Analytical bead force model for the 3DFM

Leandra Vicci
Microelectronic Systems Laboratory
Department of Computer Science

ABSTRACT

The magnetic force experienced by a soft magnetic bead is proportional to the strength of the field at the location of the bead. It is also proportional to the gradient of the field at that location. In general the analytical calculation of this force is intractable. However if the generation of the field can be abstractly modeled by a small number of magnetic monopoles, an analytical solution is straightforward. This model is arguably a reasonable approximation for some magnetic pole tip configurations of a three dimensional force microscope (3DFM) [Cumm]. This research note presents a derivation of the analytical solution.

1 The monopole model of the 3DFM

To be sure, there are pole geometries for which the monopole approximation is inappropriate. However the symmetric tetrahedral and hexapolar geometries are of particular interest for this instrument, and the monopole approximation is entirely reasonable in these cases. I will not provide any proof or error estimates here, but will give intuitive reasons why the approximation is "pretty good" for the tetrahedral and hexapole geometries of existing designs. First let us look at the tetrahedral design.

Here we have the magnetic bead located in the center of a regular tetrahedron. Four cylindrical magnetic iron cores converge towards it from the centers of the four faces of the tetrahedron (Figure 1). The inner ends of the cores are tapered to a fine point with spherical ends, forming four pole tips. The poles are excited such that the sum of the fluxes leaving their tips vanishes (because magnetic monopoles cannot actually exist). The high permeability of the iron provides that the magnetic potential is essentially constant over the extent of each pole tip. Therefore, in a substantial solid angle in the direction of the bead, a spherical magnetomotive isopotential surface is established identical to that of a monopole located at the center of curvature of the pole tip.

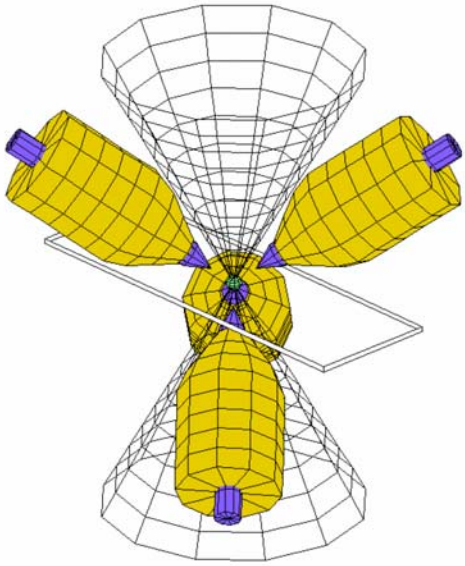
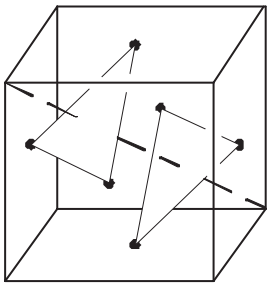


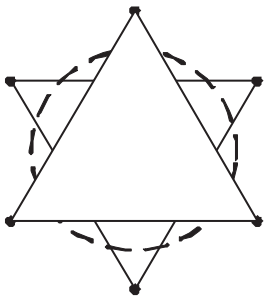
Figure 1

In the face centered cubic (FCC) hexapole geometry (Figure 2), the physical correspondence is not as neat because the pole tips are not themselves hemispherical. In this case we have six pole tips located at the centers of the six faces of a cube. The bead is located in the volumetric center of the cube. Magnetic flux is conducted to the tips through cores comprising tapered foil or thin magnetic films arranged in two closely spaced parallel planes. The tapers terminate in circular pole tips which present cylindrical profiles in the direction of the bead. Thus, the isopotential shape is cylindrical rather than spherical at a pole tip surface. However field simulations (Figure 3) show that in a solid angle in the direction of the bead, the isopotential shape quickly becomes spherical as one approaches the bead. The center of the quasi spherical isopotential surface patch touching the bead is approximately at the center of curvature of the pole tip. Therefore, a magnetic monopole at

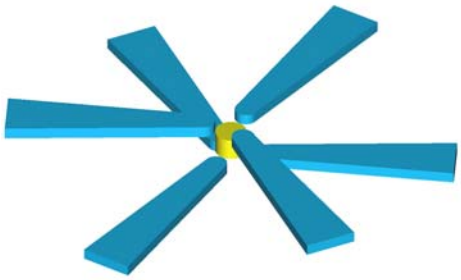
this location provides a reasonable approximation of the field from this pole at the bead location. Again, the pole excitations must provide that the sum of the pole fluxes vanishes.



Optical axis (dashed line) is perpendicular to two planes, each containing three face-centered-cubic (FCC) points.



These points form a pair of parallel equilateral triangles with a cylindrical working volume between them.



Magnetic flux is conducted by thin film cores in two parallel planes to the pole tips having centers at the FCC locations

Figure 2: The face centered cubic hexapole geometry

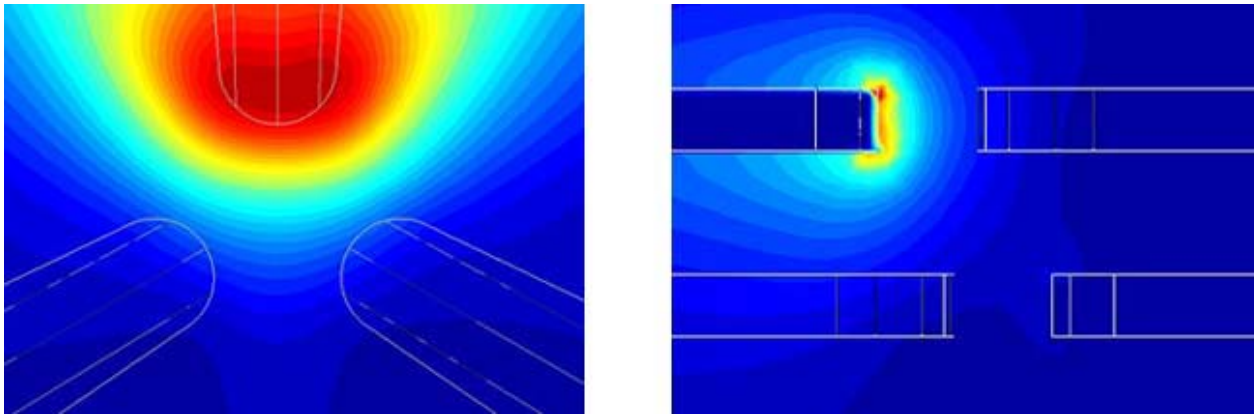


Figure 3: Field simulation results of the FCC geometry

To the extent that these monopole approximations are sufficiently accurate, the problem remains to calculate the field and its gradient at the bead location from an n-monopole model.

2 Notation

We use both 3-space vectors and matrices here, including a $1 \times n$ "vector" of pole excitations (strengths). To minimize confusion, it is useful to employ a notation which clearly distinguishes between spatial and matrix vectors. Accordingly, let us represent spatial vectors in italic bold face, e.g., "$\boldsymbol{B}$", and *all* matrices in roman type, e.g. "q", while scalars will be in the conventional italic type face, e.g., "r". A matrix of spatial vectors will be represented by bold roman typeface, e.g. "$\mathbf{f}$". Finally, unit vectors will be represented by the letter "$\boldsymbol{u}$".

3 Magnetic field at the origin

The bead is located at the origin, so we calculate the field there generated by n monopoles of excitation strength q_j at locations $\boldsymbol{r}_j$, $j = 1 \ldots n$.

$$\boldsymbol{B} = \sum_{j=1}^{n} \frac{q_j \boldsymbol{u}_j}{r_j^2}, \tag{1}$$

where r_j is the distance to the j^{th} monopole, and $\boldsymbol{u}_j$ is the unit vector towards that monopole.

4 Magnetic force on the bead

To within a constant depending on bead parameters, the force on the bead is [Vicc],

$$\begin{aligned}
\boldsymbol{F} &= \nabla(\boldsymbol{B} \cdot \boldsymbol{B}) \\
&= \nabla \left(\sum_{j=1}^{n} \frac{q_j \boldsymbol{u}_j}{r_j^2} \cdot \sum_{k=1}^{n} \frac{q_k \boldsymbol{u}_k}{r_j^2} \right) \\
&= \nabla \left(\sum_{j,k} \frac{q_j q_k \boldsymbol{u}_j \cdot \boldsymbol{u}_k}{r_j^2 r_k^2} \right) \\
&= \sum_{j,k} q_j q_k \boldsymbol{u}_j \cdot \boldsymbol{u}_k \nabla(r_j^{-2} r_k^{-2})
\end{aligned} \tag{2}$$

The gradient is calculated as follows:

$$\begin{aligned}
\nabla(r_j^{-2} r_k^{-2}) &= r_j^{-2} \nabla r_k^{-2} + r_k^{-2} \nabla r_j^{-2} = r_j^{-2}(-2r_k^{-3})\nabla r_k + r_k^{-2}(-2r_j^{-3})\nabla r_j \\
&= \frac{-2}{r_j^2 r_k^2} \left(\frac{\nabla r_j}{r_j} + \frac{\nabla r_k}{r_k} \right) = \frac{-2}{r_j^2 r_k^2} \left(\frac{\boldsymbol{u}_j}{r_j} + \frac{\boldsymbol{u}_k}{r_k} \right).
\end{aligned} \tag{3}$$

For convenience above, I used vectors pointing from the field point towards the n source points. Conventional field formulas use vectors pointing from the source point to the field point (cf. [Jack]). Therefore, the directions of the unit vectors in equation(3) should be reversed which we accomplish by a change of sign. Substituting equation(3) into equation(2) with this change of sign gives,

$$\boldsymbol{F} = \sum_{j,k} \frac{2 q_j q_k \boldsymbol{u}_j \cdot \boldsymbol{u}_k}{r_j^2 r_k^2} \left(\frac{\boldsymbol{u}_j}{r_j} + \frac{\boldsymbol{u}_k}{r_k} \right). \tag{4}$$

It is useful to separate the summand of equation(4) into a purely geometrical factor $\boldsymbol{f}_{j,k}$ and a purely excitational factor $(q_j q_k)$. Thus for one term of the summand we write,

$$\boldsymbol{F}_{j,k} = q_j \, \boldsymbol{f}_{j,k} \, q_k, \quad \text{where} \tag{5}$$

$$\boldsymbol{f}_{j,k} = \frac{2 \boldsymbol{u}_j \cdot \boldsymbol{u}_k}{r_j^2 r_k^2} \left(\frac{\boldsymbol{u}_j}{r_j} + \frac{\boldsymbol{u}_k}{r_k} \right). \tag{6}$$

5 Matrix formulation

Let us now represent the geometry dependent factors $\boldsymbol{f}_{j,k}$ as an $n \times n$ matrix $\mathbf{f}$ of spatial vectors. Also let the n pole excitations be represented by the $1 \times n$ matrix q. Then we may rewrite equation (4) in a matrix form,,

$$\boldsymbol{F} = \mathrm{q}\ \mathbf{f}\ \tilde{\mathrm{q}}, \tag{7}$$

where $\tilde{\mathrm{q}}$ is the transpose of q. It is to be understood that the element-wise operations of equation(7) are taken as scalar-vector-scalar products.

6 Examples

Consider a tetrahedral geometry with pole tips located at $[\pm x, 0, -z]$ and $[0, \pm y, z]$. If $x = y = \sqrt{2/3}$ and $z = \sqrt{1/3}$, this will be a regular tetrahedron with unit distance from each pole to the bead. In this case we have,

$$\mathbf{f} = \begin{pmatrix} [3.27, 0.00, -2.31] & [0.00, 0.00, 0.77] & [-0.54, -0.54, 0.00] & [-0.54, 0.54, 0.00] \\ [0.00, 0.00, 0.77] & [-3.27, 0.00, -2.31] & [0.54, -0.54, 0.00] & [0.54, 0.54, 0.00] \\ [-0.54, -0.54, 0.00] & [0.54, -0.54, 0.00] & [0.00, 3.27, 2.31] & [0.00, 0.00, -0.77] \\ [-0.54, 0.54, 0.00] & [0.54, 0.54, 0.00] & [0.00, 0.00, -0.77] & [0.00, -3.27, 2.31] \end{pmatrix},$$

where each matrix element is a spatial vector of the form $[x, y, z]$.

In the FCC hexapole geometry, with xyz-axis-aligned pole tip locations and all pole tips being the same distance r from the origin, many of the off-diagonal dot products vanish, and the remaining off diagonal vector terms also cancel, leaving a particularly tidy

$$\mathbf{f} = \frac{1}{r^5} \begin{pmatrix} [-4, 0, 0] & [0, 0, 0] & [0, 0, 0] & [0, 0, 0] & [0, 0, 0] & [0, 0, 0] \\ [0, 0, 0] & [4, 0, 0] & [0, 0, 0] & [0, 0, 0] & [0, 0, 0] & [0, 0, 0] \\ [0, 0, 0] & [0, 0, 0] & [0, -4, 0] & [0, 0, 0] & [0, 0, 0] & [0, 0, 0] \\ [0, 0, 0] & [0, 0, 0] & [0, 0, 0] & [0, 4, 0] & [0, 0, 0] & [0, 0, 0] \\ [0, 0, 0] & [0, 0, 0] & [0, 0, 0] & [0, 0, 0] & [0, 0, -4] & [0, 0, 0] \\ [0, 0, 0] & [0, 0, 0] & [0, 0, 0] & [0, 0, 0] & [0, 0, 0] & [0, 0, 4] \end{pmatrix}.$$

REFERENCES

[Cumm] Cummings, J., *Development and Applications of Magneto-optical Scanning Probe Microscopy*, UNCCH Department of Biomedical Engineering Ph.D. Thesis, 2001.

[Jack] Jackson, J.D., *Classical Electrodynamics*, Wiley, 1962.

[Vicc] Vicci, L., *A 3D Magnetic Force Manipulator DC Prototype*, UNCCH Department of Computer Science Technical Report TR01-031, 2001.

Appendix A: Matlab codes

This appendix contains Matlab codes for generating the **f** matrix and for using it to calculate bead forces. Also included are two specific geometry generators for the regular tetraheral and FCC geometries.

A.1 tetra(r)

This function returns the face center positions of a regular tetrahedron oriented in the manner of the 3DFM DC prototype [Vicc] with distances r from the origin, which is located at the geometric center of the tetrahedron.

```
function p = tetra(r)
% tetra(r) -- Returns pole locations of xy-axis aligned tetrahedral pole
% geometry with pole distances r from the field point at the origin.

rt13 = sqrt(1/3);
rt23 = sqrt(2/3);
p = r.*[rt23,0,-rt13; -rt23,0,-rt13; 0,rt23,rt13; 0,-rt23,rt13];
```

A.2 hexa(r)

This function returns the FCC positions of an axis-aligned cubic hexapole geometry with tip distances r from the origin, which is located in the geometric center of the cube.

```
function p = hexa(r)
% hexa(r) -- returns pole locations for axis aligned cubic geometry
% with pole distance r from the field point at the origin.

p = r.*[-1,0,0;1,0,0;0,-1,0;0,1,0;0,0,-1;0,0,1];
```

A.3 fmatrix(p, d)

This function generates the **f** matrix. It is quite general in that it accepts a list p of spatial vectors representing the arbitrary positions of an arbitrary number of monopoles.

```
function f = fmatrix(p, d)
% f = fmatrix(p) -- returns the f matrix for a list
% of n pole position vectors p, one vector per row.
% If d is given, a TeX matrix-formatted string with
% d digits of fractional precision is returned instead
% of the matrix itself.    The symmetric f matrix is
% structured as an (n x 3 x n) array where the middle
% index refers to the spatial axes, [x,y,z].
```

```
size(p);
n = ans(1);
for i = 1:n
    r(i) = norm(p(i,:));
    u(i,:)  = p(i,:)/r(i);
end

for j = 1:n
    for k = 1:n
        2*u(j,:)*u(k,:)'/(r(j)*r(k))^2;
        f(j,:,k) = ans*(u(j,:)/r(j) + u(k,:)/r(k));
    end
end

if nargin == 2
    % construct the TeX formatting strings
    fmt1 = sprintf('%%s[%%.%1df,%%.%1df,%%.%1df]&', d, d, d);
    fmt2 = sprintf('%%s[%%.%1df,%%.%1df,%%.%1df]\\\\cr\\n  ', d, d, d);
    s = sprintf ('%s\n  ', '\left(\matrix{');
    % fill in the matrix elements
    for j = 1:n
        for k = 1:n
            if k < n
                s = sprintf(fmt1, s, f(j,1,k), f(j,2,k), f(j,3,k));
            else
                s = sprintf(fmt2, s, f(j,1,k), f(j,2,k), f(j,3,k));
            end
        end
    end
    s = sprintf('%s}\\right)\n', s);
    f = s;
end

return
% Examples:
fmatrix(tetra(1))       % unit tetrahedron
fmatrix(tetra(1),2)     % ...  in TeX format
fmatrix(hexa(1))        % unit FCC
fmatrix(hexa(1),0)      % ...  in TeX format
fmatrix(hexa(1.3195))   % 1.3195^5 = 4 => unit diagonals
```

A.4 Bforce(f, Q)

Given an **f** matrix and a list **Q** of excitations, this function returns a list of forces calculated for a "unit" bead, i.e., one in which the bead force coefficient [Vicc]

$$\frac{\pi d^3}{2\mu_0}\left(\frac{\mu_r - 1}{\mu_r + 2}\right) = 1,$$

where d is the bead diameter, $\mu_r = \mu/\mu_0$ is its relative permeability, and $\mu_0 = 4\pi \times 10^{-7}$ [H/m] is the permeability of free space.

```
function F = Bforce(f, Q)
% F = Bforce(f, Q) -- Returns a list F of force vectors from a
% list Q of excitations for an n-pole geometry matrix f, which
% is structured as an (n x 3 x n) array where the middle index
% refers to the spatial axes, [x,y,z].

size(f);
n = ans(1); % the number of poles specified in f
size(Q);
if ans(2) ~= n
    error('number of poles:  incompatibility between f and Q.');
end
m = ans(1); % the number of listed excitations

for i = 1:m % process the excitation list
    q = Q(i,:);
    for h = 1:3, F(i,h) = q*squeeze(f(:,h,:))*q'; end
end

return
% Example
Bforce(fmatrix(tetra(1)), [[-1,1/3,1/3,1/3] % towards one pole
        [-1,  1,  0,  0]                    % halfway between two poles
        [-1,1/2,1/2,  0]])                  % three poles excited
```