基底状態スピングラスイジングモデルの秩序変数連立方程式化とその数値解

― 複雑系、相互作用系ネットワーク理論と組み合わせ最適化問題、P=NP 問題解法への言及 ―

斉藤　晃（akira311049@gmail.com）

要約

基底状態においてスピングラスイジングモデルの秩序変数を連立方程式で表すことができた。この連立方程式は基底状態のみで成り立つ式であるが、食物連鎖や組織、経済の社会現象などをモデル化する相互に影響を及ぼし合うネットワーク系から、経済ロスを最小化し、複雑な条件からあらゆる最適なマッチングを実現する、組み合わせ最適化問題まで、基底状態を正確に把握することで得られる恩恵は幅広い。また、スピングラスイジングモデルが基本的なモデルであるがゆえに、複雑系や P≠NP 問題等[1]、自然界や数学的な問題への言及も可能となる。ここではそれらの問題の基盤として、そして組み合わせ最適化問題や複雑系など社会現象、社会問題を解析する手法として、基底状態スピングラスイジングモデルの解法としての、連立方程式化と連立方程式解の数値計算結果を示す。

1　前書き

イジングモデルは最もシンプルで基本的な相互作用モデルである。お互いのスピン（ノード）が（-1or1）または（0or1）（以後 0or1 で述べる）によって温度 T による t=1/(Tk)（k はボルツマン定数）で定義される t（時間発展のパラメーター）と相互作用エネルギーがかけられ、初期値 t=0 から始まり、無限遠 t=∞の基底状態と呼ばれる状態に帰着する。この基底状態は系のエネルギーが最も低く、秩序変数は 0or1 をとり、相互作用の結果、系が最終どのような状態に落ち着くかを表す。複雑系では相互作用の結果としての状態であり、組み合わせ最適化問題では解となる。現実の世界では相互作用系の最終結果や組み合わせ最適化解はあらゆる分野で使うことができる。食物連鎖の結果、広告が集団に与える影響、組織のパフォーマンスやそのためのチーム編成、薬の副作用や相互影響、経済や投資のモデル化、社会問題とその解決策などの相互作用系や、複雑化した条件下の配送ルートや荷積みなどの運送運輸、条件が多様で把握が困難だった人員配置やシフト作成、人材育成・教育などの分野、情報や人とのマッチングとしてのマーケティングや広告、検索エンジンや SNS などの情報産業、莫大な数の組み合わせから目的の組み合わせを得る、創薬や遺伝子解析、品種改良などの生物化学農業分野等、ロスをなくし生産性を向上させる組み合わせ最適化分野があげられる。スピングラスイジングモデルはイジングモデルの中でも各相互作用（正負が混ざった）をランダムに（個別に指定して）取るモデルであり[1]、様々な相互作用系や組み合わせ最適化問題をモデル化することができる。このスピングラスイジングモデルの基底状態の解を連立方程式化することができた。連立方程式は数値計算で解を数値的に得ることができるので、連立方程式のソルバー（解を得るアルゴリズム）の精度と速

度で多様な問題を解くことができることに等しい。以下ではその連立方程式化と python ライブラリでの連立方程式数値計算結果を述べる。

2　結果

$1/k_B T \equiv t = \infty$ （基底状態）においてスピングラスイジングモデルの秩序変数を連立方程式で表せた。（状態変数 n_i は 0 または 1 ）

$$f_{i(\langle n \rangle)} = \left(e^{-\lambda_i^{(+)} t} + e^{-\lambda_i^{(-)} t}\right) \langle n_i \rangle - e^{-\lambda_i^{(-)} t} = 0 \quad (1)$$

$$-\lambda_i^{(+)} = -\left|\varepsilon_{ii}^{(+)}\right| - \sum_{j \neq i} \left|\varepsilon_{ij}^{(+)}\right| \langle n_j \rangle$$

$$-\lambda_i^{(-)} = -\left|\varepsilon_{ii}^{(-)}\right| - \sum_{j \neq i} \left|\varepsilon_{ij}^{(-)}\right| \langle n_j \rangle$$

$\varepsilon_{ij}^{(+)}$は正のε_{ij}、$\varepsilon_{ij}^{(-)}$は負のε_{ij}である。

$\langle n \rangle$ についての連立方程式の解は、$1/k_B T \equiv t = \infty$ （基底状態）でのスピングラスイジングモデルの秩序変数の値であり、組み合わせ最適化問題の解である。

(1)式は数値計算によって解くことができる。実際に python の SciPy などの科学技術計算ライブラリには連立方程式を解くソルバーが複数用意されており、連立方程式を指定するだけでその解を高精度に得ることができる。連立方程式化できたため、相互作用系ネットワークや組み合わせ最適化問題の解はその連立方程式ソルバーの精度と速度で得ることができる。

3　理論

N 個の各ノード $i = 1 \sim N$ に以下の状態を持つ状態変数があるとする。

$$n_i = 0 \; or \; 1 \quad (2)$$

系のハミルトニアンが以下で表せるモデルをイジングモデルという[1]。（イジングモデルは一般に $n_x = -1 \; or \; 1$ だが簡単な計算で $n_x = 0 \; or \; 1$ として扱うことができる）

$$-H = \sum_{i \leq j} \varepsilon_{i,j} n_i n_j \quad (3)$$

ただし、$\varepsilon_{i,j}$は以下の範囲の任意の値を取る（スピングラスイジングモデル[1]）。

$$\varepsilon_{i,j} = -1 \sim 1 \quad (4)$$

秩序変数 $\langle n_x \rangle$ は以下で求まる。

$$\langle n_i \rangle = \frac{Z_i}{Z} \quad (5)$$

$$Z = \sum_{\{n\}} e^{-Ht} \;, Z_i = \sum_{\{n\}} n_i e^{-Ht} \;, t = \frac{1}{k_B T} \quad (6)$$

(5)式、(6)式から

$$\langle n_i \rangle = \langle \bar{n}_\imath e^{\varepsilon_{i,i} t} \prod_{j=1 \neq i}^{N} e^{\varepsilon_{i,j} n_j t} \rangle \quad (7)$$

$n_j = 0 or 1$ なので以下と同等である。

$$\langle n_i \rangle = \langle \bar{n}_\imath e^{\varepsilon_{i,i} t} \prod_{j=1 \neq i}^{N} \left((e^{\varepsilon_{i,j} t} - 1) n_j + 1 \right) \rangle \quad (8)$$

右辺はn_jの掛け算の期待値となるが、$t = \infty$では $\langle n_a n_b \rangle = \langle n_a \rangle \langle n_b \rangle$ であるので、

$$\langle n_i \rangle = \langle \bar{n}_\imath \rangle e^{\varepsilon_{i,i} t} \prod_{j=1 \neq i}^{N} \left((e^{\varepsilon_{i,j} t} - 1) \langle n_j \rangle + 1 \right) \quad (9)$$

$\langle \bar{n}_\imath \rangle = 1 - \langle n_i \rangle$ と $n_j = 0 or 1$ より、

$$\langle n_i \rangle = (1 - \langle n_i \rangle)\, e^{\varepsilon_{i,i} t} \prod_{j=1 \neq i}^{N} e^{\varepsilon_{i,j} \langle n_j \rangle t} \quad (10)$$

$\varepsilon_{ij}^{(+)}$ を正のε_{ij}、$\varepsilon_{ij}^{(-)}$ を負のε_{ij}とし、以下の $-\lambda_i^{(+)}$、$-\lambda_i^{(-)}$ を導入する。

$$-\lambda_i^{(+)} = -\left|\varepsilon_{ii}^{(+)}\right| - \sum_{j \neq i} \left|\varepsilon_{ij}^{(+)}\right| \langle n_j \rangle \quad (11)$$

$$-\lambda_i^{(-)} = -\left|\varepsilon_{ii}^{(-)}\right| - \sum_{j \neq i} \left|\varepsilon_{ij}^{(-)}\right| \langle n_j \rangle \quad (12)$$

これを用いて式を整理すると以下を得る。

$$f_{i(\langle n \rangle)} = \left(e^{-\lambda_i^{(+)} t} + e^{-\lambda_i^{(-)} t} \right) \langle n_i \rangle - e^{-\lambda_i^{(-)} t} = 0 \quad (13)$$

(13)式はすべての $i = 1 \sim N$ について成り立ち、N個の $\langle n_i \rangle$ の連立方程式である。つまり $t = \infty$（基底状態）での秩序変数 $\langle n_i \rangle$ を連立方程式で表すことができた。

4　連立方程式数値計算

（13)式により、 $t = \infty$（基底状態）において秩序変数 $\langle n_i \rangle$ を連立方程式で表すことができた。これにより連立方程式の数値計算でスピングラスイジングモデルの解を求めることがでる。解の精度と速度はその連立方程式のソルバーの精度と速度による。python の科学技術計算ライブラリ SciPy を使ってサイト数（ノード数）N=6 で厳密解と比較した結果が以下である。

【N＝6 で 1000 回試算した結果】

サイト（ノード）数	正解率	エネルギー比率
6	98.66%	99.97%
12	98.33%	99.76%

5　考察

基底状態においてスピングラスイジングモデルの秩序変数を連立方程式で表すことができた。これにより連立方程式ソルバーの持つ精度や時間でイジングモデルの解を得ることができる。実際に数値計算をしたところ系のエネルギーが近い値での解を得ることができた。組み合わせ最適化問題について述べると、他のソルバーとは違い、どんな複雑で多様な組み合わせ条件でも、すべて相互作用の重ね合わせにより表現でき、ソルバーのパフォーマンスに影響しないという汎用的な特徴がある。例えば、配送の最適化問題で地域特性や天気、ドライバーの経験等、どんな条件も足し合わせて、すべて相互作用としてまとめ上げることができ、同様の計算速度や精度で解を得ることができる。汎用ソルバーであるからこそ、複雑化する現代ビジネス課題、社会課題の解決につながるものと期待している。

また、数学の未解決問題で、P≠NP 問題がある。P≠NP 問題は、コンピュータ科学の未解決問題で、すべての問題に対してその解が効率的に確認できるならば、その解は同様に効率的に見つけることができるかという問いを提起している。ここで「効率的」とは、問題の大きさに対して多項式時間内に計算できることを意味する。「多項式時間」は、問題を解くアルゴリズムの効率性を表す指標で、問題の大きさ n に対し n の k 乗（k は定数）の計算時間がかかるアルゴリズムを指す。つまり、P≠NP 問題は、「解の存在を高速に確認できる問題（NP）の全てが、高速に解を見つけることが可能な問題（P）に等しいか否か」を問うている[1]。今、この問題に対して、スピングラスイジングモデルの基底状態を連立方程式で表すことができた。スピングラスイジングモデルの基底状態を求める問題はすべての NP 問題に変換可能な NP 完全問題と同等であるため[1]、この連立方程式が「多項式時間」で厳密に解けるなら、P=NP であることを示せる可能性も見えてきた。

以上のように、スピングラスイジングモデルの基底状態での連立方程式化は、今まで困難だった様々な分野でのビジネス、社会課題の解決手段として大きな可能性を秘めている。ご興味をお待ちの方は下記メールアドレスまでご連絡いただきたい。

【問い合わせ】

e-mail	宛先
akira311049@gmail.com	齊藤　晃

6　引用文献

[1] 西森 秀稔 岩波書店 新物理学選書 スピングラス理論と情報統計力学 2016,2,10