# AxFEM: Micro-Gyro Simulation and Modeling

David Bindel

25 Jul 2012

# Why not Comsol, Ansys, Coventorware, ...?

Pros for standard FEA packages:

*Finite elements are like ants. They're weak on their own, but you sure can get a lot of them.*

But:

- Figures of interest are differences
  - If $|\omega_1 - \omega_2|/|\omega_1| < 10^{-p}$, lose $p$ digits in computing $\omega_1 - \omega_2$
  - Now consider MRIG tolerances
- Two keys to speed and accuracy:
  - Express differences directly
  - Preserve structure (exact or approximate)
- Careful numerics is an enabling technology
  - Optimization, reliability analysis, model fitting, ...

# Phase I: Basic technology

- Simulates axisymmetric micro-gyro mechanics
  - Modal analysis, angular gains, loss mechanisms
- Efficient computational formulations (using structure)
  - 2.5-dimensional finite element formulation
  - Fast solvers for Bryan's factor, loss mechanisms
  - Build in symbolics and incremental computation support
- Supports optimization and fitting to experiment
  - Parameterized device descriptions
  - Fast shape and material sensitivity analysis
- Tested and validated against data in the literature

Basically done (deal2lab/AxFEM) – release in September.

# Phase II: Physics

Still missing some things:

- Non-axisymmetric effects
  - Model variations from axisymmetry by Fourier expansion
  - Maintain speed of 2.5D simulations!
- Loss mechanism details
  - How much do substrate approximations matter?
  - How much is surface loss?
- Models of coupling via actuation / sensing
  - Mechanical couplings can contribute to damping
  - These can break symmetry, too!

# Phase II: Model Fitting

- Results are only as good as inputs
  - Garbage in, garbage out!
  - Testing and code validation alone don't help
- Goal: Automatically reconcile model with measurement
  - How much due to discretization? Minimize this!
  - How much could be variations in fabrication?
  - How much could be simplified physics?
- Approach:
  - Linear sensitivity for fitting optimization
  - Stochastic analysis for very uncertain parameters
  - Bound worst case for "small-but-unknown" effects
  - All methods use fast simulator as a building block!

# Phase II: Robust Optimization

- Goal: Optimize performance in an imperfect world
  - Imperfect fab $\Longrightarrow$ optimize for good yield
  - Imperfect models $\Longrightarrow$ minimize “distance to reality”
- Approach:
  - Local gradient-based optimization
  - Penalties based on sensitivity, measures of model quality
  - Response-surface-based global optimization if time permits
- Again, fast simulation is critical!

# Projected Timeline

- Sep: Initial code release (and public repository)
- Next three months
  - Non-axisymmetric effects (in progress now)
  - Initial optimization/fitting demos
- Next six months
  - Stochastic sensitivity analysis code
  - Basic (empirical) surface loss models
- Next nine months
  - Connection to process simulation
  - Bounds on substrate approximation (in progress now)

# Fourier expansion of geometric imperfections

Model real domain as distortion of ideal

$$\psi : \Omega_{\text{ideal}} \to \Omega_{\text{actual}}$$

Expand

$$\psi(r, z, \theta) = (r, z, \theta) + \alpha_0(r, z) + \sum_{m=1}^{\infty} \left( \alpha_m(r, z) \cos(m\theta) + \beta_m(r, z) \sin(m\theta) \right)$$

Mask misalignment corresponds to mostly $m = 1$ distortion

# Fourier picture

Imperfections perturb from ideal decoupling in Fourier space:

$$K^{\text{real}} = \begin{bmatrix} K_{00}^{\text{ideal}} + \delta K_{00} & \delta K_{01} & \delta K_{02} & \cdots \\ \delta K_{01}^{T} & K_{11}^{\text{ideal}} + \delta K_{11} & \delta K_{12} & \cdots \\ \delta K_{01}^{T} & \delta K_{12}^{T} & K_{22}^{\text{ideal}} + \delta K_{22} & \cdots \\ \vdots & \vdots & \ddots & \end{bmatrix}$$

$\delta K_{ij} \propto$ the $m = |j - i|$ terms in distortion function.

# Fourier-based solver

Two approaches. Both reduce to 2D solves.

- Series expansion (Schrödinger-Rayleigh)
  - First order only incorporates $m = 0$ distortion term
  - Second-order term involves solve with block-diagonal ideal
- Direct solve with structured acceleration
  - Jacobi-Davidson for subspace construction
  - Precondition with block-diagonal ideal

# deal2lab

- Uses open-source `deal.ii` FE framework.
- Programmatic meshes for geometry parameterization:

```
function hrglip(r_hemisphere,thickness,
                r_lip,
                r_anchor,
                h_anchor)
```

- Fast solvers for angular gain, loss mechanisms.
- Fast sensitivity with respect to parameters.

# Kinematic assumptions

Use 2.5D formulation for basic modal computations:

$$\mathbf{u}_1(r,z) = \begin{bmatrix} u_r(r,z)\cos(m\theta) \\ u_\theta(r,z)\sin(m\theta) \\ u_z(r,z)\cos(m\theta) \end{bmatrix}, \quad \mathbf{u}_2(r,z) = \begin{bmatrix} -u_r(r,z)\sin(m\theta) \\ u_\theta(r,z)\cos(m\theta) \\ -u_z(r,z)\sin(m\theta) \end{bmatrix}.$$

- Only mesh cross-section, 3 DOF per node.
- 2D connectivity $\implies$ fast direct solvers.
- Geometric degeneracy preserved by the discretization.

# Computational pattern

- Solve for mode with no damping or rotation:

$$(-\omega_0^2 M_{uu} + K_{uu})u_0 = 0.$$

- First-order perturbation theory for damping and rotation.

# Thermoelastic damping

Compute mechanical mode + induced temperature fluctuation:

$$
\begin{aligned}
(-\omega_0^2 M_{uu} + K_{uu}) u_0 &= 0 \\
(i\omega_0 C_{\theta\theta} + K_{\theta\theta})\theta_0 &= -i\omega_0 C_{\theta u} u_0.
\end{aligned}
$$

First-order correction to eigenvalue (generalized Zener):

$$
\delta(\omega^2) = -\frac{u_0^T K_{u\theta}\theta_0}{u_0^T M_{uu} u_0}.
$$

# Anchor loss

Incorporating numerical radiation BCs gives:

$$\left(-\omega^2 M_{uu} + K_{uu} + G(\omega)\right) u = 0$$

where $G(\omega)$ approximates a DtN map (e.g. via PML).

Perturbation approach: ignore $G$ to get $(\omega_0, u_0)$. Then

$$\delta(\omega^2) = \frac{u_0^T G(\omega_0) u_0}{u_0^T M_{uu} u_0}.$$

# Bryan's factor

Angular gain for a given mode is

$$\mathrm{BF} = \frac{\text{Angular rate of pattern relative to body}}{\text{Angular rate of vibrating body}} = \frac{1}{m}\left(\frac{u^T B u}{u^T M u}\right),$$

where $M$ is the standard FE mass matrix and $B$ is

$$B_{IJ} = \int_\Omega N_I(r,\theta) N_J(r,\theta) \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

# Geometric sensitivities

Basic strategy is standard[1]:

- Differentiate node positions w.r.t. geometric parameters
- Differentiate FE matrices w.r.t. node positions
- Differentiate $\omega$, $Q$, BF w.r.t. FE matrices
- Apply chain rule

[1] Haslinger and Mäkinen. 2003. *Introduction to shape optimization theory, approximation, and computation*.

# Testing strategy

- Unit tests for basic functionality (run automatically on build)
- Convergence tests
- Validation tests compare against results in the literature
- Finite difference checks for sensitivity computations

# Validation testing

1. F. I. Niordson, *Free Vibrations of Thin Elastic Spherical Shells*, International Journal of Solids and Structures, 20 (7), 1984, pp. 667–687.
2. J.J.Hwang C.S.Chou C.O.Chang, *Precession of Vibrational Modes of a Rotating Hemispherical Shell*, Transactions of the ASME, 119, 1997
3. S. Y. Choi, Y. H. Na, and J. H. Kim *Thermoelastic Damping of Inextensional Hemispherical Shell*, World Academy of Science, Engineering and Technology, 56, 2009.
4. S. J. Wong, C.H. Fox, S. McWilliam, C.P. Fell, R. Eley *A preliminary investigation of thermo-elastic damping in silicon rings*. J. Micromech. Microeng. 14, 2004, S108–S113

# Summary

Initial code is working:

- Fast computation of Bryan's factor, $Q_{TED}$
- Anchor loss computations work separately
- Sensitivity analysis works
- Includes unit tests and validation test suite

Some things still needed:

- Full documentation
- Removal of some known performance bottlenecks
- Integration of anchor loss code into deal2lab
- Framework for surface loss modeling