^T} & \frac{\partial^2\theta}{\partial\mathbf{x}_D\partial\mathbf{x}_D^T}\end{bmatrix}, \tag{A.42}$$

in which

$$\begin{aligned}\frac{\partial^2\theta}{\partial\mathbf{x}_C\partial\mathbf{x}_A^T}&=\left(\frac{\partial^2\theta}{\partial\mathbf{x}_A\partial\mathbf{x}_C^T}\right)^T,\frac{\partial^2\theta}{\partial\mathbf{x}_C\partial\mathbf{x}_D^T}=\left(\frac{\partial^2\theta}{\partial\mathbf{x}_D\partial\mathbf{x}_C^T}\right)^T,\\ \frac{\partial^2\theta}{\partial\mathbf{x}_C\partial\mathbf{x}_B^T}&=\left(\frac{\partial^2\theta}{\partial\mathbf{x}_B\partial\mathbf{x}_C^T}\right)^T,\frac{\partial^2\theta}{\partial\mathbf{x}_D\partial\mathbf{x}_A^T}=\left(\frac{\partial^2\theta}{\partial\mathbf{x}_A\partial\mathbf{x}_D^T}\right)^T,\\ \frac{\partial^2\theta}{\partial\mathbf{x}_D\partial\mathbf{x}_B^T}&=\left(\frac{\partial^2\theta}{\partial\mathbf{x}_B\partial\mathbf{x}_D^T}\right)^T,\frac{\partial^2\theta}{\partial\mathbf{x}_A\partial\mathbf{x}_B^T}=\left(\frac{\partial^2\theta}{\partial\mathbf{x}_B\partial\mathbf{x}_A^T}\right)^T,\end{aligned} \tag{A.43}$$

due to the Hessian is symmetric.

Appendix B

Derivation of Displacement-based Strain Components and Their Spatial Discretization for Solid-Shell Elements

B.1 The Strain Components

The natural Green–Lagrangian components are

$$\begin{aligned}\varepsilon_{ij} &= \frac{1}{2}\left(g_{ij} - (g_{ij})_0\right) \\ &= \frac{1}{2}\left(\mathbf{x}_{,i}^T\mathbf{x}_{,j} - \mathbf{X}_{,i}^T\mathbf{X}_{,j}\right) \\ &= \frac{1}{2}\left(\mathbf{X}_{,i}^T\mathbf{U}_{,j} + \mathbf{X}_{,j}^T\mathbf{U}_{,i} + \mathbf{U}_{,i}^T\mathbf{U}_{,j}\right), \quad i,j = \xi,\eta,\zeta.\end{aligned} \tag{B.1}$$

For the in-plane natural strain components ($i,j = \alpha,\beta = \xi,\eta$), which characterize the membrane and bending deformation, substituting $\mathbf{X} =$

$\mathbf{X}_o + \zeta\mathbf{X}_n$ and $\mathbf{U} = \mathbf{U}_o + \zeta\mathbf{U}_n$ into Eq.(B.1) yields

$$\begin{aligned}
\mathbf{X}_{,\alpha}^T\mathbf{U}_{,\beta} &= \left(\mathbf{X}_o + \zeta\mathbf{X}_n\right)_{,\alpha}^T\left(\mathbf{U}_o + \zeta\mathbf{U}_n\right)_{,\beta} \\
&= \mathbf{X}_{o,\alpha}^T\mathbf{U}_{o,\beta} + \zeta\left(\mathbf{X}_{n,\alpha}^T\mathbf{U}_{o,\beta} + \mathbf{X}_{o,\alpha}^T\mathbf{U}_{n,\beta}\right) + \zeta^2\mathbf{X}_{n,\alpha}^T\mathbf{U}_{n,\beta}, \\
\mathbf{X}_{,\beta}^T\mathbf{U}_{,\alpha} &= \mathbf{X}_{o,\beta}^T\mathbf{U}_{o,\alpha} + \zeta\left(\mathbf{X}_{n,\beta}^T\mathbf{U}_{o,\alpha} + \mathbf{X}_{o,\beta}^T\mathbf{U}_{n,\alpha}\right) + \zeta^2\mathbf{X}_{n,\beta}^T\mathbf{U}_{n,\alpha}, \\
\mathbf{U}_{,\alpha}^T\mathbf{U}_{,\beta} &= \mathbf{U}_{o,\alpha}^T\mathbf{U}_{o,\beta} + \zeta\left(\mathbf{U}_{n,\alpha}^T\mathbf{U}_{o,\beta} + \mathbf{U}_{o,\alpha}^T\mathbf{U}_{n,\beta}\right) + \zeta^2\mathbf{U}_{n,\alpha}^T\mathbf{U}_{n,\beta}.
\end{aligned} \tag{B.2}$$

Combining all terms in Eq.(B.2), the in-plane natural strain components $\varepsilon_{\alpha\beta}$ are

$$\begin{aligned}
\varepsilon_{\alpha\beta} = &\frac{1}{2}\left(\mathbf{X}_{o,\alpha}^T\mathbf{U}_{o,\beta} + \mathbf{X}_{o,\beta}^T\mathbf{U}_{o,\alpha} + \mathbf{U}_{o,\alpha}^T\mathbf{U}_{o,\beta}\right) \\
&+ \zeta\cdot\frac{1}{2}\left(\mathbf{X}_{n,\alpha}^T\mathbf{U}_{o,\beta} + \mathbf{X}_{o,\alpha}^T\mathbf{U}_{n,\beta} + \mathbf{X}_{n,\beta}^T\mathbf{U}_{o,\alpha}\right. \\
&\qquad\left. + \mathbf{X}_{o,\beta}^T\mathbf{U}_{n,\alpha} + \mathbf{U}_{n,\alpha}^T\mathbf{U}_{o,\beta} + \mathbf{U}_{o,\alpha}^T\mathbf{U}_{n,\beta}\right) \\
&+ \zeta^2\cdot\frac{1}{2}\left(\mathbf{X}_{n,\alpha}^T\mathbf{U}_{n,\beta} + \mathbf{X}_{n,\beta}^T\mathbf{U}_{n,\alpha} + \mathbf{U}_{n,\alpha}^T\mathbf{U}_{n,\beta}\right),
\end{aligned} \tag{B.3}$$

where the ζ^2 terms are neglected. The natural membrane strain components are

$$\varepsilon_{m\alpha\beta} = \frac{1}{2}\left(\mathbf{X}_{o,\alpha}^T\mathbf{U}_{o,\beta} + \mathbf{X}_{o,\beta}^T\mathbf{U}_{o,\alpha} + \mathbf{U}_{o,\alpha}^T\mathbf{U}_{o,\beta}\right), \tag{B.4}$$

which can be expressed in Voigt notation, i.e.

$$\left\{\begin{array}{c}\varepsilon_{m\xi\xi}\\ \varepsilon_{m\eta\eta}\\ \gamma_{m\xi\eta}\end{array}\right\} = \left[\begin{array}{c}\mathbf{X}_{o,\xi}^T\mathbf{U}_{o,\xi} + \frac{1}{2}\mathbf{U}_{o,\xi}^T\mathbf{U}_{o,\xi}\\ \mathbf{X}_{o,\eta}^T\mathbf{U}_{o,\eta} + \frac{1}{2}\mathbf{U}_{o,\eta}^T\mathbf{U}_{o,\eta}\\ \mathbf{X}_{o,\xi}^T\mathbf{U}_{o,\eta} + \mathbf{X}_{o,\eta}^T\mathbf{U}_{o,\xi} + \mathbf{U}_{o,\xi}^T\mathbf{U}_{o,\eta}\end{array}\right]. \tag{B.5}$$

The natural bending strain components are

$$\begin{aligned}
\varepsilon_{b\alpha\beta} = &\frac{1}{2}\left(\mathbf{X}_{n,\alpha}^T\mathbf{U}_{o,\beta} + \mathbf{X}_{o,\alpha}^T\mathbf{U}_{n,\beta} + \mathbf{X}_{n,\beta}^T\mathbf{U}_{o,\alpha}\right. \\
&\left. + \mathbf{X}_{o,\beta}^T\mathbf{U}_{n,\alpha} + \mathbf{U}_{n,\alpha}^T\mathbf{U}_{o,\beta} + \mathbf{U}_{o,\alpha}^T\mathbf{U}_{n,\beta}\right),
\end{aligned} \tag{B.6}$$

which can be expressed in Voigt notation, i.e.

$$\left\{\begin{array}{c}\varepsilon_{b\xi\xi}\\ \varepsilon_{b\eta\eta}\\ \gamma_{b\xi\eta}\end{array}\right\}=\left[\begin{array}{c}\mathbf{X}_{n,\xi}^{T}\mathbf{U}_{o,\xi}+\mathbf{X}_{o,\xi}^{T}\mathbf{U}_{n,\xi}+\mathbf{U}_{o,\xi}^{T}\mathbf{U}_{n,\xi}\\ \mathbf{X}_{n,\eta}^{T}\mathbf{U}_{o,\eta}+\mathbf{X}_{o,\eta}^{T}\mathbf{U}_{n,\eta}+\mathbf{U}_{o,\eta}^{T}\mathbf{U}_{n,\eta}\\ \left(\begin{array}{c}\mathbf{X}_{n,\xi}^{T}\mathbf{U}_{o,\eta}+\mathbf{X}_{o,\xi}^{T}\mathbf{U}_{n,\eta}+\mathbf{X}_{n,\eta}^{T}\mathbf{U}_{o,\xi}\\ +\mathbf{X}_{o,\eta}^{T}\mathbf{U}_{n,\xi}+\mathbf{U}_{n,\xi}^{T}\mathbf{U}_{o,\eta}+\mathbf{U}_{o,\xi}^{T}\mathbf{U}_{n,\eta}\end{array}\right)\end{array}\right]. \tag{B.7}$$

For the natural shear components ($i=\zeta, j=\xi,\eta$), substituting $\mathbf{X}=\mathbf{X}_o+\zeta\mathbf{X}_n$ and $\mathbf{U}=\mathbf{U}_o+\zeta\mathbf{U}_n$ into Eq.(B.1) yields

$$\begin{aligned}\mathbf{X}_{,\zeta}^{T}\mathbf{U}_{,j}&=\left(\mathbf{X}_o+\zeta\mathbf{X}_n\right)_{,\zeta}^{T}\left(\mathbf{U}_o+\zeta\mathbf{U}_n\right)_{,j}=\mathbf{X}_n^{T}\mathbf{U}_{o,j}+\zeta\mathbf{X}_n^{T}\mathbf{U}_{n,j},\\ \mathbf{X}_{,j}^{T}\mathbf{U}_{,\zeta}&=\left(\mathbf{X}_o+\zeta\mathbf{X}_n\right)_{,j}^{T}\left(\mathbf{U}_o+\zeta\mathbf{U}_n\right)_{,\zeta}=\mathbf{X}_{o,j}^{T}\mathbf{U}_n+\zeta\mathbf{X}_{n,j}^{T}\mathbf{U}_n,\\ \mathbf{U}_{,\zeta}^{T}\mathbf{U}_{,j}&=\mathbf{U}_n^{T}\mathbf{U}_{o,j}+\zeta\mathbf{U}_n^{T}\mathbf{U}_{n,j}.\end{aligned} \tag{B.8}$$

Combining all terms in Eq. (B.8), one can get

$$\begin{aligned}\gamma_{\zeta j}&=2\varepsilon_{\zeta j}\\ &=\mathbf{X}_n^{T}\mathbf{U}_{o,j}+\mathbf{X}_{o,j}^{T}\mathbf{U}_n+\mathbf{U}_n^{T}\mathbf{U}_{o,j}+\zeta(\mathbf{X}_n^{T}\mathbf{U}_{n,j}+\mathbf{X}_{n,j}^{T}\mathbf{U}_n+\mathbf{U}_n^{T}\mathbf{U}_{n,j}),\end{aligned} \tag{B.9}$$

where the ζ terms are neglected. The natural shear strain components are

$$\left\{\begin{array}{c}\gamma_{\zeta\xi}\\ \gamma_{\zeta\eta}\end{array}\right\}=\left[\begin{array}{c}\mathbf{X}_n^{T}\mathbf{U}_{o,\xi}+\mathbf{X}_{o,\xi}^{T}\mathbf{U}_n+\mathbf{U}_n^{T}\mathbf{U}_{o,\xi}\\ \mathbf{X}_n^{T}\mathbf{U}_{o,\eta}+\mathbf{X}_{o,\eta}^{T}\mathbf{U}_n+\mathbf{U}_n^{T}\mathbf{U}_{o,\eta}\end{array}\right]. \tag{B.10}$$

Similarly, for the natural thickness strain component ($i=\zeta, j=\zeta$), one can get

$$\begin{aligned}\varepsilon_{\zeta\zeta}&=\left(\mathbf{X}_{,\zeta}^{T}\mathbf{U}_{,\zeta}+\mathbf{X}_{,\zeta}^{T}\mathbf{U}_{,\zeta}+\mathbf{U}_{,\zeta}^{T}\mathbf{U}_{,\zeta}\right)/2\\ &=\mathbf{X}_n^{T}\mathbf{U}_n+\mathbf{U}_n^{T}\mathbf{U}_n/2.\end{aligned} \tag{B.11}$$

B.2 Spatial Discretization

The displacemnts in Eq.(5.1) of Chapter 5 can be rewritten as

$$\mathbf{U}=\left(\mathbf{N}_o+\zeta\mathbf{N}_n\right)\mathbf{d}. \tag{B.12}$$

For instance, the nodal dofs vector of the bilinear solid-shell element is

$$\mathbf{d} = \begin{bmatrix} \mathbf{U}_{o1}^T & \mathbf{U}_{n1}^T & \mathbf{U}_{o2}^T & \mathbf{U}_{n2}^T & \mathbf{U}_{o3}^T & \mathbf{U}_{n3}^T & \mathbf{U}_{o4}^T & \mathbf{U}_{n4}^T \end{bmatrix}^T, \tag{B.13}$$

the self-defined $\mathbf{N}_o$ and $\mathbf{N}_n$ are

$$\mathbf{N}_o = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 & 0 \end{bmatrix} \otimes \mathbf{I}, \tag{B.14}$$

and

$$\mathbf{N}_n = \begin{bmatrix} 0 & N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 \end{bmatrix} \otimes \mathbf{I}, \tag{B.15}$$

respectively. $\mathbf{I}$ is the third-order identity matrix and $\otimes$ is the Kronecker product. The nodal dofs vector of the biquadratic solid-shell element can be extended from the Eqs.(B.13), (B.14), and (B.15) in bilinear solid-shell element. By introducing Eq.(B.12) into Eqs.(B.5), (B.7), (B.10) and (B.11), the natural membrane strain is discretized as

$$\begin{Bmatrix} \varepsilon_{m\xi\xi} \\ \varepsilon_{m\eta\eta} \\ \gamma_{m\xi\eta} \end{Bmatrix} = \begin{bmatrix} \mathbf{X}_{o,\xi}^T \mathbf{N}_{o,\xi} \mathbf{d} + \frac{1}{2} \mathbf{d}^T \mathbf{N}_{o,\xi}^T \mathbf{N}_{o,\xi} \mathbf{d} \\ \mathbf{X}_{o,\eta}^T \mathbf{N}_{o,\eta} \mathbf{d} + \frac{1}{2} \mathbf{d}^T \mathbf{N}_{o,\eta}^T \mathbf{N}_{o,\eta} \mathbf{d} \\ \mathbf{X}_{o,\xi}^T \mathbf{N}_{o,\eta} \mathbf{d} + \mathbf{X}_{o,\eta}^T \mathbf{N}_{o,\xi} \mathbf{d} + \mathbf{d}^T \mathbf{N}_{o,\xi}^T \mathbf{N}_{o,\eta} \mathbf{d} \end{bmatrix}, \tag{B.16}$$

the natural bending strain is discretized as

$$\begin{Bmatrix} \varepsilon_{b\xi\xi} \\ \varepsilon_{b\eta\eta} \\ \gamma_{b\xi\eta} \end{Bmatrix} = \begin{bmatrix} \mathbf{X}_{n,\xi}^T \mathbf{N}_{o,\xi} \mathbf{d} + \mathbf{X}_{o,\xi}^T \mathbf{N}_{n,\xi} \mathbf{d} + \mathbf{d}^T \mathbf{N}_{o,\xi}^T \mathbf{N}_{n,\xi} \mathbf{d} \\ \mathbf{X}_{n,\eta}^T \mathbf{N}_{o,\eta} \mathbf{d} + \mathbf{X}_{o,\eta}^T \mathbf{N}_{n,\eta} \mathbf{d} + \mathbf{d}^T \mathbf{N}_{o,\eta}^T \mathbf{N}_{n,\eta} \mathbf{d} \\ \left(\begin{array}{l} \mathbf{X}_{n,\xi}^T \mathbf{N}_{o,\eta} \mathbf{d} + \mathbf{X}_{o,\xi}^T \mathbf{N}_{n,\eta} \mathbf{d} + \mathbf{X}_{n,\eta}^T \mathbf{N}_{o,\xi} \mathbf{d} + \\ \mathbf{X}_{o,\eta}^T \mathbf{N}_{n,\xi} \mathbf{d} + \mathbf{d}^T \mathbf{N}_{n,\xi}^T \mathbf{N}_{o,\eta} \mathbf{d} + \mathbf{d}^T \mathbf{N}_{o,\xi}^T \mathbf{N}_{n,\eta} \mathbf{d} \end{array} \right) \end{bmatrix}, \tag{B.17}$$

the natural shear strain is discretized as

$$\begin{Bmatrix} \gamma_{\zeta\xi} \\ \gamma_{\zeta\eta} \end{Bmatrix} = \begin{bmatrix} \mathbf{X}_n^T \mathbf{N}_{o,\xi} \mathbf{d} + \mathbf{X}_{o,\xi}^T \mathbf{N}_n \mathbf{d} + \mathbf{d}^T \mathbf{N}_n^T \mathbf{N}_{o,\xi} \mathbf{d} \\ \mathbf{X}_n^T \mathbf{N}_{o,\eta} \mathbf{d} + \mathbf{X}_{o,\eta}^T \mathbf{N}_n \mathbf{d} + \mathbf{d}^T \mathbf{N}_n^T \mathbf{N}_{o,\eta} \mathbf{d} \end{bmatrix}, \tag{B.18}$$

and the natural thickness strain is discretized as

$$\varepsilon_{\zeta\zeta} = \mathbf{X}_n^T \mathbf{N}_n \mathbf{d} + \mathbf{d}^T \mathbf{N}_n^T \mathbf{N}_n \mathbf{d}/2. \tag{B.19}$$

Appendix C

Derivation of Strain Transformation Matrix

The Green–Lagrangian strain tensor $\boldsymbol{\varepsilon}$ can be expressed using different basis vectors. In particular,

$$\boldsymbol{\varepsilon} = \varepsilon_{ij}\mathbf{e}_i\mathbf{e}_j = \varepsilon_{mn}\tilde{\mathbf{e}}_m\tilde{\mathbf{e}}_n, \tag{C.1}$$

where $\mathbf{e}_i$ and $\mathbf{e}_j$ $(i, j = \xi, \eta, \zeta)$ are the basis vectors of the natural coordinate frame, and $\tilde{\mathbf{e}}_m$ and $\tilde{\mathbf{e}}_n$ $(m, n = \tilde{X}, \tilde{Y}, \tilde{Z})$ are the basis vectors of the local Cartesian coordinate frame. These basis vectors can be computed by: $\mathbf{e}_\xi = \mathbf{X}_{o,\xi}$, $\mathbf{e}_\eta = \mathbf{X}_{o,\eta}$ and $\mathbf{e}_\zeta = \mathbf{X}_n$, as well as $\tilde{\mathbf{e}}_{\tilde{X}} = (\mathbf{X}_{o2} - \mathbf{X}_{o1})/|(\mathbf{X}_{o2} - \mathbf{X}_{o1})|$, $\tilde{\mathbf{e}}_{\tilde{Z}} = \mathbf{X}_n$ and $\tilde{\mathbf{e}}_{\tilde{Y}} = \tilde{\mathbf{e}}_{\tilde{X}} \times \tilde{\mathbf{e}}_{\tilde{Z}}$. $\mathbf{X}_{o1}$ and $\mathbf{X}_{o2}$ are the initial position of node 1 and node 2 in an element or a stencil. In this study, the initial state of the origami panels is flat.

The components of the strain tensor in the local Cartesian coordinates can be computed by

$$\varepsilon_{mn} = \tilde{\mathbf{e}}_m \cdot \varepsilon_{ij}\mathbf{e}_i\mathbf{e}_j \cdot \tilde{\mathbf{e}}_n = c_{mi}c_{nj}\varepsilon_{ij} \tag{C.2}$$

where $c_{mi} = \tilde{\mathbf{e}}_m \cdot \mathbf{e}_i$ is the cosine of the angle between the local Cartesian basis vector $\tilde{\mathbf{e}}_m$ and the natural basis vector $\mathbf{e}_i$.

The membrane strain $\boldsymbol{\varepsilon}_m = \begin{bmatrix} \varepsilon_{m\tilde{X}\tilde{X}} & \varepsilon_{m\tilde{Y}\tilde{Y}} & 2\varepsilon_{m\tilde{X}\tilde{Y}} \end{bmatrix}^T$ and bending strain $\boldsymbol{\varepsilon}_b = \begin{bmatrix} \varepsilon_{b\tilde{X}\tilde{X}} & \varepsilon_{b\tilde{Y}\tilde{Y}} & 2\varepsilon_{b\tilde{X}\tilde{Y}} \end{bmatrix}^T$ in the local Cartesian coordinates are

related to those in the natural coordinate frame by the following transformation

$$\left\{ \begin{array}{c} \varepsilon_{m\tilde{X}\tilde{X}} \\ \varepsilon_{m\tilde{Y}\tilde{Y}} \\ 2\varepsilon_{m\tilde{X}\tilde{Y}} \end{array} \right\} = \mathbf{T}_{mb} \left\{ \begin{array}{c} \varepsilon_{m\xi\xi} \\ \varepsilon_{m\eta\eta} \\ 2\varepsilon_{m\xi\eta} \end{array} \right\}, \left\{ \begin{array}{c} \varepsilon_{b\tilde{X}\tilde{X}} \\ \varepsilon_{b\tilde{Y}\tilde{Y}} \\ 2\varepsilon_{b\tilde{X}\tilde{Y}} \end{array} \right\} = \mathbf{T}_{mb} \left\{ \begin{array}{c} \varepsilon_{b\xi\xi} \\ \varepsilon_{b\eta\eta} \\ 2\varepsilon_{b\xi\eta} \end{array} \right\}, \quad \text{(C.3)}$$

where $\mathbf{T}_{mb}$ is

$$\begin{bmatrix} c_{\xi\tilde{X}}c_{\xi\tilde{X}} & c_{\xi\tilde{Y}}c_{\xi\tilde{Y}} & c_{\xi\tilde{X}}c_{\xi\tilde{Y}} \\ c_{\eta\tilde{X}}c_{\eta\tilde{X}} & c_{\eta\tilde{Y}}c_{\eta\tilde{Y}} & c_{\eta\tilde{X}}c_{\eta\tilde{Y}} \\ c_{\xi\tilde{X}}c_{\eta\tilde{X}} + c_{\eta\tilde{X}}c_{\xi\tilde{X}} & c_{\xi\tilde{Y}}c_{\eta\tilde{Y}} + c_{\eta\tilde{Y}}c_{\xi\tilde{Y}} & c_{\xi\tilde{X}}c_{\eta\tilde{Y}} + c_{\xi\tilde{Y}}c_{\eta\tilde{X}} \end{bmatrix}^{-1}. \quad \text{(C.4)}$$

The transverse shear strain $\boldsymbol{\gamma} = \begin{bmatrix} \gamma_{\tilde{Z}\tilde{X}} & \gamma_{\tilde{Z}\tilde{Y}} \end{bmatrix}^T$ in the local Cartesian coordinates can be transformed by

$$\left\{ \begin{array}{c} \gamma_{\tilde{Z}\tilde{X}} \\ \gamma_{\tilde{Z}\tilde{Y}} \end{array} \right\} = \mathbf{T}_s \left\{ \begin{array}{c} \gamma_{\zeta\xi} \\ \gamma_{\zeta\eta} \end{array} \right\}, \quad \text{(C.5)}$$

where

$$\mathbf{T}_s = \begin{bmatrix} c_{\zeta\tilde{Z}}c_{\xi\tilde{X}} & c_{\zeta\tilde{Z}}c_{\xi\tilde{Y}} \\ c_{\zeta\tilde{Z}}c_{\eta\tilde{X}} & c_{\zeta\tilde{Z}}c_{\eta\tilde{Y}} \end{bmatrix}^{-1} \quad \text{(C.6)}$$

Lastly, the thickness strain component $\varepsilon_{\tilde{Z}\tilde{Z}}$ is transformed by

$$\varepsilon_{\tilde{Z}\tilde{Z}} = T_t \varepsilon_{\zeta\zeta}, \quad \text{(C.7)}$$

where

$$T_t = (c_{\zeta\tilde{Z}}c_{\zeta\tilde{Z}})^{-1}. \quad \text{(C.8)}$$

Appendix D

Corotational Transformation

Consider a point $\tilde{\mathbf{x}}$ on the shell stencil in the current corotational frame with origin at $\mathbf{x}_o$. The triangle-centered and edge-based stencils respectively take $\mathbf{x}_1$ and $\mathbf{x}_2$ as their origins (see Figures 3.2 and 3.5). The point $\tilde{\mathbf{x}}$ is related to its counterpart $\mathbf{x}$ in the global Cartesian frame by the transformation

$$\tilde{\mathbf{x}} = \begin{bmatrix} \mathbf{n}_{\tilde{x}} & \mathbf{n}_{\tilde{y}} & \mathbf{n}_{\tilde{z}} \end{bmatrix}^T (\mathbf{x} - \mathbf{x}_o), \tag{D.1}$$

where $\mathbf{n}_{\tilde{x}}$, $\mathbf{n}_{\tilde{y}}$ and $\mathbf{n}_{\tilde{z}}$ are unit vectors of the local coordinate axes $(\tilde{x}, \tilde{y}, \tilde{z})$. For the triangle-centered stencil, the $\tilde{x}$-axis direction is defined as $\mathbf{n}_{\tilde{x}} = \mathbf{x}_{12}/\|\mathbf{x}_{12}\|$ (see Figures 3.1 and 3.2), while for the edge-based stencil, it is defined as $\mathbf{n}_{\tilde{x}} = \mathbf{x}_{23}/\|\mathbf{x}_{23}\|$ (see Figure 3.1). The $\tilde{z}$-axis direction $\mathbf{n}_{\tilde{z}}$ is computed using Eq.(3.6) for the edge stencil and Eq.(3.18) for the triangle-centered stencil. The $\tilde{y}$-axis direction, $\mathbf{n}_{\tilde{y}} = \mathbf{n}_{\tilde{z}} \times \mathbf{n}_{\tilde{x}}$, is determined to according to the right-hand rule. Similarly, points in the initial corotational frame can be transformed using the same approach.

Appendix E

Gradient of Normal Direction of Corotational Shell Stencil

The direction of $\tilde{z}$-axis, which corresponds to the normal direction of the corotational shell stencil, is given by

$$\mathbf{n}_{\tilde{z}} = \frac{\mathbf{n}_c}{\|\mathbf{n}_c\|}, \tag{E.1}$$

where $\mathbf{n}_c$ is the unnormalized normal vector. The gradient of the $\tilde{z}$-axis direction is then expressed as

$$\frac{\partial \mathbf{n}_{\tilde{z}}}{\partial (\mathbf{x}^e)^T} = (\mathbf{I} - \mathbf{n}_{\tilde{z}} \mathbf{n}_{\tilde{z}}^T) \frac{1}{\|\mathbf{n}_c\|} \frac{\partial \mathbf{n}_c}{\partial (\mathbf{x}^e)^T}. \tag{E.2}$$

For the corotational edge-based hinge bending model, let $\mathbf{p} = \mathbf{x}_{P1}$, $\mathbf{q} = \mathbf{x}_{Q4}$ as the triangle altitude vectors and $\mathbf{s} = \mathbf{x}_{23}/\|\mathbf{x}_{23}\|$ as the edge direction vector. The gradient of $\mathbf{n}_c$ is

$$\frac{\partial \mathbf{n}_c}{\partial (\mathbf{x}^e)^T} = \frac{\partial \left(\frac{\mathbf{p}}{\|\mathbf{p}\|}\right)}{\partial (\mathbf{x}^e)^T} + \frac{\partial \left(\frac{\mathbf{q}}{\|\mathbf{q}\|}\right)}{\partial (\mathbf{x}^e)^T}, \tag{E.3}$$

where the gradients of the triangle altitude directions are

$$\frac{\partial \left(\frac{\mathbf{p}}{\|\mathbf{p}\|}\right)}{\partial (\mathbf{x}^e)^T} = \left(\frac{\|\mathbf{p}\|\mathbf{I} - \mathbf{p}\mathbf{p}^T}{\|\mathbf{p}\|^3}\right) \frac{\partial \mathbf{p}}{\partial (\mathbf{x}^e)^T} \text{ and } \frac{\partial \left(\frac{\mathbf{q}}{\|\mathbf{q}\|}\right)}{\partial (\mathbf{x}^e)^T} = \left(\frac{\|\mathbf{q}\|\mathbf{I} - \mathbf{q}\mathbf{q}^T}{\|\mathbf{q}\|^3}\right) \frac{\partial \mathbf{q}}{\partial (\mathbf{x}^e)^T}. \tag{E.4}$$

The components of $\partial \mathbf{p}/\partial(\mathbf{x}^e)^T$ are

$$\begin{aligned} \frac{\partial \mathbf{p}}{\partial(\mathbf{x}_1)^T} = \mathbf{I} - \mathbf{s}\mathbf{s}^T, \frac{\partial \mathbf{p}}{\partial(\mathbf{x}_2)^T} = -\left(1 - s_p\right)\left(\mathbf{I} - \mathbf{s}\mathbf{s}^T\right) + \frac{\mathbf{s}\mathbf{p}^T}{\|\mathbf{x}_{23}\|}, \\ \frac{\partial \mathbf{p}}{\partial(\mathbf{x}_3)^T} = -s_p\left(\mathbf{I} - \mathbf{s}\mathbf{s}^T\right) - \frac{\mathbf{s}\mathbf{p}^T}{\|\mathbf{x}_{23}\|}, \frac{\partial \mathbf{p}}{\partial(\mathbf{x}_4)^T} = \mathbf{0}, \end{aligned} \tag{E.5}$$

and the components of $\partial \mathbf{q}/\partial(\mathbf{x}^e)^T$ are

$$\begin{aligned} \frac{\partial \mathbf{q}}{\partial(\mathbf{x}_1)^T} = \mathbf{0}, \frac{\partial \mathbf{q}}{\partial(\mathbf{x}_2)^T} = -\left(1 - s_q\right)\left(\mathbf{I} - \mathbf{s}\mathbf{s}^T\right) + \frac{\mathbf{s}\mathbf{q}^T}{\|\mathbf{x}_{23}\|}, \\ \frac{\partial \mathbf{q}}{\partial(\mathbf{x}_3)^T} = -s_q\left(\mathbf{I} - \mathbf{s}\mathbf{s}^T\right) - \frac{\mathbf{s}\mathbf{q}^T}{\|\mathbf{x}_{23}\|}, \frac{\partial \mathbf{q}}{\partial(\mathbf{x}_4)^T} = \mathbf{I} - \mathbf{s}\mathbf{s}^T. \end{aligned} \tag{E.6}$$

Here, the coefficients $s_p = \mathbf{x}_{21} \cdot \mathbf{x}_{23}/\|\mathbf{x}_{23}\|^2$ and $s_q = \mathbf{x}_{24} \cdot \mathbf{x}_{23}/\|\mathbf{x}_{23}\|^2$ are defined according to Eq.(2.13) in Chapter 2.

For corotational FVM/smoothed hinge bending models, the normal $\mathbf{n}_c$ is

$$\mathbf{n}_c = \mathbf{x}_{12} \times \mathbf{x}_{13}, \tag{E.7}$$

so the gradient components of $\partial \mathbf{n}_c/\partial(\mathbf{x}^e)^T$ are

$$\begin{aligned} \frac{\partial \mathbf{n}_c}{\partial(\mathbf{x}_1)^T} = (\mathbf{x}_3 - \mathbf{x}_2)^\times, \frac{\partial \mathbf{n}_c}{\partial(\mathbf{x}_3)^T} = (\mathbf{x}_2 - \mathbf{x}_1)^\times, \frac{\partial \mathbf{n}_c}{\partial(\mathbf{x}_5)^T} = \mathbf{0}, \\ \frac{\partial \mathbf{n}_c}{\partial(\mathbf{x}_2)^T} = (\mathbf{x}_1 - \mathbf{x}_3)^\times, \frac{\partial \mathbf{n}_c}{\partial(\mathbf{x}_4)^T} = \mathbf{0}, \frac{\partial \mathbf{n}_c}{\partial(\mathbf{x}_6)^T} = \mathbf{0}, \end{aligned} \tag{E.8}$$

where the operator $\times$ applied to a vector $\mathbf{v} = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}^T$ yields the skew-symmetric matrix

$$\mathbf{v}^\times = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix}, \tag{E.9}$$

which corresponds to the cross product $\mathbf{v} \times (*)$, and $(*)$ is a vector.

Finally, by vectorizing $(\partial \mathbf{n}_{\tilde{z}} / \partial (\mathbf{x}^e)^T)^T$, on can obtain

$$\frac{\partial \mathbf{n}_{\tilde{z}}}{\partial \mathbf{x}^e} = vec\left(\left(\frac{\partial \mathbf{n}_{\tilde{z}}}{\partial (\mathbf{x}^e)^T}\right)^T\right). \tag{E.10}$$

Appendix F

Quadratic Thin Plate/Shell and Accuracy Discrepancy

In this appendix, the terminology comes from the Section 3.2.3.

Quadratic Thin Plate (QTP). When the discrete curvature operator is quantified directly in the global Cartesian frame, the bending energy of the EP model is

$$\Psi_b^{QTP} = \frac{A_{\mathcal{E}}}{2} k_b (\mathbf{x}^s)^T (\mathbf{L}_n^T \mathbf{L}_n \otimes \mathbf{I}) \mathbf{x}^s, \tag{F.1}$$

where the discrete curvature operator is

$$\mathbf{L}_n = \frac{2}{(h_1 + h_4)} \left[\frac{1}{h_1} \quad -\left(\frac{\|\mathbf{x}_{P3}\|}{\|\mathbf{x}_{23}\| h_1} + \frac{\|\mathbf{x}_{Q3}\|}{\|\mathbf{x}_{23}\| h_4} \right) \quad -\left(\frac{\|\mathbf{x}_{P2}\|}{\|\mathbf{x}_{23}\| h_1} + \frac{\|\mathbf{x}_{Q2}\|}{\|\mathbf{x}_{23}\| h_4} \right) \quad \frac{1}{h_4} \right]. \tag{F.2}$$

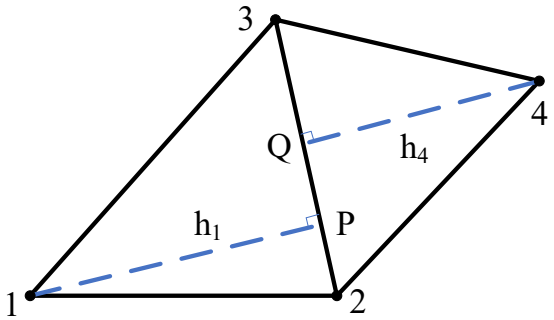


Figure F.1: Geometric quantities for the cotangent formula.

Furthermore, $\mathbf{L}_n$ can be regrouped as

$$\begin{aligned}
&\frac{2}{(h_1+h_4)\|\mathbf{X}_{23}\|}\left[\frac{\|\mathbf{X}_{23}\|}{h_1} \quad -\left(\frac{\|\mathbf{X}_{P3}\|}{h_1}+\frac{\|\mathbf{X}_{Q3}\|}{h_4}\right) \quad -\left(\frac{\|\mathbf{X}_{P2}\|}{h_1}+\frac{\|\mathbf{X}_{Q2}\|}{h_4}\right) \quad \frac{\|\mathbf{X}_{23}\|}{h_4}\right] \\
&=\frac{1}{A_{\mathcal{E}}}\left[\frac{\|\mathbf{X}_{P2}\|}{h_1}+\frac{\|\mathbf{X}_{P3}\|}{h_1} \quad -\frac{\|\mathbf{X}_{P3}\|}{h_1}-\frac{\|\mathbf{X}_{Q3}\|}{h_4} \quad -\frac{\|\mathbf{X}_{P2}\|}{h_1}-\frac{\|\mathbf{X}_{Q2}\|}{h_4} \quad \frac{\|\mathbf{X}_{P2}\|}{h_4}+\frac{\|\mathbf{X}_{P3}\|}{h_4}\right]
\end{aligned} \tag{F.3}$$

where the row vector can be expressed by these cotangents (see Figure F.1)

$$\begin{aligned}
cot\angle 12P &= \frac{\|\mathbf{X}_{P2}\|}{h_1}, cot\angle 13P = \frac{\|\mathbf{X}_{P3}\|}{h_1}, \\
cot\angle 42Q &= \frac{\|\mathbf{X}_{Q2}\|}{h_4}, cot\angle 43Q = \frac{\|\mathbf{X}_{Q3}\|}{h_4}.
\end{aligned} \tag{F.4}$$

Therefore, the operator $\mathbf{L}_n$ can be expressed in a cotangent formula [8, 9, 144, 203]. The Hessian of the QTP's bending energy is

$$\frac{\partial^2 \Psi_b^{QTP}}{\partial \mathbf{x}^s \partial (\mathbf{x}^s)^T} = k_b A_{\mathcal{E}} \mathbf{L}_n^T \mathbf{L}_n \otimes \mathbf{I}, \tag{F.5}$$

and the gradient of the QTP's bending energy is

$$\frac{\partial \Psi_b^{QTP}}{\partial \mathbf{x}^s} = (\frac{\partial^2 \Psi_b^{QTP}}{\partial \mathbf{x}^s \partial (\mathbf{x}^s)^T})\mathbf{x}^s. \tag{F.6}$$

Quadratic Thin Shell (QTS). The curvature operator in Eq.(F.1) can also be extended to the bending energy of the ES model. Thus, the bending energy of QTS is

$$\Psi_b^{QTS} = \frac{1}{2} A_{\mathcal{E}} k_b \varepsilon_b^2, \tag{F.7}$$

where the curvature change ε_b is

$$\varepsilon_b = \kappa_d - K_d = \mathbf{L}_n \mathbb{m}^T \mathbf{x}^s - \mathbf{L}_n \mathbb{N}^T \mathbf{X}^s. \tag{F.8}$$

The gradient of the QTS's bending energy is

$$\frac{\partial \Psi_b^{QTS}}{\partial \mathbf{x}^s} = A_{\mathcal{E}} k_b \frac{\partial \varepsilon_b}{\partial \mathbf{x}^s} \varepsilon_b, \tag{F.9}$$

where the gradient of the curvature change can be generalized from Eq.(3.30), i.e.,

$$\frac{\partial \varepsilon_b}{\partial \mathbf{x}^s} = (\mathbf{x}^s)^T \frac{\partial \mathbb{m}}{\partial \mathbf{x}^s} \mathbf{L}_n^T + \mathbb{m} \mathbf{L}_n^T. \tag{F.10}$$

The Hessian of QTS' bending energy is the same as the Hessian of QTP'. QTP/QTS can be seen as a variation of the EP/ES that quantifies the curvature operator in the global Cartesian frame.

Accuracy Discrepancy. In comparison, the quadratic shell model [8] is a linearized mean curvature version of the discrete shell model [66], whose detailed numerical formulations and isotropic bending rigidity can be found in [183, 182]. By expressing the bending energy in the quadratic shell model using the terminology adopted in this thesis, it can be concluded that the bending energy in the EP/QTP model is one third of that in the quadratic shell model. This discrepancy primarily arises from the use of $A_{\mathcal{E}}$ in the present model, which is the total area of triangles T_{123} and T_{432}. Notably, $A_{\mathcal{E}}$ serves not only as the integration area but also as a weighting factor in the curvature operator (see Eq. (F.1) and Eq. (F.3)). In contrast, the quadratic shell model employs one third of this area. The cubic shell model [59] builds on a foundation laid out from quadratic shell [8], so the accuracy discrepancy holds. Also, the quadratic shell model can be seen as an initially flat version of the cubic shell model.

Appendix G

Comparation of Different Hinge-Bending Energies

To facilitate the comparison of different hinge-bending energies, the initial curvature is assumed to be zero. The present hinge-bending energy is

$$\Psi_b = \frac{1}{2} k_b A_{\mathcal{E}} \left(\kappa_b\right)^2, \tag{G.1}$$

where the bending rigidity is $k_b = Eh^3/(12(1-\nu^2))$, the integration area is $A_{\mathcal{E}} = (h_C + h_D) l_{AB}/2$, and the directional curvature is

$$\kappa_b \simeq \frac{2\theta}{h_C + h_D} = \frac{\theta\, l_{AB}}{A_{\mathcal{E}}}.$$

Therefore, the present hinge-bending energy can be rewritten as

$$\Psi_b = k_b \frac{l_{AB}}{h_C + h_D} \theta^2. \tag{G.2}$$

In the following comparisons, the terminologies of the present hinge-bending energy are used consistently.

In the model proposed by Grinspun et al. [66], the hinge-bending energy is

$$\Psi_b = \frac{1}{2} k_b \frac{A_{\mathcal{E}}}{3} \left(\kappa_b\right)^2, \tag{G.3}$$

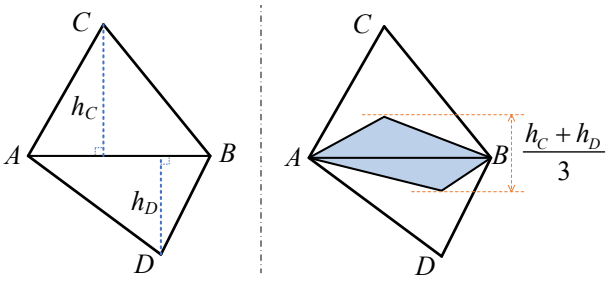


Figure G.1: Diamond region (shaded in blue) represents the integration area used in the bending model of Grinspun et al. [66].

where the directional curvature is $\kappa_b \simeq \theta l_{AB}/(A_{\mathcal{E}}/3)$ and the integration area is $A_{\mathcal{E}}/3$. This bending energy can be rewritten as

$$\Psi_b = 3k_b \frac{l_{AB}}{h_C + h_D}\theta^2. \tag{G.4}$$

The main difference from the present model lies in the choice of integration area: the present model employs the full area of the two adjacent triangles, whereas the model of Grinspun et al. uses only one-third of that area (see Figure G.1).

In the model proposed by Bridson et al. [19], the hinge-bending energy is not given explicitly. However, the nodal bending force (i.e., the gradient of the bending energy with respect to the nodal position) is given by

$$\frac{\partial \Psi_b}{\partial \mathbf{x}_i} = k_b \frac{l_{AB}}{h_C + h_D} \sin\left(\frac{\theta}{2}\right) \frac{\partial \theta}{\partial \mathbf{x}_i}. \tag{G.5}$$

For small hinge angles, $\sin(\theta/2) \simeq \theta/2$, and the bending force can be approximated as

$$\frac{\partial \Psi_b}{\partial \mathbf{x}_i} = k_b \frac{l_{AB}}{h_C + h_D} \frac{\theta}{2} \frac{\partial \theta}{\partial \mathbf{x}_i}. \tag{G.6}$$

Integrating this expression with respect to θ yields the corresponding bending energy:

$$\Psi_b = \frac{1}{4} k_b \frac{l_{AB}}{h_C + h_D}\theta^2. \tag{G.7}$$

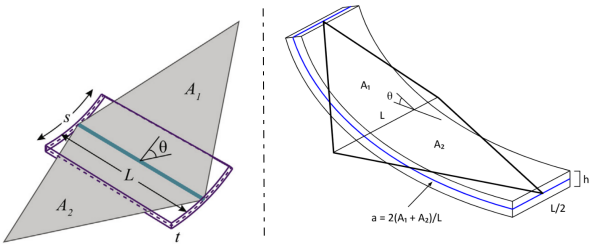


Figure G.2: The equivalent stiffness of the bending hinge is obtained using a quadrilateral plate. The left image is extracted from [207], and the right image is extracted from [41].

In the model of Woodruff et al. [207], the strain energy stored in the plate is set equal to the strain energy in the bending hinge

$$\Psi_b = \frac{1}{2} k_b' \theta^2, \tag{G.8}$$

in order to determine the equivalent stiffness coefficient k_b' through the equivalent model shown in the left image of Figure G.2. The hinge-bending energy yields

$$\Psi_b = \frac{1}{2} k_b' \theta^2 = \frac{1}{2} \frac{Eh^3}{6} \frac{l_{AB}^2}{A_{\mathcal{E}}} \theta^2 = 2k_b \frac{l_{AB}}{h_C + h_D} \theta^2. \tag{G.9}$$

This idea originates from Dudte et al. [41], which use the equivalent model shown in the right image of Figure G.2, i.e.

$$\Psi_b = \frac{1}{2} k_b' \theta^2 = \frac{1}{2} \frac{1}{48} \frac{l_{AB}^2}{A_{\mathcal{E}}} Eh^3 \theta^2 = \frac{1}{4} k_b \frac{l_{AB}}{h_C + h_D} \theta^2, \tag{G.10}$$

which has the same formulation as the model of Bridson et al. [19].

By comparing the bending energy formulations in Eqs.(G.4), (G.7), (G.9) and (G.10) with the present formulation in Eq.(G.2), it is evident that each model differs from the present one by a constant coefficient, which is the source of the observed accuracy discrepancy in Chapter2.

Appendix H

The Gradient and Hessian of the Solid-Shell Strain Energies

In this appendix, the gradient and Hessian of the bilinear solid-shell strain energies are presented in detail. The corresponding expressions for the biquadratic element can be extended from the bilinear case and are therefore omitted.

Membrane. The gradient of the membrane strain energy is

$$\frac{\partial \Psi_m^e}{\partial \mathbf{d}} = A \frac{\partial \boldsymbol{\varepsilon}_m^T}{\partial \mathbf{d}} \mathbf{D}_m \boldsymbol{\varepsilon}_m, \tag{H.1}$$

and the Hessian of the membrane strain energy is

$$\frac{\partial^2 \Psi_m^e}{\partial \mathbf{d}^T \partial \mathbf{d}} = A \frac{\partial \boldsymbol{\varepsilon}_m^T}{\partial \mathbf{d}} \mathbf{D}_m \frac{\partial \boldsymbol{\varepsilon}_m}{\partial \mathbf{d}^T} + A \boldsymbol{\varepsilon}_m^T \mathbf{D}_m^T vec(\frac{\partial^2 \mathbf{E}_\mathrm{m}}{\partial \mathbf{d} \partial \mathbf{d}^T}), \tag{H.2}$$

where $vec()$ is the vectorization operator, which has been defined in Chapter2. Here, the gradient and Hessian of membrane strain $\boldsymbol{\varepsilon}_m$ is transformed by

$$\frac{\partial \boldsymbol{\varepsilon}_m}{\partial \mathbf{d}^T} = \mathbf{T}_{mb} \frac{\partial \mathbb{e}_m}{\partial \mathbf{d}^T}, \frac{\partial^2 \boldsymbol{\varepsilon}_m}{\partial \mathbf{d}^T \partial \mathbf{d}} = \mathbf{T}_{mb} \frac{\partial^2 \mathbb{e}_m}{\partial \mathbf{d}^T \partial \mathbf{d}}, \tag{H.3}$$

with the gradient of natural membrane strain

$$\frac{\partial \mathbb{e}_m}{\partial \mathbf{d}^T} = \begin{bmatrix} \mathbf{X}_{o,\xi}^T \mathbf{N}_{o,\xi} + \mathbf{d}^T \mathbf{N}_{o,\xi}^T \mathbf{N}_{o,\xi} \\ \mathbf{X}_{o,\eta}^T \mathbf{N}_{o,\eta} + \mathbf{d}^T \mathbf{N}_{o,\eta}^T \mathbf{N}_{o,\eta} \\ \mathbf{X}_{o,\xi}^T \mathbf{N}_{o,\eta} + \mathbf{X}_{o,\eta}^T \mathbf{N}_{o,\xi} + \mathbf{d}^T \mathbf{N}_{o,\xi}^T \mathbf{N}_{o,\eta} + \mathbf{d}^T \mathbf{N}_{o,\eta}^T \mathbf{N}_{o,\xi} \end{bmatrix}, \tag{H.4}$$

and the Hessian of natural membrane strain

$$\frac{\partial^2 \mathbb{e}_m}{\partial \mathbf{d}^T \partial \mathbf{d}} = \begin{bmatrix} \mathbf{N}_{o,\xi}^T \mathbf{N}_{o,\xi} \\ \mathbf{N}_{o,\eta}^T \mathbf{N}_{o,\eta} \\ \mathbf{N}_{o,\xi}^T \mathbf{N}_{o,\eta} + \mathbf{N}_{o,\eta}^T \mathbf{N}_{o,\xi} \end{bmatrix}. \tag{H.5}$$

Bending. The gradient of the bending strain energy is

$$\frac{\partial \Psi_b^e}{\partial \mathbf{d}} = A \frac{\partial \boldsymbol{\varepsilon}_b^T}{\partial \mathbf{d}} \mathbf{D}_b \boldsymbol{\varepsilon}_b, \tag{H.6}$$

and the Hessian of the bending strain energy is

$$\frac{\partial^2 \Psi_b^e}{\partial \mathbf{d}^T \partial \mathbf{d}} = A \frac{\partial \boldsymbol{\varepsilon}_b^T}{\partial \mathbf{d}} \mathbf{D}_b \frac{\partial \boldsymbol{\varepsilon}_b}{\partial \mathbf{d}^T} + A \boldsymbol{\varepsilon}_b^T \mathbf{D}_b^T vec(\frac{\partial^2 \boldsymbol{\varepsilon}_b}{\partial \mathbf{d} \partial \mathbf{d}^T}). \tag{H.7}$$

Here, the gradient and Hessian of bending strain $\mathbf{E}_b$ is transformed by

$$\frac{\partial \boldsymbol{\varepsilon}_b}{\partial \mathbf{d}^T} = \mathbf{T}_{mb} \frac{\partial \mathbb{e}_b}{\partial \mathbf{d}^T}, \frac{\partial^2 \boldsymbol{\varepsilon}_b}{\partial \mathbf{d}^T \partial \mathbf{d}} = \mathbf{T}_{mb} \frac{\partial^2 \mathbb{e}_b}{\partial \mathbf{d}^T \partial \mathbf{d}}, \tag{H.8}$$

with the gradient of natural bending strain

$$\frac{\partial \mathbb{e}_b}{\partial \mathbf{d}^T} = \begin{bmatrix} \mathbf{X}_{n,\xi}^T \mathbf{N}_{o,\xi} + \mathbf{X}_{o,\xi}^T \mathbf{N}_{n,\xi} + \mathbf{d}^T \mathbf{N}_{o,\xi}^T \mathbf{N}_{n,\xi} + \mathbf{d}^T \mathbf{N}_{n,\xi}^T \mathbf{N}_{o,\xi} \\ \mathbf{X}_{n,\eta}^T \mathbf{N}_{o,\eta} + \mathbf{X}_{o,\eta}^T \mathbf{N}_{n,\eta} + \mathbf{d}^T \mathbf{N}_{o,\eta}^T \mathbf{N}_{n,\eta} + \mathbf{d}^T \mathbf{N}_{n,\eta}^T \mathbf{N}_{o,\eta} \\ \begin{pmatrix} \mathbf{X}_{n,\xi}^T \mathbf{N}_{o,\eta} + \mathbf{X}_{o,\xi}^T \mathbf{N}_{n,\eta} + \mathbf{X}_{n,\eta}^T \mathbf{N}_{o,\xi} + \mathbf{X}_{o,\eta}^T \mathbf{N}_{n,\xi} \\ + \mathbf{d}^T \mathbf{N}_{n,\xi}^T \mathbf{N}_{o,\eta} + \mathbf{d}^T \mathbf{N}_{o,\eta}^T \mathbf{N}_{n,\xi} + \mathbf{d}^T \mathbf{N}_{o,\xi}^T \mathbf{N}_{n,\eta} + \mathbf{d}^T \mathbf{N}_{n,\eta}^T \mathbf{N}_{o,\xi} \end{pmatrix} \end{bmatrix}, \tag{H.9}$$

and the Hessian of natural bending strain is

$$\frac{\partial^2 \mathbb{e}_b}{\partial \mathbf{d}^T \partial \mathbf{d}} = \begin{bmatrix} \mathbf{N}_{o,\xi}^T \mathbf{N}_{n,\xi} + \mathbf{N}_{n,\xi}^T \mathbf{N}_{o,\xi} \\ \mathbf{N}_{o,\eta}^T \mathbf{N}_{n,\eta} + \mathbf{N}_{n,\eta}^T \mathbf{N}_{o,\eta} \\ \mathbf{N}_{n,\xi}^T \mathbf{N}_{o,\eta} + \mathbf{N}_{o,\eta}^T \mathbf{N}_{n,\xi} + \mathbf{N}_{o,\xi}^T \mathbf{N}_{n,\eta} + \mathbf{N}_{n,\eta}^T \mathbf{N}_{o,\xi} \end{bmatrix}. \tag{H.10}$$

Shear. By introducing the ANS method, the gradient of the shear strain energy is

$$\frac{\partial \Psi_s^e}{\partial \mathbf{d}} = A \frac{\partial \bar{\boldsymbol{\gamma}}^T}{\partial \mathbf{d}} \mathbf{D}_s \bar{\boldsymbol{\gamma}}, \tag{H.11}$$

and the Hessian of the shear strain energy is

$$\frac{\partial^2 \Psi_s^e}{\partial \mathbf{d}^T \partial \mathbf{d}} = A \frac{\partial \bar{\boldsymbol{\gamma}}^T}{\partial \mathbf{d}} \mathbf{D}_s \frac{\partial \bar{\boldsymbol{\gamma}}}{\partial \mathbf{d}^T} + A \boldsymbol{\gamma}^T \mathbf{D}_s^T vec(\frac{\partial^2 \bar{\boldsymbol{\gamma}}}{\partial \mathbf{d} \partial \mathbf{d}^T}). \tag{H.12}$$

Here, the gradient and Hessian of the assumed shear strain $\bar{\mathbf{E}}_s$ is transformed by

$$\frac{\partial \bar{\boldsymbol{\gamma}}}{\partial \mathbf{d}^T} = \mathbf{T}_s \frac{\partial \bar{\mathbb{e}}_s}{\partial \mathbf{d}^T}, \frac{\partial^2 \bar{\boldsymbol{\gamma}}}{\partial \mathbf{d}^T \partial \mathbf{d}} = \mathbf{T}_s \frac{\partial^2 \bar{\mathbb{e}}_s}{\partial \mathbf{d}^T \partial \mathbf{d}}. \tag{H.13}$$

The gradient of the assumed natural shear strain is

$$\frac{\partial \bar{\mathbb{e}}_s}{\partial \mathbf{d}^T} = \begin{bmatrix} \frac{(1-\eta)}{2} \frac{\gamma_{\zeta\xi}}{\partial \mathbf{d}^T}\Big|_{\xi=0,\eta=-1} + \frac{(1+\eta)}{2} \frac{\gamma_{\zeta\xi}}{\partial \mathbf{d}^T}\Big|_{\xi=0,\eta=+1} \\ \frac{(1-\xi)}{2} \frac{\gamma_{\zeta\eta}}{\partial \mathbf{d}^T}\Big|_{\xi=-1,\eta=0} + \frac{(1+\xi)}{2} \frac{\gamma_{\zeta\eta}}{\partial \mathbf{d}^T}\Big|_{\xi=+1,\eta=0} \end{bmatrix}, \tag{H.14}$$

and the Hessian of the assumed natural shear strain is

$$\frac{\partial^2 \bar{\mathbb{e}}_s}{\partial \mathbf{d}^T \partial \mathbf{d}} = \begin{bmatrix} \frac{(1-\eta)}{2} \frac{\partial^2 \gamma_{\zeta\xi}}{\partial \mathbf{d}^T \partial \mathbf{d}}\Big|_{\xi=0,\eta=-1} + \frac{(1+\eta)}{2} \frac{\partial^2 \gamma_{\zeta\xi}}{\partial \mathbf{d}^T \partial \mathbf{d}}\Big|_{\xi=0,\eta=+1} \\ \frac{(1-\xi)}{2} \frac{\partial^2 \gamma_{\zeta\eta}}{\partial \mathbf{d}^T \partial \mathbf{d}}\Big|_{\xi=-1,\eta=0} + \frac{(1+\xi)}{2} \frac{\partial^2 \gamma_{\zeta\eta}}{\partial \mathbf{d}^T \partial \mathbf{d}}\Big|_{\xi=+1,\eta=0} \end{bmatrix}, \tag{H.15}$$

with the gradient of natural shear strain is

$$\frac{\partial \mathbb{e}_s}{\partial \mathbf{d}^T} = \begin{Bmatrix} \frac{\gamma_{\zeta\xi}}{\partial \mathbf{d}^T} \\ \frac{\gamma_{\zeta\eta}}{\partial \mathbf{d}^T} \end{Bmatrix} = \begin{bmatrix} \mathbf{X}_n^T \mathbf{N}_{o,\xi} + \mathbf{X}_{o,\xi}^T \mathbf{N}_n + \mathbf{d}^T \mathbf{N}_n^T \mathbf{N}_{o,\xi} + \mathbf{d}^T \mathbf{N}_{o,\xi}^T \mathbf{N}_n \\ \mathbf{X}_n^T \mathbf{N}_{o,\eta} + \mathbf{X}_{o,\eta}^T \mathbf{N}_n + \mathbf{d}^T \mathbf{N}_n^T \mathbf{N}_{o,\eta} + \mathbf{d}^T \mathbf{N}_{o,\eta}^T \mathbf{N}_n \end{bmatrix}, \tag{H.16}$$

and the Hessian of natural shear strain is

$$\frac{\partial^2 \mathbb{e}_s}{\partial \mathbf{d}^T \partial \mathbf{d}} = \begin{Bmatrix} \frac{\partial^2 \gamma_{\zeta\xi}}{\partial \mathbf{d}^T \partial \mathbf{d}} \\ \frac{\partial^2 \gamma_{\zeta\eta}}{\partial \mathbf{d}^T \partial \mathbf{d}} \end{Bmatrix} = \begin{bmatrix} \mathbf{N}_n^T \mathbf{N}_{o,\xi} + \mathbf{N}_{o,\xi}^T \mathbf{N}_n \\ \mathbf{N}_n^T \mathbf{N}_{o,\eta} + \mathbf{N}_{o,\eta}^T \mathbf{N}_n \end{bmatrix}. \tag{H.17}$$

Thickness. By introducing the ANS method, the gradient of the thickness strain energy is

$$\frac{\partial \Psi_t^e}{\partial \mathbf{d}} = A \frac{\partial \bar{\varepsilon}_t}{\partial \mathbf{d}} D_t \bar{\varepsilon}_t, \tag{H.18}$$

and the Hessian of the thickness strain energy is

$$\frac{\partial^2 \Psi_t^e}{\partial \mathbf{d}^T \partial \mathbf{d}} = A \frac{\partial \bar{\varepsilon}_t}{\partial \mathbf{d}} D_t \frac{\partial \bar{\varepsilon}_t}{\partial \mathbf{d}^T} + A \varepsilon_t D_t vec(\frac{\partial^2 \bar{\varepsilon}_t}{\partial \mathbf{d} \partial \mathbf{d}^T}). \tag{H.19}$$

Here, the gradient and Hessian of the assumed thickness strain is

$$\frac{\partial \bar{\varepsilon}_t}{\partial \mathbf{d}^T} = \frac{\partial \bar{\varepsilon}_{\zeta\zeta}}{\partial \mathbf{d}^T}, \frac{\partial^2 \bar{\varepsilon}_t}{\partial \mathbf{d}^T \partial \mathbf{d}} = \frac{\partial^2 \bar{\varepsilon}_{\zeta\zeta}}{\partial \mathbf{d}^T \partial \mathbf{d}}, \tag{H.20}$$

due to alignment of the natural normal direction with the local Cartesian $\tilde{Z}$-axis. The gradient of assumed natural thickness strain is

$$\frac{\partial \bar{\varepsilon}_{\zeta\zeta}}{\partial \mathbf{d}^T} = \sum_i N_i (\mathbf{X}_n^T \mathbf{N}_n + \mathbf{d}^T \mathbf{N}_n^T \mathbf{N}_n)|_i, \tag{H.21}$$

and the Hessian of assumed natural thickness strain is

$$\frac{\partial^2 \bar{\varepsilon}_{\zeta\zeta}}{\partial \mathbf{d}^T \partial \mathbf{d}} = \sum_i N_i (\mathbf{N}_n^T \mathbf{N}_n)|_i. \tag{H.22}$$

Index i labels the element nodes in the natural coordinate frame.

Appendix I

Remark on Smoothed Hinge Membrane Model

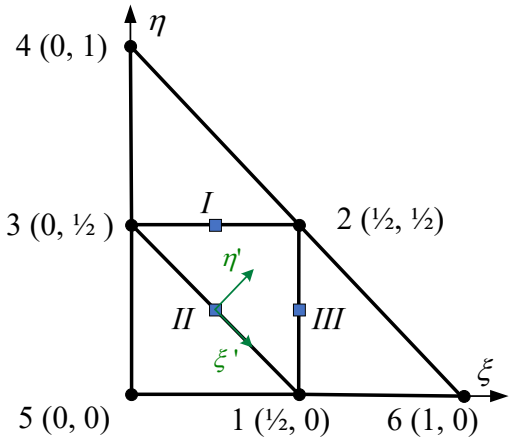


Figure I.1: A triangle-centered stencil in the ξ–η frame. I, II and III in blue are the edge midpoints of the central triangle T_{123}. The ξ'–η' frame is obtained by a clockwise rotation by an angle 45° of the ξ–η frame and then a translation of the origin to II $(1/4, 1/4)$.

I.1 Six-node Interpolation Scheme

In this scheme, all six nodes in Figure I.1 are used to fit a quadratic surface. The standard quadratic interpolated functions are

$$\begin{aligned} N_1 = 4\zeta\xi,\ N_2 = 4\xi\eta,\ N_3 = 4\eta\zeta, \\ N_4 = \eta(2\eta - 1),\ N_5 = \zeta(2\zeta - 1),\ N_6 = \xi(2\xi - 1), \end{aligned} \tag{I.1}$$

where the nature coordinates $\xi, \eta \in [0, 1]$ and $\zeta = 1 - \xi - \eta$.

With $\partial\zeta/\partial\xi = \partial\zeta/\partial\eta = -1$, the partial derivatives of the interpolation functions are

$$\begin{aligned}
\frac{\partial N_1}{\partial \xi} &= 4(1-2\xi-\eta), & \frac{\partial N_1}{\partial \eta} &= -4\,\xi,\\
\frac{\partial N_2}{\partial \xi} &= 4\,\eta, & \frac{\partial N_2}{\partial \eta} &= 4\,\xi,\\
\frac{\partial N_3}{\partial \xi} &= -4\,\eta, & \frac{\partial N_3}{\partial \eta} &= 4(1-\xi-2\eta),\\
\frac{\partial N_4}{\partial \xi} &= 0, & \frac{\partial N_4}{\partial \eta} &= 4\eta-1,\\
\frac{\partial N_5}{\partial \xi} &= 4(\xi+\eta)-3, & \frac{\partial N_5}{\partial \eta} &= 4(\xi+\eta)-3,\\
\frac{\partial N_6}{\partial \xi} &= 4\xi-1, & \frac{\partial N_6}{\partial \eta} &= 0.
\end{aligned} \tag{I.2}$$

Substituting $(\xi,\eta) = (\frac{1}{4},\frac{1}{4})$ into (I.2) gives

$$\begin{aligned}
\left(\tfrac{\partial N_1}{\partial \xi}, \tfrac{\partial N_1}{\partial \eta}\right) &= (1,\,-1), & \left(\tfrac{\partial N_2}{\partial \xi}, \tfrac{\partial N_2}{\partial \eta}\right) &= (1,\,1),\\
\left(\tfrac{\partial N_3}{\partial \xi}, \tfrac{\partial N_3}{\partial \eta}\right) &= (-1,\,1), & \left(\tfrac{\partial N_4}{\partial \xi}, \tfrac{\partial N_4}{\partial \eta}\right) &= (0,\,0),\\
\left(\tfrac{\partial N_5}{\partial \xi}, \tfrac{\partial N_5}{\partial \eta}\right) &= (-1,\,-1), & \left(\tfrac{\partial N_6}{\partial \xi}, \tfrac{\partial N_6}{\partial \eta}\right) &= (0,\,0).
\end{aligned} \tag{I.3}$$

The interpolated position derivatives at $(\xi,\eta) = (\frac{1}{4},\frac{1}{4})$ are

$$\left[\begin{array}{cc} \frac{\partial \mathbf{x}}{\partial \xi} & \frac{\partial \mathbf{x}}{\partial \eta} \end{array}\right]\Big|_{(1/4,1/4)} = \left[\mathbf{x}_1+\mathbf{x}_2-\mathbf{x}_3-\mathbf{x}_5 \quad -\mathbf{x}_1+\mathbf{x}_2+\mathbf{x}_3-\mathbf{x}_5\right]. \tag{I.4}$$

Similarly, the interpolated position derivatives at $(\xi,\eta) = (\frac{1}{2},\frac{1}{4})$ and $(\xi,\eta) = (\frac{1}{4},\frac{1}{2})$ are

$$\begin{aligned}
\left[\begin{array}{cc} \frac{\partial \mathbf{x}}{\partial \xi} & \frac{\partial \mathbf{x}}{\partial \eta} \end{array}\right]\Big|_{(1/2,1/4)} &= \left[-\mathbf{x}_1+\mathbf{x}_2-\mathbf{x}_3+\mathbf{x}_6 \quad -2\mathbf{x}_1+2\mathbf{x}_2\right],\\
\left[\begin{array}{cc} \frac{\partial \mathbf{x}}{\partial \xi} & \frac{\partial \mathbf{x}}{\partial \eta} \end{array}\right]\Big|_{(1/4,1/2)} &= \left[2\mathbf{x}_2-2\mathbf{x}_3 \quad -\mathbf{x}_1+\mathbf{x}_2-\mathbf{x}_3+\mathbf{x}_4\right].
\end{aligned} \tag{I.5}$$

I.2 Four-node Interpolation Scheme

In this scheme, nodes 1, 2, 3 and 5 in Figure I.1 are used to fit a quadratic surface. Consider ξ'–η' frame obtained by a clockwise rotation by an angle 45° of the ξ–η frame and then a translation of the origin to $(1/4, 1/4)$. Then the coordinate transform between the two frames is

$$\begin{bmatrix} \xi' \\ \eta' \end{bmatrix} = \mathbf{R}(45^\circ) \left(\begin{bmatrix} \xi \\ \eta \end{bmatrix} - \begin{bmatrix} 1/4 \\ 1/4 \end{bmatrix} \right), \tag{I.6}$$

where the standard rotation matrix is

$$\mathbf{R}(45^\circ) = \begin{bmatrix} \cos 45^\circ & -\sin 45^\circ \\ \sin 45^\circ & \cos 45^\circ \end{bmatrix}. \tag{I.7}$$

In terms of the coordinates (ξ, η), the coordinates (ξ', η') can be expressed as

$$\xi' = \tfrac{\xi - \eta}{\sqrt{2}}, \qquad \eta' = \tfrac{\xi + \eta}{\sqrt{2}} - \tfrac{\sqrt{2}}{4}. \tag{I.8}$$

In the ξ'–η' frame, the node coordinates are

$$\begin{aligned} (\xi', \eta')_1 &= \left(\tfrac{\sqrt{2}}{4}, 0 \right), \quad (\xi', \eta')_2 = \left(0, \tfrac{\sqrt{2}}{4} \right), \\ (\xi', \eta')_3 &= \left(-\tfrac{\sqrt{2}}{4}, 0 \right), \quad (\xi', \eta')_5 = \left(0, -\tfrac{\sqrt{2}}{4} \right), \end{aligned} \tag{I.9}$$

where the corresponding position vectors are $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$ and $\mathbf{x}_5$ respectively. Fit a four-parameter quadratic ansatz for $\mathbf{x}$ in the ξ'–η' frame, i.e.

$$\mathbf{x}(\xi', \eta') = \mathbf{a} + \mathbf{b}\,\xi' + \mathbf{c}\,\eta' + \mathbf{d}\,\xi'^2. \tag{I.10}$$

Let $\mathbf{x}_i = \mathbf{x}((\xi', \eta')_i)$ for $i \in \{1, 2, 3, 5\}$, the coefficient vectors are

$$\begin{aligned} \mathbf{a} &= \tfrac{1}{2}(\mathbf{x}_2 + \mathbf{x}_5), \quad \mathbf{b} = \sqrt{2}\,(\mathbf{x}_1 - \mathbf{x}_3), \\ \mathbf{c} &= \sqrt{2}\,(\mathbf{x}_2 - \mathbf{x}_5), \quad \mathbf{d} = 4\,(\mathbf{x}_1 + \mathbf{x}_3 - \mathbf{x}_2 - \mathbf{x}_5). \end{aligned} \tag{I.11}$$

Position derivatives in the (ξ', η') coordinates are

$$\begin{bmatrix} \frac{\partial \mathbf{x}}{\partial \xi'} \\ \frac{\partial \mathbf{x}}{\partial \eta'} \end{bmatrix} = \begin{bmatrix} \mathbf{b} + 2\mathbf{d}\,\xi' \\ \mathbf{c} \end{bmatrix}, \tag{I.12}$$

which can be transformed back to the (ξ, η) coordinates by

$$\begin{aligned} \begin{bmatrix} \frac{\partial \mathbf{x}}{\partial \xi} \\ \frac{\partial \mathbf{x}}{\partial \eta} \end{bmatrix} &= R_{45}^{\top} \begin{bmatrix} \frac{\partial \mathbf{x}}{\partial \xi'} \\ \frac{\partial \mathbf{x}}{\partial \eta'} \end{bmatrix} \\ &= \tfrac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{b} + 2\mathbf{d}\,\xi' \\ \mathbf{c} \end{bmatrix} \\ &= \tfrac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{b} + \mathbf{c} + \sqrt{2}\mathbf{d}\,(\xi - \eta) \\ -\mathbf{b} + \mathbf{c} + \sqrt{2}\mathbf{d}\,(\xi - \eta) \end{bmatrix}. \end{aligned} \tag{I.13}$$

Substituting $(\xi, \eta) = (\frac{1}{4}, \frac{1}{4})$ into (I.13) and using (I.11) gives

$$\left[\begin{array}{cc} \frac{\partial \mathbf{x}}{\partial \xi} & \frac{\partial \mathbf{x}}{\partial \eta} \end{array} \right] \Big|_{(1/4,1/4)} = \left[\mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 - \mathbf{x}_5 \quad -\mathbf{x}_1 + \mathbf{x}_2 + \mathbf{x}_3 - \mathbf{x}_5 \right]. \tag{I.14}$$

Similarly, the interpolated position derivatives at $(\xi, \eta) = (\frac{1}{2}, \frac{1}{4})$ and $(\xi, \eta) = (\frac{1}{4}, \frac{1}{2})$ are

$$\begin{aligned} \left[\begin{array}{cc} \frac{\partial \mathbf{x}}{\partial \xi} & \frac{\partial \mathbf{x}}{\partial \eta} \end{array} \right] \Big|_{(1/2,1/4)} &= \left[-\mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 + \mathbf{x}_6 \quad -2\mathbf{x}_1 + 2\mathbf{x}_2 \right], \\ \left[\begin{array}{cc} \frac{\partial \mathbf{x}}{\partial \xi} & \frac{\partial \mathbf{x}}{\partial \eta} \end{array} \right] \Big|_{(1/4,1/2)} &= \left[2\mathbf{x}_2 - 2\mathbf{x}_3 \quad -\mathbf{x}_1 + \mathbf{x}_2 - \mathbf{x}_3 + \mathbf{x}_4 \right]. \end{aligned} \tag{I.15}$$

By comparing (I.4) and (I.14), as well as (I.5) and (I.15), it is clear that the gradient at the edge midpoint of central triangle depends only on its four surrounding nodes and the two interpolation schemes give the same position derivatives at the center of the triangle. Therefore, the six-node interpolation scheme and the four-node interpolation scheme are equivalent in terms of evaluating the position gradients at the triangle center.

Appendix J

Bending Energy of the Annular Sector

Geometry and notation. Let α be the half-apex angle of a cone. For any point on the cone, ρ denotes the geodesic distance from the apex, and d is the axial distance measured along the cone's axis (see Figure J.1). The developed sector spans a polar angle $\varphi \in [0, \gamma]$ and a radial interval $\rho \in [\rho_1, \rho_2]$ (see Figure J.2).

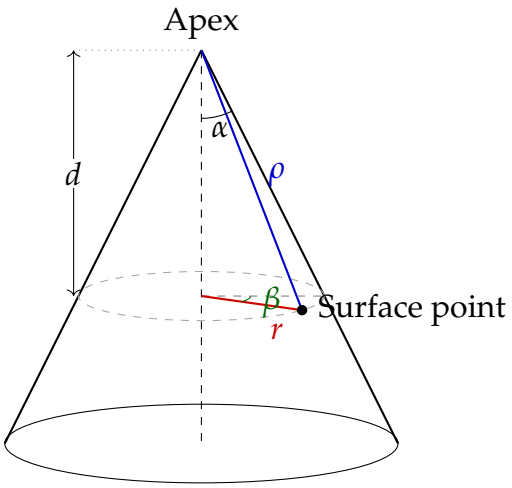


Figure J.1: Geometry of the cone.

Principal Curvature. Considering the cone parametrized by the axial distance d and the azimuthal angle β:

$$\mathbf{r}(d, \beta) = (d \tan\alpha \cos\beta,\ d \tan\alpha \sin\beta,\ d), \tag{J.1}$$

its first derivatives are

$$\frac{\partial \mathbf{r}}{\partial d} = (\tan\alpha\cos\beta,\ \tan\alpha\sin\beta,\ 1), \tag{J.2}$$

$$\frac{\partial \mathbf{r}}{\partial \beta} = (-d\tan\alpha\sin\beta,\ d\tan\alpha\cos\beta,\ 0). \tag{J.3}$$

The first fundamental form is

$$\mathbf{a} = \begin{pmatrix} E & F \\ F & G \end{pmatrix}, \tag{J.4}$$

where E, F, G are

$$E = \frac{\partial \mathbf{r}}{\partial d} \cdot \frac{\partial \mathbf{r}}{\partial d} = \tan^2\alpha + 1 = \sec^2\alpha, \tag{J.5}$$

$$F = \frac{\partial \mathbf{r}}{\partial d} \cdot \frac{\partial \mathbf{r}}{\partial \beta} = 0, \tag{J.6}$$

$$G = \frac{\partial \mathbf{r}}{\partial \beta} \cdot \frac{\partial \mathbf{r}}{\partial \beta} = d^2\tan^2\alpha. \tag{J.7}$$

The unit normal vector $\mathbf{n}$ is

$$\begin{aligned} \mathbf{n} = \frac{\frac{\partial \mathbf{r}}{\partial d} \times \frac{\partial \mathbf{r}}{\partial \beta}}{\left|\frac{\partial \mathbf{r}}{\partial d} \times \frac{\partial \mathbf{r}}{\partial \beta}\right|} &= \frac{(-d\tan\alpha\cos\beta,\ -d\tan\alpha\sin\beta,\ d\tan^2\alpha)}{d\tan\alpha\sqrt{1+\tan^2\alpha}} \\ &= \frac{1}{\sec\alpha}(-\cos\beta,\ -\sin\beta,\ \tan\alpha). \end{aligned} \tag{J.8}$$

The second derivatives are

$$\begin{aligned} \frac{\partial^2 \mathbf{r}}{\partial d^2} &= \mathbf{0}, \\ \frac{\partial^2 \mathbf{r}}{\partial d \partial \beta} &= (-\tan\alpha\sin\beta, \tan\alpha\cos\beta, 0), \\ \frac{\partial^2 \mathbf{r}}{\partial \beta^2} &= (-d\tan\alpha\cos\beta, -d\tan\alpha\sin\beta, 0). \end{aligned} \tag{J.9}$$

The second fundamental form is

$$\mathbf{b} = \begin{pmatrix} L & M \\ M & N \end{pmatrix}, \tag{J.10}$$

where L, M, N are

$$L = \frac{\partial^2 \mathbf{r}}{\partial d^2} \cdot \mathbf{n} = 0, \tag{J.11}$$

$$M = \frac{\partial^2 \mathbf{r}}{\partial d \partial \beta} \cdot \mathbf{n} = 0, \tag{J.12}$$

$$N = \frac{\partial^2 \mathbf{r}}{\partial \beta^2} \cdot \mathbf{n} = \frac{d \tan \alpha}{\sec \alpha} = d \sin \alpha. \tag{J.13}$$

The principal curvatures κ_1, κ_2 are the eigenvalues of the shape operator (Weingarten map) $\mathbf{S} = \mathbf{a}^{-1}\mathbf{b}$. Given that the coordinate system is orthogonal ($F = 0$) and conjugate ($M = 0$), both fundamental form matrices are diagonal:

$$\mathbf{a} = \begin{pmatrix} E & 0 \\ 0 & G \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} L & 0 \\ 0 & N \end{pmatrix}. \tag{J.14}$$

The shape operator is then computed as

$$\mathbf{S} = \mathbf{a}^{-1}\mathbf{b} = \begin{pmatrix} 1/E & 0 \\ 0 & 1/G \end{pmatrix} \begin{pmatrix} L & 0 \\ 0 & N \end{pmatrix} = \begin{pmatrix} L/E & 0 \\ 0 & N/G \end{pmatrix}. \tag{J.15}$$

The principal curvatures are the diagonal elements (eigenvalues) of $\mathbf{S}$:

$$\kappa_1 = \frac{N}{G} = \frac{d \sin \alpha}{d^2 \tan^2 \alpha} = \frac{1}{d \tan \alpha \sec \alpha}, \tag{J.16}$$

$$\kappa_2 = \frac{L}{E} = 0. \tag{J.17}$$

One can get the non-zero principal curvature of the cone as

$$\kappa = \frac{1}{d \tan \alpha \sec \alpha} \tag{J.18}$$

$$= \frac{1}{\tan \alpha \, (\rho \cos \alpha) \, \sec \alpha} \qquad \big(\text{using } d = \rho \cos \alpha\big) \tag{J.19}$$

$$= \frac{1}{\rho \, \tan \alpha}. \tag{J.20}$$

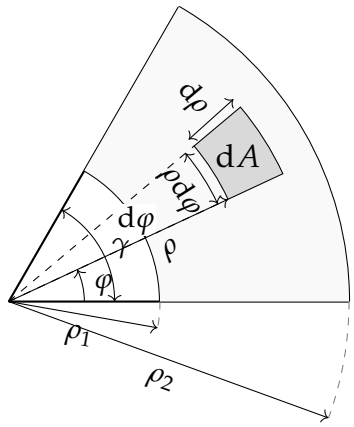


Figure J.2: Developed annular sector showing the area element.

Area of the annular sector. The area element of a developed annular sector is

$$\mathrm{d}A = \rho\,\mathrm{d}\rho\,\mathrm{d}\varphi, \qquad \varphi \in [0,\gamma],\ \rho \in [\rho_1,\rho_2], \tag{J.21}$$

so that the total area is

$$A_{\text{sector}} = \int_{\rho_1}^{\rho_2}\int_0^{\gamma} \rho\,\mathrm{d}\varphi\,\mathrm{d}\rho. \tag{J.22}$$

Bending energy of the annular sector. The bending energy of the annular sector is

$$\Psi_b = \int_{\rho_1}^{\rho_2}\int_0^{\gamma} \frac{k_b}{2}\,\kappa^2\,\rho\,\mathrm{d}\varphi\,\mathrm{d}\rho \tag{J.23}$$

with bending rigidity $k_b = Eh^3/[12(1-\nu^2)]$.

By substituting (J.20) into (J.23) and integrating over the area, one gets

$$\begin{aligned}\Psi_b &= \int_{\rho_1}^{\rho_2}\int_0^{\gamma} \frac{k_b}{2}\left(\frac{1}{\rho\,\tan\alpha}\right)^2 \rho\,\mathrm{d}\varphi\,\mathrm{d}\rho \\ &= \frac{k_b}{2\,\tan^2\alpha}\int_{\rho_1}^{\rho_2}\int_0^{\gamma}\frac{1}{\rho}\,\mathrm{d}\varphi\,\mathrm{d}\rho \\ &= -\frac{k_b\,\gamma}{2}\,\cot^2\alpha\,\ln\left(\frac{\rho_1}{\rho_2}\right).\end{aligned} \tag{J.24}$$

Using the following material and geometric parameters: $E = 4 \times 10^9$, $\nu = 0$, $h = 0.1$, $\alpha = \pi/4$, $\gamma = \pi/4$, $\rho_1 = 0.1$ and $\rho_2 = 0.105$ one gets $\Psi_b \approx 6.3866 \times 10^{-6}$.

Bibliography

[1] M. F. Ausserer and S. W. Lee. "An eighteen-node solid element for thin shell analysis". In: *International Journal for Numerical Methods in Engineering* 26.6 (1988), pp. 1345–1364.

[2] D. Baraff and A. Witkin. "Large steps in cloth simulation". In: *Proc. SIGGRAPH '98*. 1998, 43–54.

[3] D. Baraff, A. Witkin, and M. Kass. "Untangling cloth". In: *ACM Trans. Graph.* 22.3 (July 2003), 862–870.

[4] K.-J. Bathe and E. N. Dvorkin. "A formulation of general shell elements—the use of mixed interpolation of tensorial components". In: *International Journal for Numerical Methods in Engineering* 22.3 (1986), pp. 697–722.

[5] C. Becker, B. Bao, D. D. Karnaushenko, V. K. Bandari, B. Rivkin, Z. Li, M. Faghih, D. Karnaushenko, and O. G. Schmidt. "A new dimension for magnetosensitive e-skins: active matrix integrated micro-origami sensor arrays". In: *Nature communications* 13.1 (2022), p. 2121.

[6] G. van den Bergen. "Efficient collision detection of complex deformable models using AABB trees". In: *J. Graph. Tools* 2.4 (Jan. 1998), 1–13.

[7] V. L. Bergmann and S. Mukherjee. "A hybrid strain finite element for plates and shells". In: *International Journal for Numerical Methods in Engineering* 30.2 (1990), pp. 233–257.

[8] M. Bergou, M. Wardetzky, D. Harmon, D. Zorin, and E. Grinspun. "A quadratic bending model for inextensible surfaces". In: *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*. SGP '06. 2006, 227–230.

[9] M. Bergou, M. Wardetzky, D. Harmon, D. Zorin, and E. Grinspun. "Discrete quadratic curvature energies". In: *ACM SIGGRAPH 2006 Courses*. SIGGRAPH '06. 2006, 20–29.

[10] H. Bertiche, M. Madadi, and S. Escalera. "Neural Cloth Simulation". In: *ACM Trans. Graph.* 41.6 (Nov. 2022).

[11] H. Bertiche, M. Madadi, and S. Escalera. "PBNS: physically based neural simulation for unsupervised garment pose space deformation". In: *ACM Trans. Graph.* 40.6 (Dec. 2021).

[12] P. Betsch, F. Gruttmann, and E. Stein. "A 4-node finite shell element for the implementation of general hyperelastic 3D-elasticity at finite strains". In: *Computer Methods in Applied Mechanics and Engineering* 130.1 (1996), pp. 57–79. ISSN: 0045-7825.

[13] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, et al. "π_0: A Vision-Language-Action Flow Model for General Robot Control". In: *arXiv* (2024).

[14] A. I. Bobenko and P. Schröder. "Discrete Willmore flow". In: *Proceedings of the Third Eurographics Symposium on Geometry Processing*. SGP '05. Vienna, Austria: Eurographics Association, 2005, 101–es.

[15] S. Bouaziz, S. Martin, T. Liu, L. Kavan, and M. Pauly. "Projective dynamics: fusing constraint projections for fast simulation". In: *ACM Trans. Graph.* 33.4 (July 2014).

[16] D. E. Breen, D. H. House, and P. H. Getto. "A Particle-Based Computational Model of Cloth Draping Behavior". In: *Scientific Visualization of Physical Phenomena*. Ed. by N. M. Patrikalakis. Tokyo: Springer Japan, 1991, pp. 113–134.

[17] D. E. Breen, D. H. House, and P. H. Getto. "A physically-based particle model of woven cloth". In: *The Visual Computer* 8 (1992), pp. 264–277.

[18] D. E. Breen, D. H. House, and M. J. Wozny. "Predicting the drape of woven cloth using interacting particles". In: *Proc. SIGGRAPH '94*. 1994, 365–372.

[19] R. Bridson, S. Marino, and R. Fedkiw. "Simulation of clothing with folds and wrinkles". In: *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. SCA '03. 2003, 28–36.

[20] R. Bridson, R. Fedkiw, and J. Anderson. "Robust treatment of collisions, contact and friction for cloth animation". In: *ACM Trans. Graph.* 21.3 (2002), 594–603.

[21] G. E. Brown, M. Overby, Z. Forootaninia, and R. Narain. "Accurate dissipative forces in optimization integrators". In: *ACM Trans. Graph.* 37.6 (Dec. 2018).

[22] T. Buffet, D. Rohmer, L. Barthe, L. Boissieux, and M.-P. Cani. "Implicit untangling: a robust solution for modeling layered clothing". In: *ACM Trans. Graph.* 38.4 (July 2019).

[23] R. Burgoon, Z. J. Wood, and E. Grinspun. "Discrete Shells Origami". In: *Proceedings of the 1st International Conference on Computers and Their Applications (CATA 2006)*. Corpus ID: 14440608. Seattle, Washington, USA: ISCA, Mar. 2006, pp. 180–187.

[24] B. Chen and M. Govindaraj. "A Physically Based Model of Fabric Drape Using Flexible Shell Theory". In: *Textile Research Journal* 65.6 (1995), pp. 324–330.

[25] H.-Y. Chen, P. Kry, and E. Vouga. *Locking-free Simulation of Isometric Thin Plates*. 2019.

[26] H.-Y. Chen, A. Sastry, W. M. van Rees, and E. Vouga. "Physical simulation of environmentally induced thin shell deformation". In: *ACM Trans. Graph.* 37.4 (2018).

[27] S. Chen, J. Hu, and J. Teng. "A finite-volume method for contact drape simulation of woven fabrics and garments". In: *Finite Elements in Analysis and Design* 37.6 (2001), pp. 513–531.

[28] Y. Chen, R. Peng, and Z. You. "Origami of thick panels". In: *Science* 349.6246 (2015), pp. 396–400.

[29] Y. Chen, T. Xie, C. Yuksel, D. Kaufman, Y. Yang, C. Jiang, and M. Li. "Multi-Layer Thick Shells". In: *ACM SIGGRAPH 2023 Conference Proceedings*. SIGGRAPH '23. Los Angeles, CA, USA: Association for Computing Machinery, 2023.

[30] Z. Chen, H.-Y. Chen, D. M. Kaufman, M. Skouras, and E. Vouga. "Fine Wrinkling on Coarsely Meshed Thin Shells". In: 40.5 (2021).

[31] K.-J. Choi and H.-S. Ko. "Stable but responsive cloth". In: *ACM Trans. Graph.* 21.3 (2002), pp. 1–19.

[32] F. Cirak, M. Ortiz, and P. Schröder. "Subdivision surfaces: a new paradigm for thin-shell finite-element analysis". In: *International Journal for Numerical Methods in Engineering* (2000), pp. 2039 –2072.

[33] D. Clyde, J. Teran, and R. Tamstorf. "Modeling and data-driven parameter estimation for woven fabrics". In: *Proceedings of the ACM SIGGRAPH / Eurographics Symposium on Computer Animation*. SCA '17. Los Angeles, California: Association for Computing Machinery, 2017.

[34] J. R. Collier, B. J. Collier, G. O'Toole, and S. M. S. and. "Drape Prediction by Means of Finite-element Analysis". In: *The Journal of The Textile Institute* 82.1 (1991), pp. 96–107.

[35] R. Courant, K. Friedrichs, and H. Lewy. "Über die partiellen Differenzengleichungen der mathematischen Physik". In: *Mathematische annalen* 100.1 (1928), pp. 32–74.

[36] M. A. Crisfield. *Non-Linear Finite Element Analysis of Solids and Structures: Advanced Topics*. John Wiley & Sons, Inc., 1997.

[37] G. Daviet, F. Bertails-Descoubes, and L. Boissieux. "A hybrid iterative solver for robustly capturing coulomb friction in hair dynamics". In: *ACM Trans. Graph.* 30.6 (Dec. 2011), 1–12.

[38] S. Deng. *Nonlinear fabric mechanics including material nonlinearity, contact, and an adaptive global solution algorithm*. North Carolina State University, 1994.

[39] M. A. Dias, L. H. Dudte, L. Mahadevan, and C. D. Santangelo. "Geometric Mechanics of Curved Crease Origami". In: *Phys. Rev. Lett.* 109 (11 2012), p. 114301.

[40] S. Doll, K. Schweizerhof, R. Hauptmann, and C. Freischläger. "On volumetric locking of low-order solid and solid-shell elements for finite elastoviscoplastic deformations and selective reduced integration". In: *Engineering Computations* 17.7 (2000), pp. 874–902.

[41] L. H. Dudte, E. Vouga, T. Tachi, and L. Mahadevan. "Programming curvature using origami tessellations". In: *Nature materials* 15.5 (2016), pp. 583–588.

[42] E. N. Dvorkin and K. Bathe. "A continuum mechanics based four-node shell element for general non-linear analysis". In: *Engineering Computations* 1.1 (1984), pp. 77–88.

[43] B. Eberhardt, A. Weber, and W. Strasser. "A fast, flexible, particle-system model for cloth draping". In: *IEEE Computer Graphics and Applications* 16.5 (1996), pp. 52–59.

[44] J. Eischen, S. Deng, and T. Clapp. "Finite-element modeling and control of flexible fabric parts". In: *IEEE Computer Graphics and Applications* 16.5 (1996), pp. 71–80.

[45] E. English and R. Bridson. "Animating developable surfaces using nonconforming elements". In: *ACM Trans. Graph.* 27.3 (Aug. 2008), 1–5.

[46] O. Etzmuß, M. Keckeisen, and W. Straßer. "A Fast Finite Element Solution for Cloth Modelling". In: *Proceedings of the 11th Pacific Conference on Computer Graphics and Applications*. PG '03. 2003, pp. 244–251.

[47] J. A. Faber, A. F. Arrieta, and A. R. Studart. "Bioinspired spring origami". In: *Science* 359.6382 (2018), pp. 1386–1391.

[48] X. Feng, W. Huang, W. Xu, and H. Wang. "Learning-Based Bending Stiffness Parameter Estimation by a Drape Tester". In: *ACM Trans. Graph.* 41.6 (Nov. 2022).

[49] Z. Ferguson, P. Jain, D. Zorin, T. Schneider, and D. Panozzo. "High-Order Incremental Potential Contact for Elastodynamic Simulation on Curved Meshes". In: *ACM SIGGRAPH 2023 Conference Proceedings*. SIGGRAPH '23. Los Angeles, CA, USA: Association for Computing Machinery, 2023.

[50] Z. Ferguson, M. Li, T. Schneider, F. Gil-Ureta, T. Langlois, C. Jiang, D. Zorin, D. M. Kaufman, and D. Panozzo. "Intersection-free rigid body dynamics". In: *ACM Trans. Graph.* 40.4 (July 2021).

[51] E. Filipov, K. Liu, T. Tachi, M. Schenk, and G. Paulino. "Bar and hinge models for scalable analysis of origami". In: *International Journal of Solids and Structures* 124 (2017), pp. 26–45.

[52] E. T. Filipov, T. Tachi, and G. H. Paulino. "Origami tubes assembled into stiff, yet reconfigurable structures and metamaterials". In: *Proceedings of the National Academy of Sciences* 112.40 (2015), pp. 12321–12326.

[53] F. G. Flores and E. Oñate. "Wrinkling and folding analysis of elastic membranes using an enhanced rotation-free thin shell triangular element". In: *Finite Elements in Analysis and Design* 47.9 (2011), pp. 982–990.

[54] F. G. Flores and E. Oñate. "A basic thin shell triangle with only translational DOFs for large strain plasticity". In: *International Journal for Numerical Methods in Engineering* 51.1 (2001), pp. 57–83.

[55] F. G. Flores and E. Oñate. "Improvements in the membrane behaviour of the three node rotation-free BST shell triangle using an assumed strain approach". In: *Computer Methods in Applied Mechanics and Engineering* 194.6 (2005), pp. 907–932.

[56] M. Fratarcangeli, V. Tibaldo, and F. Pellacini. "Vivace: a practical gauss-seidel method for stable soft body dynamics". In: *ACM Trans. Graph.* 35.6 (Dec. 2016).

[57] J. Fulton and H. Schaub. "Deployment Dynamics Analysis of an Origami-Folded Spacecraft Structure with Elastic Hinges". In: *Journal of Spacecraft and Rockets* 59.2 (2022), pp. 401–420.

[58] L. Gan, N. Ly, and G. Steven. "A Study of Fabric Deformation Using Nonlinear Finite Elements". In: *Textile Research Journal* 65.11 (1995), pp. 660–668.

[59] A. Garg, E. Grinspun, M. Wardetzky, and D. Zorin. "Cubic shells". In: *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. SCA '07. 2007, 91–98.

[60] M. Geilinger, D. Hahn, J. Zehnder, M. Bächer, B. Thomaszewski, and S. Coros. "ADD: analytically differentiable dynamics for multi-body systems with frictional contact". In: *ACM Trans. Graph.* 39.6 (Nov. 2020).

[61] Y. Gingold, A. Secord, J. Y. Han, E. Grinspun, and D. Zorin. "A discrete model for inelastic deformation of thin shells". In: *Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. SCA '04. 2004.

[62] S. Gottschalk, M. C. Lin, and D. Manocha. "OBBTree: a hierarchical structure for rapid interference detection". In: *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive*

Techniques. SIGGRAPH '96. New York, NY, USA: Association for Computing Machinery, 1996, 171–180.

[63] A. Grigorev, M. J. Black, and O. Hilliges. "HOOD: Hierarchical Graphs for Generalized Modelling of Clothing Dynamics". In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 16965–16974.

[64] E. Grinspun, M. Desbrun, K. Polthier, P. Schröder, and A. Stern. "Discrete differential geometry: an applied introduction". In: *ACM SIGGRAPH 2006 Courses*. SIGGRAPH '06. 2006.

[65] E. Grinspun, Y. Gingold, J. Reisman, and D. Zorin. "Computing discrete shape operators on general meshes". In: *Computer Graphics Forum (Proc. Eurographics'06)* 25.3 (2006), pp. 547–556.

[66] E. Grinspun, A. N. Hirani, M. Desbrun, and P. Schröder. "Discrete shells". In: *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. SCA '03. 2003, 62–67.

[67] E. Grinspun, P. Krysl, and P. Schröder. "CHARMS: a simple framework for adaptive simulation". In: 21.3 (2002), 281–290.

[68] Y. Guo, W. Gati, H. Naceur, and J. Batoz. "An efficient DKT rotation free shell element for springback simulation in sheet metal forming". In: *Computers & Structures* 80.27 (2002), pp. 2299–2312.

[69] D. Harmon, E. Vouga, R. Tamstorf, and E. Grinspun. "Robust treatment of simultaneous collisions". In: *ACM Trans. Graph.* 27.3 (Aug. 2008), 1–4. ISSN: 0730-0301.

[70] R. Hauptmann and K. Schweizerhof. "A systematic development of 'solid-shell' element formulations for linear and non-linear analyses employing only displacement degrees of freedom". In: *International Journal for Numerical Methods in Engineering* 42.1 (1998), pp. 49–69.

[71] R. Hauptmann, K. Schweizerhof, and S. Doll. "Extension of the 'solid-shell' concept for application to large elastic and large elastoplastic deformations". In: *International Journal for Numerical Methods in Engineering* 49.9 (2000), pp. 1121–1141.

[72] Y. Hu, Y. Zhou, K. Kwok, and K. Sze. "Simulating flexible origami structures by finite element method". In: *International Journal of Mechanics and Materials in Design* 17 (2021), 801–829.

[73] K. Huang, F. M. Chitalu, H. Lin, and T. Komura. "GIPC: Fast and Stable Gauss-Newton Optimization of IPC Barrier Energy". In: *ACM Trans. Graph.* 43.2 (Mar. 2024).

[74] K. Huang, X. Lu, H. Lin, T. Komura, and M. Li. "StiffGIPC: Advancing GPU IPC for Stiff Affine-Deformable Simulation". In: *ACM Trans. Graph.* (May 2025).

[75] W. Huang, Z. Hao, J. Li, D. Tong, K. Guo, Y. Zhang, H. Gao, K. J. Hsia, and M. Liu. *A tutorial on simulating nonlinear behaviors of flexible structures with the discrete differential geometry (DDG) method*. 2025.

[76] P. Hubbard. "Collision detection for interactive graphics applications". In: *IEEE Transactions on Visualization and Computer Graphics* 1.3 (1995), pp. 218–230.

[77] D. A. Huffman. "Curvature and Creases: A Primer on Paper". In: *IEEE Trans. Comput.* 25.10 (Oct. 1976), 1010–1019.

[78] S. Huh, D. N. Metaxas, and N. I. Badler. "Collision resolutions in cloth simulation". In: *Proceedings Computer Animation 2001. Fourteenth Conference on Computer Animation (Cat. No. 01TH8596)*. IEEE. 2001, pp. 122–127.

[79] A. Jamalimehr, M. Mirzajanzadeh, A. Akbarzadeh, and D. Pasini. "Rigidly flat-foldable class of lockable origami-inspired metamaterials with topological stiff states". In: *Nature communications* 13.1 (2022), p. 1816.

[80] C. Jiang, T. Gast, and J. Teran. "Anisotropic elastoplasticity for cloth, knit and hair frictional contact". In: *ACM Trans. Graph.* 36.4 (July 2017).

[81] N. Jin, W. Lu, Z. Geng, and R. P. Fedkiw. "Inequality cloth". In: *Proceedings of the ACM SIGGRAPH/Eurographics symposium on computer animation*. 2017, pp. 1–10.

[82] J. M. Kaldor, D. L. James, and S. Marschner. "Simulating knitted cloth at the yarn level". In: *ACM Trans. Graph.* 27.3 (Aug. 2008), 1–9.

[83] C. Kane, J. E. Marsden, M. Ortiz, and M. West. "Variational integrators and the Newmark algorithm for conservative and dissipative mechanical systems". In: *International Journal for Numerical Methods in Engineering* 49.10 (2000), pp. 1295–1325.

[84] T. J. Kang and W. R. Y. and. "Drape Simulation of Woven Fabric by Using the Finite-element Method". In: *The Journal of The Textile Institute* 86.4 (1995), pp. 635–648.

[85] L. Kavan, D. Gerszewski, A. W. Bargteil, and P.-P. Sloan. "Physics-inspired upsampling for cloth simulation in games". In: *ACM Trans. Graph.* 30.4 (July 2011).

[86] L. Kharevych, W. Yang, Y. Tong, E. Kanso, J. E. Marsden, P. Schröder, and M. Desbrun. "Geometric, variational integrators for computer animation". In: *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. SCA '06. 2006, 43–51.

[87] M. Kilian, S. Flöry, Z. Chen, N. J. Mitra, A. Sheffer, and H. Pottmann. "Curved folding". In: *ACM Trans. Graph.* 27.3 (Aug. 2008), 1–9.

[88] C. H. Kim, K. Y. Sze, and Y. H. Kim. "Curved quadratic triangular degenerated- and solid-shell elements for geometric nonlinear analysis". In: *International Journal for Numerical Methods in Engineering* 57.14 (2003), pp. 2077–2097.

[89] J. H. Kim. *Fabric mechanics analysis using large deformation orthotropic shell theory*. North Carolina State University, 1991.

[90] M. Kim, N.-J. Sung, S.-J. Kim, Y.-J. Choi, and M. Hong. "Parallel cloth simulation with effective collision detection for interactive AR application". In: *Multimedia Tools and Applications* 78.4 (2019), pp. 4851–4868.

[91] S. Krishnan, M. Gopi, M. Lin, D. Manocha, and A. Pattekar. "Rapid and Accurate Contact Determination between Spline Models using ShellTrees". In: *Computer Graphics Forum* 17.3 (1998), pp. 315–326.

[92] P. Krysl. "Robust flat-facet triangular shell finite element". In: *International Journal for Numerical Methods in Engineering* 123.10 (2022), pp. 2399–2423.

[93] T.-H. Kwok. "Geometry-based thick origami simulation". In: *Journal of Mechanical Design* 143.6 (2021), p. 061701.

[94] L. Lan, G. Ma, Y. Yang, C. Zheng, M. Li, and C. Jiang. "Penetration-free projective dynamics on the GPU". In: *ACM Trans. Graph.* 41.4 (July 2022).

[95] Q. Le, Y. Deng, J. Bu, B. Zhu, and T. Du. "Second-Order Finite Elements for Deformable Surfaces". In: *SIGGRAPH Asia 2023 Conference Papers*. 2023.

[96] F. Lechenault, B. Thiria, and M. Adda-Bedia. "Mechanical Response of a Creased Sheet". In: *Phys. Rev. Lett.* 112 (24 2014), p. 244301.

[97] T. G. Leong, C. L. Randall, B. R. Benson, N. Bassik, G. M. Stern, and D. H. Gracias. "Tetherless thermobiochemically actuated microgrippers". In: *Proceedings of the National Academy of Sciences* 106.3 (2009), pp. 703–708.

[98] J. Li, G. Daviet, R. Narain, F. Bertails-Descoubes, M. Overby, G. E. Brown, and L. Boissieux. "An implicit frictional contact solver for adaptive cloth simulation". In: *ACM Trans. Graph.* 37.4 (July 2018).

[99] M. Li, Z. Ferguson, T. Schneider, T. Langlois, D. Zorin, D. Panozzo, C. Jiang, and D. M. Kaufman. "Incremental Potential Contact: Intersection- and Inversion-free Large Deformation Dynamics". In: *ACM Trans. Graph. (SIGGRAPH)* 39.4 (2020).

[100] M. Li, D. M. Kaufman, and C. Jiang. "Codimensional Incremental Potential Contact". In: *ACM Trans. Graph. (SIGGRAPH)* 40.4 (2021).

[101] Y. Li, T. Du, K. Wu, J. Xu, and W. Matusik. "DiffCloth: Differentiable Cloth Simulation with Dry Frictional Contact". In: *ACM Trans. Graph.* 42.1 (Oct. 2022).

[102] J. Liang, M. C. Lin, and V. Koltun. "Differentiable cloth simulation for inverse problems". In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2019.

[103] Q. Liang. "Corotational Hinge-based Thin Plates/Shells". In: *Computer Graphics Forum (Eurographics 2025)* 44.2 (2025).

[104] Q. Liang. "Modeling and Simulating Origami Structures using Bilinear Solid-Shell Element". In: *Proceedings of the SIGGRAPH Asia 2025 Technical Communications*. Hong Kong, 2025.

[105] Q. Liang. "Smoothed-Hinge Model for Cloth Simulation". In: *Eurographics/ ACM SIGGRAPH Symposium on Computer Animation - Posters*. Vancouver, 2024.

[106] H. Liu and R. D. James. "Design of origami structures with curved tiles between the creases". In: *Journal of the Mechanics and Physics of Solids* 185 (2024), p. 105559.

[107] K Liu, G. Paulino, et al. "Highly efficient nonlinear structural analysis of origami assemblages using the MERLIN2 software". In: *Origami* 7 (2018), pp. 1167–1182.

[108] K. Liu and G. H. Paulino. "Nonlinear mechanics of non-rigid origami: an efficient computational approach". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 473.2206 (2017), p. 20170348.

[109] T. Liu, A. W. Bargteil, J. F. O'Brien, and L. Kavan. "Fast simulation of mass-spring systems". In: *ACM Trans. Graph.* 32.6 (Nov. 2013).

[110] T. Liu, S. Bouaziz, and L. Kavan. "Quasi-Newton Methods for Real-Time Simulation of Hyperelastic Materials". In: *ACM Trans. Graph.* 36.3 (May 2017).

[111] X. H. Liu and K. Y. Sze. "A corotational interpolatory model for fabric drape simulation". In: *International Journal for Numerical Methods in Engineering* 77.6 (2009), pp. 799–823.

[112] X. Long, Q. Zhao, K. Zhang, Z. Zhang, D. Wang, Y. Liu, Z. Shu, Y. Lu, S. Wang, X. Wei, W. Li, W. Yin, Y. Yao, J. Pan, Q. Shen, R. Yang, X. Cao, and Q. Dai. *A Survey: Learning Embodied Intelligence from Physical Simulators and World Models*. 2025. arXiv: 2507.00917.

[113] J. Lu and C. Zheng. "Dynamic cloth simulation by isogeometric analysis". In: *Computer Methods in Applied Mechanics and Engineering* 268 (2014), pp. 475–493.

[114] Z. Lu, X. He, Y. Guo, X. Liu, and H. Wang. "Projective Peridynamic Modeling of Hyperelastic Membranes With Contact". In: *IEEE Transactions on Visualization and Computer Graphics* 30.8 (2024), pp. 4601–4614.

[115] M. Ly, J. Jouve, L. Boissieux, and F. Bertails-Descoubes. "Projective dynamics with dry frictional contact". In: *ACM Trans. Graph.* 39.4 (Aug. 2020).

[116] M. Macklin, K. Erleben, M. Müller, N. Chentanez, S. Jeschke, and T. Y. Kim. "Primal/dual descent methods for dynamics". In: *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. SCA '20. Virtual Event, Canada: Eurographics Association, 2020.

[117] M. Macklin, M. Müller, and N. Chentanez. "XPBD: position-based simulation of compliant constrained dynamics". In: *Proceedings of the 9th International Conference on Motion in Games*. MIG '16. Burlingame, California: Association for Computing Machinery, 2016, 49–54.

[118] R. H. MacNeal. "Derivation of element stiffness matrices by assumed strain distributions". In: *Nuclear Engineering and Design* 70.1 (1982), pp. 3–12.

[119] S. Martin, B. Thomaszewski, E. Grinspun, and M. Gross. "Example-based elastic materials". In: *ACM Trans. Graph.* 30.4 (July 2011).

[120] M. Meyer, G. Debunne, M. Desbrun, and A. H. Barr. "Interactive animation of cloth-like objects in virtual reality". In: *The Journal of Visualization and Computer Animation* 12.1 (2001), pp. 1–12.

[121] D. Misseroni, P. P. Pratapa, K. Liu, B. Kresling, Y. Chen, C. Daraio, and G. H. Paulino. "Origami engineering". In: *Nature Reviews Methods Primers* 4.1 (2024), p. 40.

[122] J. Mitani and T. Igarashi. "Interactive Design of Planar Curved Folding by Reflection". In: *Pacific Graphics Short Papers*. Ed. by B.-Y. Chen, J. Kautz, T.-Y. Lee, and M. C. Lin. Kaohsiung, Taiwan: The Eurographics Association, 2011, pp. 77–81.

[123] K. Mundilova. "Gluing and creasing paper along curves: computational methods for analysis and design". PhD thesis. Massachusetts Institute of Technology, 2024.

[124] M. Müller, B. Heidelberger, M. Hennix, and J. Ratcliff. "Position based dynamics". In: *Journal of Visual Communication and Image Representation* 18.2 (2007), pp. 109–118.

[125] R. Narain, M. Overby, and G. E. Brown. "ADMM ⊇ projective dynamics: fast simulation of general constitutive models". In: *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. SCA '16. Zurich, Switzerland: Eurographics Association, 2016, 21–28.

[126] R. Narain, T. Pfaff, and J. F. O'Brien. "Folding and crumpling adaptive sheets". In: *ACM Trans. Graph.* 32.4 (July 2013).

[127] R. Narain, A. Samii, and J. F. O'Brien. "Adaptive anisotropic remeshing for cloth simulation". In: *ACM Trans. Graph.* 31.6 (Nov. 2012).

[128] Newton Contributors. *Newton: GPU-accelerated physics simulation for robotics, and simulation research*. Newton a Series of LF Projects, LLC, 2025. URL: https://github.com/newton-physics/newton.

[129] J. Nocedal and S. J. Wright. *Numerical optimization*. Springer, 1999.

[130] E Oñate and M Cervera. "Derivation of thin plate bending elements with one degree of freedom per node: a simple three node triangle". In: *Engineering computations* 10.6 (1993), pp. 543–561.

[131] M. Ortiz and L. Stainier. "The variational formulation of viscoplastic constitutive updates". In: *Computer Methods in Applied Mechanics and Engineering* 171.3 (1999), pp. 419–444.

[132] M. A. Otaduy, R. Tamstorf, D. Steinemann, and M. Gross. "Implicit Contact Handling for Deformable Objects". In: *Computer Graphics Forum* 28.2 (2009), pp. 559–568.

[133] M. Overby, G. E. Brown, J. Li, and R. Narain. "ADMM ⊇ Projective Dynamics: Fast Simulation of Hyperelastic Models with Dynamic Constraints". In: *IEEE Transactions on Visualization and Computer Graphics* 23.10 (2017), pp. 2222–2234.

[134] E. Oñate and F. Zárate. "Rotation-free triangular plate and shell elements". In: *International Journal for Numerical Methods in Engineering* 47.1-3 (2000), pp. 557–603.

[135] H. C. Park, C. Cho, and S. W. Lee. "An efficient assumed strain element model with six DOF per node for geometrically non-linear shells". In: *International Journal for Numerical Methods in Engineering* 38.24 (1995), pp. 4101–4122.

[136] Ž. Penava, D. Šimić-Penava, and Ž Knezic. "Determination of the elastic constants of plain woven fabrics by a tensile test in various directions". In: *Fibres & Textiles in Eastern Europe* 22.2 (104) (2014).

[137] E. A. Peraza Hernandez, D. J. Hartl, E. Akleman, and D. C. Lagoudas. "Modeling and analysis of origami structures with smooth folds". In: *Computer-Aided Design* 78 (2016). SPM 2016, pp. 93–106.

[138] T. Pfaff, M. Fortunato, A. Sanchez-Gonzalez, and P. Battaglia. "Learning mesh-based simulation with graph networks". In: *International conference on learning representations*. 2020.

[139] T. Pfaff, R. Narain, J. M. de Joya, and J. F. O'Brien. "Adaptive tearing and cracking of thin sheets". In: *ACM Trans. Graph.* 33.4 (2014).

[140] R. Phaal and C. R. Calladine. "A simple class of finite elements for plate and shell problems. I: Elements for beams and thin flat plates". In: *International Journal for Numerical Methods in Engineering* 35.5 (1992), pp. 955–977.

[141] R. Phaal and C. R. Calladine. "A simple class of finite elements for plate and shell problems. II: An element for thin shells, with only translational degrees of freedom". In: *International Journal for Numerical Methods in Engineering* 35.5 (1992), pp. 979–996.

[142] T. H. H. Pian and S. W. Lee. "Notes on finite elements for nearly incompressible materials". In: *AIAA Journal* 14.6 (1976), pp. 824–826.

[143] T. H. Pian. "Finite elements based on consistently assumed stresses and displacements". In: *Finite Elements in Analysis and Design* 1.2 (1985), pp. 131–140.

[144] U. Pinkall and K. Polthier. "Computing Discrete Minimal Surfaces and Their Conjugates". In: *Experimental Mathematics* 2.1 (1993), pp. 15–36.

[145] J. M. Pizana, A. Rodríguez, G. Cirio, and M. A. Otaduy. "A Bending Model for Nodal Discretizations of Yarn-Level Cloth". In: *Computer Graphics Forum* 39.8 (2020), pp. 181–189.

[146] X. Provot. "Collision and self-collision handling in cloth model dedicated to design garments". In: *Computer Animation and Simulation'97: Proceedings of the Eurographics Workshop in Budapest, Hungary, September 2–3, 1997*. Springer. 1997, pp. 177–189.

[147] X. Provot et al. "Deformation constraints in a mass-spring model to describe rigid cloth behaviour". In: *Graphics interface*. Canadian Information Processing Society. 1995, pp. 147–147.

[148] A. Quaglino. "A framework for creating low-order shell elements free of membrane locking". In: *International Journal for Numerical Methods in Engineering* 108.1 (2016), pp. 55–75.

[149] A. Quaglino. "A framework for creating low-order shell elements free of membrane locking". In: *International Journal for Numerical Methods in Engineering* 108.1 (2016), pp. 55–75.

[150] M. Rabinovich, T. Hoffmann, and O. Sorkine-Hornung. "Modeling curved folding with freeform deformations". In: *ACM Trans. Graph.* 38.6 (Nov. 2019).

[151] J Reisman, E. Grinspun, and D. Zorin. *A note on the triangle-centered quadratic interpolation discretization of the shape operator: Technical report*. Tech. rep. NYU and Columbia, 2007.

[152] V. Romero, M. Ly, A. H. Rasheed, R. Charrondière, A. Lazarus, S. Neukirch, and F. Bertails-Descoubes. "Physical validation of simulators in computer graphics: a new framework dedicated to slender elastic structures and frictional contact". In: *ACM Trans. Graph.* 40.4 (2021).

[153] D. Rus and C. Sung. "Spotlight on origami robots". In: *Science Robotics* 3.15 (2018), eaat0938.

[154] F. Sabourin and M. Brunet. "Analysis of plates and shells with a simplified three node triangular element". In: *Thin-Walled Structures* (1995). Finite Elements for Thin-Walled Structures, pp. 209–223.

[155] R. A. Sauer, Z. Zou, and T. J. Hughes. "A simple and efficient hybrid discretization approach to alleviate membrane locking in isogeometric thin shells". In: *Computer Methods in Applied Mechanics and Engineering* 424 (2024), p. 116869.

[156] M. Schenk and S. D. Guest. "Geometry of Miura-folded metamaterials". In: *Proceedings of the National Academy of Sciences* 110.9 (2013), pp. 3276–3281.

[157] M. Schenk, S. D. Guest, et al. "Origami folding: A structural engineering approach". In: *Origami* 5 (2011), pp. 291–304.

[158] C. Schreck, D. Rohmer, S. Hahmann, M.-P. Cani, S. Jin, C. C. L. Wang, and J.-F. Bloch. "Nonsmooth Developable Geometry for Interactively Animating Paper Crumpling". In: *ACM Trans. Graph.* 35.1 (Dec. 2016).

[159] S.-G. Shin and C.-O. Lee. "Splitting basis techniques in cloth simulation by isogeometric analysis". In: *Computer Methods in Applied Mechanics and Engineering* 362 (2020), p. 112871.

[160] J. L. Silverberg, A. A. Evans, L. McLeod, R. C. Hayward, T. Hull, C. D. Santangelo, and I. Cohen. "Using origami design principles to fold reprogrammable mechanical metamaterials". In: *Science* 345.6197 (2014), pp. 647–650.

[161] J. C. Simo and M. S. Rifai. "A class of mixed assumed strain methods and the method of incompatible modes". In: *International Journal for Numerical Methods in Engineering* 29.8 (1990), pp. 1595–1638.

[162] J. Simo, D. Fox, and M. Rifai. "On a stress resultant geometrically exact shell model. Part II: The linear theory; computational aspects". In: *Computer Methods in Applied Mechanics and Engineering* 73.1 (1989), pp. 53–92.

[163] J. C. Simo and D. D. Fox. "On a stress resultant geometrically exact shell model. Part I: Formulation and optimal parametrization". In: *Computer Methods in Applied Mechanics and Engineering* 72.3 (1989), pp. 267–304.

[164] J. C. Simo, D. D. Fox, and M. S. Rifai. "On a stress resultant geometrically exact shell model. Part III: Computational aspects of the nonlinear theory". In: *Computer Methods in Applied Mechanics and Engineering* 79.1 (1990), pp. 21–70.

[165] J. Solomon, E. Vouga, M. Wardetzky, and E. Grinspun. "Flexible Developable Surfaces". In: *Computer Graphics Forum* 31.5 (2012), pp. 1567–1576.

[166] O. Stein, E. Grinspun, and K. Crane. "Developability of triangle meshes". In: *ACM Trans. Graph.* 37.4 (July 2018).

[167] A. Stern and M. Desbrun. "Discrete geometric mechanics for variational time integrators". In: *ACM SIGGRAPH 2006 Courses*. SIGGRAPH '06. 2006, 75–80.

[168] A. Stern and E. Grinspun. "Implicit-Explicit Variational Integration of Highly Oscillatory Problems". In: *Multiscale Modeling & Simulation* 7.4 (2009), pp. 1779–1794.

[169] T. Stuyck and B. A. Barsky. *Cloth Simulation for Computer Graphics*. Morgan & Claypool Publishers, 2018. ISBN: 1681734117.

[170] K. Y. Sze. "Three-dimensional continuum finite element models for plate/shell analysis". In: *Progress in Structural Engineering and Materials* 4.4 (2002), pp. 400–407.

[171] K. Y. Sze, W. K. Chan, and T. H. H. Pian. "An eight-node hybrid-stress solid-shell element for geometric non-linear analysis of elastic shells". In: *International Journal for Numerical Methods in Engineering* 55.7 (2002), pp. 853–878.

[172] K. Y. Sze and A. Ghali. "Hybrid hexahedral element for solids, plates, shells and beams by selective scaling". In: *International Journal for Numerical Methods in Engineering* 36.9 (1993), pp. 1519–1540.

[173] K. Y. Sze and Y. C. Hu. "Assumed natural strain and stabilized quadrilateral Lobatto spectral elements for C0 plate/shell analysis". In: *International Journal for Numerical Methods in Engineering* 111.5 (2017), pp. 403–446.

[174] K. Y. Sze and X. H. Liu. "A co-rotational grid-based model for fabric drapes". In: *International Journal for Numerical Methods in Engineering* 57.11 (2003), pp. 1503–1521.

[175] K. Y. Sze and L. Q. Yao. "A hybrid stress ANS solid-shell element and its generalization for smart structure modelling. Part I—solid-shell element formulation". In: *International Journal for Numerical Methods in Engineering* 48.4 (2000), pp. 545–564.

[176] K. Y. Sze and Y. X. Zhou. "An Efficient Rotation-free Triangle for Drape/Cloth Simulations — Part I: Model Improvement, Dynamic Simulation and Adaptive Remeshing". In: *International Journal of Computational Methods* 13 (2016), p. 1650021.

[177] K. Sze. "Admissible matrix formulation — from orthogonal approach to explicit hybrid stabilization". In: *Finite Elements in Analysis and Design* 24.1 (1996). Mixed and Hybrid Finite Element Methods Part II, pp. 1–30. ISSN: 0168-874X.

[178] K. Sze and X. Liu. "A new skeletal model for fabric drapes". In: *International Journal of Mechanics and Materials in Design* 2 (2005), pp. 225–243.

[179] K. Sze and X. Liu. "Fabric drape simulation by solid-shell finite element method". In: *Finite Elements in Analysis and Design* 43.11 (2007), pp. 819–838.

[180] K. Sze, X. Liu, and S. Lo. "Popular benchmark problems for geometric nonlinear analysis of shells". In: *Finite Elements in Analysis and Design* 40.11 (2004), pp. 1551–1569.

[181] T. Tachi. "Simulation of rigid origami". In: *Origami* 4.08 (2009), pp. 175–187.

[182] R. Tamstorf. *Derivation of discrete bending forces and their gradients*. Tech. rep. Walt Disney Animation Studios, 2013.

[183] R. Tamstorf and E. Grinspun. "Discrete bending forces and their Jacobians". In: *Graph. Models* 75.6 (2013).

[184] C. Tang, P. Bo, J. Wallner, and H. Pottmann. "Interactive Design of Developable Surfaces". In: *ACM Trans. Graph.* 35.2 (Jan. 2016).

[185] D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer. "Elastically deformable models". In: *Proc. SIGGRAPH '87*. 1987, 205–214.

[186] B. Thomaszewski, S. Pabst, and W. Straßer. "Continuum-based Strain Limiting". In: *Computer Graphics Forum* 28.2 (2009), pp. 569–576.

[187] B. Thomaszewski, M. Wacker, and W. Straßer. "A consistent bending model for cloth simulation with corotational subdivision finite elements". In: *Proceedings of the 2006 ACM SIGGRAPH / Eurographics Symposium on Computer Animation*. SCA '06. 2006, 107–116.

[188] S. Timoshenko, S. Woinowsky-Krieger, et al. *Theory of plates and shells*. McGraw-hill New York, 1959.

[189] K. D. Tsiknis. "Better cloth through unbiased strain limiting and physics-aware subdivision". PhD thesis. University of British Columbia, 2006.

[190] N. Umetani, D. M. Kaufman, T. Igarashi, and E. Grinspun. "Sensitive couture for interactive garment modeling and editing". In: *ACM Trans. Graph.* 30.4 (July 2011).

[191] A. Vergauwen, L. D. Laet, and N. D. Temmerman. "Computational modelling methods for pliable structures based on curved-line folding". In: *Computer-Aided Design* 83 (2017), pp. 51–63.

[192] P. Volino and N. M.-T. and. "Accurate Garment Prototyping and Simulation". In: *Computer-Aided Design and Applications* 2.5 (2005), pp. 645–654.

[193] P. Volino, M. Courchesne, and N. Magnenat Thalmann. "Versatile and efficient techniques for simulating cloth and other deformable objects". In: *Proc. SIGGRAPH '95*. 1995, 137–144.

[194] P. Volino and N. Magnenat-Thalmann. "Resolving surface collisions through intersection contour minimization". In: *ACM Trans. Graph.* 25.3 (July 2006), 1154–1159.

[195] P. Volino, N. Magnenat-Thalmann, and F. Faure. "A simple approach to nonlinear tensile stiffness for accurate cloth simulation". In: *ACM Trans. Graph.* 28.4 (2009).

[196] S. Waitukaitis, R. Menaut, B. G.-g. Chen, and M. van Hecke. "Origami Multistability: From Single Vertices to Metasheets". In: *Phys. Rev. Lett.* 114 (5 2015), p. 055503.

[197] B. Wang, Z. Ferguson, T. Schneider, X. Jiang, M. Attene, and D. Panozzo. "A Large-scale Benchmark and an Inclusion-based Algorithm & nbsp;for Continuous Collision Detection". In: *ACM Trans. Graph.* 40.5 (Sept. 2021).

[198] H. Wang. "A chebyshev semi-iterative approach for accelerating projective and position-based dynamics". In: *ACM Trans. Graph.* 34.6 (Nov. 2015).

[199] H. Wang. "Defending continuous collision detection against errors". In: *ACM Trans. Graph.* 33.4 (July 2014).

[200] H. Wang, J. O'Brien, and R. Ramamoorthi. "Multi-resolution isotropic strain limiting". In: *ACM Trans. Graph.* 29.6 (Dec. 2010).

[201] H. Wang, J. F. O'Brien, and R. Ramamoorthi. "Data-driven elastic models for cloth: modeling and measurement". In: *ACM SIGGRAPH 2011 Papers*. SIGGRAPH '11. Vancouver, British Columbia, Canada: Association for Computing Machinery, 2011.

[202] Z. Wang, Y. Yang, and H. Wang. "Stable Discrete Bending by Analytic Eigensystem and Adaptive Orthotropic Geometric Stiffness". In: *ACM Trans. Graph.* 42.6 (Dec. 2023).

[203] M. Wardetzky, M. Bergou, D. Harmon, D. Zorin, and E. Grinspun. "Discrete quadratic curvature energies". In: *Computer Aided Geometric Design* 24.8 (2007), pp. 499–518.

[204] Z. Y. Wei, Z. V. Guo, L. Dudte, H. Y. Liang, and L. Mahadevan. "Geometric Mechanics of Periodic Pleated Origami". In: *Phys. Rev. Lett.* 110 (21 2013), p. 215501.

[205] C. Weischedel. *A discrete geometric view on shear-deformable shell models*. Georg-August-Universität Göttingen., 2012.

[206] Z. Wen, P. Lv, F. Feng, and H. Duan. "A generalized geometric mechanics theory for multi-curve-fold origami: Vertex constrained universal configurations". In: *Journal of the Mechanics and Physics of Solids* 192 (2024), p. 105829.

[207] S. R. Woodruff and E. T. Filipov. "A bar and hinge model formulation for structural analysis of curved-crease origami". In: *International Journal of Solids and Structures* 204-205 (2020), pp. 114–127.

[208] S. R. Woodruff and E. T. Filipov. "Structural Analysis of Curved Folded Deployables". In: *Earth and Space 2018: Engineering for Extreme Environments*. Reston, VA: American Society of Civil Engineers (ASCE), 2018, pp. 793–803.

[209] P. Wriggers. *Nonlinear Finite Element Methods*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008.

[210] H. Wu and T. Kim. "An Eigenanalysis of Angle-Based Deformation Energies". In: *Proc. ACM Comput. Graph. Interact. Tech.* 6.3 (Aug. 2023).

[211] Q. Xie, K. Y. Sze, and Y. X. Zhou. "Drape simulation using solid-shell elements and adaptive mesh subdivision". In: *Finite Elements in Analysis and Design* 106 (2015), pp. 85–102.

[212] H. Yasuda and J. Yang. "Reentrant Origami-Based Metamaterials with Negative Poisson's Ratio and Bistability". In: *Phys. Rev. Lett.* 114 (18 2015), p. 185502.

[213] J. Ye and J. Zhao. "The intersection contour minimization method for untangling oriented deformable surfaces". In: *Proceedings of*

the 11th ACM SIGGRAPH / Eurographics Conference on Computer Animation. EUROSCA'12. Lausanne, Switzerland: Eurographics Association, 2012, 311–316.

[214] S. Ye, Y. Ge, K. Zheng, S. Gao, S. Yu, G. Kurian, S. Indupuru, Y. L. Tan, C. Zhu, J. Xiang, A. Malik, K. Lee, W. Liang, N. Ranawaka, J. Gu, Y. Xu, G. Wang, F. Hu, A. Narayan, J. Bjorck, J. Wang, G. Kim, D. Niu, R. Zheng, Y. Xie, J. Wu, Q. Wang, R. Julian, D. Xu, Y. Du, Y. Chebotar, S. Reed, J. Kautz, Y. Zhu, L. J. Fan, and J. Jang. *World Action Models are Zero-shot Policies*. 2026.

[215] C. Yuksel. "A Fast & Robust Solution for Cubic & Higher-Order Polynomials". In: *ACM SIGGRAPH 2022 Talks*. SIGGRAPH '22. Vancouver, BC, Canada: Association for Computing Machinery, 2022. ISBN: 9781450393713.

[216] G. Zachmann. "Minimal hierarchical collision detection". In: *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*. VRST '02. Hong Kong, China: Association for Computing Machinery, 2002, 121–128. ISBN: 1581135300.

[217] H. Zhang, H. Feng, J.-L. Huang, and J. Paik. "Generalized modeling of origami folding joints". In: *Extreme Mechanics Letters* 45 (2021), p. 101213.

[218] Y. Zhou and K. Sze. "A geometric nonlinear rotation-free triangle and its application to drape simulation". In: *International Journal for Numerical Methods in Engineering* 89.4 (2012), pp. 509–536.

[219] Y. Zhu and E. T. Filipov. "A Bar and Hinge Model for Simulating Bistability in Origami Structures With Compliant Creases". In: *Journal of Mechanisms and Robotics* 12.2 (Feb. 2020), p. 021110.

[220] O. C. Zienkiewicz, R. L. Taylor, and J. M. Too. "Reduced integration technique in general analysis of plates and shells". In: *International Journal for Numerical Methods in Engineering* 3.2 (1971), pp. 275–290.

[221] D. Zorin. "Curvature-based Energy for Simulation and Variational Modeling". In: *Proceedings of the International Conference on Shape Modeling and Applications 2005*. SMI '05. 2005, 198–206.