

磁気異方性媒質中の磁場と磁化の方向及び磁気抵抗の角度依存性

2014.12.6 鈴木 実

磁化容易方向など

磁気モーメントあるいは磁化を持つ物質では磁化の方向により磁性エネルギーが異なる。つまり、異方的な磁気エネルギーを有する。このような異方的磁気エネルギーが生じる理由として、形状によるもの、および結晶構造によるものがある。

形状による場合として、例えば細長い棒では、磁気モーメントが、他の部分の磁気モーメントの作る磁場に平行であるほうがエネルギーが低いので、棒の長手方向に磁気モーメントは揃いやすい。

結晶構造に依存する場合は、電子の結合状態が重要な役割を担う。結晶構造によって価電子の軌道が決まるので、軌道角運動量の方向と大きさが決まる。その時、この軌道角運動量とスピンはスピン軌道相互作用によるエネルギーが最も小さくなる方向にスピンの揃い易くなる。したがって、磁気モーメントは結晶構造のある一定の方向を向きやすくなる。

磁化が最も向き易い方向を磁化容易方向 (an easy direction of magnetization), 最も向き易い面内となる面を磁化容易面 (an easy plane of magnetization), 磁化が最も向きにくい方向を磁化困難方向 (a hard direction of magnetization), 最も向きにくい面内となる面を磁化困難面 (a hard plane of magnetization) という。

異方性エネルギー

磁化が存在することによる異方性エネルギー (anisotropy energy) は、磁化と磁化容易軸との間の角度 ϕ に依存する。磁化容易方向はしばしば結晶の基本並進ベクトルに一致し、一般に複数存在する。そのため、そういう場合は磁化容易軸と呼ばれる。 ϕ_i を i 番目の磁化容易軸との角度とすると、異方性エネルギー E_a は一般に

$$E_a(\phi_i) = K_0 + K_1 F_1(\phi_1) + K_2 F_2(\phi_2) + \dots \quad (1)$$

と表される。

具体的に、最も簡単な1軸磁気異方性 (uniaxial magnetic anisotropy) の場合は、

$$E_a = K_U \sin^2 \phi \quad (2)$$

と表される。 $K_U > 0$ の場合、 $\phi = 0$ の方向が磁化容易軸、それと直交する平面が磁化困難面である。逆に $K_U < 0$ の場合、 $\phi = 0$ の方向が磁化困難軸、それと直交する平面が磁化容易面である。

磁場が存在する場合、磁場と磁化の相対角度 φ に依存する静磁エネルギー (magnetostatic energy) あるいはゼーマンエネルギー (Zeeman energy) E_Z が生じる。したがって、全エネルギー E は

$$E = E_a + E_Z = -K \sin^2(\theta + \varphi) - \mu_0 M H \cos(\alpha - \theta) \quad (3)$$

となる。ここで、 θ は磁化困難軸 (z 軸) と磁場 \mathbf{H} の間の角、 α は磁化困難軸 (z 軸) と磁化 \mathbf{M} の間の角、 φ は \mathbf{H} と \mathbf{M} の間の角である。すなわち、

$$\alpha = \theta + \varphi \quad (4)$$

である。

磁場と磁化の間の角度

磁気異方性が存在する場合、磁場 \mathbf{H} に対して磁化 \mathbf{M} の方向は傾く。 \mathbf{H} と \mathbf{M} の間の角 φ は式 (3) のエネルギーを最小にする値である。したがって、式 (3) を微分することにより、

$$\frac{\partial E}{\partial \varphi} = -2K \sin(\theta + \varphi) \cos(\theta + \varphi) + \mu_0 MH \sin \varphi = 0 \quad (5)$$

が成り立つので、これより、

$$\sin \varphi = \frac{K}{\mu_0 MH} \sin 2(\theta + \varphi) \quad (6)$$

が得られる。(6) は $\varphi \ll 1$ の場合に文献 [1] の式 (5) の上の式に一致する。(ただし、文献 [1] ではミスプリントがあり係数 2 の位置が異なる。)

これから、 φ を得るには、

$$\sin \varphi_n = \frac{K}{\mu_0 MH} \sin 2(\theta + \varphi_{n-1}) \quad (7)$$

として、 $\varphi_0 = 0$ から始める逐次法により近似解を得ることができる。あるいは、Newton 法により数値的に解を得ることができる (c のプログラム例を添付)。 $K/\mu_0 MH$ の値を 0.1 から 2.0 まで変えた時の θ と φ の関係を図 1 に示す。

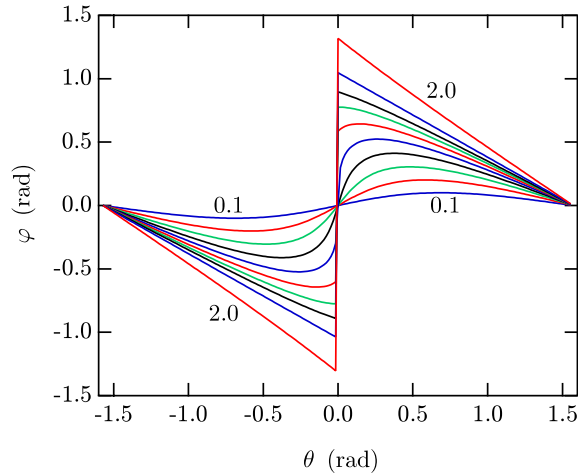


図 1: θ と φ の関係。 $K/\mu_0 MH = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 2.0$ の場合。

強磁性体の磁気抵抗 $\Delta\rho$ は $M_{\perp} = M \cos 2\alpha$ に比例する。磁気異方性がある場合には、磁気抵抗の角度依存性は $\gamma = K/\mu_0 MH$ の大きさに依存して、 $\cos 2\alpha$ からずれてくる。図 2 に 2 つの例を示す。

文献

[1] N. Naftalis, N. Haham, J. Hoffman, M. S. J. Marshall, C. H. Ahn, and L. Klein, Phys. Rev. B **86**, 184402 (2012).

プログラム例

以下の C プログラムで図 1 と図 2 のグラフを計算することができる。 `phi-theta-2.c` は、式 (5) を Newton 法で解く。根を $-\pi/2 < \varphi < \pi/2$ に求めるために、 $g(\varphi) = -a \sin 2(\theta + \varphi) + \sin \varphi$ と $d^2g/d\varphi^2 = 4a \sin 2(\theta + \varphi) - \sin \varphi$

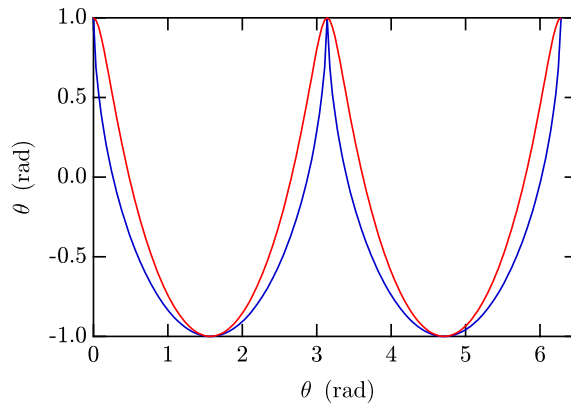


図 2: red は $K/\mu_0MH = 0.3$ の場合, green は $K/\mu_0MH = 0.5$ 場合である.

の関係調べて Newton 法の初期値を選択するルーチンをとっている.

`aniso-magres.c` は, 磁気抵抗の角度依存性を示す項が $\cos(2\varphi)$ に比例するので, その部分を計算するプログラムである.

```

/* phi-theta-2.c */
/* M. Suzuki 2014.12.2, 2019.4.1 */
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include <string.h>

double g(double x, double a, double b)          /* x - phi, a - K/MH, b - theta */
{
    double z;
    z=a*sin(2*(b+x))-sin(x);
    return z;
}

double g2(double x, double a, double b)         /* 2d derivative of g; x - phi, a - K/MH, b - theta */
{
    double z;
    z=-4*a*sin(2*(b+x))+sin(x);
    return z;
}

double aniso_magn(double a, double b)          /* a - K/MH, b - theta */
{
    int i, j, n;
    double dx, f1, f, x, x1, y, z;
    double alpha, beta, pi;

    pi=M_PI;
    n=50;
    dx=pi/4/n;
    while(b>pi/2) b-=pi;
    while(b<-pi/2) b+=pi;
    if(b<0)
    {
        x=0;
        while(g(x, a, b)<0)

```

```

        {
            x-=dx;
        }
        alpha=x;
        while(g2(x, a, b)>0)
        {
            x-=dx;
        }
        beta=x;
    }
    else
    {
        x=0;
        while(g(x, a, b)>0)
        {
            x+=dx;
        }
        alpha=x;
        while(g2(x, a, b)<0)
        {
            x+=dx;
        }
        beta=x;
    }
    x=(alpha+beta)/2;
    z=1.0;
    j=0;
    while(fabs(z)>0.001 && j<10)
    {
        f=a*sin(2*(x+b))-sin(x);
        f1=2*a*cos(2*(x+b))-cos(x);
        x1=x-f/f1;
        if(f1==0) {x=0; break;}
        z=x1-x;
        x=x1;
        j++;
    }
    return x;
}

int main()
{
    FILE *fp;
    char filenameout[12][30]={"phi-theta-0.1.txt", "phi-theta-0.2.txt", "phi-theta-0.3.txt",
                             "phi-theta-0.4.txt", "phi-theta-0.5.txt", "phi-theta-0.6.txt",
                             "phi-theta-0.7.txt", "phi-theta-0.8.txt", "phi-theta-0.9.txt",
                             "phi-theta-1.0.txt", "phi-theta-2.0.txt"};

    int i, j, n;
    double a[12]={0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 2.0};
    double dth, th0, th, pi;
    double z;

    n=200;
    pi=M_PI;
    dth=pi/n;
    th0=-0.5*pi;

    for(j=0;j<11;j++)
    {
        printf("a=%lf\n", a[j]);
        fp=fopen(filenameout[j], "w");
    }
}

```

```

        for(i=0;i<n;i++)
        {
            th=th0+dth*i;
            z=aniso_magn(a[j],th);
            printf("%d\t %lf \t %lf\n", i, th, z);
            fprintf(fp, "%d\t %lf \t %lf\n", i, th, z);
        }
        fclose(fp);
    }
}

/* aniso-magres.c */
/* M.Suzuki 2014/12/6, 2019/4/2 */
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include <string.h>

double g(double x, double a, double b)          /* x - phi, a - g, b - theta */
{
    double z;
    z=a*sin(2*(b+x))-sin(x);
    return z;
}

double g2(double x, double a, double b)
{
    double z;
    z=-4*a*sin(2*(b+x))+sin(x);
    return z;
}

double aniso_magn(double a, double b)          /* x - phi, a - g, b - theta */
{
    int i, j, n;
    double dx, f1, f, x, x1, y, z;
    double alpha, beta, pi;

    pi=M_PI;
    n=50;
    dx=pi/4/n;
    while(b>pi/2) b-=pi;
    while(b<-pi/2) b+=pi;
    if(b<0)
    {
        x=0;
        while(g(x, a, b)<0)
        {
            x-=dx;
        }
        alpha=x;
        while(g2(x, a, b)>0)
        {
            x-=dx;
        }
        beta=x;
    }
}

```

```

else
{
    x=0;
    while(g(x, a, b)>0)          /* x - phi, a - g, b - theta */
    {
        x+=dx;
    }
    alpha=x;
    while(g2(x, a, b)<0)       /* second derivative of g(x) */
    {
        x+=dx;
    }
    beta=x;
}
x=(alpha+beta)/2;
z=1.0;
j=0;
while(fabs(z)>0.001 && j<10)
{
    f=a*sin(2*(x+b))-sin(x);
    f1=2*a*cos(2*(x+b))-cos(x);
    x1=x-f/f1;
    if(f1==0) {x=0; break;}
    z=x1-x;
    x=x1;
    j++;
}
return x;
}

```

```

int main()
{
    FILE *fp;
    char *filenameout[2]={"0.5.txt", "0.3.txt"};
    int i, j, n;
    double a[2]={0.5, 0.3};
    double c, dx, phi, pi, th, x, y, z;

    pi=M_PI;
    n=180;
    dx=2*pi/n;

    for(j=0;j<2;j++)
    {
        fp=fopen(filenameout[j], "w");
        for(i=0;i<n+1;i++)
        {
            th=dx*i;
            phi=aniso_magn(a[j], th);
            fprintf(fp, "%lf\t %lf\n", th, cos(2*(th+phi)));
        }
        fclose(fp);

        fp=fopen(filenameout[j], "w");
        for(i=0;i<n+1;i++)
        {
            th=dx*i;
            phi=aniso_magn(a[j], th);
            fprintf(fp, "%lf\t %lf\n", th, cos(2*(th+phi)));
        }
        fclose(fp);
    }
}

```

}
}

2019.4.2 以上
改訂