DIGITAL PROXY SIMULATION FOR ROBOTIC HARDWARE

Xi Chen, Ph.D., Douglas E. Barker, Ph.D., James A. Bacon, James D. English, Ph.D.
Energid Technologies, Cambridge, MA

25 August 2014
Why Proxy Simulation

• Field tests are common in preparation for robotic missions
  – Maintaining people for mission planning, operation, remote monitoring may be taxing
  – Use of alternate locations is often deficient in truth data

• Need a digital simulation
  – Serves as a proxy of any number of robots in an environment
  – Provides ground truth data for comparison
What Capabilities Are Needed

• Easily Configurable Environments
• Real-time Dynamic Simulation
• Rendering & Sensor Simulation
• Control Architecture
What Capabilities Are Needed

- Easily Configurable Environments
- Real-time Dynamic Simulation
- Rendering & Sensor Simulation
- Control Architecture

No less than 30 FPS
What Capabilities Are Needed

- Easily Configurable Environments
- Real-time Dynamic Simulation
- Rendering & Sensor Simulation
- Control Architecture
What Capabilities Are Needed

- Easily Configurable Environments
- Real-time Dynamic Simulation
- Rendering & Sensor Simulation
- Control Architecture
Digital Proxy Simulation

- Easily Configurable Environments
- Real-time Dynamic Simulation
- Rendering & Sensor Simulation
- Control Architecture
Environment Configuration

- Add objects (robots, rocks, terrains, etc.) to the simulation
- SolidWorks Converter
  - Physical extents
  - Mass properties
  - Joint configurations
- GeoTIFF Loader for DEM models and orthophoto
Environment Configuration

- **Large Terrain Loading**
  - Build a database of terrains using the Virtual Planet Builder
  - Can be paged by the Open Scene Graph (OSG) at runtime
  - Use axis aligned bounding box (AABB) to determine tile

Rover traversing Mojave Desert (~20km range)
Environment Configuration

- Scene Configuration XML to assemble objects
- Shape primitive plugin for changing BV or adding objects

A simulation assembled by a scene configuration XML with a terrain, a rover, and two rocks with BV, and a cone added by the shape plugin.
Contact Forces

- High-fidelity force modeling is crucial for digital proxy simulation of vehicles
- Impact forces using penetration depth calculation

\[ f_{B \rightarrow A} = (k_p - k_d v) n_{B \rightarrow A} \]  
Non-linear spring model
Contact Forces

• High-fidelity force modeling is crucial for digital proxy simulation of vehicles

• Coulomb friction
  – Static: $|\mathbf{f}_{sf}| \leq \mu_s |\mathbf{f}_n|$ if $\mathbf{v}_{pB/A} = 0$
  – Kinetic: $\mathbf{f}_{kf} = -\mu_k |\mathbf{f}_n| \frac{\mathbf{v}_{pB/A}}{|\mathbf{v}_{pB/A}|}$ if otherwise

  – Methods based on breaking spring and modeling through time-step
    • Initially, if $\mathbf{v}_{pB/A} = 0$, $\mathbf{f}_{sf} = -k_s \delta_{B\rightarrow A} - \lambda_s \mathbf{v}_{pB/A}$
    • If $|\mathbf{f}_{sf}| \leq \mu_s |\mathbf{f}_n|$, still in static mode, otherwise, in kinetic mode
    • The damping is still considered in kinetic mode
      $\mathbf{f}_{sf} = -\lambda_s \mathbf{v}_{pB/A}$
Advanced Friction

- High-fidelity terrain-wheel interaction is crucial for digital proxy simulation of vehicles
- Advanced friction using Bekker theory
  - Input: sinkage, slip ratio, and slip angle
  - Output:
    - Drawbar pull: \[ F_x = rb \int_{\theta_0}^{\theta} (\tau_x(\theta) \cos \theta - \sigma(\theta) \sin \theta) d\theta \]
    - Side force: \[ F_y = rb \int_{\theta_0}^{\theta} (rb \tau_y(\theta) + R_b (r - z(\theta) \cos \theta)) d\theta \]
    - Vertical force: \[ F_z = rb \int_{\theta_0}^{\theta} (\tau_z(\theta) \cos \theta + \sigma(\theta) \cos \theta) d\theta \]
    - Wheel moment: \[ M_w = r^2 b \int_{\theta_0}^{\theta} \tau_x(\theta) d\theta \]
  - Soil characteristics: such as cohesion, internal friction angle, cohesive modulus, frictional modulus, shear deformation modulus, sinkage component, and sinkage ratio
Limitations and Improvements

• Penetration depth is based on shapes (too crude). Hard to estimate sinkage and slippage.

• Alternative: particle simulation with GPU computation.
  – Real-time
  – Finer forces, more natural
  – Terrain shape changeable
  – Require good graphics card
  – Will compare the two
Dynamic Simulation

- From force to acceleration and new states
- In Actin, all objects are manipulators. A manipulator is a tree of links with mass properties and kinematics (D-H parameters)

![Manipulator Diagram]

- Integration methods
  - Composite rigid body algorithm for DOF < 12, runs $O(N^3)$
    \[
    \tau = M(q)\ddot{q} + C(q)\dot{q} + G(q) + DA_b + B
    \]
  - Featherstone’s articulated-body algorithm, runs $O(N)$, but with a large constant
- Capable of faster-than-real-time. In DPS, >= 30fps
Real-time Ray-tracing

- High-fidelity fast rendering and sensor simulation
- Ray-tracing is a method of creating physics-based photo-realistic images of a virtual scene
  - Traditionally an offline method. GPU computation in real-time.
  - True reflections are faithfully reproduced using ray tracing.

Ray-tracing performance of two NVIDIA graphics cards on the proxy simulation scene
Lighting, Shadow, and Camera

- Accurate sensor models with realistic lighting and shadows
- Lights (directional and positional) can be added, removed, and edited. Shadows are faithfully rendered.
Lidar Simulation

- Ray-casting is slow
- OpenGL Approach
  - z-buffer
    \[ z = (2^N - 1) \cdot \left( a + \frac{b}{d} \right) \Rightarrow d = \frac{b (2^N - 1)}{z - a (2^N - 1)} \]
  - Accuracy
    \[ \delta = -\frac{d_1 d_2}{(2^N - 1)b} \approx -\frac{d^2}{(2^N - 1)b} = 2.4 \text{mm} \]
    for \( d = 350 \text{m}, N = 24 \text{bit} \)
  - Distortion Correction
    \[ s_i = \frac{d_{\text{Near}}}{z_{\text{Near}}} = \sqrt{\tan^2(\theta_i) + \tan^2(\phi_i) + 1} \]
    \[ d_i = \frac{b (2^N - 1) \cdot s_i}{z_i - a (2^N - 1)} \]
    also added noise
• Local planning and control are necessary for algorithm testing and are convenient to add

• Demonstration: Path planning and control added to support lunar mission using A* search and path following
Conclusion

• The digital proxy simulation was successfully applied to dynamically simulate NASA rovers, military vehicles, research vehicles, and commercial exploration robots in real-time.

• The digital proxy simulation fills the need to seamlessly serve as a replacement for real hardware and can be applied to simulate any articulated robotic vehicles with complex sensors and control systems in complex environments.
Thank you