ASPECT Phase Change Simulation Technical Work

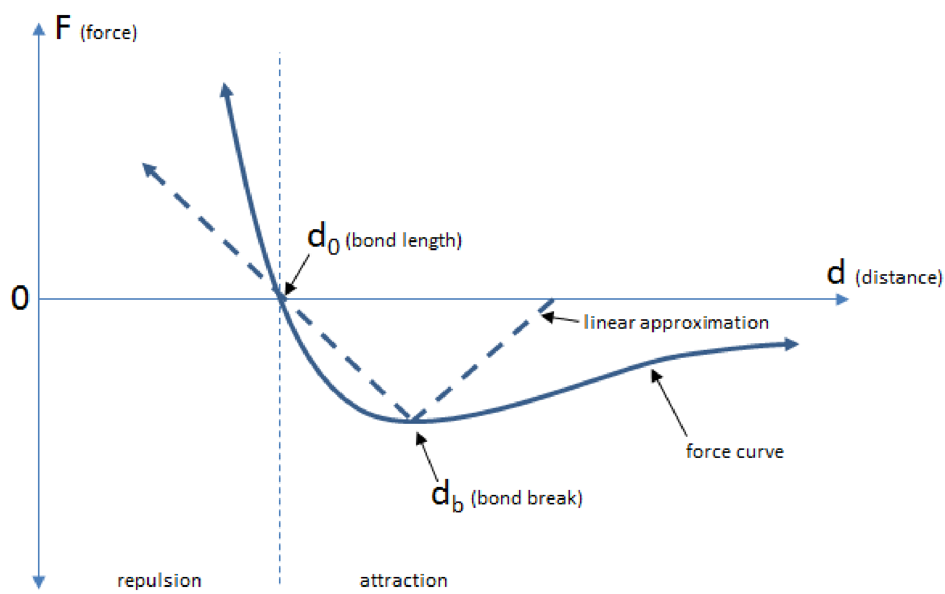
We continue to refine and add functionalities to the C++ dynamically-linked library (DLL) that was developed in Year 1. This technology integrates the Novint Falcon device with the Unity game engine, via a wrapper around Novint's Haptic Device Abstraction Layer (HDAL) library. Previously we added capabilities to read device position and button information from the device within Unity, and send computed forces from Unity to the device. In Year 2 we have added various haptic effects to enable our Molecular Forces simulation.

Spring Forces

We provide the ability to add multiple spring force effects, specifying the position, rest length, spring constant, and damping coefficient for each spring. The spring force is calculated based on Hooke's law, with a damping effect to reduce vibrations. Linear spring forces are the basis of our intermolecular force and surface effects described below.

Intermolecular Forces

We have developed a piecewise linear approximation of an intermolecular force (IMF) curve. Multiple IMF effects can be added, simulating the repulsive and attractive forces of multiple molecules in a scene. The user specifies the position (molecular position), bond length (spring rest length), maximum length (inversion point beyond which the bond is "broken"), strength (spring constant), and damping coefficient (heuristic to reduce vibrations). The figure below illustrates this piecewise linear approximation, compared to a schematic IMF curve.



The main features of the force curve are maintained: a repulsive force when the distance *d* is less than the bond length d_0, an attractive force when *d* is greater than d_0, and a falling off of the attractive force past a given distance d_b where the bond "breaks." The advantages of the piecewise linear approximation are ease of specification by the user (position, bond length, and bond strength), and ease of calculation, while maintaining main features of the force curve. Intermolecular forces are used to simulate microscopic forces for solids, liquids, and gases in our simulation.

Surfaces

We have integrated our haptic interface with Unity's physics engine to enable force feedback to simulate touching the surface of 3D objects in the scenario. We use the standard haptic interface approach of using spring forces to simulate surfaces. Unity's physics engine is used for collision detection between the haptic probe object and the objects in the scene. The collision surface contact point and surface normal are passed to the haptic device, along with a spring coefficient and damping coefficient to determine the perceived hardness or softness of the surface. The surface contact point and surface normal define a plane used as a local approximation of the surface. The contact point and normal are updated at the rate of the Unity physics engine (typically 50 Hz), whereas the calculated spring force felt by the user is updated at the much higher haptic update rate (typically 1000 Hz). The spring force is applied along the plane normal, based on the distance of the haptic probe below the plane. In this manner adding haptics to feel the surfaces of complex scenes can be easily integrated by relying on Unity's existing physics engine for collision detection, while retaining the ability to feel stable forces. Surfaces are used to simulate macroscopic solids in our simulation.

Deformable Objects

We have also integrated haptics with a dynamic mesh deformation Unity plugin to enable a visual representation of the hardness or softness of surfaces, along with a haptic representation. Our solution includes two probe objects, one which collides with the deformable mesh to induce deformations, and another invisible probe which collides with an invisible non-deformable proxy of the deformable object to generate forces as with a standard surface. Deformable objects are used to simulate macroscopic solids with different degrees of softness in our simulation.

Viscosity

We also provide the ability to apply a viscosity effect when the haptic probe is in the interior of designated objects in the scenario. The viscosity effect applies a damping force opposite to the direction of the current velocity vector of the haptic device, proportional to the velocity magnitude. The amount of viscosity is controlled by a damping coefficient parameter. This effect is used to simulate macroscopic fluids in our simulation.