arXiv:2304.13550v1 [cs.DM] 26 Apr 2023

将块顺序自动机网络转换为具有同构极限动力学的更小并行网络

Pacˆome Perrotin[1,2] 和 Sylvain Sen´e[1,2]

[1] 公立大学，法国马赛[2] Aix-Marseille大学，法国国家科学研究中心，LIS，马赛，法国

摘要。我们提出一种算法：给定一个自动机网络和一个块顺序更新调度，可以生成一个规模相同或更小、且在并行更新调度下具有相同极限动力学的自动机网络。随后，我们关注一类共享唯一自动机路径的自动机环，并将其称为切向循环；我们证明，对该算法加以限制，可以将这类网络在块顺序更新调度下的任意实例，约化为该类中规模更小的并行网络，并在保持其极限动力学不变的同时，刻画所执行的约化次数。我们还证明，经主算法约化的任意切向循环都会被转换为一个规模等于初始网络最大环规模的网络。最后，我们证明，受限算法可以直接将块顺序双循环刻画为并行双循环。

1 引言

自动机网络经典地用于对基因调控网络进行建模 [9,16][10,2,4]。在这些应用中，自动机网络的动力学有助于理解生物系统可能如何演化。因此，人们有动机改进对自动机网络动力学的计算与刻画。鉴于当前研究的网络结构、局部函数和更新调度具有极大的多样性，这一问题很难处理。我们不考虑一般情形下的问题，而是寻找能够产生更简单动力学的族或性质，以便对其进行刻画 [7,8]。

我们感兴趣的是研究自动机网络的极限动力学，即它们随时间演化所呈现的极限环和不动点，尤其是因为底层动力系统的这些渐近行为可能对应于真实的生物学现象，例如细胞类型、组织或生物节律的基因表达模式。更准确地说，与可能的配置本身相比，我们不太关注网络中随时间传递和计算的信息。因此，对于给定的网络族，我们的目标之一是统计其中包含的不动点和极限环数量。

本文提出一种算法：给定一个自动机网络和一种块顺序更新调度，可生成一个自动机网络，其

规模相同或更小，且在并行更新调度下具有同构极限动力学。第2节给出定义，第3节详细介绍该算法。第4节研究该算法应用于切向循环（TC）的可行性。TC 是一组环，它们在一条共享的自动机路径上相交。

为何聚焦于 TC？环是基本的回溯模式，是观察复杂动力学所必需的 [14]。它们存在于许多生物调控网络中 [17]，且其孤立状态已得到充分理解 [6,12]。从理论上讲，环会产生数量呈指数增长的极限环，这与生物系统的观测行为不符 [9]。减少极限环数量的唯一方法，是约束孤立环所带来的自由度；而从纯结构角度看，这只能通过环的相交来实现。这自然引出了 TC 这一简单的相交情形。特别是，双循环（两个孤立环的相交）是目前已有完整刻画的最大一类相交环 [12,5]；本文将这一结果推广到块顺序更新调度。此外，从生物学角度看，双循环也存在于生物调控网络中，似乎会抑制其极限行为 [3]。

2 定义

设 Σ 为一个有限字母表。我们将 Σ^n 记作字母表 Σ上所有长度为 n 的单词的集合，其中对于所有 $1 \leq i \leq n$ 和 $x \in \Sigma^n$， x_i 是该单词的第 i个字母。自动机网络（AN）是一个函数 F： $\Sigma^n \to \Sigma^n$，其中 n是网络的大小。 F 的一个配置是 Σ^n上的一个单词。全局函数 F 可以分解为分别局部于每个自动机的函数：$\forall k$、 f_k ： $\Sigma^n \to \Sigma$；全局函数可以重新定义为每个局部函数的并行应用： $\forall 1 \leq i \leq n$、 $F(x)_i = f_i(x)$。为方便起见，自动机集合 $\{1, \ldots, n\}$ 记作 S，有时将其视为字母集合而非数字集合。对于复杂性问题，我们认为局部函数始终编码为电路。

对于（i， j）任意一对自动机，当且仅当存在一个配置 $x \in \Sigma^n$，其中存在 i 的状态变化，使得 f_j（x）的状态发生变化时， i 才被称为影响 j。更正式地说，当且仅当存在 x、 $x' \in \Sigma^n$，使得 $\forall k$、 $x_k = x'_k \Leftrightarrow k \neq i$ 以及 f_j（x） $\neq f_j$（x'）时， i 影响 j。

通常将自动机网络 F 表示为一个有向图，其中自动机作为节点；当且仅当 i 影响 j时，（i， j）是一条边。该有向图称为交互有向图，记作 G_I（F） =（S， E），其中E 为边集。示例 1中描述的自动机网络在图1中以交互有向图的形式展示。

示例 1。令 F： $\mathbb{B}^3 \to \mathbb{B}^3$ 为一个自动机网络，其局部函数为

$$f_a(x) = \neg x_b \vee x_c$$
$$f_b(x) = x_a$$
$$f_c(x) = \neg x_b$$

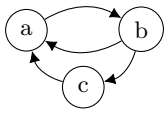


图 1。示例 1中详细介绍的AN的交互有向图。

更新调度是由 S 的非空子集组成的无限序列，这些子集称为块。这样的序列描述了应用局部函数以更新网络的顺序，并且这样的序列有不可数无穷多个。周期更新调度是由 S的非空子集组成的无限周期序列，我们直接通过其周期对其进行定义。将更新调度应用于网络的一个配置，就是并行应用序列中各个子集的局部函数，每个子集依次应用。

例如，序列 $\pi = (S)$ 是并行更新调度。它是周期性的，并且将其应用于一个配置与应用 F 无法区分。序列 $(\{1\}, \ldots, \{n\})$ 也是一个周期更新调度，并意味着按顺序逐一应用每个局部函数。

形式上，将周期更新调度 Δ 应用于配置 $x \in \Sigma^n$ 记作函数 F_Δ，其定义为按照 Δ 指定的顺序复合应用局部函数。对于任意子集 $X \subseteq S$，将 X 更新为 x 记作 $F_X(x)$，其定义为

$$\forall i \in S,\ F_X(x)_i = \begin{cases} f_i(x) \text{ if } i \in X \\ x_i \quad \text{otherwise} \end{cases}.$$

示例 2 给出了在某个非平凡更新调度下执行示例 1中详细介绍的网络的例子。

示例 2。设 $\Delta = (\{b, c\}, \{a\}, \{a, b\})$ 是一个周期更新调度，且 $x = 000$ 是一个初始配置。对于示例 1中详细介绍的AN F，我们有：

$$\begin{aligned} F_\Delta(000) &= (F_{\{a,b\}} \circ F_{\{a\}} \circ F_{\{b,c\}})(000) \\ &= (F_{\{a,b\}} \circ F_{\{a\}})(001) \\ &= F_{\{a,b\}}(101) = 111. \end{aligned}$$

块顺序更新调度是一种周期更新调度，其中一个周期内的所有子集构成 S 的划分；也就是说，序列中的每个自动机都恰好更新一次。如果序列中的每个子集的基数均为 1，则称该更新调度为顺序更新。对于包含自动机 S 的任意自动机网络，并行更新调度和 $|S|!$ 种不同的顺序更新调度都是块顺序的。块顺序更新调度是公平更新调度，因为应用该调度时，每个自动机的更新次数都相同。

对自动机网络应用块顺序更新调度，也可以用 [15,1] 中引入的更新有向图来表示。对于 F 一个自动机网络和

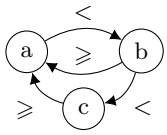


图 2. 自动机网络的更新 示例 1 中详述的有向图，针对 $\Delta = (\{a\}, \{b\}, \{c\})$。

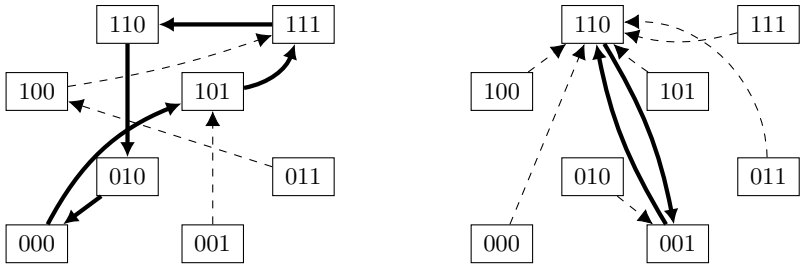


图 3. 示例 1 中详述的自动机网络 F的两种动力学。左侧为 F在并行更新调度下的动力学。右侧为 F在更新调度 $\Delta = (\{a\}, \{b\}, \{c\})$ 下的动力学。粗箭头表示极限动力学。

Δ 一个块顺序更新调度， F_Δ的更新有向图记为$G_U(F_\Delta)$，是对该网络交互有向图的注释图，其中，若在 Δ 中 u 的更新时间严格早于 v ，则任意边 (u, v) 会标注为 $<$ ，否则标注为 $\geqslant$。示例 1 中详述的自动机网络的更新有向图如图 2 所示。

给定一个自动机网络 F 和一个周期更新调度 Δ，我们将 F_Δ 的动力学定义为一个有向图，其所有构型 $x \in \Sigma^n$ 均为节点；当且仅当 $F_\Delta(x) = y$时，(x, y) 是该动力学的一条边。我们将任何由互不相同的构型 $(x_1, x_2, \ldots, x_k)$ 组成的序列称为长度为 k的极限环，前提是对所有 $1 \leq i < k$，均有$F_\Delta(x_i) = x_{i+1}$ ，且 $F_\Delta(x_k) = x_1$。长度为一的极限环称为不动点。 F_Δ 的极限动力学是仅包含该动力学的极限环和不动点的子图。图 3 着重显示了示例 1 中定义的网络的极限动力学。

由于网络的动力学是一个图，其规模相对于自动机的数量呈指数增长，因此，直接计算一族网络的极限动力学是一项计算开销高的过程。

3 算法

在本节中，我们将介绍一种算法，它可以将任意采用块顺序更新调度 Δ 的自动机网络 F转换为另一个自动机网络 F'，使得 F_Δ 的极限动力学与 F' 在并行更新调度 π 下的极限动力学同构。此外， F' 的规模始终不大于 F。

该算法由两部分组成：首先，我们借助自动机网络领域广为人知的一种算法，将网络并行化。其次，我们根据第一步产生的冗余，从网络中移除自动机。

首先，我们陈述通常使用的算法：给定一个自动机网络 F和一个块顺序更新调度 Δ，该算法会生成一个定义在同一组自动机上的新自动机网络 F'，使得 F_Δ 和 F'_π 具有相同的精确动力学。

算法1 F_Δ的并行化算法

输入
  F 网络在 S上的局部函数 , 编码为电路
  Δ S上的块顺序更新调度
输出
  F 并行网络在 S上的局部函数，以电路编码
对于满足 u 在 Δ 中先于 v 的(u, v)，执行
  在 f_v中应用替换 $x_u \mapsto \theta_u$ ▷ θ 是一个临时符号
令 $X \leftarrow S$
当 $|X| > 0$ 时执行
  令 $s \in X$，使得 f_s 不包含任何 θ 符号
  $X \leftarrow X \setminus \{s\}$
  对于 $s' \in X$，执行
    如果 $f_{s'}$ 包含 θ_s，则
      在 $f_{s'}$中应用替换 $\theta_s \mapsto f_s$
返回 F

算法1分两轮进行替换。首先，对于每条 <-边（u、 <、 v），将影响自动机 u替换为v 的局部函数中的令牌符号 θ_u。然后按照正确的顺序，将所有这些令牌符号替换为相应的局部函数（此处为 f_u）：也就是说，如果某个函数包含令牌字符，就绝不会在替换中使用它。这样一来，即使网络中包含由 <-边构成的复杂树，也能按正确顺序进行替换。该算法总能返回结果，且运行时间为多项式时间。

性质1. 算法1总能返回结果，且运行时间为多项式时间。

证明. 令 <-图表示图 $G_U(F_\Delta)$ 中仅保留<-边后得到的子集。 F_Δ 的 <-图始终是一棵树（或多棵互不连通的树）：否则， $G_U(F_\Delta)$中就会存在由 <-边构成的环，这意味着存在一组自动机，它们都严格先于各自的出邻居更新，这是不可能的。

算法1将在 <图中的每条边上放置一个 θ 符号。在第二次循环中，选中的 s 始终是 <图中某棵树的叶节点。应用替换会将该叶节点从 <图中移除。根据树的结构，所有 <边都会被移除，算法随即终止。

为了证明该算法可以在多项式时间内执行，请注意，所有局部函数都编码为电路。因此，只需将每个局部函数复制到一个大型电路中，并在其中执行每次替换即可。任何替换 $x_u \mapsto \theta_u$ 都可以通过重命名相应的输入门来完成。任何替换 $\theta_s \mapsto f_s\ (x)$ 都可以通过将对应于 θ_s 的输入门替换为与计算局部函数 f_s 的电路的输出门相连来完成。对于 F_{Δ}的更新有向图中的每条 $<$边，都要执行这些替换；具体而言，只需对 Δ所提供的偏序中的每一对执行一次替换，替换次数不会超过 n^2。然后，将所得电路为输出网络中的每个自动机各复制一份，最终总大小不超过 k^2，其中 k 表示输入的大小。□

注释 1。如果局部函数以公式形式编码，则该算法不是多项式算法；这是文献中经常被忽视的细节，因为在文献中，该并行化算法总是假定为多项式算法。

定理 1。对于任意 F_{Δ} 算法1 返回一个网络 F'，使得 F_{Δ} 的动力学与 F'_{π} 的动力学相同。

证明。设我们考虑某个配置 $x \in \Sigma^n$，并计算其在两个系统中的像x'。设我们考虑 Δ 中的初始块 X_0。对于 X_0 中的任意自动机，其局部函数在 F' 中保持不变，因此 $F_{\Delta}(x)|_{X_0} = F'(x)|_{X_0}$。假设对于某个 k， $F_{\Delta}(x)|_{X_0 \cup \ldots \cup X_k} = F'(x)|_{X_0 \cup \ldots \cup X_k}$，下面证明在包含下一个块 X_{k+1} 时该结论仍然成立。

设 $v \in X_{k+1}$。根据 Δ 中更新的性质，对于满足 $u \in X_0 \cup \ldots \cup X_k$ 的任意 x_u， f_v 将使用 $F_{\Delta}(x)$ 中的值进行更新；否则则使用 x 中的值。在 F' 中，局部函数 f'_v 内，对于影响 v 的任意 $u \in X_0 \cup \ldots \cup X_k$，替换已将 x_u 替换为 $f'_u(x)$，这意味着 v 将使用 $F'(x)$ 中的 u 值进行更新。综上可得，$f_v(x) = f'_v(x)$ 且 $F_{\Delta}(x)|_{X_0 \cup \ldots \cup X_{k+1}} = F'(x)|_{X_0 \cup \ldots \cup X_{k+1}}$，由递推可得，对于任意 x， $F_{\Delta}(x) = F'(x)$。□

算法2是我们对这一过程的贡献，它会移除对网络极限动力学而言并非必要的自动机。该算法分两步进行：首先，识别局部函数等价的自动机对（允许相差某个函数）。换言之，如果自动机 u 可以通过函数 g 由另一个自动机 v的局部函数计算得出，那么 u 就是非必要的，并且可以在网络中将所有对 x_u 的引用替换为 $g(x_v)$，而不会影响结果。当然，这只有在假设 u 和 v 同步更新时才成立；应用算法1后，正是这种情况。其次，该算法会迭代移除任何在网络中没有影响的自动机，也就是说，在网络交互图中没有可达邻居的自动机。这些自动机不属于任何环，也不会导向任何环，因此不会影响吸引子。

算法2是非确定性的；当证明任意一对自动机（u、 v）的局部函数可通过某个可逆函数g 等价时： $\Sigma \to \Sigma$，任一自动机都可以替代另一个自动机的影响，而无需

算法2 F_Δ的并行化算法，可缩减规模

输入
F 网络在 S上的局部函数 ，编码为电路
Δ 在 S上的更新调度

输出
F' 子集 S 上并行网络的局部函数，
编码为电路

令 $F' \leftarrow$ 对 F_Δ 应用算法1
令 $G_I(F')$ 为 F' 的交互有向图
对于 $(u, v), \in S^2$ 执行
如果 $\forall x \in \Sigma^n, f_u(x) = g(f_v(x))$ 则 ▷ 对于某个 $g: \Sigma \to \Sigma$
对于 $(u, w) \in E(G_I(F'))$ 执行
在 f_w 中应用替换 $x_u \mapsto g(x_v)$
当 $\exists u \in S$ 使得 u 在 G'_I 中没有可访问的邻居时执行
$S \leftarrow S \setminus \{u\}$ ▷ u 从网络中移除
返回 F'

优先选择。因此，可能得到多个结果网络，但正如后文将展示的那样，它们的极限动力学都是等价的。

虽然显然算法2总会终止，但其复杂性超出了确定性多项式范围，因为应用该算法意味着要解决 coNP-完全判定问题：对于所有可能的自动机对以及每个可能的函数 $g: \Sigma \to \Sigma$，判断两个布尔公式是否始终相等。因此，多项式时间实现该算法将（至少）意味着 P $=$ NP。考察那些能够轻易指出冗余的受限网络类时，这一极端结论会有所缓和，本文余下部分讨论的正是这种情况。

定理 2。对于任意 F_Δ，算法2返回一个网络 F'，使得 F_Δ 和 F'_π 的极限动力学同构。

证明。根据定理 1，将算法1应用于 F_Δ 后返回的网络 F' 与 F_Δ 具有相同的动力学。

算法2执行两类修改。

第一种操作是：如果发现某个自动机 u 与另一个自动机 v 的局部函数在某个 $g: \Sigma \to \Sigma$ 意义下等价，即 $f_u = g \circ f_v$，则用后者替换前者的影响。对于任意配置 x， $f_u(x)$ 与 $g(f_v(x))$ 的值始终相等。因此，在 u 的每个出邻居的局部函数中，用 $g(x_v)$ 替换变量 x_u 将导致相同的极限行为。完成此替换后，自动机 u 不再对网络产生任何影响。此外，它在 $G_I(F')$ 中原有的所有出邻居现在都成为 v 的出邻居。

第二个操作是迭代地移除不影响任何自动机的自动机。设 u 为这样一个被删除的自动机。考虑 G 中的一个极限环（x^1、x^2、.....、 x^k）。根据极限环的定义，对于任意 i，都有 $G(x^i) = x^{i+1}$，

$G(x^k) = x^1$，且 $x^i = x^j \Rightarrow i = j$。考虑某个 i 的分量 x^i_u 。由于 u 不影响任何自动机，因此 x^{i+1} 是 $x^i|_{S\setminus\{u\}}$ 的函数。由于整个序列是非周期的，子配置序列 $x^i|_{S\setminus\{u\}}$ 也是非周期的，并且吸引子在 F' 中得以保留。□

4 切向循环规模的约简

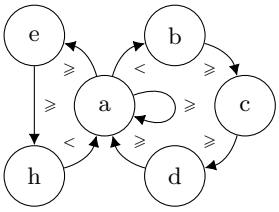
本节将刻画我们的算法在特定网络族上实现的规模约简。我们将由任意数量的环组成的自动机网络称为切向循环（TC），其中所有环共享一条由自动机组成的唯一通路 $\{C_1, C_2, \ldots, C_k\}$，该通路称为切线。切线的第一个自动机是唯一一个具有多个入邻居的自动机，称为中心自动机。图 4 展示了一个 TC 的部分结构，其中包含三个环以及一条长度为 0 的切线（环之间仅共享一个节点）。

4.1 块顺序切向循环的约简

算法2在 TC 上提供的规模约简可能相当显著，因为即使是并行更新的 TC，也可以通过尽可能合并不同的环来缩减规模。因此，该算法的约简能力不仅限于消除块顺序到并行更新转换中固有的冗余。事实上，图 4 给出了一个并行 TC 的示例，通过应用算法2，其规模得到了大幅约简。但是，经过这一过程，算法2的最终结果不再是 TC。

如上文所述，在自动机循环之后，TC 是具有生物学意义的复杂自动机网络中最简单的下一类情形。孤立循环和双循环都是 TC 的例子。为说明在块顺序更新调度下研究 TC 可以直接约简为在并行更新调度下研究 TC，我们提供了一种算法，将块顺序更新调度下的任意 TC 转换为并行更新调度下的 TC，使它们的极限动力学同构，并且其中心自动机的局部函数等价。具体做法是提前停止算法2的过程，以保留网络的 TC 形状。

算法2和算法3之间的唯一差异在于，后者限制了其执行的约简操作。如果发现两个局部函数在某个函数 g 意义下等价，则当且仅当这些局部函数是网络中心自动机的前一个局部函数的重复项时，算法3才会移除一个节点。移除作为某个环一部分的任意函数的重复项会合并两个环，使网络不再是切向循环，并且会使约简操作更难计数。由于算法3是算法2的一个变体，只执行较少的约简操作，因此定理 2 仍然适用于算法3。算法2和算法3的应用示例见图 4，二者之间的差异也得到了突出显示。



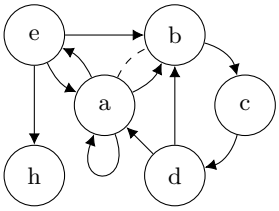
$f_a(x) = x_a \vee x_d \vee \neg x_h$
$f_b(x) = \neg x_a$
$f_c(x) = x_b$
$f_d(x) = x_c$
$f_e(x) = x_a$
$f_h(x) = x_e$

初始网络，
$\Delta = (\{h\}, \{a, c, d, e\}, \{b\})$

不适用

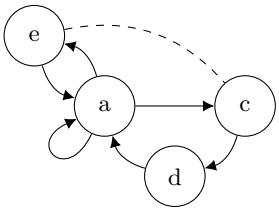
$f_a(x) = x_a \vee x_d \vee \neg\theta_h$
$f_b(x) = \neg\theta_a$
...

算法1，第一次循环（c、d, e 和 h 不变）



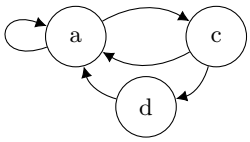
$f'_a(x) = x_a \vee x_d \vee \neg x_e$
$f'_b(x) = \neg(x_a \vee x_d \vee \neg x_e)$
$f'_c(x) = x_b$
$f'_d(x) = x_c$
$f'_e(x) = x_a$
$f'_h(x) = x_e$

算法1，第二次循环



$f'_a(x) = x_a \vee x_d \vee \neg x_e$
$f'_c(x) = \neg x_a$
$f'_d(x) = x_c$
$f'_e(x) = x_a$

算法2 在删除 h 并将 b 合并到 a。这是算法3结束处



$f'_a(x) = x_a \vee x_d \vee \neg x_c$
$f'_c(x) = x_a$
$f'_d(x) = \neg x_c$

算法2结束后将 e 合并到 c之后

图4. 算法1、2和3在一个示例网络上的应用。算法的不同步骤使用水平线表示并分隔。在每个步骤中，交互图或更新图以及局部函数均为右侧所示操作的结果。由于初始网络是一个TC，第四步表示算法3返回的结果，即一个规模更小的TC。第五步表示算法2返回的结果，该结果不是TC。交互有向图中的虚线连接了局部函数在否定函数意义下等价的自动机。只有第一张图表示为更新有向图，因为其他所有网络均采用并行更新。

算法3 块顺序更新调度下 TC 的并行化算法 F
更新调度 Δ，并可能缩小规模

输入
  F 网络的局部函数，定义在 S上 ，编码为电路
  Δ S上的更新调度
输出
  F' S的子集上的并行网络的局部函数，
  编码为电路
令 $F' \leftarrow$ 对 F_Δ应用算法1
令 $G_I(F')$为 F'的交互有向图
对于 $(u, v) \in S^2$，如果 u或 v在 $G_I(F')$ 中具有多个入邻居，则执行
  如果 $\forall x \in \Sigma^n, f_u(x) = g(f_v(x))$，则 ▷ 对于某个 $g: \Sigma \to \Sigma$
    对于 $(u, w) \in E(G_I(F'))$，执行
      在 f_w 中应用替换 $x_u \mapsto g(x_v)$
对于 $u \in S$ 执行
  如果 u在 G_I（F'）中没有可访问的邻居，则
    $S \leftarrow S \setminus \{u\}$ ▷ 从网络中移除 u
返回 F'

定理3. 设 F 为一个TC， Δ 为一个块顺序更新调度。算法3在 F_Δ 上执行的规模约简次数等于<条边在 F_Δ的更新有向图中的数量，且结果为一个TC。

证明. 算法1对 F_Δ的更新有向图中的每一条 <边执行一次替换。在本证明中，我们将说明，每一种可能的变换都意味着从网络中恰好移除一个节点。

对于任意这样的边（u， $<$， v），存在两种情况。 u要么是中心自动机，要么不是。无论哪种情况， $u \neq v$ ，因为相反情况将意味着某个自动机在其自身之前被严格更新。

如果我们假设 u 是中心自动机，这意味着 f_v 是一个只依赖于 x_u 的局部函数。因此，对于某个 $g: \Sigma \to \Sigma$，它可以写作 $f_v(x) = g(x_u)$。应用算法1后，我们得到 $f_v(x) = g(f_u(x))$，这意味着可以通过算法3移除 u 或 v 中的任意一个（但在此时不能同时移除二者）。

如果我们假设 u 不是中心自动机，这意味着 f_v 是一个包含 x_u 的任意公式，而 f_u 是某个 g 和某个 $w \in S$ 所对应的形如 $f_u(x) = g(x_w)$ 的函数。注意，根据 F 是 TC 这一假设， w 要么是路径中的前一个自动机，要么是中心自动机 v。因此，应用算法1会将 f_v 中对 x_u 的任何提及替换为 $g(x_w)$。此前， u 只有一个可访问的邻居，因为它属于连接到中心自动机的路径。这使得 x_u 在 F 的交互有向图中没有任何可访问的邻居，因此会被算法 3 移除。如果被移除的边属于一个环，这意味着该环的规模将缩小。如果该边属于切线，这意味着切线的规模将缩小。

因此，我们得到约简的数量至少等于网络更新有向图中 $<$-边的数量。现在假设除与 $<$-边相关的约简之外，还移除了某个额外的自动机 u 。首先注意，如果 u 没有可访问的邻居，那么在应用算法1之前它必然也没有可访问的邻居，因为在这两种情况下，外部自动机都不会彼此断开。现在假设 f_u 与某个 f_v 在某个 g 意义下等价。 u 和 v 都不可能是中心自动机，因为该函数的任何复制都会在第一种情况下得到处理。这证明约简的数量恰好等于 (F, Δ) 的更新有向图中 $<$-边的数量。

现在说明算法3的结果是一个 TC。如果初始网络中有一个中心自动机，那么算法结束时仍然存在唯一的中心自动机，即使原中心自动机在某次选定的约简中被移除了。在前一个网络中离开中心自动机的路径，在结果中仍然会离开中心自动机，数量相同，并且仍然共享某条切线。这些路径的长度以及切线都可能变小，但它们仍然终止于中心自动机。□

如果在 P $\neq$ NP 的假设下，算法2在最坏情况下无法做到多项式时间，那么算法3可以简化为以下规则：给定一个具有块顺序更新调度的 TC，通过将每个环缩减其更新有向图中包含的 $<$-边数量，我们可以得到等价并行 TC。该过程是二次的，因为我们只需检查由 Δ 定义的偏序所确定的可能的 $<$-边，其数量不超过 n^2。

4.2 进一步缩减并行布尔 TC

将算法2充分应用于布尔 TC（即定义在布尔字母表上的 TC），可以大幅缩小网络规模。由于任何非中心自动机的局部函数都是一元函数，因此，只要一对非中心局部函数受到同一个自动机的影响，它们就等价，差异至多为某个 $g\colon \Sigma \to \Sigma$。例如，如果中心自动机影响着另外三个自动机，而这三个自动机分别是三条链的起点，那么就可以将它们合并为一个。继续这一压缩过程，最终得到的网络规模仅与初始 TC 中最长环的长度相当。

对于非布尔 TC，这一过程并不简单，因为链上的局部函数可能无法通过模运算等方式求逆。尽管如此，仍然可以优化这类网络，但需要使用一组更复杂的替换。利用模组和输出函数已在一般情形下证明了这一点 [13]。以下定理对应于布尔情形，其证明采用了更经典的方法。图 4 最后两个步骤展示了该定理的一个应用示例。

定理 4. 设 F 是一个布尔 TC。对 F_π 应用算法2，将生成网络 F'，其规模等于 F 中最大环的规模。

证明。从初始 TC F 出发，所有直接受切线末端的自动机影响、但不是 u 的自动机，其局部

函数为 $f_v(x) = g(x_u)$ $f_w(x) = h(x_u)$ ....., 其中 g、h、.....: $\Sigma = \{0,1\} \to \Sigma$。由于根据假设，它们所表示的自动机都受到某个自动机的影响，因此所有这些函数 g、h、..... 都不是常函数。因此，它们只能定义为 x_u 的恒等函数或否定函数。结果，这些自动机中除一个以外的所有自动机都将被算法移除，因为它们在某个 g 意义下都是等价的。

通过仅取上一轮迭代所得的那个自动机所影响的所有自动机（不包括中心自动机），可以重复这一相同的论证。每一步都会将距离中心自动机相同的所有自动机合并。因此，在这一过程结束时，无论迭代过程中选择如何合并自动机，所得的自动机网络都将由 k 个自动机组成，其中 k 为 F 的最大环长度。□

5 应用：析取双循环

作为该算法的一个应用，也作为展示该算法缩减给定网络规模能力的示例，我们将转向析取双循环这一类。请注意，由于合取环与析取环具有同构动力学，因此该结果对合取双环同样成立 [12,11]。

在析取自动机网络中，若 x_u 在 f_v 中作为正变量出现，则边 (u, v) 带正号。若 x_u 在 f_v 中作为负变量出现，则边 (u, v) 带负号。若一个环包含偶数条负边，则称其为正环；否则称为负环。

析取双环是一种自动机网络，其交互有向图由两个在一个自动机处相交的自动机环组成。这个中心自动机的局部函数是一个析取子句。这类网络的定义非常简单，是继由单个环组成的布尔自动机环这一类网络之后，自然而直观的下一步。

这两类网络在并行更新调度 [12,7] 下均已得到刻画；也就是说，给定有关环大小、符号以及任意整数 k 的基本参数，已有明确公式（定义为可在多项式时间内计算的函数），可用于计算此类网络在并行更新调度下大小为 k 的极限环数量等。在本节中，我们将这种刻画扩展到块序列等价物，并说明如何应用我们的算法将网络约化为同一网络类中一个规模更小的实例。

此外，由于布尔自动机环和析取双循环都是 TC，我们的方法可以简化为以下简单规则：给定一个 TC F 和一个块顺序更新调度 Δ，计算更新有向图 $G_U(F_\Delta)$ 中的 $<$-边数量；对于每个环，在保持其符号不变的同时，从其大小中减去其中包含的此类边的数量；最终网络在并行更新调度下，与初始网络在 Δ 下具有同构极限动力学。这是定理3的一个简单应用，也是由算法3推导出的经验法则。

我们用 $DC(s, s', a, b)$ 表示环大小为 a、b，符号为 s、s' 的析取双循环。

定理5。设 $D = DC(s, s', a, b)$ 为析取双循环，Δ 为块顺序更新调度。对于 $G_U(F_\Delta)$ 中大小为 a (b) 的环上 $<$边的数量 A (B)，D_Δ 的极限动力学与 D'_π 的极限动力学同构，其中 $D' = DC(s, s', a - A, b - B)$。

证明。这是定理3的直接应用。 □

6 结论

本文提出了一种新颖的算法，可缩小自动机网络的规模，具体而言，是将网络的更新调度从块顺序更新转换为并行更新，同时保持极限动力学同构。尽管该算法在一般情况下计算开销高，但我们研究了自动机循环交集这一特定族，在该族上应用此算法十分简便。通过这项研究，我们发现所有块顺序切向循环的极限动力学都与并行切向循环的极限动力学同构。最后，我们将这一结论应用于布尔自动机双循环，以刻画其在块顺序更新调度下的行为。

现在我们已经清楚，并行更新调度与块顺序更新调度之间的区别在于，后者改变了信息沿网络各个部分传递的时序。特别是，切向循环等结构可以直接转换为具有更短环的等价并行网络。我们希望了解这种转换对更一般的网络族可能产生何种影响，以及是否存在其他网络族，使块顺序更新调度能够导出仍属于该网络族的等价并行网络。

展望未来，我们希望刻画出更多可以从网络中移除的冗余，以帮助计算其动力学。例如，我们目前对自动机循环的更复杂复合感兴趣，并且已经发现了一些等价关系，表明许多网络在极限动力学方面是等价的：自动机网络的复杂部分可以沿着环移动，而不会影响网络的极限动力学。

孤立路径也是缩减规模的有力候选对象。孤立路径是从一个环通向另一个环、但只能被经过一次的路径。尽管在许多情况下可以在不改变网络极限动力学的情况下完全缩减这些路径，例如当孤立路径是从网络的一部分通往另一部分的唯一途径时，我们当前的算法仍会保留此类路径。当多个孤立路径从相同部分离开并汇入相同部分时，我们必须谨慎处理，因为必须保持整个网络中信息的同步性。

致谢。本研究部分得到 ANR-18-CE40-0002 FANs 项目（PP 与 SS）、ECOS-SudCE19E02 SyDySy 项目（PP 与 SS）以及 STIC- AmSud 22-STIC-02 CAMA 项目（SS）的资助。

参考文献

1. J. Aracena、E. Goles、A. Moreira 和 L. Salinas。关于布尔网络中更新调度的鲁棒性。 *Biosystems*，97:1–8，2009。2. M.I. Davidich 和 S. Bornholdt。布尔网络模型预测裂殖酵母的细胞周期序列。 *PLoS One*，3:e1672，2008。3. J. Demongeot、A. Elena、M. Noual、S. Sen´e 和 F. Thuderoz。“Immunetworks”、相交电路与动力学。 *J. Theor. Biol.*，280:18–33，2011。4. J. Demongeot、E. Goles、M. Morvan、M. Noual 和 S. Sen´e。吸引盆作为生物复杂系统抵御边界条件的鲁棒性测度。*PLoS One*，5:e11793，2010。5. J. Demongeot、T. Melliti、M. Noual、D. Regnault 和 S. Sen´e。 *Automata and Complexity: Essays presented to Eric Goles on the occasion of his 70th birthday*，*Emergence*第 42 卷， *Complexity, Computation*，章节《关于布尔自动机孤立循环和相切双环的动力学》，第 145–178 页。Springer，2022。6. J. Demongeot、M. Noual 和 S. Sen´e。关于正、负布尔自动机电路的吸引子数量。载于 *Proceedings of WAINA'10*，第 782–789 页。IEEE Press，2010。7. J. Demongeot、M. Noual 和 S. Sen´e。布尔自动机电路动力学的组合学。 *Discr. Appl. Math.*，160:398–415，2012。8. Z. Gao、X. Chen 和 T. Ba。sar。合取布尔网络的稳定性结构。 *Automatica*，89:8–20，2018。9. S. A. Kauffman。随机构建的遗传网络中的代谢稳定性与表观遗传。 *J. Theor. Biol.*，22:437–467，1969。10. L. Mendoza 和 E. R. Alvarez-Buylla。 *Arabidopsis thaliana* 花形态发生遗传调控网络的动力学。 *J. Theor. Biol.*，193:307–319，1998。11. M. Noual。电路与相交电路的动力学。 *Proceedings of* 载于 *LATA'12* 7183 *LNCS* 433 444 2011 ，第 卷，页码 – 。Springer，。12. M. Noual。 *Updating Automata Networks*。博士论文，´里昂高等师范学院，2012。13. P. Perrotin。 *Simulation entre modèles de calcul naturel et modularité des réseauxd'automate*。博士论文，艾克斯-马赛大学，2021。14. F. Robert。有限集合上的迭代与收缩细胞自动机。 *Linear Algebra and its Applications*，29:393–412，1980。15. L. Salinas。 *Estudio de modelos discretos: estructura y dinámica*。博士论文，智利大学，圣地亚哥，智利，2008。16. R. Thomas。遗传控制电路的布尔形式化。 *J. Theor. Biol.*，42 563 585，1973：–。17. R. Thomas。关于系统逻辑结构与其产生多个稳态或持续振荡能力之间的关系。载于 *Numerical methods in the study of critical phenomena* 180 193 1981 ，页码 – 。Springer，。