Physical Modeling of Heterogeneous Embedded Deformable Object Deformation

Sara Farag and Wael Abdelrahman *
Center of Intelligent Systems Research (CISR)
Deakin University

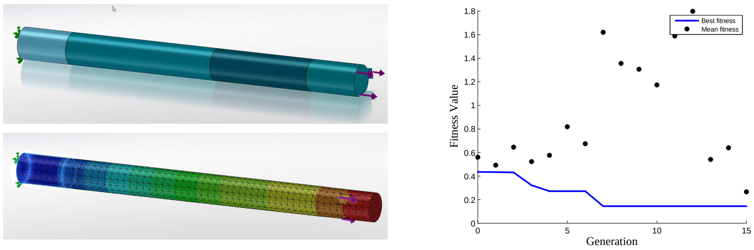

Figure 1: *(Right) A 1D heterogeneous bar deformed under external forces (shown as red arrows) and fixed from one side (green fixtures at the left end). The material distribution affects its behavior and optimised values are required of the shape (interpolation) functions. (Left) Using a genetic algorithm to estimate the shape functions.*

1 Introduction

Simulation of the interactions with deformable models is important in many applications such as medical training and tissue engineering. To physically model the 3D object, both the inner and outer segments need to be considered. This implies dealing with different materials and hence different deformation behavior. Thus, a physically-based simulation needs to augment the behavior of embedded materials when the materials are in direct physical contact, and produce a plausible net result in both visual and haptic cues.

A straightforward solution is to model the object as a mass spring system (MSS) and change stiffness in different material regions. Although this is a direct and fast approach, it is not usually convergent or matches the constitutional laws. Other approaches use finite element methods (FEM) to simulate the interactions. For static simulation the governing equation is:

$$K_{ab} u_1^b = F^a \tag{1}$$

where the stiffness matrix (for the 1D case) is given by:

$$K_{ab} = A \int_0^L \frac{2\mu(1-\nu)}{1-2\nu} \frac{\partial N^a(x_1)}{\partial x_1} \frac{\partial N^b(x_1)}{\partial x_1} dx_1 \tag{2}$$

Here μ is the shear modulus, ν is the Poisson's ratio, A is the cross section, L is the 1D bar length, u_1^b is the displacement vector, F^a is the external force vector, and N^a and N^b are the shape functions. The shape functions are used in the interpolation of displacements along the object elements. In classic FEM, linear or quadratic shape functions are used for the 1D case where $-1 \leq \xi \leq 1$ Eq. (3). However, they do not reflect the material distribution of the object because they are static and designed independently to be generic [Nesme et al. 2009].

$$\begin{aligned} N^1(\xi) &= 0.5(1-\xi) \\ N^2(\xi) &= 0.5(1+\xi) \end{aligned} \tag{3}$$

We propose a shape function estimation techniques using genetic algorithms. The technique uses empirical data sets as a fitness function to judge the accuracy of the solution.

*{sfara, wmam@deakin.edu.au}

2 Our Approach

In order to use an optimisation technique such as genetic algorithms [Mitchell 1998], we need to have an individual representation (chromosome) and a fitness function to evaluate the performance of the individuals. The algorithm also has tuning parameters which are the mutation and the crossover functions. For the estimation of the shape function, we represented the individual as a vector with length equal to the number of different materials. The fitness function used is a critical part of the system as it needs to be matching realistic behavior and, quick to calculate as well. The parametric methods such as the generic functions used in FEM are not computationally efficient [Nesme et al. 2009]. Thus, we propose using a data-driven approach with collected data sets using a robot arm and force sensor.

The fitness function is defined as the average Euclidean distance between the displacement vector values that an individual scores against a force vector with and the pre-computed data set of displacements by the robot arm. Thus, mathematically speaking for an individual represented as $[\phi_1, \phi_2, \ldots, \phi_n]$ we define its fitness as:

$$fit(ind) = \frac{\sum_1^m (\sqrt{\sum_1^n (\mathbf{u}_{ind}^i - \mathbf{u}_{target}^i)^2}/n)}{m} \tag{4}$$

Where m is the number of external applied forces, n is the length of chromosome vector, $\mathbf{u}_{ind}$ is the displacement vector generated from using ϕ_S which are the shape functions derivatives in Eqs. (1) and (2) and $\mathbf{u}_{target}$ is the displacement vector generated empirically by the robot arm.

For the 1D case the optimised model behavior is more realistic than the generic shape functions and no need to enforce constraints as in [Nesme et al. 2009].

References

MITCHELL, M. 1998. *An Introduction to Genetic Algorithms*. MIT.

NESME, M., KRY, P. G., JEŘÁBKOVÁ, L., AND FAURE, F. 2009. Preserving topology and elasticity for embedded deformable models. In *ACM Transactions on Graphics (Proc. of SIGGRAPH)*, ACM. to appear.

SIGGRAPH 2010, Los Angeles, California, July 25 – 29, 2010.
ISBN 978-1-4503-0210-4/10/0007