Haptics
ME7960, Sect. 007
Lect. 4: Haptic Rendering
Basics

Spring 2011
Prof. William Provancher
University of Utah
Salt Lake City, UT USA

We would like to acknowledge the many colleagues whose course materials were borrowed and adapted in putting together this course, namely: Drs. Allison Okamura (JHU), Katherine Kuchenbecker (U Penn), Francois Conti, Federico Barbagli, and Kenneth Salisbury (Stanford), Ed Colgate (Northwestern), Hong Tan (Purdue), Blake Hannaford and Ganesh Sankaranarayanan (U. Washington), and Karon MacLean (UBC).

Today’s Class

- Finish up → Physical Haptic Systems
  - Actuators for haptic/tactile systems
    - Actuator types
    - Actuator examples
  - Sensors
  - Amplifier
  - Control Card
  - Overview of Control Architecture
    - Overall system view
    - Kinematics
- Haptic Rendering Basics
- Overview of Lab code template
- No Readings today

Basic procedure for force feedback with an impedance-type device

- Read the position of the user through the haptic display
- See if there is a collision with objects in the virtual environment
  - If there is, calculate forces
- Send corresponding torque commands to motors, and change the virtual environment state

Static Rigid Body Interaction

- The virtual environment pretends that the user is holding onto a fictional rigid body though the haptic device handle
- This rigid body interacts with other “rigid” bodies in the virtual environment.
- With impedance control, nothing is perfectly rigid: $F = kx$
Rendering a Simple Wall
Penalty-Based Method
(think of like a compression spring)

- Set up coordinate systems and terminology:
  
  **HIP** = Haptic Interaction Point
  (This is the physical location of the haptic device’s handle and is the same as \( \vec{p}_{user} \) in the next slides)
  
  \[
  \vec{p}_{user} = [x_{user}, y_{user}, z_{user}]
  \]

---

Wall aligned with axes

If \( x_{user} > x_{wall} \), \( F = k (x_{wall} - x_{user}) \)

Stiffness \( k > 0 \)

When the tool is not a point

If \( x_{user} + r > x_{wall} \), \( F = k (x_{wall} - x_{user} - r) \)

Stiffness \( k > 0 \)
Hard sphere (3D)

\[ r = \sqrt{(x_u - x_s)^2 + (y_u - y_s)^2 + (z_u - z_s)^2} \]

\[ \hat{r} = \frac{1}{r} \begin{bmatrix} x_u - x_s \\ y_u - y_s \\ z_u - z_s \end{bmatrix} \]

if \( r < R \), \( F = k(R - r)\hat{r} \)

Non-aligned wall

\[ \vec{r} = \begin{bmatrix} x_u - x_w \\ y_u - y_w \\ z_u - z_w \end{bmatrix} \]

if \((\vec{r} \cdot \hat{n}) > 0 \rightarrow \) no collision,
if \((\vec{r} \cdot \hat{n}) < 0 \rightarrow \) collision

\[ d = |(\vec{r} \cdot \hat{n})| \]

\[ F = kd\hat{n} \]

(Does this work? Check yourself)

Inside a Box

\[ F_x = 0 \]
\[ F_y = 0 \]

if \( x_{user} > x_{wall-max} \)

\[ F_x = F_x + k(x_{wall-max} - x_{user}) \]

if \( x_{user} < x_{wall-min} \)

\[ F_x = F_x + k(x_{wall-min} - x_{user}) \]

if \( y_{user} > y_{wall-max} \)

\[ F_y = F_y + k(y_{wall-max} - y_{user}) \]

if \( y_{user} < y_{wall-min} \)

\[ F_y = F_y + k(y_{wall-min} - y_{user}) \]

Outside a Box

\[ F = 0 \]

if \([ (x_{user} < x_{wall-max}) \& (x_{user} > x_{wall-min}) \]

\& \([ y_{user} < y_{wall-max} ) \& (y_{user} > y_{wall-min} ) \])

Then... what force should be displayed??
Limitations of “Penalty-Based” Methods

- (a) Lack of locality
- (b) Force discontinuities (snap-thru)
- (c) “Pop-thru” of thin objects

Basics of springs

\[ F = K \cdot x \]

Where \( x \) \equiv amount spring is deflected

“God-Object” and Proxy
(implemented like a tension spring)

- Keep track of virtual object on the surface
  - simple to compute the proxy position from the HIP given a previous position
- A related concept is the “God Object” and proxies (see Zilles & Salisbury 1997 and Ruspini & Khatib 2001)

Implementing a virtual wall with a proxy
What You Feel

- Object has limited stiffness, but it is not “deformable” in a global sense
- Display proxy to reinforce visual feedback of stiff surface
  - Never show the point penetrating the surface, even if it is
  - Psychophysical studies have shown that this makes the surface appear stiffer/harder
  - Proxies are useful for implementing other haptic phenomenon
    - e.g., friction, virtual coupling

System Requirements

- Why do you need to run fast (> 500 Hz usually)?
- Why do you need high position resolution?
- Why do you need high force resolution?

- To prevent the feeling of “steps” and instability
- We do this for you automatically in the code template we provide for use with the Novint Falcon