FEM cheatsheet

Tao Du

November 2018

Disclaimer: this is a cheatsheet I made after reading Eftychios' SIGGRAPH tutorial on FEM. My summary is quite sketchy and should in no way be considered as a complete or rigorous description of FEM materials. I find Eftychios's tutorial extremely useful, but I keep forgetting important concepts in his tutorial if I don't do FEM for a while. So I write this cheatsheet to help me quickly refresh my knowledge.

1 Strain

You have an undeformed elastic body. You deform it and observe the deformed elastic body. Let $X \in \mathbb{R}^3$ be a point in the undeformed elastic body. Let $\phi : \mathbb{R}^3 \to \mathbb{R}^3$ map X to its location x in the deformed body: $x = \phi(X)$.

Let $F : \mathbb{R}^3 \to \mathbb{R}^{3\times 3}$ be the Jacobian of ϕ. We call F the **deformation gradient**. For any X, F describes how it deforms locally — it is the first-order approximation of the deformation by definition.

While F precisely defines the deformation, it is not always an intuitive measure of the strain. Consider a rigid transform $x = \phi(X) = RX + b$ where R is a rotation and b is a translation. Here $F = R$, but the elastic body itself does not undergo any deformation at all. As a result, people manually designed a few **strain tensors**:

- the Green strain E is defined as $E := 0.5(F^\top F - I)$. It discards the rigid transform part from the deformation. You can verify it by letting $F = R$.
- the Cauchy strain tensor $\epsilon := 0.5(F + F^\top) - I$ is also widely used. It is a linear approximation of E and no longer discards the whole rigid transform from the deformation, but it is easier to compute.

2 Material Model

When an elastic body deforms, it generates potential energy. Let $\Psi : \mathbb{R}^{3\times 3} \to \mathbb{R}$ be the **energy density function**. The amount of energy after it undergoes deformation ϕ is defined as:

$$W = \int_X \Psi(F(X))dX$$

Note that Ψ is a function of the deformation gradient, which stresses out the assumption that the energy purely depends on the local deformation.

The property of different materials is fully described by Ψ. Intuitively, given certain amount of deformation described by F, Ψ shows how much energy is generated. Equivalently, people sometimes model Ψ using E or ϵ, which are also functions of F.

3 Stress

We have related strain to energy. Now we connect energy to stress.

Consider a particular point X_i and the corresponding $x_i = \phi(X_i)$. The elastic force generated from the potential energy W is defined as:

$$f_{ik} = -\frac{\partial W}{\partial x_{ik}} = -\int_X \frac{\partial \Psi(F)}{\partial F} : \frac{\partial F(X)}{\partial x_{ik}} dX$$

where $k = 1, 2, 3$ represents the index of elements in the elastic force. We call the first term inside the integral the **stress tensor** P, which is a 3×3 matrix:

$$f_{ik} = -\int_X P(F) : \frac{\partial F(X)}{\partial x_{ik}} dX$$

Note that P purely depends on the material model and $\frac{\partial F}{\partial x_{ik}}$ solely relies on the deformation.

4 Discretization

4.1 Discretizing ϕ

We sample n points $X_1, X_2, \cdots, X_n$ and define $x_i := \phi(X_i), i = 1, 2, \cdots, n$. For tetrahedra or hexahedra, X_i will simply be their corners. The goal is to approximate ϕ by interpolating x_i. Let $N_i : \mathbb{R}^3 \to \mathbb{R}^+$, which is usally called a **basis function**. The mapping function ϕ is then replaced with the following discretized version:

$$\phi(X) \approx \sum_i N_i(X) x_i$$

Note that here X can be any point in the undeformed body, it does not have to be limited in the sef of X_is. ϕ is still a continuous function.

4.2 Discretizing F and P

The deformation gradient is the Jacobian of ϕ. Note that x_i is constant, so

$$F(X) = \sum_i x_i (\nabla N_i(X))^\top$$

The gradient of N_i is a 3D column vector. Again, F is also a continuous function defined on all Xs.

Now you know how to compute F. Since P is a function of F only, you also know how to compute P.

4.3 Discretizing nodal force f_i

FEM simulation is a Laplacian method: you track the states of lots of particles, you compute the force at these particles, and you use the force to update their states. All we need to know is how to compute the nodal force at x_i. We already know how to compute $P(F)$, so the remaining task is to compute $\frac{\partial F(X)}{\partial x_i}$. To avoid confusion, I rewrite F as:

$$F(X) = \sum_j x_j (\nabla N_j(X))^\top$$

The derivative of F with respect to x_i is a $3 \times 3 \times 3$ tensor. I will show you how to compute each element. Consider $\frac{\partial F}{\partial x_{i1}}$, where x_{i1} is the first element of x_i. Note that N_j does not depend on x_i at all, we have:

$$\frac{\partial F(X)}{\partial x_{i1}} = e_1 (\nabla N_i(X))^\top$$

where $e_1 = (1, 0, 0)^\top$. In general, we have

$$\frac{\partial F(X)}{\partial x_{ik}} = e_k (\nabla N_i(X))^\top$$

where $k = 1, 2, 3$. As you can see, the right hand side no longer depends on any x_i.

Finally, we need to discretize the integral in the definition of f_i. The whole space of X is first discretized with finite elements (typically tet meshes or hex meshes), and the integral is then replaced with a sum over integrals defined in each element. I will skip this part and assume we are now dealing with the integral inside a single tet or hex.

4.3.1 Tet element

For a single tet, I have only 4 X_is: X_1, X_2, X_3, and X_4. People realize that a linear mapping $\phi(X) = AX + b$ can be found such that $x_i = \phi(X_i), \forall i$. So you don't really need basis functions in this case. Let's compute A and b:

$$A \begin{bmatrix} X_2 - X_1 & X_3 - X_1 & X_4 - X_1 \end{bmatrix} = \begin{bmatrix} x_2 - x_1 & x_3 - x_1 & x_4 - x_1 \end{bmatrix}$$

Let $D_m := [X_2 - X_1, X_3 - X_1, X_4 - X_1]$ and $D_s := [x_2 - x_1, x_3 - x_1, x_4 - x_1]$. We have $A = D_s D_m^{-1}$ and $b = x_1 - AX_1$.

Computing P Now $F(X) = A$ becomes a constant matrix in each tet, so $P(F) = P(A)$ is also constant.

Computing $\frac{\partial F}{\partial x_{ik}}$ Since $F = A = D_s D_m^{-1}$, D_m is constant, and D_s is linear on x_i, we conclude F is linear on x_i and $\frac{\partial F}{\partial x_{ik}}$ is constant:

$$\frac{\partial F}{\partial x_{ik}} = \frac{\partial D_s}{\partial x_{ik}} D_m^{-1}$$

$\frac{\partial D_s}{\partial x_{ik}}$ is trivial, so we don't list it here.

Putting it together, we conclude everything inside the integral is constant, so

$$f_{ik} = -P : \frac{\partial F}{\partial x_{ik}} V$$

where V is the volume of this tet element.

4.3.2 Hex element

This is the case where the basis function really plays a role. In this case, ϕ is not linear, so F is not constant, and therefore P is not constant. We need to discretize the integral with a few sample points inside the hex element. Assume we use $Y_1, Y_2, \cdots, Y_m$ to approximate the integral (there are systematic rules to choose them, which we will explain shortly):

$$f_{ik} \approx -\frac{dV}{m} \sum_{Y_j} P(F(Y_j)) : \frac{\partial F(Y_j)}{\partial x_{ik}}$$

We already explained how to evaluate P and $\frac{\partial F}{\partial x_{ik}}$ at arbitrary X, so we can compute f_{ik} without any problems given $Y_1, Y_2, \cdots, Y_m$.

Here is a sketchy description of how to choose Y_i: consider approximating an integral of a function f over $[-1, 1]$: $\int_{-1}^{1} f(x)dx$. If I am asked to use one sample to compute this integral, a reasonable choice would possibly be $\int_{-1}^{1} f(x)dx \approx \int_{-1}^{1} f(0)dx = 2f(0)$. Of course, more samples lead to more accurate estimation. For some reason, people find using the sum of two samples at $\pm 1/\sqrt{3}$, i.e., $f(-1/\sqrt{3}) + f(1/\sqrt{3})$, can reasonably approximate this integral. This motivates people to sample 8 Y_is in the hex defined by $X_1, X_2, \cdots, X_8$ in the following way: we first scale and shift the hex element such that it is defined on $[-1, 1]^3$, then sample two points at $\pm 1/\sqrt{3}$ along each axis.

5 Forward Simulation

- Initialize X_i, e.g., the corners of your initial tet or hex mesh.
- Initialize $x_i = X_i$ (could be other values as well).
- Initialize $v_i = 0$ (could be other values as well).
- Compute f_i for each x_i using X_i and x_i.
- Add external forces to f_i if necessary.
- Update $x_i \leftarrow x_i + v_i dt$.
- Update $v_i \leftarrow v_i + f_i/m_i dt$.

6 Static Force Equilibrium

We now understand the forward process of FEM: given x_i, we know how to compute f_i at each node. Now we explain how to do the other direction: given desired f_i, figure out what x_i is needed. If you think of f_is as a (possibly) nonlinear function of x_is, then it is essentially a root finding problem:

$$\mathbf{f}(\mathbf{x}) = \mathbf{f}_{ext}$$

Here $\mathbf{f}$ and $\mathbf{x}$ stacks all f_is and x_is. Note that each f_i is a function defined on all x_is.

To apply Newton's method, we need to know how to compute the derivative of $\mathbf{f}$ w.r.t $\mathbf{x}$. Here we show how to compute $\frac{\partial f_{ik}}{\partial x_{jl}}$:

$$f_{ik} = -\int_X P(F) : \frac{\partial F(X)}{\partial x_{ik}} dX$$
$$\frac{\partial f_{ik}}{\partial x_{jl}} = -\int_X \frac{\partial P(F)}{\partial x_{jl}} : \frac{\partial F(X)}{\partial x_{ik}} dX$$

We don't have $\frac{\partial^2 F}{\partial x_{ik} \partial x_{jl}}$ because $\frac{\partial F}{\partial x_{ik}}$ does not depend on x. Now let's take a look at the first term:

$$\frac{\partial P}{\partial x_{jl}} = \sum_s \sum_t \frac{\partial P}{\partial F_{st}} \frac{\partial F_{st}}{\partial x_{jl}}$$

We already know how to compute $\frac{\partial F_{st}}{\partial x_{jl}}$. The first part $\frac{\partial P}{\partial F_{st}}$ depends on the material model, and it is a constant value for linear materials. For tet mesh, you replace the integral with multiplying V; for hex mesh, you replace it with the sum over Y_is.