

相变的模拟及其可视化

汤子凡¹, 李向华¹

(1. 云南大学 物理与天文学院, 云南 昆明 650504)

摘要: 本文主要利用 Metropolis 算法和 Wolff 算法模拟二维 Ising 模型的顺磁-铁磁相变, 计算了模型的磁化强度、能量和比热随温度的变化曲线。对比两个算法的模拟结果, 发现 Wolff 算法精度和收敛速度比 Metropolis 算法更佳。在体系尺寸变大时, Wolff 算法可以牺牲速度提高精度, 而 Metropolis 则不能。Wolff 算法的多粒子翻转策略使其演化效率更高。

关键词: Ising 模型; Monte Carlo 模拟; Wolff 算法; Metropolis 算法

中图分类号: O 4-1

文献标识码: A

文章编号: 1000-0712 XXXXXX

【DOI】 10.16854/j.cnki.1000-0712. XXXXXX

相, 被用来称作物质的一种存在形式, 而物质可以从一种存在形式变化到另一种显著不同的存在形式, 称为相变。相变分为经典相变和量子相变。常见的相变都是经典相变, 是由热力学参量发生变化导致的, 例如随外界压强和温度的变化, 水会发生固-液-气相变。而量子相变则是通过改变掺杂、外磁场和外电场等参数影响物质。量子相变多在零温下才能观测到, 因为低温才能抑制物质分子的热涨落。

对经典相变研究已经较为成熟。磁性物质的顺磁-铁磁相变颇具代表性, 其开端可以追溯到 19 世纪末, P.Curie 发现当磁石加热到一定温度时, 原来的磁性就会消失, 其温度被命名为材料的 Curie 里温度。后来, Wilhelm Lenz 在 1920 年提出了一种描述铁磁性物质的内部的原子自旋状态及其与宏观磁矩的关系的模型。1924 年, Lenz 的学生 Ising 求解了该模型, 发现这种模型不会发生相变^[1], 这模型被后人称为一维伊辛模型, 是最简单的统计力学模型之一。20 世纪 30-40 年代, Bragg 等使用平均场近似理论对二维 Ising 点阵模型进行了研究^{[2][3]}。1933 年, Ehrenfest 提出了相变的分类标准, 一级相变即相变点附近体系的热力学函数连续, 但对温度的一阶导数不连续或发散; 二级相变即热力学函数对温度的一阶导数连续, 二阶导数不连续或者发散^[4]。1937 年, Landau 建立了描述二级相变的对称性破缺理论^[5]。在这理论框架下, 磁化强度被认为是 Ising 模型的序参量。高温下, 每个格点的自旋磁矩由于热运

动而具有指向无序性, 这时宏观磁矩没有任何方向特殊性。但在低温下, 由于自旋取向时体系的能量最低, 所以自旋之间的相互作用使宏观磁矩具有了某个空间方向的指向 (例如 +1), 并且体系不能自发地从该指向跃变到另一个指向 (例如 +1 到 -1), 所以序参量的空间对称性被打破了。

1944 年 Onsager 严格求解了二维 (2D) Ising 模型在没有外磁场时的解析解, 即 Onsager 解^[6], 当温度大于临界温度时, 体系呈现高温顺磁性, 反之呈现低温铁磁性, 其能量的二阶导数 (即比热的一阶导数) 不连续, 是二级相变。自此之后, 研究三维伊辛模型中的相变临界点和相的分类, 成为统计物理中一项非常重要的任务, 但多年的研究一直没有取得突破性进展。

统计物理所研究的对象是微观的、具有阿伏伽德罗常数量级 ($\sim 10^{23}$ 个) 的系统, 如此多的粒子, 自然无法用经典力学的方式求解每一个粒子的运动方程, 所以系综理论被建立了起来。另一方面, 随着计算机的发展, 人们自然想通过计算机模拟粒子的行为, 以此验证 2D Ising 模型以及给出 3D Ising 模型的大致图景。但是, 在计算机模拟中, 如果想要通过较少量的粒子体系展现大量粒子的行为, 概率论必须介入。所以, 统计物理所研究的模型天然是 Monte-Carlo 方法 (MC) 大显身手的舞台。本文主要利用 Metropolis 算法^[7] 和 Wolff 算法^[8] 求解无外场下的 2D Ising 模型, 并将二者的结果对比分析。

收稿日期: 2023-6-25; 修改日期: 2023-6-25

基金项目: XXXXXX 基金 (xxxxxxx) 资助

作者简介: 汤子凡 (1999—), 男, 云南昆明人, 云南大学物理系本科生。 (第一作者)

通信作者: 李向华, E-mail: xhli@ynu.edu.cn

1 无外场下的 2D Ising 模型

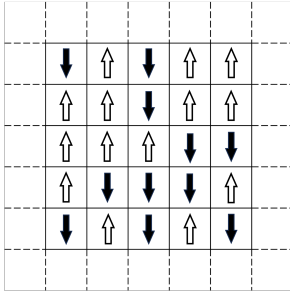


图 1 Ising 模型示意图

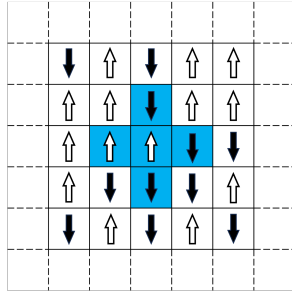


图 2 Von Neumann 邻居

2D Ising 模型是描述磁性系统相变的简化模型. 如图 1 所示, 设体系有 N 个周期性排列的粒子 (或格点), 每个格点的自旋 s_i 只有两个取向: 向上 (定为 +1) 或向下 (定为 -1). 理论上, 每个粒子自旋会受到该系统中所有粒子自旋的共同影响, 随距离增加, 相互影响逐渐降低. 简单起见, 假设在二维情形下, 每个粒子只受到其 Von Neumann 邻居即上下左右四个格点的影响^[9] (见图 2). 再假设每对粒子的相互作用能为 $-J$ (只讨论铁磁系统, $J > 0$), 则写出该系统的 Hamilton 量为

$$H = -J \sum_{\langle i, j \rangle} s_i s_j \quad (1)$$

其中 $\langle i, j \rangle$ 表示互为近邻. 由于自旋同向时系统能量最低, 所以当 $s_i \cdot s_j > 0$ 时, $H > 0$, 反之小于零. 对系统所有格点的 (正则) 配分函数求和即是体系的配分函数,^[10]

$$Z_N = \sum_{s_1} \sum_{s_2} \cdots \sum_{s_N} e^{-H/kT} = \sum_{s_i} e^{-H/kT}, \quad (2)$$

其中 $\{s_i\}$ 代表体系某一组自旋配置, 对应于系统的一个可能的 Hamilton 量. 磁化强度写为

$$\bar{M} = N\mu\bar{s}, \quad (3)$$

, 其中 \bar{s} 是体系的自旋平均值,

$$\bar{s} = \frac{1}{N} \sum_{i=1}^N s_i \quad (4)$$

可见磁化强度正比于体系的自旋平均值. 简单起见, 本文将序参量选为 \bar{s} .

Onsager 给出的精确解是自旋为^[6]

$$\bar{s} = \begin{cases} 0, & T > T_c \\ [1 - \sinh(2\beta J)^{-4}]^{1/8}, & T \leq T_c \end{cases} \quad (5)$$

能量为

$$\bar{E} = -J \cot(2\beta J) \times \left[1 + \frac{2}{\pi} A \cdot B(\lambda) \right]. \quad (6)$$

其中

$$\begin{cases} A \equiv 2 \tanh(2\beta J)^2 - 1 \\ B(\lambda) \equiv \int_0^{\pi/2} \frac{d\phi}{\sqrt{1 - \lambda^2 \sin^2 \phi}} \\ \lambda \equiv \frac{2 \sinh(2\beta J)}{\cosh(2\beta J)^2} \end{cases} \quad (7)$$

\bar{s} 和 \bar{E} 随温度 T 的变化曲线绘制在了图 4, 图 11 等图中.

2 方法

在上一节中提到的 N 个格点的自旋系统, 其所有可能的自旋配置有 2^N 种, 想要计算所有情况是不可能的. 更重要的是, 按照配分函数表达式, 随着状态能量的升高, 某一状态 μ 实际出现的概率 $p_\mu = e^{-E/kT}$ 是指数衰减的. 所以, 应该按照不同状态出现的概率来确定我们的抽样密度.

2.1 Metropolis 算法 (Single-Spin-Flip Dynamics)

Metropolis 算法模拟 Ising 模型相变的过程如下^[11]

1. 在给定温度下, 随机给定一套初始自旋配置

$$\{s_i\} \equiv s_1, s_2, \dots, s_N \quad (8)$$

2. 利用随机数, 随机选择一个格点, 计算翻转这个格点带来的体系总能量变化

$$\begin{aligned} \Delta E &= E_{\text{final}} - E_{\text{initial}} \\ &= \left(-J(-\sigma_m) \sum_n \sigma_n \right) - \left(-J\sigma_m \sum_n \sigma_n \right) \\ &= 2J\sigma_m \sum_n \sigma_n \end{aligned} \quad (9)$$

3. 如果能量差 $\Delta E < 0$, 说明翻转该自旋将造成能量降低, 那么翻转该格点自旋; 如果能量差大于 0, 以接受概率 $A = e^{-\Delta E/kT} \in [0, 1]$ 接受该自旋翻转, 即产生一个随机数 r , 如果 $r \leq A$, 则接受, 反之则不接受;
4. 重复第 2, 3 步足够长 (称为 Markov 链长), 然后得到体系自旋在该温度下的最终配置.

5. 改变温度, 重复第 24 步.

需要注意的是, 这循环有两层, 最外层改变环境温度, 内部层在改变抽样格点. 尽管微观上讲, 具体的某个格点的自旋每次采样时可能会继续变化, 但是宏观热力学量 (如能量和自旋平均值) 最终将收敛到一个固定值. 随机采样并模拟 Ising 模拟演化的过程实际上是一个 Markov 过程. Markov 过程要求满足两点^[12]

1. 各态历经 (Ergodicity): 假设时间足够长, 无论从哪一个状态出发, 都有概率演化到系统其他所有状态. 事实上这是系综理论的最基本假设, Ising 模型的演化过程自然满足该条件.
2. 细致平衡 (Detailed Balance): 对于一个态 ν , 其出现的概率 p_ν 和其演化为另一态 μ 的概率 $P(\nu \rightarrow \mu)$ 的乘积, 等于态 μ 出现的概率乘以其演化为态 ν 的概率 $P_{\mu \rightarrow \nu}$, 即

$$p_\mu P(\mu \rightarrow \nu) = p_\nu P(\nu \rightarrow \mu) \quad (10)$$

代入 p_μ, p_ν 的表达式, 得

$$\frac{P(\mu \rightarrow \nu)}{P(\nu \rightarrow \mu)} = \frac{p_\nu}{p_\mu} = e^{-\beta(E_\nu - E_\mu)} \quad (11)$$

而演化概率 $P_{\mu \rightarrow \nu}$ 又拆分成两部分: 选择概率 $g_{\mu \rightarrow \nu}$ 和接受概率 $A(\mu \rightarrow \nu)$.

$$P(\mu \rightarrow \nu) = g(\mu \rightarrow \nu)A(\mu \rightarrow \nu) \quad (12)$$

在 Metropolis 算法中, 选择其他任一个 (自旋格点相差正好为 ± 1) 状态的概率 g 被简单的定为相等, 即

$$g(\mu \rightarrow \nu) = \frac{1}{N}. \quad (13)$$

联立上式, 可得

$$\begin{aligned} \frac{P(\mu \rightarrow \nu)}{P(\nu \rightarrow \mu)} &= \frac{g(\mu \rightarrow \nu)A(\mu \rightarrow \nu)}{g(\nu \rightarrow \mu)A(\nu \rightarrow \mu)} \\ &= \frac{A(\mu \rightarrow \nu)}{A(\nu \rightarrow \mu)} \\ &= e^{-\beta(E_\nu - E_\mu)} \end{aligned} \quad (14)$$

理论上, 可以任取 A 的值. 但 Metropolis 给出较优的取值为

$$A(\mu \rightarrow \nu) = \begin{cases} e^{-\beta(E_\nu - E_\mu)} & , \text{如果 } E_\nu - E_\mu > 0 \\ 1 & , \text{其它} \end{cases} \quad (15)$$

即如果 $\Delta E = E_\nu - E_\mu > 0$, 则以概率 $e^{-\beta\Delta E}$ 翻转. 如果 $\Delta < 0$, 则 $A(\mu \rightarrow \nu) = e^{-\beta\Delta E} > 1$, 由于概率不可能大于 1, 所以取为 1. 这被称为 Metropolis 接受准则. 这样取的原因是, 在该准则下每个格点的翻转概率要更大多, 更高的翻转概率意味着更高的演化效率.

由于 Metropolis 算法每次只考虑单个粒子的翻转与否, 所以这种算法也被称为“单粒子翻转算法”, 其缺点是

1. 这算法收敛的相当慢, 需要更长的 Markov 链.
2. 由于每次只翻转一个粒子, 会出现某一个态无论翻转哪一个自旋都会抬高能量, 即容易掉入亚稳态并难以跳出来, 这在机器学习里被称为学习率太低而陷入局部最优解.

2.2 Wolff 算法 (Cluster-Flip Dynamics)

由于单粒子翻转算法的缺点, 人们在 Metropolis 算法的基础上加以改进, 得到了多粒子翻转算法. 其中 Wolff 算法的接受度最高, 其和 Metropolis 算法的区别在于每次考虑是否翻转时, 选择一个初始格点 (称为种子), 然后再考虑是否将周围的同自旋格点加入 (称为簇的生长). 设翻转前为 μ 态, 翻转后为 ν 态. 在 μ 态时, 将同自旋的格点加入该簇的概率为 P_{add} . 那么当簇形成后 (设簇含 c 个格点), 周围那些自旋同向但未被加入的格点 (设有 m 个) 的概率为 $(1 - P_{\text{add}})^m$. 反之, 当该簇翻转形成 ν 态后, (对于同一个种子) 同一个簇形成后, 未加入同向格点 (设有 n 个) 的概率是 $(1 - P_{\text{add}})^n$. 那么选择概率为

$$g_{\mu \rightarrow \nu} = (P_{\text{add}})^c (1 - P_{\text{add}})^m, \quad (16)$$

$$g_{\nu \rightarrow \mu} = (P_{\text{add}})^c (1 - P_{\text{add}})^n. \quad (17)$$

由细致平衡条件

$$\begin{aligned} \frac{P(\mu \rightarrow \nu)}{P(\nu \rightarrow \mu)} &= \frac{g(\mu \rightarrow \nu)A(\mu \rightarrow \nu)}{g(\nu \rightarrow \mu)A(\nu \rightarrow \mu)} \\ &= (1 - P_{\text{add}})^{m-n} \frac{A(\mu \rightarrow \nu)}{A(\nu \rightarrow \mu)} \\ &= e^{-\beta(E_\nu - E_\mu)} \end{aligned} \quad (18)$$

从 $\mu \rightarrow \nu$, 能量增加了 $+2Jm$, 从 $\nu \rightarrow \mu$ 能量改变了 $-2Jn$. 所以 $\Delta E = E_\nu - E_\mu = 2J(m - n)$, 上式

成为

$$\frac{A(\mu \rightarrow \nu)}{A(\nu \rightarrow \mu)} = [e^{2\beta J} (1 - P_{\text{add}})]^{n-m} \quad (19)$$

为简化结果, Wolff 令

$$P_{\text{add}} = 1 - e^{-2\beta J} \quad (20)$$

这样, $A(\nu \rightarrow \mu) = A(\mu \rightarrow \nu) = 1$, 即每次只要选定了种子, 一定会有格点被翻转. 这一过程的示意图见图 3.

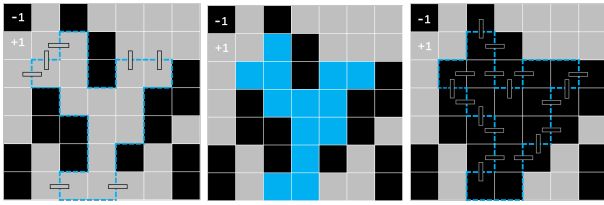


图 3 Wolff 算法的簇生长示意图.

灰色: 自旋为 +1, 黑色: 自旋为 -1. 在最左图中, 蓝色线代表某个簇的生长结果, 可见有 7 个点没有被加入簇. 在右图中, 同样的簇则有 18 个点未被加入, 同自旋之间的带表示“键”. 当簇翻转时键将被破坏, 释放能量, 能量差为

$$\Delta E = 2J(7 - 18) = -22J.$$

可见, Wolff 算法不仅翻转的接受概率要比 Metropolis 更高, 而且翻转是以簇为单位进行的, 这大大提高了计算机模拟的演化效率. Wolff 算法的一般步骤如下^[13]

1. 在给定温度下, 随机给定一套初始自旋配置;
2. 随机选取种子;
3. 令簇生长: 考虑种子的所有近邻, 以概率为 $P_{\text{add}} = 1 - e^{2\beta J}$ 加入簇.
4. 对新加入的格点, 重复第 3 步, 直到没有新的格点加入, 此时簇停止了生长.
5. 翻转簇.
6. 重复第 2-4 步足够长.
7. 改变温度, 重复第 2-6 步.

3 结果与分析

在 MATLAB 上编写 Metropolis 算法和 Wolff 算法模拟 2D Ising 模型随温度变化的演化, 计算其序参量 \bar{s} 、能量 E 和比热 C_V .

3.1 序参量

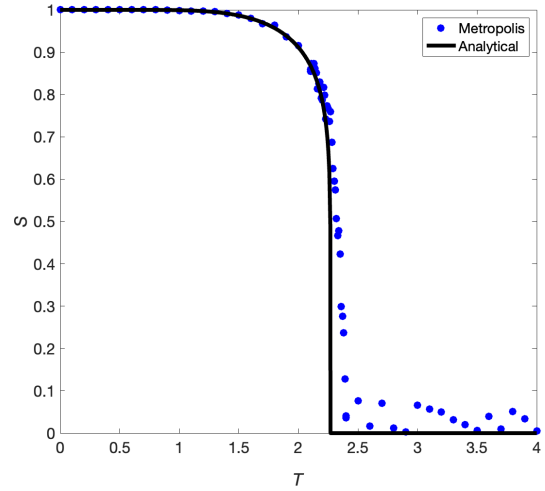


图 4 Metropolis 算法, $\bar{s} - T$ 曲线图.

体系大小 $L = 100$, Markov 链长 $l_M = 500k$, 用时 $t = 159s$.

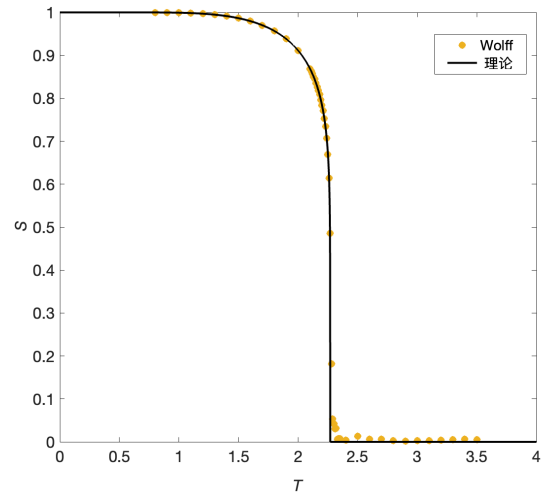


图 5 Wolff 算法, $\bar{s} - T$ 曲线图.

体系大小 $L = 400$, 取样次数 $l_w = 1000$, 用时 $t = 20$ 小时

模拟结果较好的是图 4 和图 5, 参数见图注. Metropolis 算法在温度高于临界点 $T_c = 2.2692$ 后的模拟精度十分糟糕, 对比可知, Wolff 算法有着更高的精度. 事实上, 对于 Wolff 算法, 增加体系的尺寸可以显著提高精度 (图 8, 图 9). 而 Metropolis 算法若增加尺寸则误差越来越大, $L = 40, 400$ 的模拟见图 6 和图 7. 理论上, Ising 模型没有边界, 是无限大的体系, 所以对于计算机模拟, 增大尺寸应该得到更好的模拟精度, 对 Wolff 算法来说确实如此, 但是 Metropolis 算法却不是. 这说明 Metropolis 的单粒子

翻转策略在大尺寸下演化效率确实有限, 无法穷尽所有的可能状态, 所以计算统计值会出现误差.

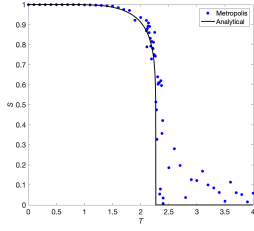


图 6 Metropolis, $L=40$,
 $l = 500k$

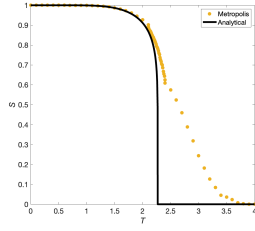


图 7 Metropolis, $L=400$,
 $l_M = 500k$

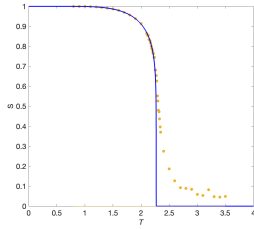


图 8 Wolff, $L = 40$,
 $l_W = 1000, t = 109s$, 高于
临界点的值偏高

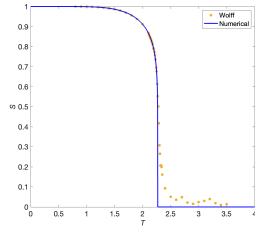


图 9 Wolff, $L = 100$,
 $l_W = 1000, t = 609s$, 比
 $L = 40$ 时误差低

在小尺寸下 Metropolis 算法和 Wolff 算法结果都比较糟糕, 不同之处在于 Metropolis 是杂乱的随机误差, 是没有统计到尽可能多的状态造成的; 而 Wolff 算法的系统误差, 是体系尺寸太小造成的, 增加尺寸会提高精度.

3.2 能量与比热

其他热力学参量, 例如平均能量、比热如图 10, 图 11 所示. Metropolis 算法对能量的模拟有小幅误差, 这误差体现在对能量求导后被放大. 相比之下, Wolff 算法的精度要好很多. 从比热的变化曲线来看, 能量对温度的一阶导数连续, 但是在 $T = T_c$ 处, 比热的二阶导数显然不连续, 这也验证了 Ising 模型的顺磁-铁磁相变属于二级相变.

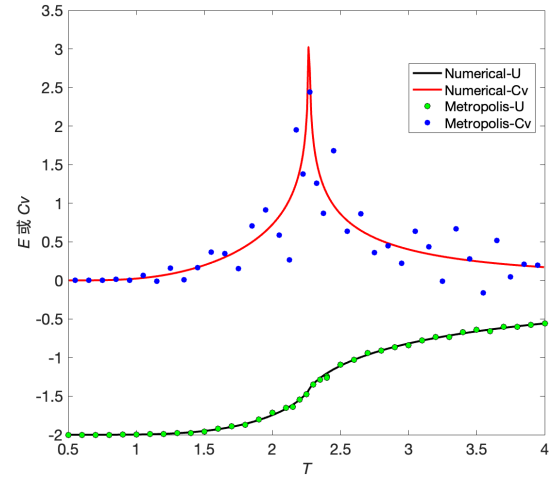


图 10 Metropolis $E/C_V - T$ 曲线. $L = 100, l_m = 500k$,
 $t = 108s$, 能量的小误差反映在比热上被放大

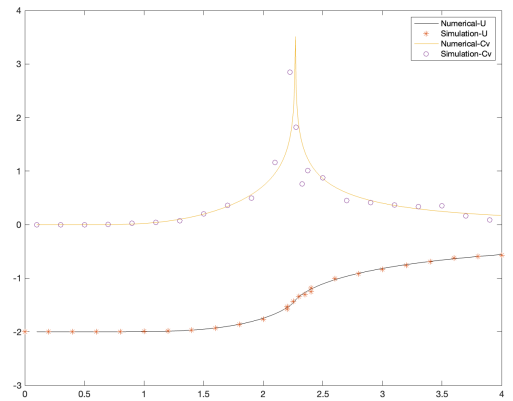


图 11 Wolff $E/C_V - T$ 曲线. $L = 400, l_M = 1000, t = 20h$

3.3 远离临界点时的收敛速度

以自旋平均值 \bar{s} 为例, 对比二者在 $T = 1$ (图 12, 图 13) 和 $T = 3$ (见图 14, 图 15) 的收敛情况, 不难发现同样的体系尺寸下 Metropolis 算法都需要比 Wolff 算法更多的迭代次数才收敛 (注意 Metropolis 收敛曲线的横坐标单位是 10^5), Metropolis 用时也长于 Wolff. 在 $L = 400$ 时, 曾尝试使用 Metropolis 算法迭代更多次数, 二百五十六万步时仍然不收敛. 这些结果有力地说明粒子的多次翻转不等效于多粒子 (簇) 的单次翻转, 又一次证明了 Wolff 算法的多粒子翻转策略演化效率远胜于单粒子翻转.

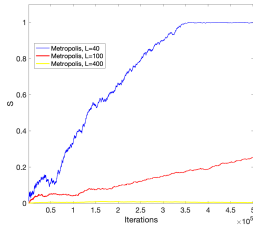


图 12 Metropolis 收敛曲线.
 $T = 1, L = 100, t = 44s$

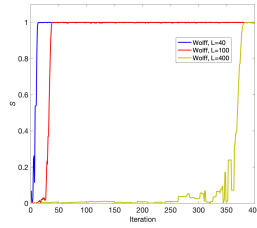


图 13 Wolff 收敛曲线.
 $T = 1, L = 400, t = 76s$

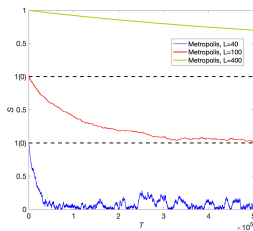


图 14 Metropolis 收敛曲线.
 $T = 3, L = 100, t = 46s$

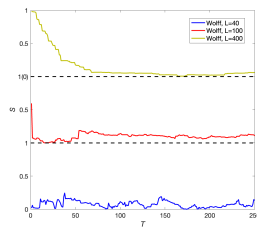


图 15 Wolff 收敛曲线.
 $T = 3, L = 400, t = 0.9s$

3.4 临界点附近的稳定性

Wolff 算法在大尺寸时耗时最多的是在临界点附近, 由于 Metropolis 算法在临界点附近的模拟结果误差大于 Wolff 算法, 故对比二者所用时间没有意义, 但可以对比二者在 $T = 2.26$ 时候的自旋平均值 (理论值为 $\bar{s} = 0.6134$)。分别在 $L = 100$ 和 400 时进行模拟 (这是两个算法表现最佳的尺寸大小), 计算自旋均值, 结果如表 1 所示。显然, Wolff 算法的误差更小, 稳定程度远高于 Metropolis 算法。

4 结论与展望

从模拟结果来看, Wolff 算法精度远大于 Metropolis 算法, 如果尺寸趋于无穷, Wolff 算法可以模拟无限趋近于理想的 2D Ising 模型的演化行为, 但 Metropolis 算法误差却会随尺寸增大而增大。所以, Metropolis 算法只能在小尺寸下给出近似结果, 精确模拟需要用 Wolff 算法。

在远离临界温度时, Wolff 算法的收敛速度远快于 Metropolis, 并且可以在有限步数内使大尺寸体系收敛, Metropolis 算法则可能不收敛。

在临界温度附近, Wolff 算法可以模拟格点尽可能多的自旋配置情况, 从而给出一个稳定的序参量值, 虽然所用时间非常长但误差小。而 Metropolis 算

法虽然时间占优, 但稳定性差、误差大。这些结论都说明, 不论是从收敛速度还是计算精度上, Wolff 算法都更胜一筹, Wolff 算法的多粒子翻转策略演化效率远胜于单粒子翻转。

利用 Monte Carol 方法生成的各个温度下的格点自旋配置, 可以作为机器学习模型中的训练数据, 然后训练模型可以自动识别 Ising 模型处于哪种相。^{[14][15]} 这在无法快速用常规方法区分相的相变过程中非常有用。

参考文献

- [1] Ising E. Beitrag zur theorie des ferromagnetismus[J]. Z. physik, 1925, 31: 253.
- [2] Bragg W L, Williams E J. The effect of thermal agitation on atomic arrangement in alloys[J]. Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character, 1934, 145(855): 699-730.
- [3] Peierls R. On ising's model of ferromagnetism[C]// Mathematical Proceedings of the Cambridge Philosophical Society: volume 32. Cambridge University Press, 1936: 477-481.
- [4] Ehrenfest P. Phasenumwandlungen im ueblichen und erweiterten sinn, classifiziert nach den entsprechenden singularitaeten des thermodynamischen potentiales[M]. NV Noord-Hollandsche Uitgevers Maatschappij, 1933.
- [5] Landau L D. On the theory of phase transitions. i.[J]. Zh. Eksp. Teor. Fiz., 1937, 11: 19.
- [6] Onsager L. A two-dimensional model with an order-disorder transition (crystal statistics i)[J]. Phys. Rev, 1944, 65(3-4): 117-149.
- [7] Metropolis N, Rosenbluth A W, Rosenbluth M N, et al. Equation of state calculations by fast computing machines [J]. The journal of chemical physics, 1953, 21(6): 1087-1092.
- [8] Wolff U. Collective monte carlo updating for spin systems[J/OL]. Phys. Rev. Lett., 1989, 62: 361-364. <https://link.aps.org/doi/10.1103/PhysRevLett.62.361>.
- [9] 李晓寒. 二维 Ising 模型重正化群方法及数值模拟分析[D]. 重庆: 重庆大学数理学院, 2005.
- [10] 林宗涵. 热力学与统计物理学[M]. 北京大学出版社, 2007: 465-466.
- [11] Newman M E, Barkema G T. Monte carlo methods in statistical physics[M]. Clarendon Press, 1999: 46-49.

- [12] Newman M E, Barkema G T. Monte carlo methods in statistical physics[M]. Clarendon Press, 1999: 35-42.
- [13] Newman M E, Barkema G T. Monte carlo methods in statistical physics[M]. Clarendon Press, 1999: 91-96.
- [14] Carrasquilla J, Melko R G. Machine learning phases of matter[J]. Nature Physics, 2017, 13(5): 431-434.
- [15] Dong X Y, Pollmann F, Zhang X F, et al. Machine learning of quantum phase transitions[J]. Physical Review B, 2019, 99(12): 121104.

表 1 $T_c = 2.26$ 下两种算法的误差对比

	1	2	3	4	5	平均值	标准差	相对误差
Metropolis	0.7843	0.7469	0.7412	0.7776	0.7719	0.7644	0.0192	0.31%
Wolff	0.6120	0.6207	0.6161	0.6159	0.6119	0.6153	0.0036	24.6%

Simulation of phase transition and its visualization

TANG Zi-fan, LI Xiang-hua

(College of Physics and Astronomy, Yunnan University, Kunming 650504, China)

Abstract: The Metropolis Algorithm and Wolff Algorithm are used to simulate the paramagnetic-ferromagnetic phase transition of Ising Model. Spin, energy and specific heat capacity vs. temperature are calculated. Compare the result, Wolff Algorithm is considered to have a faster convergence speed and a better precision than the Metropolis Algorithm. The precision of Wolff Algorithm could be improved by enlarge the size of the system, but Metropolis could not do so. Wolff Algorithm's cluster-flip strategy give the simulation a higher efficiency.

Key words: Ising Model; Monte Carlo simulation; Wolff Algorithm; Metropolis Algorithm