CSE 621: Physics-based Modeling, Simulation, and Computing

Hong Qin
Department of Computer Science
State University of New York at Stony Brook
Stony Brook, New York 11794--4400
Tel: (631)632-8450; Fax: (631)632-8334
qin@cs.sunysb.edu
http://www.cs.sunysb.edu/~qin
The Main Theme of CSE621

• Physics-based modeling, dynamic simulation, and scientific computing for computer science, information science, and engineering

• Their widespread applications in the entire spectrum of visual computing discipline

• Our course objective is to demonstrate that physics-based modeling and computing is a fundamental and enabling computational framework that can facilitate visual information processing in general.
Our Approach is Unique!

- The Physics-based approach is unique, natural, and integrative!
- We are driven by and are focusing on various visual computing fields such as graphics (image synthesis), visualization, computer-aided geometric design, biomedical image processing, vision (image analysis), human-computer interaction, and virtual environments.
- Our research topics are centered on physics-based modeling, simulation, and scientific computing methodology and associated computational (computer science and engineering) methods for tackling theoretical and practical problems in widespread areas of visual computing.
Key Theory, Algorithms, and Techniques

- The rich theory of mathematical physics
- Geometric and solid modeling based on PDEs and energy optimization
- Deformation-centered geometric design techniques
- Deformable models for shape estimation and reverse engineering
- Variational analysis, optimization methods, level-set methods
- Numerical techniques with finite-difference and finite-element algorithms
- Differential equations for initial-value and boundary-value problems
- Force-driven haptic interaction
- Dynamic sculpting system
- Animation of flexible objects and their interactions
- Simulation of physical worlds and natural phenomena
- Energy, PDE, and optimization methods for graphics applications (including texture synthesis, texture mapping, surface parameterization and mapping, shape comparison, object retrieval, etc.)
- Novel theory for new types of splines
- Many many more topics beyond the above mentioned list
Motivation

• Natural phenomena are characterized by physical laws
• Deformation (as well as time-varying events) is ubiquitous ranging from macro-scale to nano-scale
• Graphics and visual computing aim to model and simulate physical worlds including objects, phenomena, behaviors, etc.
• Every component of graphics (including modeling, rendering, animation, and interactive techniques) and the entire visual computing field is relevant to physical laws
• Many geometric problems can be formulated by and transformed to physical problems (PDEs, energy optimization)
• Geometric design requirements are naturally represented by using optimization techniques
• Graphical display mimics what is happening in real-world (projection, photorealistic appearance, etc. are based on physics, especially, optics)
• For animation and interactive techniques, physical laws are also utmost significant
• Physics-based modeling gives rise to a large variety of applications in graphics, geometric design, dynamic simulation, visualization, medical image processing, etc.
• Physics-based modeling has been a very powerful tool for us to tackle many real problems in many real applications
Application Areas

- Geometric representation by means of physics and energy
- Interactive and dynamic editing for geometric design
- Digital geometric processing (meshes, points), denoising, smoothing, fairing, hole-filling, etc.
- Virtual surgery simulation using finite element methods for surgery training
- Physics-based haptic interface
- Animation such as morphing, smoking, firing, etc.
- You will see more applications later…
Geometric Processing

- Applying signal and image processing algorithms to geometry

De-noising

Enhancement
Shape Smoothing/Denoising

Figure 3: (A) Sphere partially corrupted by normal noise. (B) Sphere (A) after 10 non-shrinking smoothing steps. (C) Sphere (A) after 50 non-shrinking smoothing steps. (D) Sphere (A) after 200 non-shrinking smoothing steps. Surfaces are flat-shaded to enhance the faceting effect.

Figure 4: (A) Boundary surface of voxels from a CT scan. (B) Surface (A) after 10 non-shrinking smoothing steps. (C) Surface (A) after 50 non-shrinking smoothing steps. (D) Surface (A) after 100 non-shrinking smoothing steps. $k_{pa} = 0.1$ and $\lambda = 0.6307$ in (B), (C), and (D). Surfaces are flat-shaded to enhance the faceting effect.
Another Example on Smoothing

- Smoothing with volumetric constraints

**Figure 1:** Local vs. global volume preservation: The middle image shows the original noisy surface. On the left 100 smoothing iterations with the local strategy, on the right 100 iterations with global rescaling.
Real-time Simulation
Parameterization based on PDEs
Spherical Parameterization

Pawn (154 vertices)  Rabbit (543 vertices)  Triceratops (1,727 vertices)

Tutte Laplacian  Conformal Laplacian
More Parameterization Examples
Spherical parameterization (closed genus-0 surface)

- For closed genus-0 surface, the natural domain is the sphere

(courtesy of Hugues Hoppe)
Flow Simulation (Navier-Stokes Equation)
Natural Phenomena
Fluid Simulation
Cloud Animation

- No heat involved
    - Procedural modeling
  - Dobashi et al. 2002
    - CA model with humidity

- Simple linear relation between buoyancy force and temperature
  - Kajiya and Herzen 1984
  - Overby et al. 2000

- Completely physical based
  - Harris et al. 2003
    - Grid-based method with Euler’s equation
    - Based on heat transfer equation, experimental equations on buoyant force, phase transition, saturation, pressure lapse rate
    - Fully implemented on GPU
Particles
Flow Simulation

Image courtesy of Vee Hirsch, NASA Ames
Computer Art with Physical Interface
Scattered Data Fitting

From points to splines

- Reverse engineering
- Geometric modeling and processing

(points $P$) → surface $S$ (courtesy of Hugues Hoppe)
Sketch-based Shape Editing

(b) Translation in the direction of the plane normal
Interactive Manipulation

(a) 90 degree rotation around an axis through the plane center that spans an angle of 45 degrees with the plane normal
Shape Deformation and Editing
Motion Synthesis (Animation)
Bio-applications
Virtual Surgery
Brain Deformation

- Medicine
- Simulation
- Modeling
- Entertainment
Organ Deformation
Finite Element Simulation

Figure 1: Distributor Cap
Material Analysis (Virtual Prototyping)

compressive forces

displacement mapping
Material Modeling
FEM-based Deformable Objects
Animation: Morphing
PDE-driven Texture Synthesis
Natural Phenomena

- Sims 1990
- Stora 1999
Our Course

• A subset of key theory, algorithms, techniques, and applications
• Extensive topics with a main focus on our unique course mission
• Comprehensive lectures (focusing on geometric and physical intuition, good ideas, and application needs)
• Numerous slides, figures, images, and videos for easy understanding (after all, this is the nature of graphics and visualization)
• Active students’ involvements
Course Prerequisites

• Mathematical skills: calculus, linear algebra, analytic geometry, basic physics
• Computer science background: programming, basic graphics/visualization courses at the undergraduate level and the graduate entry-level
• Essentially, you need to have an undergraduate education in computer science or engineering with basic knowledge on graphics/visualization
• You need to speak to the instructor if you are not sure about your background knowledge
# Questionnaire

1. List your background courses/knowledge/education related to graphics/visualization, your current education level
2. What is the main goal/purpose for you to take this course (e.g., learn the knowledge, pursue a career in this area)
3. How does this course help your future professional career
4. Your expectations on the course
5. Your studying plan
6. Other important issues that you can think of about the course
If You are Serious

- Study my on-line, electronic course notes
- Review programming assignments carefully and start to implement them
- Think about your course project
- Write a proposal on your project and start to work on it
- Practice on the programming assignments
- Finish your project by the end of this semester
- Try to submit a paper if your project is really really new
- You are welcome to communicate with me via emails
My Contact Information

Hong Qin
Department of Computer Science
State University of New York at Stony Brook
Tel: (631)632-8450; Fax: (631)632-8334
qin@cs.sunysb.edu
http://www.cs.sunysb.edu/~qin
Office: Room 2426, CS Building
How to Get a “A”?

• NO midterm tests!
• NO final exams!
• ASSIGNMENTS and PROJECT only !!!
Important Information

- **WHEN:** MW 3:50pm --- 5:10pm
- **WHERE:** Social and Behavioral Sciences N107 (it is possible that we will move to our CS Seminar Room (Rm.1306) after the first week of our lectures).
- **OFFICE HOURS:** Monday 9:00am --- 12:30pm, Wednesday 9:00am --- 11:00am, 1:00pm --- 2:20pm, or by appointment!
- **CREDITS:** 3
- **HOW CAN I GET an “A”??
  - **NO** midterm tests!
  - **NO** final exams!
  - ASSIGNMENTS and PROJECT only !!!!
Important Information

- Two types of assignments/assignments
- Paper presentation
- Programming
Course Synopsis

- **Objectives**
  - Physics-based modeling, simulation, and scientific computing methodology
  - Integrated approach for visual computing applications

- **Concepts and framework**
- **Mathematical techniques**
- **Geometric computing**
- **Computational physics**
- **Numerical techniques**
- **A wide spectrum of applications**
CSE631 vs. CSE530

- Above entry-level graduate course
- Open to everyone!
- MS projects (thesis)
- Ph.D oriented and research driven
- Case studies through paper presentation
- Additional handouts from various books
- Pro-active involvement from students
- Different mechanism for course grade
- Group project in encouraged
- Higher standard
Course Facts

• This is an advanced graduate course!
• Can I take this course? YES, if YOU
  – Are a graduate student with CS background
  – Talk to the instructor
  – Have graphics/visualization background
  – Familiar with calculus/algebra/geometry
  – Have basic knowledge numerical analysis
  – Understand simple physical laws
Course Facts

- You do NOT need to take CSE528 or CSE530 prior to this course
- Having knowledge from CSE528 or CSE530 makes it much better
- However, you need to have taken CSE328, or CSE332, or equivalent courses elsewhere
- Lectures: paper presentation
- Class attendance is critical!
- No textbooks
Course Facts

• **Students are expected to**
  – Present several papers
  – Finish one course assignment
  – Complete one course project
  – Present your project in the class
  – Given system demonstration
  – Submit the final report

• **What projects are appropriate?**
  – Talk to Hong
  – Projects available from Hong
Pathway to Success

- Highly-motivated
- Hard-working
- Start as soon as possible
- Meet with Hong on a regular basis
- Actively interact with your fellow students
- Visit CS library frequently
Course Facts

- NOT a graphics/visualization course
- NOT a course to teach OpenGL
- Do NOT teach graphics (basic knowledge & programming skills should be acquired elsewhere)
- Learn fundamental algorithms and advanced techniques
- Study various visual computing applications
- Course projects lead to MS thesis (project) or Ph.D dissertation topics
### What Will Be Covered

<table>
<thead>
<tr>
<th>Mathematics</th>
<th>Computer graphics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geometry</td>
<td>CAGD/CAD/CAM</td>
</tr>
<tr>
<td>Numerics</td>
<td>Computer animation</td>
</tr>
<tr>
<td>Physics</td>
<td>Visualization</td>
</tr>
<tr>
<td>Dynamics</td>
<td>Computer vision</td>
</tr>
<tr>
<td>Algorithm</td>
<td>Medical imaging</td>
</tr>
<tr>
<td>Simulation</td>
<td>Virtual environments (VR, AR)</td>
</tr>
<tr>
<td>Modeling</td>
<td>Artificial life</td>
</tr>
</tbody>
</table>
Paper Resources

- Access libraries for journals and conferences’ proceedings
- Relevant journals
- Major conferences:
  - Siggraph, Visualization, Eurographics, Pacific Graphics, Graphics Interface, Solid Modeling, Shape Modeling, ACM I3D Symposium, ICCV, CVPR, MICCAI, etc.
Grading Schemes

• No midterm, No final exam.
• 100% on Assignment/Project
• Individual project (one student), or group project (two students)
• Paper reading and presentation (other people’s work, one or two papers at most): 15%
• Assignment: 5%
• Course project: 80%, plus the additional bonus
• Class attendance: 10%
Course Project

• Two-page project proposal (10%)
• Preliminary demonstration in the middle of the semester (10%)
• Oral presentation and final demonstration (20%)
• Working system + software codes (20%)
• Project report (10-15 pages) (20%)
• Target dates
  – Warm-up assignment --- preferable 3rd week, but really anytime!
  – Paper reading --- throughout the semester
  – Paper presentation --- towards the end of the semester
  – Proposal --- 4th week
  – mid-term demo --- 8th week
  – Final presentation --- at the end of the semester
  – Project report --- at the end of the semester
• Individual or group project
Project Requirements

- Interactive interface (graphics-based)
- Intuitive and easy to understand
- Efficient (fast, high-performance)
- Basic functionalities
- Examples
- Flexible and easy to generalize
- Project plan (multiple check-points and phases)
- Individual or group project
- Office hours / individual meetings
- Penalty for late submission
Outline

- Geometric and solid modeling
- Physics-based modeling
- Deformable models
- Mathematics
- Numerical solutions
- Visual computing applications
  - Graphics
  - Vision
  - Geometric design
  - Visualization
  - Virtual environments
Geometric and Solid Modeling

- Polygonal meshes
- Polynomials and splines
- Parametric curves and surfaces
- Bezier
- B-splines and NURBS
- Triangular & irregular patches
- Multisided surfaces
- Subdivision objects
- Manifold splines
- Implicit functions
- CSG and volumetric models
Geometric and Solid Modeling

• Wavelets and hierarchical models

• Special shapes
  – Ruled surface
  – Developable surface
  – Offset
  – Sweeping
  – Swung surface
  – Surface of revolution

• Solid models
  – Constructive solid geometry (CSG)
  – Boundary representation (B-rep)
  – Cell decomposition
Geometric Operations

• Trimming, Intersection
• Approximation, Fitting
• Interaction
  – Control point, weight, knot vector
• Interpolation
  – Scattered data, curve network, regular dataset
• Constraints
  – Shape-preserving, convex-preserving
• Continuity
• Optimization
• Computational geometry
• Differential geometry
• Efficient algorithm
Physical Models

- Rigid and non-rigid models
- Mass-spring lattices
- Parameterized models
- Elastic and inelastic bodies
- Dynamic B-splines & NURBS
- Dynamic subdivision models
- Particle systems and fluid models
- Superquadric geometry
- Snakes: dynamic contour models
- Symmetric models
- Finite elements
Geometric Modeling Techniques

- Interpolation
- Approximation
- Optimization
- Interaction
- Forces
- Constraints
- Dynamic sculpting
- Continuity
- Differential geometry
- Hierarchical techniques
- Level of Details (LOD)
- Simplification
Dynamic Modeling

- Mass, damping, elastic energy
- Internal and external forces
- Geometric constraints
- Optimal control of physical models
- Lagrange mechanics
- Mathematical physics
- Multi-body (rigid and non-rigid) simulation
- Local and global deformations
- Viscoelasticity, plasticity, fracture
- Thermoelasticity, heat transfer, melting
- Fluid dynamics
## Numerics

- Linear algebra & matrix computation
- Linear & nonlinear systems
- Static & dynamic problems
- Initial-value & boundary-value problems
- The finite difference method
- The finite element method
- Calculus of variations
- Direct & iterative methods
- Differential equations of equilibrium
- Numerical analysis
- Multiresolution algorithms
Visual Computing Applications

- Morphing and image warping
- Surface blending and solid rounding
- Animation and simulation
- Free-form deformation
- Reverse engineering
- Shape reconstruction
- Sparse data fitting
- Interactive sculpting
- Model simplification
- Object motion tracking
- Feature extraction and segmentation
Visual Computing Applications

- Visualization
- Variational design
- Shape interrogation and control
- Biomedical imaging
- Interface and virtual environments
- Texture mapping
- Artificial life
- Plastic surgery
- Natural phenomena
Graphics

Geometric Modeling

Mathematics

Finite Element Method

Curve and Surface Survey

- Parametric representation for curves and surfaces