

一种面向组合优化问题的高并行度模拟绝热分岔处理器

李任龙^① 郝 昕^① 陈卓俊^① 丁 玎^{*②③}

^①(湖南大学半导体学院(集成电路学院) 长沙 410082)

^②(湖南工商大学智能工程与智能制造学院 长沙 410205)

^③(湘江实验室 长沙 410205)

摘 要: 组合优化问题巨大的解空间使得基于穷举搜索的策略难以在可行时间内找到高质量解。受量子启发的伊辛机为加速求解组合优化问题提供了新的计算范式, 但实现兼具高速度、高能效和高精度的CMOS基伊辛机仍然是一项重大挑战。对这一问题, 本文基于65 nm CMOS工艺设计并流片了一款面向组合优化问题的高并行度模拟绝热分岔专用处理器。该处理器利用模拟绝热分岔算法的全并行自旋更新特性, 构建支持全连接拓扑的硬件架构, 并提出自旋流水线更新策略, 将耦合系数访问、动量更新和位置更新划分为三级流水线, 实现无气泡迭代计算。为降低全连接耦合矩阵带来的存储开销, 本文进一步设计折叠式耦合系数存储阵列, 利用矩阵对称性减少冗余存储, 使存储面积降低52%。同时, 动量更新单元和位置更新单元采用定点数实现, 在较低硬件开销下完成模拟绝热分岔演化。测试结果表明, 该芯片在200 MHz下功耗为14 mW, 求解64节点最大割问题仅需要25ns, 求解40子句、8变量的SAT问题不超过16 μ s, 验证了所提架构在组合优化加速中的有效性。

关键词: 组合优化问题; 模拟绝热分岔; 伊辛机; 专用集成电路

中图分类号: TN492

文献标识码: A

文章编号: 1009-5896(2026)00-0001-11

DOI: 10.11999/JEIT260780

CSTR: 32379.14.JEIT260780

1 引言

组合优化问题(COPs)广泛应用于各类实际场景, 包括网络优化^[1]、路径规划^[2]、超大规模集成电路设计^[3]等实际场景。这类问题的一个决定性特征是: 其解空间随问题规模呈指数级增长, 这使得基于传统冯·诺依曼架构的CPU难以在合理时间内找到高质量解。因此, 开发高效的COPs求解方法将对相关技术产生显著推动作用^[4,5]。目前已提出多种元启发式方法, 例如蚁群优化算法^[6]、遗传算法^[7]以及基于伊辛模型的模拟退火算法^[8-11]。伊辛模型是统计物理学中的经典模型, 由相互连接的自旋构成, 最初用于描述铁磁材料中的相变现象。由于该模型能够模拟自然、社会和人工系统中普遍存在的复杂现象, 因此被用于表征各类组合优化问题。伊辛模型的哈密顿量(H)代表系统总能量, 而局域哈密顿量(H_i)代表单个自旋的能量, 二者由以下方程给出:

$$H = - \sum_{i=1}^n \sum_{j=1}^n J_{i,j} \sigma_i \sigma_j + h_i \sigma_i \quad (1)$$

$$H_i = \sum_{i=1}^n J_{i,j} \sigma_i + h_i \quad (2)$$

组合优化问题可以通过耦合系数和外加磁场映射到伊辛机上^[12]。随后, 伊辛机通过执行搜索操作来生成一个相对最优的解。这一过程的工作流程如图1(a)所示。图1(b)展示了一个包含64个节点的全连接无向图, 该图对应于图1(c)所示的伊辛连接矩阵, 其中 $J_{i,j}$ 表示第 i 个自旋与第 j 个自旋之间的耦合强度, σ_i 表示第 i 个自旋的状态。

基于伊辛模型的模拟退火算法是一种启发式算法, 用于模拟物理热退火过程。它在传统爬山启发式算法的基础上引入了随机性, 使系统能够逃离局部极小值; 随着系统冷却(类似于温度下降), 这种随机性逐渐降低。当应用于伊辛模型时, 模拟退火算法会根据能量变化, 在每次迭代中随机选择一个自旋并翻转其状态。为确保收敛, 相邻的自旋不能同时更新。在全连接伊辛模型中, 每个自旋都与其他所有自旋相互作用, 因此自旋必须顺序更新, 这显著增加了求解时间。此外, 模拟退火算法通常

需要吉布斯采样来帮助系统逃离局部最优解; 若不采用吉布斯采样, 求解精度可能会受到影响。尽管已有研究利用近似计算电路来简化计算, 但对随机数生成器的依赖仍然是面积开销的主要来源。

收稿日期: 2026-06-09; 改回日期: 2026-07-13; 网络出版: 2026-07-23

*通信作者: 丁玎 dingding@hutb.edu.cn

基金项目: 湖南省重大科技攻关项目(2025QK2007), 湖南省教育厅重点项目(25A0459), 湘江实验室重点项目(25XJ02003)

Foundation Items: Major Scientific and Technological Research Project of Hunan Province (2025QK2007), Key Project of the Education Department of Hunan Province (25A0459) and Key Project of Xiangjiang Laboratory (25XJ02003)

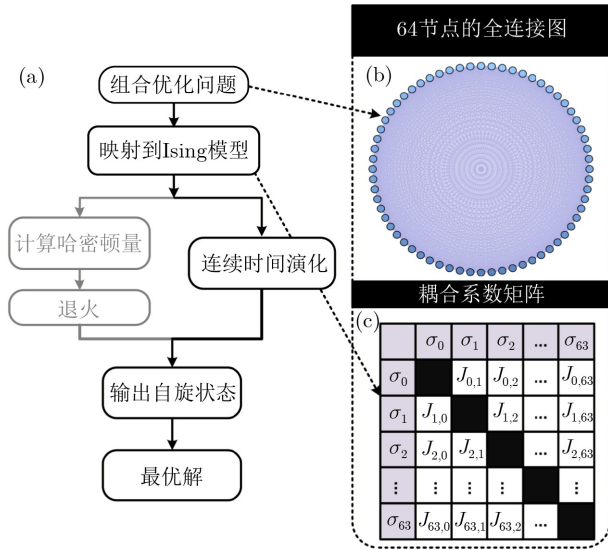


图1 伊辛机工作流程与64节点全连接问题及其伊辛交互矩阵。
(a) 伊辛机工作流程。(b) 64节点全连接问题。(c) 组合优化问题到伊辛模型的映射。

近年来,伊辛机得到了广泛研究,大致可分为两类:连续时间计算与离散时间计算^[13]。离散时间伊辛机基于数学运算来更新自旋。Yamaoka等人^[14]提出了一种退火处理器,其拓扑复杂度为最近五个邻居(在网格图基础上增加垂直连接关系),系数位宽为2比特。Su等人^[15]提出了一种数字存内计算退火处理器,使用寄存器存储自旋状态,允许在单个周期内更新自旋,并实现了较高的计算并行度。为实现片上退火功能^[14, 15],必须外部生成随机数或降低存储器供电电压,以避免陷入局部最小状态。得益于多种退火策略,这类机器在灵活的自旋连接性、大规模自旋集成以及高精度解质量方面表现出色,能够处理复杂的组合优化问题并获得高质量解。然而,其离散时间特性需要数千次迭代更新,带来了显著的开销。此外,它们对随机数生成器的依赖也导致了相当大的芯片面积消耗。

相比之下,连续时间伊辛机更接近地模拟了自然物理现象。它们不通过迭代方式求解哈密顿量;而是通过一个逐渐向基态收敛的自然过程,同时更新所有自旋。I. Ahmed等人^[16]提出了一种环形振荡器伊辛机,耦合环形振荡器的连续时间运行方式具有优势,因为它能显著降低求解组合优化问题所需的能量和时间。J. Bae等人^[17]提出了一种使用锁存器存储自旋状态的伊辛机,并通过耦合开关来改变自旋状态。图2(a)展示了模拟退火伊辛机在连续时间与离散时间计算下的演化过程。离散时间伊辛机能够达到接近最优解的能量状态,但需要较长的求解时间(TTS)。相比之下,连续时间伊辛机具有更快的哈密顿量收敛速度,但往往得到的解质量较差。

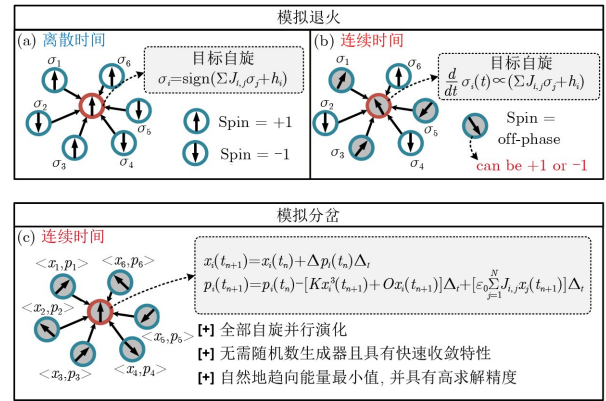


图2 离散时间模拟退火伊辛机、连续时间模拟退火伊辛机与模拟绝热分岔机。(a) 离散时间与连续时间模拟退火伊辛机的计算类型。(b) 模拟绝热分岔机

模拟绝热分岔(Simulated Adiabatic Bifurcation, SAB)是近年来提出的一种用于求解组合优化问题的启发式方法^[18]。该算法受量子绝热优化的启发,利用非线性振荡器网络,通过求解一组常微分方程,数值模拟了呈现分岔现象的经典非线性哈密顿系统的绝热演化。在每个更新周期中,该算法自然地搜索能量最小值,从而保证了较高的求解精度。此外,SAB不需要吉布斯采样来逃离局部最优解。它将每个自旋建模为一个振荡器;在迭代过程中,每个振荡器的下一个位置由其动量、当前位置、所有其他相连振荡器的位置以及耦合系数共同决定。在最后一次迭代之后,通过每个自旋位置的符号即可获得组合优化问题的解。由于具有内在的并行性,SAB算法无需额外的随机数生成器即可高效求解组合优化问题,从而提高了计算效率并降低了硬件开销。

在本工作中,我们采用模拟绝热分岔算法求解组合优化问题,实现了所有自旋状态的同时更新。我们获得了较高的求解精度,并进一步优化了系数的存储阵列。对于64节点的组合优化问题,本方法在无需随机数生成器的情况下,以0.74 μs 的求解时间实现了高质量解,功耗为14 mW。通过利用耦合矩阵的对称性,与全矩阵存储相比,我们将系数存储面积减少了48%。为确保处理不同问题的灵活性,我们的架构支持可配置的系数精度位宽,并能够针对特定问题动态激活不同的系数阵列。当求解对耦合精度要求较低的问题时,该机制有效降低了从静态随机存取存储器读取耦合系数所产生的功耗。

2 模拟绝热分岔算法

模拟绝热分岔算法凭借其固有的并行自旋更新机制,实现了快速收敛。此外,在每个更新周期中,它都会倾向于搜索能量最小值,从而保证了较

高的求解精度，如图2(b)所示。模拟绝热分岔的演化过程由以下常微分方程组控制：

$$H_{SAB} = \sum_{i=1}^n \frac{\Delta p_i^2}{2} + V(x, t) \quad (3)$$

$$V(x, t) = \sum_{i=1}^n \frac{Kx_i^4}{4} + \frac{\Delta - \alpha(t)}{2} x_i^2 - \frac{\epsilon \sum \sum J_{i,j} \sigma_i \sigma_j}{2} \quad (4)$$

$$\frac{\partial x_i}{\partial t} = \frac{\partial H_{SAB}}{\partial p_i} = \Delta p_i \quad (5)$$

$$\frac{\partial p_i}{\partial t} = -\frac{\partial H_{SAB}}{\partial x_i} \quad (6)$$

方程(3)定义了分岔系统的哈密顿量结构。如图所示，系统的总能量由所有振荡器的动量 p_i 所对应的动能以及势能 $V(x, t)$ 共同构成，其中势能 $V(x, t)$ 控制着振荡器位置 x_i 的演化。随后的方程(4)和(5)阐明了每个振荡器的动量与位置之间的动态相互作用，构成了一组常微分方程组。在方程(4)中，超参数 K 和 Δ 均设为1。参数 $\alpha(t)$ 控制着绝热演化过程，其值随时间逐渐增加。每个振荡器动量的迭代变化取决于其自身位置的三次非线性项，并叠加了来自所有其他振荡器的耦合项以及一个阻尼项。每个振荡器通过耦合强度 $J_{i,j}$ 与其他所有振荡器相互作用，该耦合强度精确对应于自旋 i 与自旋 j 之间的耦合系数。通过数值模拟，振荡器的连续状态(位置和动量)被离散化为变量 x_i 和 p_i 。利用耦合系数 $J_{i,j}$ 计算对目标振荡器 i 的累积耦合影响，从而得到其动量变化量 Δp_i 。随后，这一更新后的动量决定了该振荡器在下次迭代中的新位置。这些新位置又作为下一轮全局并行迭代周期的输入。

图3展示了模拟绝热分岔算法的伪代码。对于芯片系统而言，所需的输入包括耦合系数 $J_{i,j}$ 、 N 个节点的随机初始位置以及初始动量。常数 O 和

K 均设为1，此外还需要设定迭代步长 Δt 和总迭代次数 T 。工作流程如下：首先更新每个自旋的动量。这涉及一个乘法运算，该运算使用所有其他自旋的位置以及相应的耦合系数 $J_{i,j}$ 。运算结果加上当前自旋位置的立方(即一个非线性微扰项)，再减去当前自旋位置本身，从而得到动量变化量 Δp 。然后，将 Δp 与步长 Δt 的乘积加到动量 p 上，完成动量的更新。

自旋迭代步长决定了下一时刻自旋状态的变化量。在算法建模中，为使分岔系统能够精细演化，步长通常取较小的数值，例如0.1或0.05。然而，在硬件中实现此类小数的乘法运算会引入较大的面积与功耗开销。考虑到步长乘法在模拟绝热分岔算法的整体计算流程中仅占较小比例，并非核心计算模块，专门部署乘法器将导致硬件资源的显著浪费。此外，小步长还会导致总迭代次数显著增加，直接延长求解时间，削弱加速器在延迟方面的优势。

为了在算法建模与电路实现之间取得平衡，我们将 Δt 设为0.25。该取值有利于硬件的高效实现，仅需简单的2位右移操作，无需复杂的乘法器。针对不同复杂程度的问题，分别设定了两种迭代次数：16次和256次。对于具有预设耦合系数的“简单”最大割问题，使用16次迭代；而对于耦合系数随机生成的更具挑战性的最大割问题，则保留256次迭代。相应的结果将在后续的结论部分进行展示与分析。

3 系统架构与核心模块

3.1 系统架构

图4展示了所提出的CMOS模拟绝热分岔机的系统架构。该系统采用近存计算架构，芯片内部包含了耦合系数阵列模块(Coeff. SRAM Array)，演化模块(Evolution Block)，顶层控制模块(Top Controller)以及一个寄存器组(Reg Block)。耦合系数阵列模块内部存储了组合优化问题映射到伊辛模型中的耦合系数大小，由四个存储阵列组成，存储

Algorithm 1 Simulated Adiabatic Bifurcation

Input: $J_{i,j}, x, p, \Delta t, N, T, K, t_{stop}$
Output: $\text{sgn}(x(T))$
 1: Random initialization of x in N
 2: Random initialization of p in N
 3: **while** ($t < t_{stop}$)
 4: **for** $i = 1$ to N do
 5: $x_i(t_{n+1}) = x_i(t_n) + p_i(t_n)\Delta t$
 6: // Update Position of All Spin
 7: **end for**
 8: **for** $i = 1$ to N do
 9: $\Delta p_i(t_{n+1}) = -Kx_i^3(t_{n+1}) - O x_i(t_{n+1}) + \sum J_{i,j} x_j(t_{n+1})$
 10: $p_i(t_{n+1}) = p_i(t_n) + \Delta p_i(t_{n+1})\Delta t$
 11: // Update Momentum of All Spin
 12: **end for**
 13: $t_{n+1} = t_n + 1$
 14: // Time evolution
 15: **end while**
 16: **output** $\text{sgn}(x(t_{stop}))$

图3 模拟绝热分岔算法伪代码

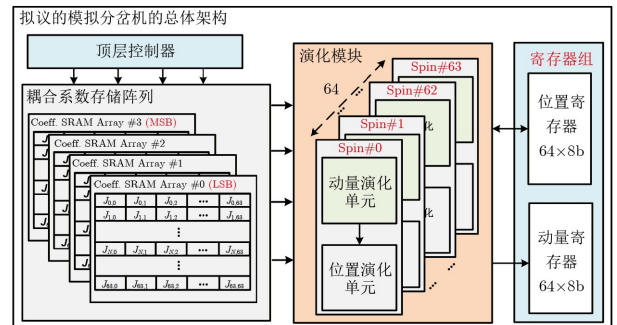


图4 模拟绝热分岔机整体架构

单元采用标准的6T SRAM结构, 每一个存储阵列大小为2016比特, 在进行耦合系数访问时, 一次将全部的存储数据读出。在演化模块中, 一共有64个自旋演化单元, 将耦合系数读出后根据其对应的坐标送入到对应的自旋单元中, 每一个自旋单元内部包含了动量演化单元(Momentum Evolution Units, MEU)、位置演化单元(Position Evolution Unit, PEU), 这个模块保证了全部的自旋是能够进行独立演化。位置演化单元中通过计算 Δp 后将下一个周期的位置信息计算出来, 这样实现周期化的连续更新。寄存器组是一个数据流中的缓冲器, 由于在动量演化、位置演化中需要其余自旋的信息, 可以通过这个寄存器组存储后进行发送, 内部由两块由寄存器组成的存储空间, 一个是位置信息的寄存器组, 另一个是动量信息的寄存器组, 二者大小都为512比特。顶层控制是芯片的控制核心, 内部涵盖了SRAM写电路、字线控制电路、输入驱动电路、模式控制电路等关键模块。顶层控制负责将耦合系数写入SRAM阵列中, 且针对不同精度的组合优化问题启用不同的阵列。在演化模块中, 顶层控制模块通过判断耦合系数的位宽以及符号位启用不同的乘法策略。

本文中的芯片对于自旋状态采用流水线更新, 每一个自旋动量下一时刻的值更新依赖全部自旋以及自身自旋的位置信息作为输入, 而自旋的位置信息又依赖上一个时刻的动量信息, 因此每一个自旋更新呈现天然的流水线更新特征。在架构设计中每一个自旋都有独立的演化模块, 全部自旋在系统开始计算时就可以并行更新, 且在整个迭代周期中不会存在气泡, 增大了系统的吞吐率。

以一个自旋为例, 其流水线更新时序图如图5所示, 三级流水线分别为访问耦合系数, 动量更新以及位置更新。由于模拟绝热分岔算法本身的特性决定, 且演化更新每一个自旋的方式就是一个流水线更新的策略。自旋更新开始时, 系统从SRAM阵列中读取耦合系数以及从寄存器组中读出其他自旋作为输入数据准备送给动量演化单元。之后将准备好的耦合系数以及位置信息进行动量演化, 计算出

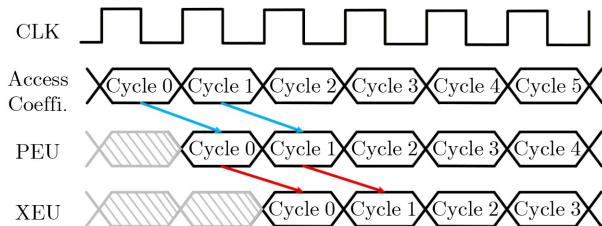


图5 模拟绝热分岔机时序图

下一个周期自旋的动量值。最后将计算出的动量结果作为输入送到位置演化模块, 计算出下一时刻自旋的位置信息。通过这种流水线操作的方式, 可以以较高的吞吐率对自旋状态进行更新, 保证了所提出处理器的低延时特性。

3.2 折叠式耦合系数存储阵列

在全连接的CMOS模拟绝热分岔机中, 由于自旋与自旋之间都存在耦合系数, 若将所有耦合系数完整的集成在芯片上, 会导致每个自旋单元对应的布线通道严重拥塞, 并且存储单元的数量与自旋数的平方成正比, 随着节点数增加, 芯片面积会急剧膨胀。针对这一问题, 本设计提出了一种折叠式存储阵列结构, 能够在完全不丢失任何耦合系数信息的前提下, 将所需存储面积。该方案的核心在于充分利用耦合系数矩阵的对称性(即 $J_{i,j} = J_{j,i}$)以及对角线元素为零($J_{i,i} = 0$)的特性, 在完整的方形矩阵中, 上三角和下三角区域存储的是相同的信息, 因此只需要保留其中一部分, 就可以完整地表示所有耦合关系。

经过折叠后, 存储阵列的物理形状变为三角形, 如图6(a)所示。然而, 三角形阵列在后端布局布线中不够规整, 容易造成走线拥塞和版图空洞, 不利于高密度集成。为了便于物理实现, 该芯片进一步将三角形阵列重构为矩形阵列, 如图6(b)所示。重构的方法是将上三角区域内的耦合系数重新排列到一个矩形SRAM中, 并为每一行和每一列分

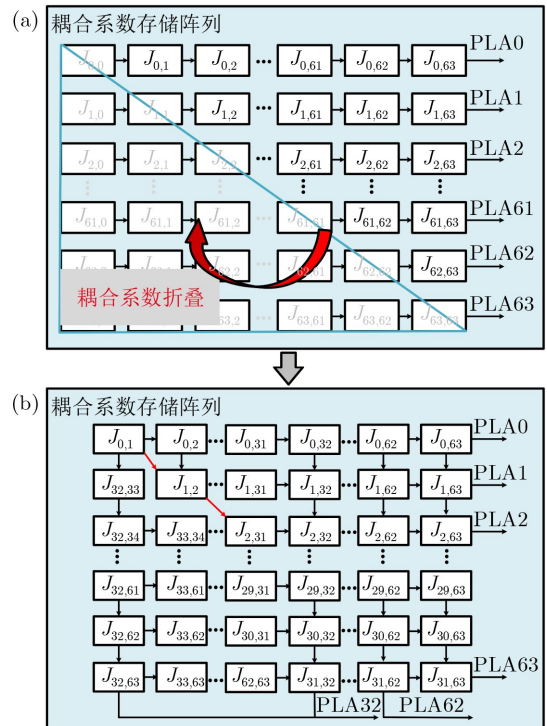


图6 折叠耦合系数存储矩阵

配对应的坐标，作为耦合系数路由时的译码标识。对于原本存储在第 i 行、第 j 列($i < j$)的耦合系数 $J_{i,j}$ ，在矩形阵列中它被放置于一个新的物理地址，但通过地址映射表可以快速恢复其原始的行列坐标。在演化第 i 个自旋时，系统需要读取所有与第 i 个自旋相关的耦合系数，即矩阵的第 i 行。由于芯片只存储了上三角部分，因此对于 $i < j$ 的系数，直接从矩形阵列中读出并路由至第 i 个演化模块；对于 $i > j$ 的系数，由于对称性，数值等于 $J_{i,j}$ ，而 $J_{i,j}$ 存储在上三角中，因此也需要从矩形阵列中读出并路由至第 i 个演化模块。通过这种基于坐标的路由策略，每个耦合系数可以被两个自旋演化单元共享，既保证了信息的完整访问，又避免了重复存储。地址映射表会根据当前演化的自旋索引，生成对应的矩形阵列读地址，并控制数据通路的方向。

在访问耦合存储阵列时，芯片采用了并行加载访问机制。传统的方式是串行读取每一个耦合系数，这会成为系统吞吐量的瓶颈。在本文提出的设计中，系统根据当前正在演化的自旋索引，通过坐标译码器同时读出对应的整行或整列耦合系数，并直接通过纵横开关网络分发至各个演化单元。由于所有耦合系数在同一个时钟周期内被并行读出并分发，计算单元可以立即开始后续的乘累加操作，无需等待。这种方式彻底消除了传统串行读取带来的访存瓶颈，显著提高了内存访问效率与系统整体吞吐量。实测表明采用并行加载访问后，每个演化周期仅需一个时钟周期即可完成所有耦合系数的读取和分发，相比串行读取的 N 个周期，访存的速度提升了 N 倍。

通过折叠式存储阵列与全耦合矩阵存储相比，面积降低了52%。与简单的三角形存储相比，矩形重构避免了芯片在布局布线中的不规则形状，降低了芯片在物理实现中布局布线的难度。并行加载访问机制使得系统的演化速度几乎不受耦合系数数量的影响，为大规模集成提供了有力支持。

3.3 动量更新单元与位置更新单元

在系统内部，每个自旋更新单元在迭代过程中依赖于两个关键模块：PEU和XEU，如图7所示。计算每个自旋单元的 Δp 需要来自其他自旋的位置信息，以及与从SRAM阵列读取的耦合系数进行的乘加运算。为了满足求解组合优化问题的精度要求，并防止迭代过程中位置和动量发生溢出，我们采用8位定点数表示 X 和 P ，其中2位为整数部分，6位为小数部分。在乘加运算过程中，每次耦合系数与位置的乘法在一个周期内完成。与此同时，微扰项和 X 的立方运算也在进行处理。由于两个操

作数均为定点格式，乘法后通过截断得到结果。另一个关键组成部分是图7(a)中 $g(x)$ 的计算，它对应于模拟绝热分岔中的线性项。在硬件实现中，该项表征了系统迭代的进程。每次更新动量和位置时， $p(t)$ 增加1。当系数 $1 - p(t)$ 变为零时，系统迭代终止，所有自旋的符号位作为组合优化问题的解输出。

4 测试结果

该芯片采用65 nm CMOS工艺技术制造，芯片面积为 $0.47 \times 1.25 \text{ mm}^2$ ，图8(a)展示了所制造芯片的显微照片，图8(b)展示相应的测试平台。在1V电源电压和200MHz时钟频率下工作时，整体功耗为14 mW。图9(a)为各模块功耗占比饼状图，自旋演化模块消耗了大部分的功耗，占整体芯片功耗的58%。图9(b)为各模块面积占比饼状图，结果表明SRAM阵列和自旋演化模块占据大部分的面积，两个模块加起来占了78%。

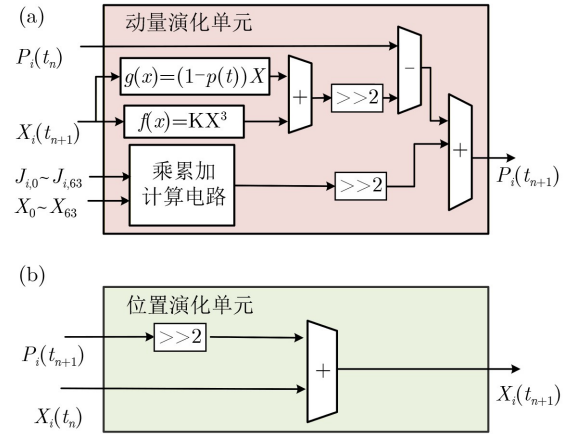
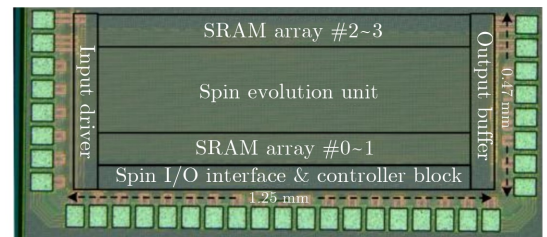
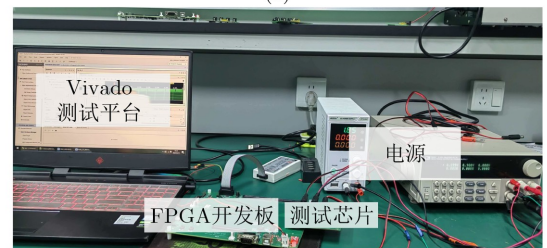


图 7 (a) 动量演化单元和 (b) 位置演化单元原理图



(a)



(b)

图 8 (a) 芯片显微图；(b)测试环境。

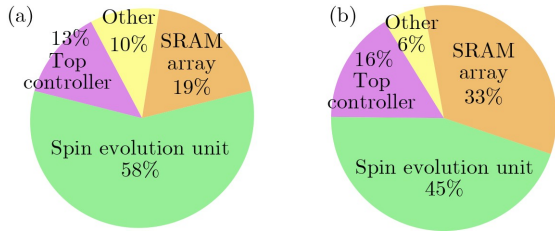


图 9 (a) 各模块功耗占比饼状图; (b) 各模块面积占比饼状图。

4.1 针对 Max-Cut 问题的应用测试

我们对两类问题进行了全面测试, 并针对大规模组合优化问题扩展了模拟绝热分岔系统的仿真。对于已知最优解的标准最大割问题, 该芯片能够一致地收敛到全局最优解。图10(a)展示了三个最大割实例(“红心”、“黑桃”、“方块”)的测量结果, 每个实例包含64个节点, 配置为 8×8 的网格结构。耦合系数被量化为2位精度, 取值范围限定为

$\{-1, 0, +1\}$ 。该芯片最多仅需5个计算周期即可达到收敛, 最终伊辛能量为 $-4\,032$, 如图10(b)所示。基态能量遵循解析表达式 $(n^4 - n^2)$, 其中 n 为最大割的行数(和列数), 这证实了该芯片能够以最短求解时间(Time to Solution, TTS)求解最大割问题。

进一步地, 我们将测试扩展到复杂度更高的预定义最大割问题, 如图11(a)和图11(d)所示。处理 108×108 图像以推导出相应的耦合系数, 随后将这些系数部署到模拟绝热分岔系统中, 如图11(b)和图11(e)所示。由图可见, 所提出求解器展现了卓越的求解精度, 同时计算时间显著减少, 如图11(c)和图11(f)所示。与理论基态进行评估时, 这对应着100%的求解准确率, 凸显了其在求解复杂优化实例时的精确性。

为了验证芯片对于随机生成的 Max-Cut 的求

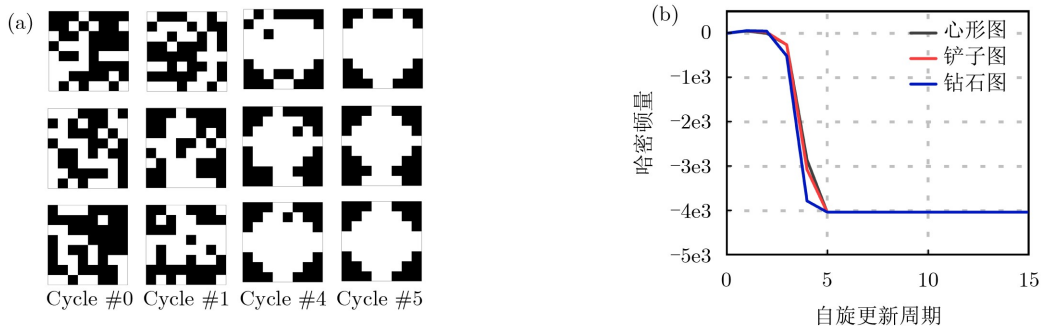


图 10 (a) 预定义最大割问题: 红心、黑桃、方块; (b) 随更新周期变化的能量演化。

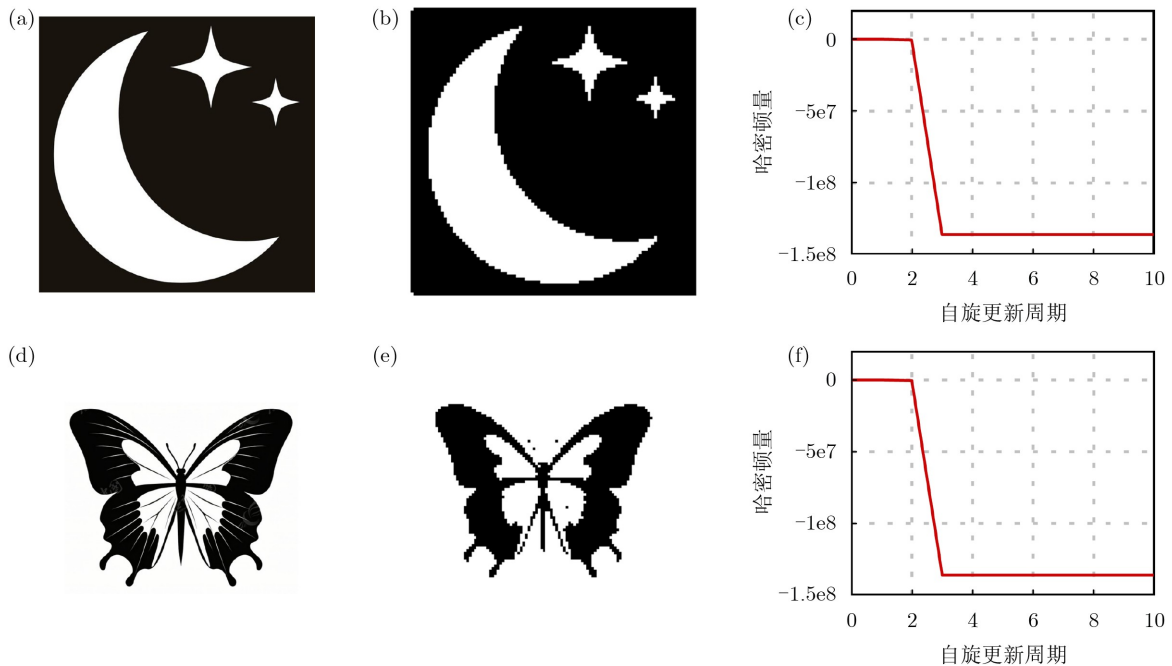


图 11 扩展最大割问题(108×108)及分岔结果。(a) 月亮与星星原始图; (b) 月亮与星星的最大割结果; (c) 月亮与星星的能量演化; (d) 蝴蝶原始图; (e) 蝴蝶的最大割结果; (f) 蝴蝶的能量演化。

解能力，本工作对随机生成的一组耦合系数的 Max-Cut 进行了求解，这种问题特性在于由于问题生成的随机性，导致该问题不一定会有全局最优解。图12(a)展示了该芯片在求解过程中在四个不同迭代时间点采集的自旋构型，由于这一类问题的耦合系数随机生成，所以最终的结果不会有一个清晰的图案。这四个时间点的自旋状态所对应的哈密顿量数值，分别标示在图12(b)的状态采样位置。从图中可以清楚地看到，随着迭代的推进，整体伊辛系统的哈密顿量呈现出明显的阶段性变化，在前25个周期内，哈密顿量迅速下降，表明系统正在快速向基态迭代。在25至125个周期之间，哈密顿量出现一定幅度的振荡。而在大约150个周期之后，哈密顿量逐渐趋于稳定，仅在平衡点附近作微小波动，这意味着系统已经基本完成了求解过程，找到了一个质量较高的近似最优解。这一演化规律与模拟绝热分岔算法的绝热分岔及后续遍历搜索机制高度吻合，进一步验证了本芯片在复杂组合优化问题上的有效性与稳定性。

为了进一步阐明模拟绝热分岔算法与模拟退火算法之间的性能差异，我们在随机生成的最大割实例上进行了蒙特卡洛仿真，并进行了系统性的比较。如图13(a)所示，在1000次独立试验中，模拟绝热分岔算法达到了平均哈密顿量值-3617.42，优于模拟退火算法的平均值-3352.12。这些结果表

明，模拟绝热分岔算法在求解随机生成的最大割问题方面具有更优越的性能。图13(b)给出了某一次模拟绝热分岔求解过程中各个自旋的位置演化曲线。从图中可以清晰观察到，当自旋更新约20个周期时，系统中部分自旋出现明显的分岔现象，原本聚集在零附近的自旋开始向正负两侧分离。随着迭代次数增加，所有自旋的位置逐渐向-1和+1两个边界点靠近，并最终在这两个边界附近作微小振荡，即系统这时已经完成收敛。

该演化过程直观地展示了模拟绝热分岔算法将自旋在绝热场中的动力学特征，自旋会通过动量更新自旋位置，且自旋以来一个随机动量逃离局部最优解，验证了芯片高效求解组合优化问题的能力。相比模拟退火需要通过大量随机翻转和缓慢降温来逐步逼近最优解，模拟绝热分岔伊辛机利用自然的绝热分岔过程，每一个自旋会独立的进行演化，这保证系统演化过程中更低的求解延迟，且每一个自旋会通过自身的微扰项跳出系统的局部最小值，这一过程保证了系统的求解精度。通过与先前工作使用的模拟退火算法相比，这进一步体现了本芯片在求解速度与精度上的双重优势。

4.2 针对3-SAT问题的应用测试

3-SAT问题是一个NP-Complete问题，求解SAT问题的方式在于判定给定的布尔表达式 $S(C_1, C_2, \dots, C_n)$ 是否存在一组赋值 $V(v_1, v_2, \dots, v_n)$

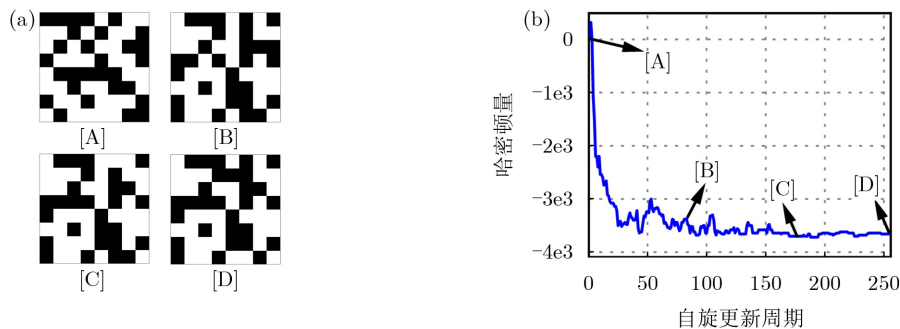


图 12 (a) 随机生成的Max-Cut问题在演化过程的自旋状态图片；(b) 伊辛系统哈密顿量随着自旋更新周期的变化。

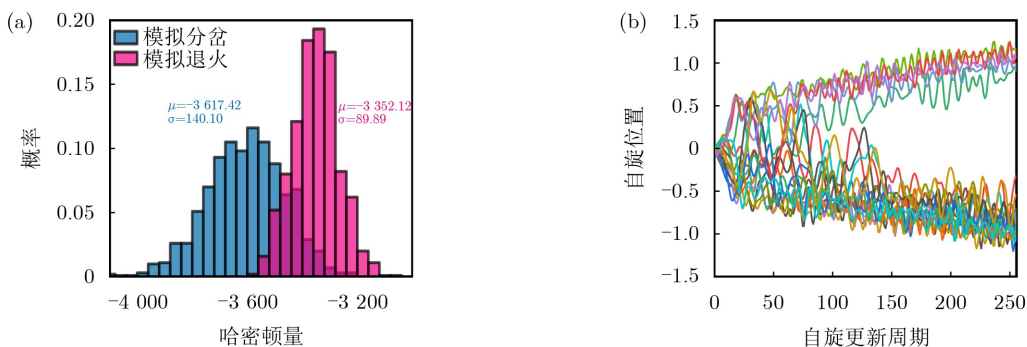


图 13 (a) 在随机生成的最大割问题上，模拟绝热分岔与模拟退火各执行1000次的蒙特卡洛仿真结果对比；(b) 选取一次求解问题中自旋位置的分岔曲线。

使其结果为真。例如, S 是一个包含三个变量和三个子句的合取范式(Conjunctive Normal Form, CNF) $S = (\neg v_1 \vee v_2 \vee v_3) \vee (v_1 \vee \neg v_2 \vee v_3) \wedge (v_1 \vee \neg v_2 \vee \neg v_3)$ 要保证一组变量逻辑构型满足布尔表达式为真。当 v_1 赋值为真, v_2 赋值为假, v_3 赋值为真时, 表达式 S 结果为真, 则称该合取范式 S 是可满足的, 反之, 当不存在一组解使得该合取范式 S 为真, 则该式 S 是不可满足的。

图14展示了本芯片在求解不同难度的3-SAT实例时的求解时间与未满足子句数量之间的动态关系, 通过将3-SAT问题转换到模拟绝热分岔模型的耦合系数进行运算。从图中可以直观地观察到, 随着求解过程的推进, 未满足子句的数量呈现出逐渐下降的趋势。当合取范式中的所有子句均被满足时, 未满足子句的数量最终降为零, 此时芯片找到了问题的一个可满足解。

为了定量衡量不同3-SAT实例的求解难易程度, 本工作引入了子句与变量比率(Clause to Variable Ratio, CTV)这一指标。CTV值定义为3-SAT问题中子句总数与变量总数之比。一般而言, 当CTV较小时, 子句约束相对稀疏, 解空间较大, 存在大量可满足解, 问题相对容易求解。当CTV较大

时, 约束条件过于密集, 解空间变得极为狭窄, 甚至可能不存在任何可满足解。对于3-SAT实例来说, 存在一个位于4.26附近的相变点。当CTV低于该阈值时, 绝大多数3-SAT问题的实例是可满足的; 反之, 当CTV高于该阈值时, 绝大多数3-SAT问题的实例是不可满足的。

本文选取的三个测试实例的CTV值分别为3、4和5, 覆盖了从容易到困难的典型范围。从实验结果可以看出, CTV为3的实例收敛速度最快, 求解时间仅需约6 μs , CTV为4的实例求解时间略有增加, 约为10 μs , 因为此时接近相变点, 解空间结构变得复杂, 局部极小值增多, 芯片需要更多时间进行探索。CTV为5的实例求解时间最长, 达到16 μs , 这主要原因是约束密度过高, 变量之间的冲突频繁, 芯片需要多次迭代才能判定是否不存在可满足解或找到一个极其隐蔽的解。上述结果与理论分析完全吻合, 进一步验证了本芯片在不同复杂度3-SAT实例上的有效性。

4.3 芯片性能总结

表1汇总了本文所提出芯片与现有全连接CMOS伊辛机在关键性能指标上的对比结果。从先前功能验证的讨论中可以看出, 本工作在求解时间方面具有显著优势, 这得益于所提出的近存计算架构与全并行自旋更新策略。与此同时, 该芯片在1 V供电电压下的工作频率达200 MHz, 功耗为14 mW。每自旋面积比其他伊辛机小1.8~2.6倍, 每自旋功耗降低1.36~17.36倍。可见, 本文所提从的芯片具有较高的面积效率和能量效率。

5 结论

本工作提出一种无需复制自旋副本即可实现全并行更新的模拟绝热分岔机。为提升吞吐率, 设计了融合耦合系数访问、动量更新与位置更新的三级流水线策略, 实现了无气泡并行更新和低延时求

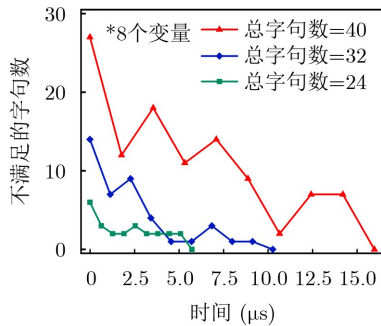


图14 求解3-SAT问题求解时间与未满足子句数量之间的动态关系。

表1 与其他全连接伊辛机的性能对比

	ISSCC'23 [19]	TCAS-I'21 [20]	ISSCC'20 [11]	TCAS-II'24 [21]	本工作
工艺	40 nm	28 nm	65 nm	90 nm	65 nm
自旋数目	512	36	512	128	64
系数位宽	8 bit	4 bit	2 bit	5 bit	1-4 bit
退火算法	RPA	RAGU	SCA	SA	SAB
工作频率	320 MHz	N/A	300 MHz	100 MHz	200 MHz
芯片面积	9 mm ²	N/A	12 mm ²	2.15 mm ²	0.59 mm ²
每自旋面积	0.018 mm ²	N/A	0.023 mm ²	0.016 mm ²	0.009 mm ²
芯片功耗	151.6 mW	137.5 mW	252 mW	106 mW	14 mW
每自旋功耗	0.3 mW	3.82 mW	0.49 mW	0.83 mW	0.22 mW
是否需要随机数	需要	需要	需要	需要	不需要

解。针对稀疏耦合系数，采用折叠式存储结构，有效降低存储面积开销。动量和位置变量均采用8位定点表示，在保证计算精度的同时兼顾资源效率。与先前全连接伊辛机相比，本分岔机在达到100%求解准确率的基础上，兼具更高能效和更低硬件开销，在边缘端组合优化领域具有较大的应用前景。

参 考 文 献

- [1] 潘钰, 胡航, 金虎, 等. 非授权频段下无人机辅助通信的轨迹与资源分配优化[J]. 电子与信息学报, 2024, 46(11): 4287–4294. doi: [10.11999/JEIT240275](https://doi.org/10.11999/JEIT240275).
PAN Yu, HU Hang, JIN Hu, *et al.* Trajectory and resource allocation optimization for unmanned aerial vehicles assisted communications in unlicensed bands[J]. *Journal of Electronics & Information Technology*, 2024, 46(11): 4287–4294. doi: [10.11999/JEIT240275](https://doi.org/10.11999/JEIT240275).
- [2] 赖李洋, 郑铭骏, 梁海成, 等. 路径规划算法的高层综合设计研究[J]. 电子与信息学报, 2024, 46(11): 4132–4140. doi: [10.11999/JEIT240210](https://doi.org/10.11999/JEIT240210).
LAI Liyang, ZHENG Peijun, LIANG Haicheng, *et al.* Case study of high level synthesis on path planning algorithm[J]. *Journal of Electronics & Information Technology*, 2024, 46(11): 4132–4140. doi: [10.11999/JEIT240210](https://doi.org/10.11999/JEIT240210).
- [3] 陈家瑞, 吴昭怡, 游勇杰, 等. 基于概率模型的集成电路寄生参数提取算法[J]. 电子与信息学报, 2025, 47(9): 3198–3207. doi: [10.11999/JEIT250458](https://doi.org/10.11999/JEIT250458).
CHEN Jiarui, WU Zhaoyi, YOU Yongjie, *et al.* A probability-based parasitic extraction algorithm for global-routed VLSI designs[J]. *Journal of Electronics & Information Technology*, 2025, 47(9): 3198–3207. doi: [10.11999/JEIT250458](https://doi.org/10.11999/JEIT250458).
- [4] 钱煜, 杨泽禹, 王然然, 等. 铁电基的存算一体组合优化求解器[J]. 电子与信息学报, 2025, 47(9): 3104–3115. doi: [10.11999/JEIT250369](https://doi.org/10.11999/JEIT250369).
QIAN Yu, YANG Zeyu, WANG Ranran, *et al.* Ferroelectric FET-based compute-in-memory solver for combinatorial optimization problems[J]. *Journal of Electronics & Information Technology*, 2025, 47(9): 3104–3115. doi: [10.11999/JEIT250369](https://doi.org/10.11999/JEIT250369).
- [5] 洪庆辉, 孙辰, 肖平旦, 等. 复值Hopfield神经网络的信号盲检测一步计算电路[J]. 电子与信息学报, 2024, 46(11): 4123–4131. doi: [10.11999/JEIT240224](https://doi.org/10.11999/JEIT240224).
HONG Qinghui, SUN Chen, XIAO Pingdan, *et al.* One-step calculation circuit of blind signal detection using complex-valued hopfield neural network[J]. *Journal of Electronics & Information Technology*, 2024, 46(11): 4123–4131. doi: [10.11999/JEIT240224](https://doi.org/10.11999/JEIT240224).
- [6] FU Fangfa, LOU Binglei, CHEN Yukun, *et al.* Improving reliability in NOCs by reconstructing location distribution of management cores[J]. *Microelectronics Journal*, 2019, 90: 133–140. doi: [10.1016/j.mejo.2019.06.001](https://doi.org/10.1016/j.mejo.2019.06.001).
- [7] LI Tonghui, DUAN Xiaofeng, LIU Kai, *et al.* Parameter extraction for photodiode equivalent circuit model based on hybrid genetic algorithm[J]. *Microelectronics Journal*, 2024, 143: 106017. doi: [10.1016/j.mejo.2023.106017](https://doi.org/10.1016/j.mejo.2023.106017).
- [8] CHEN Jingyang, YU Zhiping, and ZHU Xiaolei. A 28 nm Ising machine with adaptive majority voter and reduction algorithms for high-performance combinatorial optimization[J]. *Microelectronics Journal*, 2025, 159: 106621. doi: [10.1016/j.mejo.2025.106621](https://doi.org/10.1016/j.mejo.2025.106621).
- [9] SU Yuqi, KIM T T H, and KIM B. FlexSpin: A CMOS Ising machine with 256 flexible spin processing elements with 8-b coefficients for solving combinatorial optimization problems[J]. *IEEE Journal of Solid-State Circuits*, 2024, 59(8): 2659–2670. doi: [10.1109/JSSC.2024.3352907](https://doi.org/10.1109/JSSC.2024.3352907).
- [10] XIE Shanshan, RAMAN S R S, NI Can, *et al.* Ising-CIM: A reconfigurable and scalable compute within memory analog Ising accelerator for solving combinatorial optimization problems[J]. *IEEE Journal of Solid-State Circuits*, 2022, 57(11): 3453–3465. doi: [10.1109/JSSC.2022.3176610](https://doi.org/10.1109/JSSC.2022.3176610).
- [11] YAMAMOTO K, KAWAMURA K, ANDO K, *et al.* STATICA: A 512-spin 0.25M-weight annealing processor with an all-spin-updates-at-once architecture for combinatorial optimization with complete spin-spin interactions[J]. *IEEE Journal of Solid-State Circuits*, 2020, 56(1): 165–178. doi: [10.1109/JSSC.2020.3027702](https://doi.org/10.1109/JSSC.2020.3027702).
- [12] LIU Yixuan, WU Qiqiao, YANG Honghu, *et al.* Enhancing all-to-all RRAM Ising machines with randomized granular update strategies for solving combinatorial optimization problems[J]. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2025, 72(9): 4877–4890. doi: [10.1109/TCSI.2024.3523999](https://doi.org/10.1109/TCSI.2024.3523999).
- [13] GRIMALDI A, MAZZA L, RAIMONDO E, *et al.* Evaluating spintronics-compatible implementations of Ising machines[J]. *Physical Review Applied*, 2023, 20(2): 024005. doi: [10.1103/PhysRevApplied.20.024005](https://doi.org/10.1103/PhysRevApplied.20.024005).
- [14] YAMAOKA M, YOSHIMURA C, HAYASHI M, *et al.* A 20k-spin Ising chip to solve combinatorial optimization problems with CMOS annealing[J]. *IEEE Journal of Solid-State Circuits*, 2016, 51(1): 303–309. doi: [10.1109/JSSC.2015.2498601](https://doi.org/10.1109/JSSC.2015.2498601).
- [15] SU Yuqi, KIM H, and KIM B. CIM-spin: A scalable CMOS annealing processor with digital in-memory spin operators and register spins for combinatorial optimization problems[J]. *IEEE Journal of Solid-State Circuits*, 2022, 57(7): 2263–2273. doi: [10.1109/JSSC.2021.3139901](https://doi.org/10.1109/JSSC.2021.3139901).
- [16] AHMED I, CHIU P W, and KIM C H. A probabilistic self-

- annealing compute fabric based on 560 hexagonally coupled ring oscillators for solving combinatorial optimization problems[C]. 2020 IEEE Symposium on VLSI Circuits, Honolulu, USA, 2020: 1–2. doi: [10.1109/VLSICircuits18222.2020.9162869](https://doi.org/10.1109/VLSICircuits18222.2020.9162869).
- [17] BAE J, OH W, KOO J, *et al.* CTLE-Ising: A 1440-spin continuous-time latch-based Ising machine with one-shot fully-parallel spin updates featuring equalization of spin states[C]. 2023 IEEE International Solid-State Circuits Conference (ISSCC), San Francisco, USA, 2023: 142–144. doi: [10.1109/ISSCC42615.2023.10067622](https://doi.org/10.1109/ISSCC42615.2023.10067622).
- [18] TATSUMURA K, YAMASAKI M, and GOTO H. Scaling out Ising machines using a multi-chip architecture for simulated bifurcation[J]. *Nature Electronics*, 2021, 4(3): 208–217. doi: [10.1038/s41928-021-00546-4](https://doi.org/10.1038/s41928-021-00546-4).
- [19] KAWAMURA K, YU J, OKONOGI D, *et al.* Amorphica: 4-replica 512 fully connected spin 336MHz metamorphic annealer with programmable optimization strategy and compressed-spin-transfer multi-chip extension[C]. 2023 IEEE International Solid-State Circuits Conference (ISSCC), San Francisco, USA, 2023: 42–44. doi: [10.1109/ISSCC42615.2023.10067504](https://doi.org/10.1109/ISSCC42615.2023.10067504).
- [20] LIU Yixuan, WU Qiqiao, YANG Honghu, *et al.* Enhancing all-to-all RRAM Ising machines with randomized granular update strategies for solving combinatorial optimization problems[J]. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2025, 72(9): 4877–4890. doi: [10.1109/TCSI.2024.3523999](https://doi.org/10.1109/TCSI.2024.3523999).
- [21] CHEN Y H, CHOU C A, NIEN C F, *et al.* Design and implementation of a VLSI-based annealing accelerator for efficiently solving combinatorial optimization problems[J]. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2024, 71(9): 4291–4295. doi: [10.1109/TCSII.2024.3380609](https://doi.org/10.1109/TCSII.2024.3380609).
- 李任龙: 男, 博士生, 研究方向为专用加速器、模拟计算等。
郝 昕: 男, 硕士生, 研究方向为专用加速器、模拟计算等。
陈卓俊: 男, 教授, 研究方向为集成电路设计。
丁 玓: 女, 高工, 研究方向为存算一体芯片的设计与性能研究。

责任编辑: 廖海贝

A High-Parallelism Simulated Adiabatic Bifurcation Processor for Combinatorial Optimization Problems

LI Renlong^① HAO Xin^① CHEN Zhuojun^① DING Ding^{②③}

^①(College of Semiconductors (College of Integrated Circuits), Hunan University, Changsha 410082, China)

^②(School of Intelligent Engineering and Intelligent Manufacturing, Hunan University of Technology and Business, Changsha 410205, China)

^③(Xiangjiang Laboratory, Changsha 410205, China)

Abstract:

Objective Combinatorial optimization problems (COPs) are widely encountered in fields such as network optimization, autonomous driving path planning, VLSI design, and computational biology. The solution space of these problems grows exponentially with problem size, making it impossible for traditional von Neumann architectures (e.g., CPUs) to find high-quality solutions within a feasible time. Quantum-inspired Ising machines have emerged as a promising computing paradigm for accelerating COP solving. However, existing CMOS-based Ising machines still face significant challenges in simultaneously achieving high speed, high energy efficiency, and high accuracy. Discrete-time Ising machines often require thousands of iterations and rely heavily on random number generators, leading to large chip area and long solving times. Continuous-time Ising machines suffer from poor solution quality, frequently getting trapped in local minima. To address these limitations, this paper designs and fabricates a high-parallelism simulated adiabatic bifurcation application-specific processor for combinatorial optimization problems in 65 nm CMOS technology.

Methods The proposed processor adopts the simulated adiabatic bifurcation (SAB) algorithm, which is inspired by quantum adiabatic optimization. Unlike simulated annealing, SAB does not require Gibbs sampling or random number generators to escape local minima. The algorithm models each spin as a nonlinear oscillator and solves a set of ordinary differential equations to simulate the adiabatic evolution of a classical nonlinear Hamiltonian system exhibiting bifurcation phenomena. The processor builds a hardware architecture that

supports fully connected spin topologies and leverages the inherent fully parallel spin update characteristic of the SAB algorithm. To achieve bubble-free iterative computation, a three-stage pipelined spin update strategy is proposed, dividing the update process into coupling coefficient access, momentum update, and position update. To reduce the storage overhead introduced by the fully connected coupling matrix, a folded coupling coefficient storage array is designed, exploiting matrix symmetry to eliminate redundant storage. The momentum update unit and position update unit are implemented using 8-bit fixed-point arithmetic (2 bits for integer, 6 bits for fractional part) to achieve SAB evolution with low hardware overhead. The chip is fabricated in 65 nm CMOS technology, occupying an area of $0.47 \times 1.25 \text{ mm}^2$ and operating at a 200 MHz clock frequency and 1 V supply voltage.

Results and Discussions The chip achieves a total power consumption of only 14 mW, with the spin evolution module consuming 58% of the total power. The folded coupling coefficient storage array reduces storage area by 52% compared to full matrix storage, and the rectangular restructuring avoids irregular shapes in physical layout, reducing routing congestion and layout voids (Fig.6). The parallel loading access mechanism allows all coupling coefficients to be read and distributed within a single clock cycle, eliminating the memory access bottleneck inherent in serial reading. For predefined Max-Cut problems configured as 8×8 grid structures (64 nodes) with coupling coefficients quantized to 2-bit precision, the chip converges to the global optimum within only 5 computing cycles, achieving a final Ising energy of -4032 (Fig.9). This energy follows the analytical expression $(n_4 - n_2)(n_4 + n_2)$, confirming that the chip solves Max-Cut problems with the shortest solving time. For larger extended Max-Cut problems (108×108 nodes), the simulated adiabatic bifurcation algorithm achieves 100% accuracy relative to the theoretical ground state (Fig.10). Monte Carlo simulations over 1,000 independent trials on randomly generated Max-Cut problems demonstrate that simulated adiabatic bifurcation achieves an average Hamiltonian of -3617.42 , significantly outperforming simulated annealing which achieves only -3352.12 (Fig.11). For 3-SAT problems with 40 clauses and 8 variables, the chip solves instances with clause-to-variable ratios of 3, 4, and 5 in approximately 6 μs , 10 μs , and 16 μs , respectively (Fig.12). These results align perfectly with theoretical phase transition predictions, confirming the processor's effectiveness across varying problem complexities.

Conclusions This work proposes an simulated adiabatic bifurcation machine that enables fully parallel updates without duplicating spin copies. To improve throughput, a three-stage pipeline strategy is designed that integrates coupling-coefficient access, momentum update, and position update, achieving bubble-free parallel updating and low-latency solving. For sparse coupling coefficients, a folded storage scheme is adopted to significantly reduce memory area overhead. Both momentum and position variables are represented in 8-bit fixed-point format, ensuring sufficient computational accuracy while balancing resource efficiency. Compared with previous fully connected Ising machines, the proposed bifurcation machine achieves 100% solving accuracy, along with higher energy efficiency and lower hardware overhead, demonstrating substantial application prospects in edge-side combinatorial optimization.

Key words: Combinatorial optimization problems (COPs); Simulated adiabatic bifurcation (SAB); Ising machine; Application-Specific Integrated Circuit (ASIC)