

入門講座

状態図(相変態編)—自信を持って使うための熱力学的基礎-6

状態図から組織形成シミュレーションへ

—状態図とPhase-field法(非線形拡散方程式)の相補関係—

From Static Approach of Phase Diagram to Dynamic Simulation of Phase Decomposition

—Complementary Relation between Phase Diagram and Phase-Field Method (Non-Linear Diffusion Equation)—

小山敏幸
Toshiyuki Koyama

独立行政法人物質・材料研究機構
計算科学センター 主幹研究員

1 はじめに

状態図は相の平衡状態や準安定状態を記述する手法であり、静的相安定性の理解を主目的とした方法論である。一方、拡散方程式に基づく相分離シミュレーションは、動的相分離過程の本質の理解を主目的とする方法論である。したがって、前者は材料設計において、温度および組成空間で“何が起こり得るか”を知るための道具であり、後者はそれが、時間および位置空間において、“どのように変化していくのか”を知るための道具である。効率的な材料設計を実現するためには、この両者が必要不可欠であることは論を待たないであろう。この両者は理想的には同時に発展すべき分野であるが、この両者の学問体系は、従来、別々に発展してきたために、両者の関連性を明確に意識した論理体系の定量的整備は、ごく最近始まったに過ぎない。特に材料組織形態形成の多くの部分を記述することのできるPhase-field法¹⁻⁸⁾に基づく組織形成シミュレーション手法の構築が、現在、進展しつつあり、CALPHAD法^{9,10)}に代表される状態図の熱力学的データベース構築と融合することによって、スタティクスからダイナミクスに渡る材料設計の統一的解析法へと発展しつつある。この分野は、材料科学・工学のあらゆる分野に関連するために、非常に多様な計算モデルが存在する。ここでは、最も基本的なPhase-fieldシミュレーションとして、2元系の拡散相分離を例に取り、状態図の分野とPhase-field法(特に非線形拡散方程式)の計算が、具体的にどのように相補的に結びついているのかについて、Fe-Cu2元合金の α (bcc)相のスピノーダル分解¹¹⁾を例に取り、シミュレーションプログラム(Javaアプレット)も含めて解説する。

2 α (bcc)相(Fe-Cu合金)のスピノーダル分解の計算式

α (bcc)相(Fe-Cu合金)のスピノーダル分解の計算において、秩序変数は濃度場のみであるので、本計算のPhase-field法は、従来のCahn-Hilliardの非線形拡散方程式^{12,13)}に基づく相分離の計算手法に等しい。ここでは独立な秩序変数としてCu濃度(モル分率)を採用する。Phase-field法では、まずこの独立な秩序変数を用いて全自由エネルギーを汎関数形式に定式化し、この全自由エネルギーが効率よく減少するように、その秩序変数の発展方程式(この場合は2元系の非線形拡散方程式)が定義される。この発展方程式を数値計算することによって、秩序変数の時間および空間発展(すなわち組織形成過程)が算出される¹⁻⁵⁾。以下、使用する全自由エネルギーの計算式について説明する。

2.1 全自由エネルギーの計算式

2.1.1 化学的自由エネルギー

Fe-Cu2元系の α (bcc)相の原子1モル当たりの化学的自由エネルギーは、準正則溶体近似(磁性項を含む)に基づき、

$$G_c^{\alpha}(c, T) = G_{Fe}^{\alpha}(T)(1-c) + G_{Cu}^{\alpha}(T)c + {}^E G_m^{\alpha} + {}^{mg} G_m^{\alpha} + RT\{c \ln c + (1-c) \ln(1-c)\} \quad \dots\dots (1)$$

$${}^E G_m^{\alpha} \equiv L_{Cu, Fe}^{\alpha}(c, T) c(1-c) \quad \dots\dots\dots (2)$$

$${}^{mg} G_m^{\alpha} \equiv RT \ln(\beta^{\alpha} + 1) f(\tau) \quad \dots\dots\dots (3)$$

にて与えられる^{9,10)}。 c がCu成分のモル分率である。組織形成過程の計算であるので、 c は時間 t および組織内の位置 $\mathbf{r} = (x, y, z)$ の関数として $c(\mathbf{r}, t)$ と表記するのが正しいが、ここでは式の煩雑さを避けるために c と記している。 $G_X^{\alpha}(T)$ は純成分X(=Fe, Cu)の自由エネルギーで、温度 T の関数

である。 R はガス定数で、(1)式右辺最後の項が原子の配置のエントロピーに起因するエネルギー項である。 $E G_m^a$ は混合の過剰エンタルピー項で準正則溶体近似では(2)式にて定義され、平衡状態図の熱力学データベースにおいて、右辺の $L_{Cu,Fe}^a(c, T)$ が温度と組成の展開式として表現される^{9,10)}。 $^{mg}G_m^a$ は化学的自由エネルギー内の磁性項で、常磁性状態を基準として、相が強磁性もしくは反強磁性状態に転移した時の自由エネルギー変化量を意味している^{9,10)}。 $^{mg}G_m^a$ は(3)式にて定義され、 τ がキュリー温度 T_c^a にて規格化された温度 $\tau \equiv T/T_c^a$ で、関数 $f(\tau)$ は、

$$f(\tau) \equiv 1 - \frac{1}{D} \left\{ \frac{79\tau^{-1}}{140p} + \frac{474}{497} \left(\frac{1}{p} - 1 \right) \left(\frac{\tau^3}{6} + \frac{\tau^9}{135} + \frac{\tau^{15}}{600} \right) \right\}, \quad (\tau \leq 1)$$

$$f(\tau) \equiv -\frac{1}{D} \left(\frac{\tau^{-5}}{10} + \frac{\tau^{-15}}{315} + \frac{\tau^{-25}}{1500} \right), \quad (\tau > 1)$$

$$D \equiv \frac{518}{1125} + \frac{11692}{15975} \left(\frac{1}{p} - 1 \right) \dots\dots\dots (4)$$

にて定義される^{9,10)}。固溶体がbcc構造の場合 $p=0.40$ であり、それ以外の結晶構造では $p=0.28$ となる。 β^a はボア磁子で無次元化された1原子当たりの磁化の強さである。合金状態図の熱力学データベースにおいて、 T_c^a と β^a は組成による展開式として与えられる。

具体的にFe-Cu 2元系の α (bcc)相については、 $L_{Cu,Fe}^a(c, T)$, T_c^a , および β^a は、

$$L_{Cu,Fe}^a(c, T) = 39676.0 - 4.73222T, \quad [\text{J/mol}]$$

$$T_c^a = 1043(1-c) - 41.4c(1-c), \quad [\text{K}]$$

$$\beta^a = 2.22(1-c)$$

にて与えられている¹⁴⁾。本計算の対象は同一結晶構造(bcc)内での等温時効における拡散相分解であるので、自由エネルギーの基準0として改めて、 $G_{Fe}^a(T)(1-c) + G_{Cu}^a(T)c$ を採用しても計算結果は影響を受けない。(拡散相分離の計算では、拡散ポテンシャルの空間勾配が溶質移動の駆動力になる。拡散ポテンシャルを計算する際に自由エネルギーの組成に関する1次項は定数となり(温度一定)、さらに拡散ポテンシャルの空間勾配を計算する際に、この定数は消えてしまう)。したがって、ここでは、議論を簡単にするために、自由エネ

ルギーの基準について、 $G_{Fe}^a(T) = G_{Cu}^a(T) = 0$ と設定する。

2.1.2 界面エネルギー (濃度勾配エネルギー)

実際の界面エネルギーには、界面位置における平均場の化学的自由エネルギーも含まれるので、界面エネルギーと濃度勾配エネルギーは正確には異なるが、ここでは(界面エネルギー) = (濃度勾配エネルギー)として議論を進める。計算式は、

$$E_{surf} = \frac{1}{2} \kappa_c (\nabla c_{Fe})^2 + \frac{1}{2} \kappa_c (\nabla c_{Cu})^2$$

$$= \frac{1}{2} \kappa_c \{ \nabla(1-c) \}^2 + \frac{1}{2} \kappa_c (\nabla c)^2$$

$$= \kappa_c (\nabla c)^2 \dots\dots\dots (5)$$

にて与えられる^{1-5,12)}。スピノーダル分解の計算では濃度勾配エネルギー係数 κ_c は通常、方向に依存しない定数と仮定される。 κ_c の値は、原子間相互作用パラメータ Ω 、界面エネルギー密度 γ_s 、界面幅 d_1 、および相互作用距離¹³⁾ d_2 を用いて、近似的に見積もることが可能である。評価式には、通常、 $\kappa_c = k_1 d_1 \gamma_s V_m$ 、もしくは、 $\kappa_c = k_2 d_2^2 \Omega$ が用いられる。 k_1 と k_2 は結晶構造や温度に依存する定数であるが、推奨値などは未だ明確にはなっていない。著者は特に情報が無い場合には経験的に、 $k_1 = 1 \sim 2$ および $k_2 = 0.5$ 、かつ $d_2 = d_1/2$ を用いている。

Fe-CuのCu(bcc)整合析出を対象に、 $\gamma_s = 0.28$ (J/m²)¹⁵⁾、 $d_1 = 1$ (nm)、 $\Omega = 39676$ (J/mol) ((2)式の $L_{Cu,Fe}^a$ の0Kにおける値を採用)と仮定すると、上式から例えば

$$\kappa_c = 1.5 \times (1 \times 10^{-9}) [\text{m}] \times 0.28 [\text{J/m}^2]$$

$$\times (7.1 \times 10^{-6}) [\text{m}^3/\text{mol}]$$

$$\cong 3.2 \times 10^{-15} [\text{J} \cdot \text{m}^2/\text{mol}]$$

$$\kappa_c = 0.5 \times (0.5 \times 10^{-9})^2 [\text{m}^2] \times 39676 [\text{J/mol}]$$

$$\cong 5.0 \times 10^{-15} [\text{J} \cdot \text{m}^2/\text{mol}]$$

となり、両者はほぼ近い値となる。本計算では $\kappa_c = 5.0 \times 10^{-15}$ [J・m²/mol]を採用した。しかしいずれにしても、界面幅 d_1 の設定や界面エネルギー密度 γ_s の値自体、不明な場合が多く、 κ_c は物理定数と考えるよりも計算モデルにおけるフィッティングパラメータと見なした方が工学的には有効な場合が多い。例えば実験で得られたスピノーダル分解組織の変調構造の波長を用いて、この値を再現できるように逆問題として κ_c をシミュレーションから決定するのも有効な方法である。さらに整合相分解に限定すれば、条件(温度、組成)を変えても、 κ_c 値は通常大きく変化しないので、ある1つの実験条件で κ_c 値を定めれば良い。なお κ_c の単位に

については、ここでは化学的自由エネルギーの単位 [J/mol] との整合性から [J・m²/mol] を採用しているが、モル体積で割って [J/m] としている場合が多い点を記しておく。

2.1.3 弾性歪エネルギー

弾性歪エネルギー E_{str} は、正確には非等方弾性論に基づき、弾性場の長距離相互作用を考慮して積分方程式として解析する必要があるが、ここでは議論を簡単にするため等方弾性体を仮定する。 E_{str} は、

$$E_{str} = \eta^2 Y (c - c_0)^2 V_m \dots\dots\dots (6)$$

にて与えられる^{1-5, 12)}。 c_0 は Cu の合金組成、 Y は弾性率の関数^{1-5, 12)}、 V_m はモル体積、および η は格子ミスマッチ^{1-5, 12)} である。Fe (bcc) と Cu (bcc) の格子定数 $a_{Fe} = 0.28664$ (nm)¹⁶⁾ および $a_{Cu} = 0.29607$ (nm)¹⁷⁾ から、 $\eta = (a_{Cu} - a_{Fe}) / a_{Fe} = 3.29 \times 10^{-2}$ と評価される。また Y は以下のように計算される。まず Fe (bcc) と Cu (bcc) の弾性定数として、 $C_{11}^{Fe} = 2.33 \times 10^{11}$, $C_{12}^{Fe} = 1.35 \times 10^{11}$, $C_{44}^{Fe} = 1.18 \times 10^{11}$ (Pa)¹²⁾ および $C_{11}^{Cu} = 2.86 \times 10^{11}$, $C_{12}^{Cu} = 1.90 \times 10^{11}$, $C_{44}^{Cu} = 1.45 \times 10^{11}$ (Pa)¹⁶⁾ を採用し、次にこれを合金組成 c_0 で重み付き平均して、 $C_{11} = C_{11}^{Fe}(1 - c_0) + C_{11}^{Cu}c_0$, $C_{12} = C_{12}^{Fe}(1 - c_0) + C_{12}^{Cu}c_0$ とする。これより Y は、 $Y = C_{11} + C_{12} - 2(C_{12}^2/C_{11})$ にて計算される (ここでは変調構造の計算の際に用いられる関数 $Y_{\langle hkl \rangle}$ の $Y_{\langle 100 \rangle}$ の式を採用している^{1-5, 12)})。 V_m については純 Fe (bcc) のモル体積を用いる。純 Fe (bcc) の格子定数を a_{Fe}^a とすると、 N_{Av} をアボガドロ数として、モル体積 V_m は原子1個あたりの体積 $(a_{Fe}^a)^3 / 2$ を N_{Av} 倍すればよいので、 $V_m = N_{Av} (a_{Fe}^a)^3 / 2$ にて計算できる (bcc では単位胞に原子が2個あるので、2で割っている)。

2.1.4 全自由エネルギー汎関数と拡散ポテンシャル

以上から、全自由エネルギーは、化学的自由エネルギー、界面エネルギー (濃度勾配エネルギー)、および弾性歪エネルギーの総和として、空間における汎関数形式にて、

$$G_{sys} = \int_{\mathbf{r}} \left[\begin{array}{l} L_{Cu, Fe}^a c(1 - c) \\ + RT \{ c \ln c + (1 - c) \ln (1 - c) \\ + RT \ln (\beta^a + 1) f(\tau) + \kappa_c (\nabla c)^2 \\ + \eta^2 Y (c - c_0)^2 V_m \end{array} \right] d\mathbf{r} \dots\dots (7)$$

と表現される。 G_{sys} の c による変分は、変分原理に基づき、

$$\frac{\delta G_{sys}}{\delta c} = \frac{\partial G_c^a}{\partial c} - 2\kappa_c \nabla^2 c + 2\eta^2 Y (c - c_0) V_m \dots\dots\dots (8)$$

にて与えられ^{1-5, 12)}、これは拡散ポテンシャルと呼ばれる (弾性歪エネルギーや濃度勾配エネルギーを考慮しない条件下で定義されている場合が多いが、整合相分解を扱う場合は

これら全てのエネルギーを考慮する必要がある)。

2.2 発展方程式 (非線形拡散方程式) の計算式

計算に用いる非線形拡散方程式¹⁻⁵⁾は、

$$\frac{\partial c}{\partial t} = \nabla \cdot \left(M_c(c, T) \nabla \frac{\partial G_{sys}}{\partial c} \right) \dots\dots\dots (9)$$

にて与えられる (スピノードル分解の計算が対象であるので、濃度揺らぎ項は省略した)。 $M_c(c, T)$ は原子の拡散の易動度で、組成および温度の関数であるが¹⁻⁵⁾、合金組成 c_0 および温度 T の関数 $M_c(c_0, T)$ と近似する。(8) 式を代入すると、

$$\begin{aligned} \frac{\partial c}{\partial t} &= \nabla \cdot \left[M_c(c_0, T) \nabla \left\{ \frac{\partial G_c^a}{\partial c} - 2\kappa_c \nabla^2 c + 2\eta^2 Y (c - c_0) V_m \right\} \right] \\ &= \nabla \cdot \left[M_c(c_0, T) \left(\frac{\partial^2 G_c^a}{\partial c^2} + 2\eta^2 Y V_m \right) \nabla c \right] \\ &\quad - 2M_c(c_0, T) \kappa_c \nabla^4 c \dots\dots\dots (10) \end{aligned}$$

を得る。ここで、

$$\tilde{D} \equiv M_c(c_0, T) \left(\frac{\partial^2 G_c^a}{\partial c^2} + 2\eta^2 Y V_m \right)$$

および

$$\tilde{K} \equiv M_c(c_0, T) \kappa_c$$

と置くと、Cahn-Hilliard の非線形拡散方程式

$$\frac{\partial c}{\partial t} = \nabla \cdot (\tilde{D} \nabla c) - 2\tilde{K} \nabla^4 c \dots\dots\dots (11)$$

に一致することがわかる。ここで \tilde{D} は整合相分解における相互拡散係数に他ならない。

なお実際の相分離シミュレーションでは、(8) 式の拡散ポテンシャル $\mu_{sys} \equiv \delta G_{sys} / \delta c$ を位置の関数 (\mathbf{r}, t) として数値計算し、(9) 式を

$$\frac{\partial c}{\partial t} = \nabla \cdot \{ M_c \nabla \mu_{sys} \} = M_c \nabla^2 \mu_{sys} \dots\dots\dots (12)$$

と変形して、差分法を用いて数値計算した方が効率的である。つまり組織内の任意の点における拡散ポテンシャル $\mu_{sys}(\mathbf{r}, t)$ を (8) 式から求め、 $\mu_{sys}(\mathbf{r}, t)$ について直接差分計算を行う手法である。

2.2.1 非線形拡散方程式の無次元化

拡散方程式の無次元化について説明する。まずエネルギーは RT にて無次元化する。また2次元の計算領域の1辺の長さを L 、差分計算の分割数を N とすると、差分による空間分割セルの1辺の長さ b_1 は、 $b_1 = L/N$ である。この b_1 を用いて距離を無次元化する (差分計算を容易にするため)。また時間は b_1^2/D にて無次元化する (D は拡散係数)。本計算では

等温時効を想定しているので、易動度 $M_c(c_0, T)$ は定数 M_c となる。また拡散係数と易動度の関係式(後述)より、 M_c の次元は $[\text{m}^2/\text{s}]/[\text{J}/\text{mol}]$ である。以上より2次元 (x, y) における非線形拡散方程式を、次元と合わせて表現すると、

$$\frac{\partial c}{\partial t} = M_c \left(\frac{\partial^2 \mu_{\text{sys}}}{\partial x^2} + \frac{\partial^2 \mu_{\text{sys}}}{\partial y^2} \right) : \dots\dots\dots (13)$$

$$\left[\frac{1}{\text{s}} \right] = \left[\frac{\text{m}^2/\text{s}}{\text{J}/\text{mol}} \right] \left[\frac{\text{J}/\text{mol}}{\text{m}^2} \right] \dots\dots\dots (13)$$

となり、無次元化した方程式は、

$$\frac{\partial c}{\partial \left(\frac{t}{b_1^2/D} \right)} = \frac{M_c RT}{D} \left(\frac{\partial^2 (\mu_{\text{sys}}/RT)}{\partial (x/b_1)^2} + \frac{\partial^2 (\mu_{\text{sys}}/RT)}{\partial (y/b_1)^2} \right) \dots\dots\dots (14)$$

となる。なお濃度勾配エネルギー一定数 κ_c は $b_1^2 RT$ にて無次元化される。

2.2.2 拡散係数と易動度について

自己拡散係数 D_X^* ($X = \text{Fe}, \text{Cu}$) と $M_c(c, T)$ には

$$M_c(c, T) = \left(\frac{D_{\text{Cu}}^*}{RT} (1-c) + \frac{D_{\text{Fe}}^*}{RT} c \right) c(1-c) \dots (15)$$

の関係がある^{1-5, 12)}。 D_X^* は固溶体中におけるX原子の自己拡散係数であるので、正確には D_X^* は温度および濃度(媒質の固溶体の濃度)の関数である。しかし D_X^* の組成依存性のデータは少なく、さらにbcc構造におけるCuの自己拡散のデータは入手困難である。ここでは最も単純に、bccFe中におけるCuの不純物拡散係数 $D_{\text{Cu}}(T)$ にて D_{Cu}^* と D_{Fe}^* を代用し、かつ組成 c については合金組成 c_0 にて近似する。したがって、(15)式を变形することにより、(14)式右辺の係数部分が、

$$M_c(c_0, T) = \left(\frac{D_{\text{Cu}}(T)}{RT} (1-c_0) + \frac{D_{\text{Cu}}(T)}{RT} c_0 \right) c_0 (1-c_0) = \frac{D_{\text{Cu}}(T)}{RT} c_0 (1-c_0) \therefore \frac{M_c(c_0, T) RT}{D_{\text{Cu}}(T)} = c_0 (1-c_0) \dots\dots\dots (16)$$

と導かれる。さて αFe 中のCuの不純物拡散係数は、

$$D_{\text{Cu}} = 4.7 \times 10^{-5} \exp\left(-\frac{244000}{RT}\right), [\text{m}^2/\text{s}]^{17)}$$

にて与えられるが、実際に相分解シミュレーションを行う際

に、この値は、無次元化された計算時間 $t' \equiv t(b_1^2/D)$ を実時間 t に変換する時のみ必要になる。つまり無次元化された非線形拡散方程式は最終的に

$$\frac{\partial c}{\partial t'} = c_0(1-c_0) \left(\frac{\partial^2 (\mu_{\text{sys}}/RT)}{\partial (x/b_1)^2} + \frac{\partial^2 (\mu_{\text{sys}}/RT)}{\partial (y/b_1)^2} \right) \dots\dots\dots (17)$$

であるので、計算時間 t' にて議論する限りにおいては、拡散係数の値自体は特に必要ではない。

3 計算状態図とスピノーダル分解の計算結果

3.1 α (bcc) 相の相分離に対する準安定状態図

Fe-Cu2元系における α (bcc)相のみを考慮した場合の準安定状態図は図1にて与えられる¹¹⁾。実線がバイノーダル線、点線がスピノーダル線、および一点鎖線はキュリー温度を表している。磁気過剰自由エネルギーの効果によって、キュリー温度とスピノーダル線が交わる位置においてスピノーダル線に折れ曲がりが生じている⁹⁾。Fe-Cu系は非常に相分離傾向が大きく、図のように広いmiscibility gapが存在する(なおFe-Cu系では、 γ (fcc)相や液相においても同様な準安定のmiscibility gapが存在する)。つまり、この系の相分離の駆動力は極めて大きいので、形成されるCu-rich相は極めて微細(数nm程度)となる。近年、このナノスケールのCu析出を鉄鋼材料の力学特性改善に積極的に活用する研究が進められている¹⁸⁾。

3.2 プログラムの実行方法と計算結果の例

2次元相分解シミュレーションのプログラムはJavaアプレットにて作成されている。本解説の付録(1)にJavaアプレ

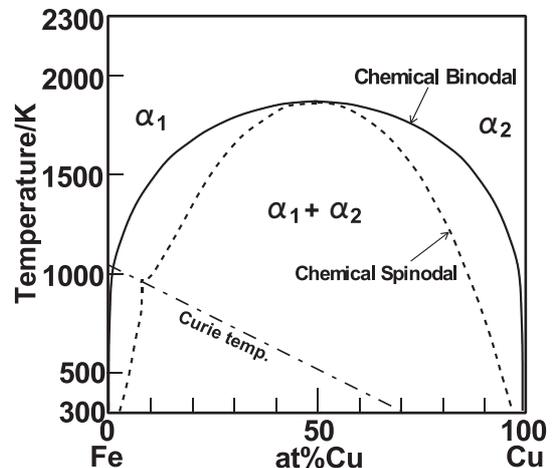


図1 Fe-Cu2元系における α (bcc)相の準安定状態図

ットのソースコード (FeCu_bcc_ferrum.java)、および付録 (2) にアプレット起動用のhtmlファイル (FeCu_bcc_ferrum.html) を示す。プログラムの内容については、プログラム内のコメント文 (各行の//以降の文) を参照していただきたい。Javaコンパイラはインターネット経由で無償にて入手できる。なお、Java言語やコンパイル環境の構築については文献 (19, 20) 等をご参照願いたい。

さてFeCu_bcc_ferrum.javaをコンパイルすると、クラスファイル (FeCu_bcc_ferrum.class) が生成される。次に生成されたクラスファイルとFeCu_bcc_ferrum.htmlをディスク上の同じディレクトリに置き、FeCu_bcc_ferrum.htmlダブルクリックして、Internet Explorer等のブラウザを起動することによって、図2の画面が現れる。図中の左列が入力するパラメータ値で、上から、合金組成 (モル分率)、時効温度 (K)、計算領域の長さ (nm)、および時間きざみ (無次元) である。画面の一番下に、スタート、ストップ、およびリセットのボタンがあり、左列の値を設定して、スタートボタンを押すことにより相分解シミュレーションが開始する。図の下の数値は計算ステップ数である (計算ステップ数に時間きざみを乗じた数が無次元の計算時間)。ストップボタンによって計算は一時停止し、リセットボタンによって、均一固溶体状態に戻る。

具体的な計算例を図3に示す。図3 (a) ~ (d) は、合金組成 $c_0=0.3$ (Fe-30at%Cu) および時効温度873Kにおけるスピノーダル分解の時間発展である。図中の相分解組織の明暗が局所的なCu組成に対応し、白~黒が0at%Cu~100at%Cuを表している。また計算領域1辺は20nmに設定してある。初期状態は図2の固溶体 ($\pm 1\%$ 程度の濃度揺らぎを乱数によって与えてある) であり、相分解の初期に均一な“まだら構造”が形成され [図3 (a) ~ (b)], その後、組織は孤立した微細なCu析出粒子が分散する組織となり [図3 (c)],

時効の進行に伴いオストワルド成長によってCu析出相が粗大化していく [図3 (d)]。合金組成や、時効温度を変えて種々の条件下でのスピノーダル分解過程を、パソコン上にてその場観察しながら計算を進めることができる。

本Javaアプレットのソースコード (FeCu_bcc_ferrum.java) を修正することによって、本プログラムは、Fe-Cr合金やCo-Cu合金などの他の2元系のプログラムに容易に書き換えることができる。特にFe-Cr合金への変換は簡単であるので、興味のある方はトライされることを薦める。

4 おわりに

以上、Fe-Cu2元系の拡散相分離シミュレーションを例にとり、平衡状態図の熱力学的データがどのように相分離シミュレーションに反映されているかについて説明した。組織形成の解析に対して、状態図に関連する熱力学的データを直接使用することが可能となったことは、合金組織学が、平衡論 (スタティクス) から動的変化過程 (ダイナミクス) までを統一かつ定量的に同時解析できるステージに入り始めたことを意味している。近年、種々の分野で活発に研究が進められているナノ・メゾスケールにおける材料設計では、デバイスのスケール自体が微視的な材料組織のスケールに重なってくるので、組織形態のコントロールが、これまでも増して重要となりつつある。この時、平衡論から動的変化過程までをシームレスに活用できるスキルを完備しておくことは、次世代の新材料・デバイス設計 (競争力を十分に維持するもの造り方法論の構築) において、決定的な武器になるのではない



図2 相分解の2次元シミュレーションの初期画面

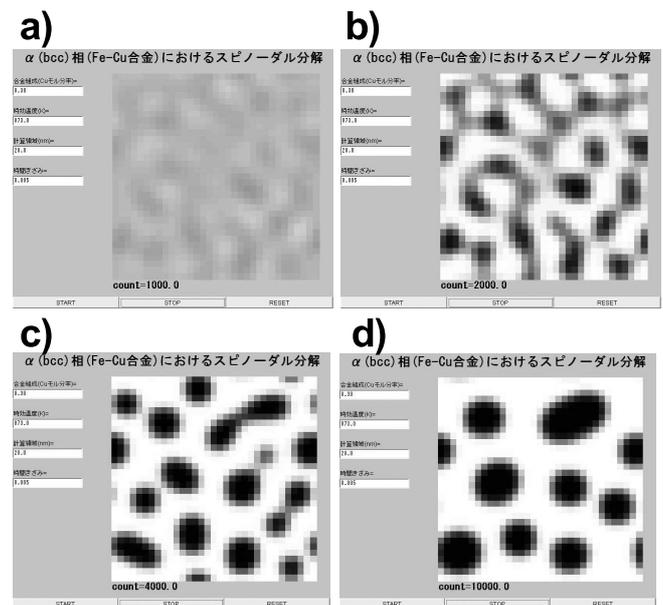


図3 Fe-30at%Cu合金の873K等温時効におけるスピノーダル分解過程の計算例

だろうか。

なお本稿では、より基礎的な事項やさらに進んだ計算理論の詳細については紙面の関係上ふれなかった。計算理論に関する基礎に関しては文献(1-5)をご参照願いたい。また各種のデモプログラムやより進んだ計算理論の詳細については、著者のホームページ (http://www.nims.go.jp/cmssc/pst/koyama/Phase-Field_Modeling.htm) にて公開しているので、ご興味のある方は参照していただきたい。

付録(1)

α (bcc) 相 (Fe-Cu 合金) のスピノーダル分解のJava アプレットソースコード (ファイル名: FeCu_bcc_ferrum.java)

```
//-----
//  $\alpha$  (bcc) 相 (Fe-Cu) のスピノーダル分解過程の
// 2次元シミュレーション
//----- by T.Koyama, (June 20, 2006)---
```

```
import java.applet.*;
import java.awt.*;
import java.awt.event.*;
```

```
public class FeCu_bcc_ferrum extends Applet
implements ActionListener, Runnable{
```

```
int ND=32; //組織1辺の分割数
int nd=ND, ndm=ND-1;
int width, height; //画面の幅と高さ
double rr=8.3145; //ガス定数
double [][] ch=new double[ND][ND];
//組織内の濃度データ配列
```

```
//グローバル変数
double temp; //温度(K)
double c0; //合金組成(モル分率)
double delt; //時間刻み(無次元)
double al; //計算領域の1辺の長さ
double time1; //計算時間(カウント数)
```

```
//グローバル変数の入力受け変数
TextField c0_TXT, delt_TXT, temp_TXT, al_TXT;
```

```
Button bt_stop, bt_start, bt_reset;
```

```
Image buff;
Graphics bg;
Thread thread1=null;
```

```
/**初期画面設定*****
```

```
public void init (){
```

```
width = getSize ().width;
//画面サイズはhtmlファイル
//(FeCu_bcc_ferrum.html)より
//読み込み
height = getSize ().height;
setBackground (Color.lightGray);
setLayout (new BorderLayout ());
```

```
buff=this.createImage (width, height);
bg=buff.getGraphics ();
```

```
//---表題表示-----
```

```
Panel p0=new Panel ();
p0.setLayout (new GridLayout (1,1,0,3));
Label Title_Label_1
=new Label (" $\alpha$ (bcc)相(Fe-Cu 合金)
におけるスピノーダル分解",
Label.CENTER);
Title_Label_1.setFont (new Font
("Courier", Font.BOLD, 26));
p0.add (Title_Label_1);
add (p0, "North");
```

```
//---定数入力一覧-----
```

```
Panel p1=new Panel ();
p1.setLayout (new GridLayout (23,1,0,0));
p1.add (new Label ("")); //空白行
p1.add (new Label ("合金組成(Cuモル分率)="));
c0_TXT=new TextField ("0.30", 4);
p1.add (c0_TXT);
c0_TXT.addActionListener (this);
p1.add (new Label ("")); //空白行
p1.add (new Label ("時効温度(K)="));
temp_TXT=new TextField ("873.0", 5);
p1.add (temp_TXT);
temp_TXT.addActionListener (this);
p1.add (new Label ("")); //空白行
p1.add (new Label ("計算領域(nm)="));
al_TXT=new TextField ("20.0", 4);
p1.add (al_TXT);
al_TXT.addActionListener (this);
p1.add (new Label ("")); //空白行
```

```

p1.add (new Label ("時間きざみ="));
delt_TXT=new TextField ("0.005", 5);
p1.add (delt_TXT);
delt_TXT.addActionListener (this);
p1.add (new Label ("")); //空白行

add (p1, "West");

//--- START STOP RESET ボタン設定-----
Panel p2=new Panel ();
p2.setLayout (new GridLayout (1,3,0,5));

bt_start=new Button ("START");
p2.add (bt_start);
bt_start.addActionListener (this);
bt_stop=new Button ("STOP");
p2.add (bt_stop);
bt_stop.addActionListener (this);
bt_reset=new Button ("RESET");
p2.add (bt_reset);
bt_reset.addActionListener (this);

add (p2, "South");

//---時間0における濃度場設定-----
time1=0.0; c0=0.3; ini_comp_field ();
}

//**** 初期濃度場設定 ****
public void ini_comp_field () {
    int i, j;
    double rnd0, fac1;

    fac1=0.01; //最大1%の初期濃度揺らぎ
    for (i=0;i<=ndm;i++){
        for (j=0;j<=ndm;j++){
            rnd0=2.0*Math.random ()-1.0;
            ch[i][j]=c0+rnd0* fac1;
        }
    }
}

//*** イベント定義 ****
public void actionPerformed (ActionEvent ev){

    String string1;

    //スタートボタンが押された時、スレッド開始
    if (ev.getSource ()==bt_start){
        //---入力定数を、変数への代入-----
        string1=c0_TXT.getText ().trim ();
        c0=new Double (string1).doubleValue ();
        string1=delt_TXT.getText ().trim ();
        delt=new Double (string1).doubleValue ();
        string1=temp_TXT.getText ().trim ();
        temp=new Double (string1).doubleValue ();
        string1=al_TXT.getText ().trim ();
        al=new Double (string1).doubleValue ();
        thread1=new Thread (this);
        thread1.start (); //スレッドスタート
    }

    //ストップボタンが押された時、スレッドを nullにする
    else if (ev.getSource ()==bt_stop){ thread1=null; }

    //リセットボタンが押された時、初期単相状態にもどす
    else if (ev.getSource ()==bt_reset){
        time1=0.0; ini_comp_field (); repaint ();
    }
}

//***シミュレーションのメインルーチン ****
public void run () {
    double [][] ck = new double[ND][ND];
        //拡散ポテンシャル
    double [][] ch2 = new double[ND][ND];
        //組織内の濃度データ予備配列
    double mu_chem, mu_str, mu_surf, mu_mag;
        //各ポテンシャル
    int i, j, k, l;
        //整数
    double c1, c2;
        //濃度 (Fe:1, Cu:2)
    double a01, a02, atomNo;
        //格子定数と、単位胞内の原子数
    double vm0;
        //モル体積
    double L0;
        //原子間相互作用パラメータ
    double L0_0, L0_1;
        //相互作用パラメータ内の定数
    double eta;
        //格子ミスマッチ
    double c11a, c12a, c44a;
        //Feの弾性定数
    double c11b, c12b, c44b;
        //Cu (bcc)の弾性定数
    double c11, c12, c44;

```

```

//合金の平均弾性率
double y100;
//弾性関数
double Da, Da0, Qa0;
//拡散係数、頻度因子、活性化エネルギー
double Mc;
//易動度関数とその微分
double cddtt;
//濃度の増分
double b1;
//差分ブロックサイズ
double kapa_c;
//濃度勾配エネルギー係数
int ip, im, jp, jm;
//整数(i+1, i-1, j+1, j-1)
double c2ip, c2im, c2jp, c2jm;
//差分ブロックにおいてc2を中心に、
//その上下左右の濃度
double ck2dev;
//拡散ポテンシャルの2階微分
double ck0, ckip, ckim, ckjp, ckjm;
//差分ブロックにてck0を中心に
//その上下左右の拡散ポテンシャル
double sumc, dc;
//濃度場の総和、平均組成からのずれ

//磁気過剰エネルギー関連パラメータ*****
double Tc, d2Tc;
//キュリー温度、およびその組成に対する微分
double tau;
//キュリー温度で規格化した温度
double ftau, dftau;
//磁気過剰エネルギー関数と
//温度に対する微分
double beta, d2beta;
//ボーア磁子、およびその組成に対する微分
double p_mag, D_mag;
//pとD

//*****
a01=0.28664; //格子定数(bcc Fe)
a02=0.29607; //格子定数(bcc Cu)
atomNo=2.0; //単位胞内の原子数

L0_0=39676.0;
//原子間相互作用パラメータ(J/mol)
L0_1=-4.73222; //L=L0_0+L0_1*T

kapa_c=5.0e-15;

//濃度勾配エネルギー係数、
//単位は[J m^2/mol]

eta=(a02-a01)/a01; //格子ミスマッチ

c11a=2.3310e+11; //bccFeの弾性定数(Pa)
c12a=1.3544e+11;
c44a=1.1783e+11;

c11b=2.86e+11; //bccCuの弾性定数(Pa)
c12b=1.90e+11;
c44b=1.45e+11;

Da0=4.70e-05;
//不純物拡散(bccFe中のCu)における
//ジャンプ頻度(m^2/s)
Qa0=2.44e+05;
//不純物拡散(bccFe中のCu)における
//活性化エネルギー(J/mol)

//-----
al=al*1.0e-9;
//(m)に変換
b1=al/(double)nd;
//差分1ブロックのサイズ

a01=a01*1.0e-09;
//Feの格子定数(mに換算)
vm0=6.02e+23*a01*a01*a01/atomNo;
//モル体積

kapa_c=kapa_c/b1/b1/rr/temp;
//濃度勾配エネルギー係数、
//(b1^2*rr*temp)で無次元化

c11a=c11a*vm0/rr/temp; //RTで無次元化
c12a=c12a*vm0/rr/temp;
c44a=c44a*vm0/rr/temp;
c11b=c11b*vm0/rr/temp;
c12b=c12b*vm0/rr/temp;
c44b=c44b*vm0/rr/temp;

c11=(1.0-c0)*c11a+c0*c11b; //合金の弾性率
c12=(1.0-c0)*c12a+c0*c12b;
c44=(1.0-c0)*c44a+c0*c44b;

y100=c11+c12-2.0*(c12*c12/c11);
//弾性率の関数Y<100>

```

```

Da=Da0*Math.exp (- Qa0/rr / temp);
//bccFe中のCuの不純物拡散係数
Mc=c0*(1.0- c0); //式(16)
p_mag=0.4; // (bcc)
D_mag=518.0/1125.0
+11692.0/15975.0* (1.0/p_mag -1.0);

//-----
while (thread1!=null){
if (((int)(time1)% 10)==0){ repaint ();}
//カウント数が10の倍数おきに濃度場描画

//***** [拡散ポテンシャルの計算] *****
for (i=0;i<=ndm;i++){
for (j=0;j<=ndm;j++){
ip=i+1; im=i-1; jp=j+1; jm=j-1;
if (i==ndm){ip=0;} if (i==0){im=ndm;}
if (j==ndm){jp=0;} if (j==0){jm=ndm;}

c2=ch[i][j]; c1=1.0- c2;
c2ip=ch[ip][j]; c2im=ch[im][j];
c2jp=ch[i][jp]; c2jm=ch[i][jm];

L0=(L0_0+L0_1*temp)/rr / temp;
//原子間相互作用パラメータ
mu_chem=L0*(c1- c2)
+Math.log (c2)- Math.log (c1);
//化学ポテンシャル差
mu_surf=- 2.0*kapa_c*(c2ip+c2im
+c2jp+c2jm-4.*c2);
//濃度勾配のポテンシャル
mu_str=2.0*eta*eta*y100*(c2- c0);
//弾性ポテンシャル

Tc=1043.0*c1-41.4*c1*c2;
//キュリー温度
d2Tc=- 1043.0-41.4*(c1- c2);
//キュリー温度のCu組成微分
if (Tc<0.){Tc=- Tc; d2Tc=- d2Tc;}
//bccの場合

beta=2.22*c1;
//1原子当たりの磁化の強さ
//(ボーア磁子で無次元化)
d2beta=- 2.22;
//betaのCu組成微分
if (beta<0.){beta=- beta; d2beta=- d2beta;}
//bccの場合

tau=temp / Tc;
//  $\tau$  の定義
if (tau<=1.0){
//f( $\tau$ )と、f( $\tau$ )の $\tau$ 微分の計算
ftau=1.0-1.0/D_mag*(79.0/140.0/p_mag / tau
+474.0/497.0*(1.0/p_mag-1.0)
*( Math.pow (tau,3.0)/6.0
+Math.pow (tau,9.0)/135.0
+Math.pow (tau,15.0)/600.0));
dftau=- 1.0/D_mag*(- 79.0/140.0
/p_mag / tau / tau
+474.0/497.0*(1.0/p_mag-1.0)
*( Math. pow (tau,2.0)/2.0
+Math.pow (tau,8.0)/15.0
+Math.pow (tau,14.0)/40.0));
}
else{
ftau=- 1.0/D_mag
*(Math.pow (tau,- 5.0)/10.0
+Math.pow (tau,- 15.0)/315.0
+Math.pow (tau,- 25.0)/1500.0);
dftau=1.0/D_mag
*(Math.pow (tau,- 6.0)/2.0
+Math.pow (tau,- 16.0)/21.0
+Math.pow (tau,- 26.0)/60.0);
}

mu_mag=ftau / (beta+1.0)*d2beta
- dftau*tau / Tc*d2Tc*Math.log (beta+1.0);
//磁気過剰エネルギーのポテンシャル

ck[i][j]=mu_chem+mu_str+mu_surf+mu_mag;
//拡散ポテンシャル(式(8))
}
}

//***** [濃度場の時間変化
(非線形拡散方程式の差分解陽解法)] *****
for (i=0;i<=ndm;i++){
for (j=0;j<=ndm;j++){
ip=i+1; im=i-1;
jp=j+1; jm=j-1;
if (i==ndm){ip=0;} if (i==0){im=ndm;}
if (j==ndm){jp=0;} if (j==0){jm=ndm;}

ck0=ck[i][j];
ckip=ck[ip][j];
ckim=ck[im][j];

```

```

ckjp=ck[i][jp];
ckjm=ck[i][jm];

ck2dev=(ckip+ckim+ckjp+ckjm)-4.0*ck0;
//拡散ポテンシャルの2階微分
cddtt=Mc*ck2dev;
//非線形拡散方程式(式(14))
ch2[i][j]=ch[i][j]+cddtt*delt;
//濃度場の時間発展
//ch2[i][j]=ch[i][j]+cddtt*delt
//+1.0e-03*(2.0*Math.random()-1.0);
//濃度場の時間発展(濃度揺らぎ導入)
}
}

/** [濃度場の収支の補正] *****
/** 数値計算であるので濃度場の収支の補正を行う
/** (実際には毎ステップ行う必要はない)****
sumc=0;
for (i=0;i<=ndm;i++){
for (j=0;j<=ndm;j++){
sumc=sumc+ch2[i][j];
}
}
dc=sumc/(double)nd/(double)nd-c0;

for (i=0;i<=ndm;i++){
for (j=0;j<=ndm;j++){
ch[i][j]=ch2[i][j]-dc;
if (ch[i][j]>=1.0){ch[i][j]=1.0-1.0e-6;}
if (ch[i][j]<=0.0){ch[i][j]=1.0e-6;}
}
}

/** [時間増加] *****
time1=time1+1.0;
//try{ thread1.sleep (100); } // 0.01 seconds
//catch ( InterruptedException e) { }
} // while
}

/** *****
public void update ( Graphics g) { paint ( g); }

/** 濃度場描画 *****
public void paint (Graphics g){
bg.clearRect (0, 0, width, height);
bg.setColor (Color.lightGray);
int xwidth=400, yheight=400;
int delxheight=200, delyheight=50;
int i, j, ii, jj, icol;
int ixmin=0, iymin=0, igx, igy, irad0;
int ixmax=xwidth, iymax=yheight;
double c, x, xmax, xmin, y, ymax, ymin, rad0;

xmin=0.0; xmax=1.0; ymin=0.0; ymax=1.0;

bg.setColor (Color.black);
//描画色を黒に設定
bg.setFont (new Font ("Courier", Font.BOLD, 20));
//フォントを設定
bg.drawString ("count="+time1,
delxheight, yheight+delyheight+30);
//文字列描画

rad0=1.0/(double)nd/2.0;
irad0=1+(int) ( ((double)ixmax
-(double)ixmin)/ (xmax-xmin) *rad0);

for (i=0;i<=nd;i++){
for (j=0;j<=nd;j++){
x=1.0/(double)nd*(double)i+rad0;
igx=(int) ( ((double)ixmax-(double)ixmin)
*(x-xmin)/(xmax-xmin)
+(double)ixmin );
y=1.0/(double)nd*(double)j+rad0;
igy=(int) ( ((double)iymax-(double)iymin)
*(y-ymin)/(ymax-ymin)
+(double)iymin );
ii=i; jj=j; if (i==nd){ii=0;} if (j==nd){jj=0;}
//icol=(int) (255.0*ch[ii][jj]);
icol=(int) (255.0*(1.0-ch[ii][jj]));
if (icol>=255){icol=255;}
if (icol<=0){icol=0;}
bg.setColor (new Color (icol,icol,icol));
bg.fillRect (delxheight+igx-irad0,
delyheight+igy-irad0,
irad0*2, irad0*2);
//bg.setColor (Color.blue);
//bg.setColor (new Color (0,0,icol));
}
}
g.drawImage (buff, 0, 0, this);
}
}

/** 以上 *****

```

付録 (2)

Java アプレット実行用のhtmlファイル (ファイル名:
FeCu_bcc_ferrum.html)

```
<html><body>
  <title>test</title>
  <applet code="FeCu_bcc_ferrum.class"
    width="640" height="520">
  </applet>
</body></html>
```

(注) 以上のFeCu_bcc_ferrum.html、FeCu_bcc_ferrum.java、およびFeCu_bcc_ferrum.classをコンパイルしたFeCu_bcc_ferrum.classの3つのファイルは、著者のホームページ (http://www.nims.go.jp/cmssc/pst/koyama/Phase-Field_Modeling.htm) から、無償でダウンロードできるので、ご興味のある方は自由に活用していただきたい。なお本プログラムの使用による、いかなる損害に対しても、プログラム製作者 (著者) ならびに日本鉄鋼協会は責任を負いませんので、悪しからずご了承願います。

参考文献

- 1) 小山敏幸：ふえらむ, 9 (2004), 240.
- 2) 小山敏幸：ふえらむ, 9 (2004), 301.
- 3) 小山敏幸：ふえらむ, 9 (2004), 376.
- 4) 小山敏幸：ふえらむ, 9 (2004), 497.
- 5) 小山敏幸：ふえらむ, 9 (2004), 905.
- 6) 小山敏幸：まてりあ, 42 (2003), 397.
- 7) 小山敏幸：まてりあ, 42 (2003), 470.
- 8) 小山敏幸：日本機械学会誌, 108 (2005), 794.
- 9) N. Saunders and A.P. Miodownik : CALPHAD, Pergamon, (1998)
- 10) 阿部太一：ふえらむ, 11 (2006) 8, 520.
- 11) 小山敏幸：第180・181回西山記念技術講座, 鉄鋼材料の組織と材質予測技術, (2004), 47.
- 12) J.E. Hilliard : Phase Transformation, ed. by H.I. Aaronson, ASM, Metals Park, Ohio, (1970), 497.
- 13) J.W. Cahn : The Selected Works of J.W. Cahn, ed. by W.C. Carter and W.C. Johnson, TMS, (1998), 29.
- 14) Q. Chen and Z. Jin : Metall.Mater.Trans. A, 26 (1995), 417.
- 15) Z.G. Yang and M. Enomoto : Mater. Sci. Eng. A, 332 (2002), 184.
- 16) T. Harry and D.J. Bacon : Acta Mater., 50 (2002), 195.
- 17) 金属データブック (改定3版), 日本金属学会編, 丸善, (1993)
- 18) 高木節雄：シンポジウム予稿 ナノCu析出を活用した鋼材特性の向上, 日本鉄鋼協会 材料の組織と特性部会・JRCM, (2006)
- 19) 峯村吉泰：Javaによる流体・熱流動の数値シミュレーション, 森北出版, (2001)
- 20) 山本芳人：Javaによる図形処理入門, 工学図書, (1998)

(2006年6月22日受付)