Measuring cellular traction forces on non-planar substrates - supplementary material

Jérôme R. D. Soiné,[1,2] Nils Hersch,[3] Georg Dreissen,[3] Nico Hampe,[3]
Bernd Hoffmann,[3] Rudolf Merkel,[3] and Ulrich S. Schwarz[1,2]

[1]*Institute for Theoretical Physics, Heidelberg University, Heidelberg, Germany*

[2]*BioQuant, Heidelberg University, Heidelberg, Germany*

[3]*Institute of Complex Systems 7: Biomechanics, Forschungszentrum Jülich GmbH, 52425 Jülich, Germany*

FEM-IMPLEMENTATION

For a given traction pattern $\mathbf{t}(\mathbf{x})$, the direct boundary value problem (BVP) formulated in Eq. 1 is now solved by means of the finite element method (FEM) as typically applied to elastic problems [1–3]. The first step in such a calculation is the transformation of the given equations into the weak form. Therefore we multiply the equation with an arbitrary field $\delta\mathbf{u}$ and integrate over the substrate volume Ω:

$$\int_{\Omega} (\nabla\delta\mathbf{u})^T : \mathrm{C} : \nabla\mathbf{u} \; dV = \int_{\partial\Omega} (\delta\mathbf{u})^T \mathbf{t} \; dS. \tag{1}$$

C represents the constant elasticity matrix for isotropic elastic materials, $\mathrm{C}_{ijkl} = \lambda\delta_{ij}\delta_{kl} + \mu(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk})$. In the following we impose a discretization scheme, by partitioning the integral over Ω into smaller elements corresponding the generated substrate mesh. We further apply a local interpolation scheme. This has the objective to reduce the infinite dimensional space of displacement solutions to a finite dimensional subspace of nodal displacement values. The weak form for a single element reads:

$$\int_{\Omega^e} (\nabla\delta\mathbf{u}^e)^T : \mathrm{C} : \nabla\mathbf{u}^e \; dV = \int_{\partial\Omega^e} (\delta\mathbf{u}^e)^T \mathbf{t} \; dS. \tag{2}$$

In our calculations, we consider hexahedral elements with eight nodal points at the element boundary. As an advantage of hexahedral elements, the volume and surface integration can be mapped to an integration over the unit cube parametrized by Cartesian coordinates, $(x_1, x_2, x_3) \rightarrow (\xi_1, \xi_2, \xi_3)$, while the coordinate transformation is determined by an individual Jacoby matrix calculated for each element. Thus the integration reads:

$$\int_{\tilde{\Omega}^e} (\nabla_\xi\delta\tilde{\mathbf{u}}^e)^T : \mathrm{C} : \nabla_\xi\tilde{\mathbf{u}}^e J_V \; d\tilde{V} = \int_{\partial\tilde{\Omega}^e} \delta\tilde{\mathbf{u}}^e \mathbf{t} J_S \; d\tilde{S}. \tag{3}$$

J represents the Jacobian and the tilde marks quantities with respect to the new coordinate $\boldsymbol{\xi}$. For subsequent interpolation of the displacement field we use interpolation functions $\Phi_n(\boldsymbol{\xi})$, which are based on an elementary set of linear shape functions. By interpolation of nodal values we can rewrite the displacement field, which now only dependents on a discrete number of degrees of freedom (DoFs) (in the hexahedral case used here, this is the number

of nodes multiplied by the dimension, $8 * 3 = 32$):

$$\tilde{\mathbf{u}}^e(\boldsymbol{\xi}) = N(\boldsymbol{\xi})\mathbf{u}^e_{node} = \begin{pmatrix} \Phi_1 & 0 & 0 & \dots & \Phi_n & 0 & 0 \\ 0 & \Phi_1 & 0 & \dots & 0 & \Phi_n & 0 \\ 0 & 0 & \Phi_1 & \dots & 0 & 0 & \Phi_n \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_n \end{pmatrix} \tag{4}$$

$\mathbf{u}^e_{node}$ is a vector of nodal displacement DoF values and $N(\boldsymbol{\xi})$ is the interpolation matrix, which interpolates the element displacement $\tilde{\mathbf{u}}^e(\boldsymbol{\xi})$ for a given configuration of nodal values $\mathbf{u}^e_{node}$. The same can be achieved for the virtual displacement $\delta\tilde{\mathbf{u}}^e(\boldsymbol{\xi}) = N(\boldsymbol{\xi})\delta\mathbf{u}^e_{node}$. Applying this to Eq. 3, we obtain

$$\delta\mathbf{u}^e_{node} \underbrace{\left[\int_\Omega (\nabla_\xi N(\boldsymbol{\xi}))^T : \mathrm{C} : \nabla_\xi N(\boldsymbol{\xi}) J_V \; d\tilde{V}\right]}_{K^e} \mathbf{u}^e_{node} = \delta\mathbf{u}^e_{node} \underbrace{\left[\int_{\partial\Omega^e} N(\boldsymbol{\xi})^T \mathbf{t} J_S \; dS\right]}_{\mathbf{R}^e} \tag{5}$$

K^e is called the element stiffness matrix and $\mathbf{R}^e$ the element load vector. Since $\delta\mathbf{u}^e_{node}$ is a vector of arbitrary values, we can reduce the problem to solving a linear algebraic system:

$$K^e\mathbf{u}^e_{node} = \mathbf{R}^e. \tag{6}$$

The components of the element stiffness matrix and the load vector are calculated numerically. Here, the integrals have been solved by means of Gauss quadrature. In this way we calculate the stiffness matrix and load vector for each element. In a subsequent step, we assemble a global system that forms the domain Ω:

$$K\mathbf{U}_{node} = \mathbf{R}. \tag{7}$$

K, $\mathbf{R}$, and $\mathbf{U}_{node}$ are the global stiffness matrix, the global load vector, and the global DoF vector. Nodal DoFs are only shared by neighboring elements and thus K is a sparse matrix, while most of the matrix entries are zero. K is further singular and hence not invertible, since we have still not introduced constraints to avoid rigid body motions. Therefore, the system must be further restrained by incorporating appropriate boundary conditions (BCs). For our direct BVP we consider the BCs illustrated in Fig. 2. The traction BC enters the system through the surface integral in the load vector $\mathbf{R}$. The zero stress BC leads to no constraints of the system. Only the remaining zero displacement BC applied to the bottom

surfaces constrains Eq. 7 and by forcing the displacement condition, we can reduce the system to

$$K_f \mathbf{U_{f,node}} = \mathbf{R_f} \tag{8}$$

where K_f is the reduced non-singular stiffness matrix, $\mathbf{U}_{f,node}$ is the global vector of unconstrained (free) DoFs, and $\mathbf{R_f}$ is the corresponding load vector. We solve the system with respect to $\mathbf{U}_{f,node}$ by using the conjugated gradient (CG) method. Alternatively, it is also possible to directly invert K by means of e.g. Gauss-elimination. In order to evaluate a displacement solutions at every position within the domain Ω, we apply interpolation via the introduced functions Φ_n with respect to obtained global DoF configuration $\mathbf{U}$.

---

[1] Dietrich Braess. *Finite Elemente Theorie: schnelle Löser und Anwendungen in der Elastizitätstheorie*. Springer Verlag, Berlin, Germany, 4th ed. edition, 2007.

[2] Mohammed Ameen. *Computational Elasticity*. Alpha Science Int. Ltd., Harrow, U.K., 2005.

[3] Javier Bonet and Richard D Wood. *Nonlinear Continuum Mechanics for Finite Element Analysis*. Cambridge University Press, 1997.