arXiv:2302.03797v1 [cs.DS] 7 Feb 2023

人类并非总能变成老鼠：对称倒位排序的判定算法与复杂性

XinTong* Yixiao Yu† Ziyi Fang‡ Haitao Jiang§
Lusheng Wang¶ Binhai Zhu‖ Daming Zhu**

摘要

通过反转对排列进行排序是基因组重排领域的一个著名问题，在过去三十年中已得到充分研究。然而，在基因组进化过程中，重复片段的参与不可避免，尤其是在反转事件中。自1997年以来，人们发现了不少生物学证据，表明在许多基因组中，逆转区域的两侧通常各有一个反向重复序列。例如，反转会将 $+a+x-y-z-a$ 变为 $+a+z+y-x-a$，其中 $+a$ 和 $-a$构成一对反向重复序列。

这种类型的反转称为对称反转，可惜的是，直到最近才受到关注。在本文中，我们研究对称倒位排序问题，该问题要求通过一系列对称反转将一条染色体 A 转换为另一条染色体 B。对称倒位排序的判定问题称为 SSR（给定输入染色体 A 和 B 时，我们使用 $SSR(A,B)$；以下优化版本也采用类似记法）；相应的优化版本（即，当 $SSR(A,B)$的答案为“是”时，求将 A转换为 B所需的最少对称反转次数）称为 $SMSR(A,B)$。本文的主要结果总结如下，其中输入是一对染色体 A 和 B，且具有 n 个重复序列。

*山东大学计算机科学与技术学院，中国青岛。电子邮箱： xtong@mail.sdu.edu.cn。

†山东大学计算机科学与技术学院，中国青岛。电子邮箱： yixiaoyu@mail.sdu.edu.cn。

‡山东大学计算机科学与技术学院，中国青岛。电子邮箱： fangziyi@mail.sdu.edu.cn。§山东大学计算机科学与技术学院，中国青岛。电子邮箱： htjiang@sdu.edu.cn。

¶香港城市大学计算机科学系，中国香港九龙。电子邮箱： cswangl@cityu.edu.hk。

‖蒙大拿州立大学吉安福特计算学院，美国蒙大拿州博兹曼市 59717。电子邮箱： bhz@montana.edu。

**山东大学计算机科学与技术学院，中国青岛。电子邮箱： dmzhu@sdu.edu.cn。

1. 我们提出一种运行时间为 $O(n^2)$ 的算法，用于解决判定问题 $SSR(A,B)$，即判断染色体 A 能否通过一系列对称反转转换为 B。我们通过将该问题转化为圆图来得到这一结果。该圆图在传统圆图的基础上进行了显著扩展，并且必须证明一系列组合性质，才能成功回答这一判定问题。2. 我们为 $SMSR(A,B)$的一个特殊 2-平衡情形设计了一种运行时间为 $O(n^2)$ 的算法：染色体 A 和 B 的重复数均为 2，且每个重复序列在 A 和 B中都以不同方向出现两次。3. 我们证明，即使输入染色体的重复数至多为 2，SMSR 仍然是 NP 困难，从而表明上述优化结果已是可能达到的最佳结果。作为副产品，我们还证明了 $circle\ graphs$上的 $minimum\ Steiner\ tree$ 问题是 NP 困难，为一个悬而未决 38 年的问题确定了复杂性。

1 引言

20世纪80年代，人们发现了大量证据，表明某些物种拥有基本相同的基因集合，但基因顺序却各不相同 [HP94, PH86]。此后，在过去三十年间，通过重排操作对排列进行排序，逐渐成为计算生物学领域备受关注的课题。Sankoff 等人通过对基因组进行一些基本操作，正式定义了基因组重排事件，例如反转、转置和易位 [SLA92], ，其中反转操作的应用最为广泛 [KS95, FLR09, WPR19]。

通过反转对排列排序这一问题的复杂性，与基因是否带符号密切相关。Watterson等人 率先研究了通过反转对无符号排列进行排序的问题 [WEH82]。1997年，Caprara证明了该问题是 NP 困难的 [Cap99]。不久之后，Berman 等人 证明了该问题是 APX 困难的 [BK99]。Kececioglu 和 Sankoff 提出了该问题的首个多项式时间近似算法，近似比为 2 [KS95]。Christie 将近似比改进到了 1.5 [Chr98]。据我们所知，通过反转对无符号排列进行排序这一问题目前最好的近似比是 1.375 [BHK02]。至于更符合实际的通过反转对带符号排列进行排序问题，Hannenhalli 和 Pevzner 为该问题提出了一种运行时间为 $O(n^4)$ 的精确算法，其中 n 是给定排列（基因组）中的基因数量 [HP99]。随后，Kaplan 等人 将时间复杂度改进到了 $O(n^2)$ [KST00]。目前最好的运行时间是 Tannier 等人 提出的 $O(n^{1.5}\sqrt{\log n})$ [TBS07]。

另一方面，有证据表明，发生反转的断点在基因组中可能具有某些特殊属性 [LCG09, San09]。早在 1997 年，一些研究就指出，哺乳动物和果蝇基因组中的断点通常与重复元件相关联 [TVO11, BBK04, APC03, SIW97]。事实上，著名的“位点特异性重组”，

它在“基因敲除”中有重要应用 [Sau87, SH88, OCM92]., 也符合这一规律。然而，重复元件为何以及如何在基因组重排中发挥重要作用，仍不清楚。最近，Wang 等人对铜绿假单胞菌、大肠杆菌、结核分枝杆菌和希瓦氏菌属等多种细菌的不同菌株进行了系统比较研究 [WW18, WLG17]。他们的研究进一步表明，在反转、转置、反向区块交换等各种重排事件中，重复序列与重排片段的两端相关联，因此在重排事件发生后，这些重复序列左右两侧的邻域保持不变。聚焦于反转事件，逆转区域通常两侧各有一个反向重复序列 [SIW97]。下面这个真实案例来自铜绿假单胞菌菌株 [WLG17]。这一现象也能更好地解释著名的“断点复用”现象（在比较人类与小鼠时，这是一个有趣的发现，并得到了详细讨论）为何会发生 [PT03]。

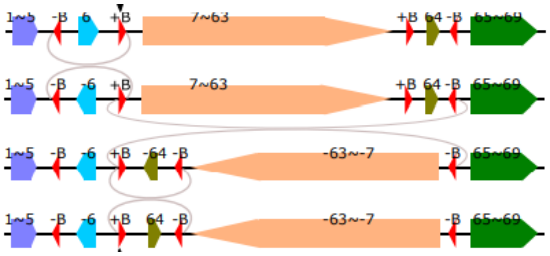

图 1：三次对称反转中，重复序列“$+B$”被使用了三次。

本文提出了一种名为对称倒位排序的新模型，该模型要求染色体上的每个倒位区域两侧都由一对互为反向的重复序列夹定。我们研究对称倒位排序判定问题（简称 SSR），即判断能否通过一系列对称反转将一条染色体转换为另一条染色体。我们设计了一种运行时间为 $O(n^2)$ 的算法来解决这一判定问题。我们还研究了优化版本（称为 SMSR），即使用最少数量的对称反转将一条染色体转换为另一条染色体。对于 SMSR 的一种特殊 2-平衡情形，我们设计了一种运行时间为 $O(n^2)$ 的算法；在该情形下，染色体的重复数为 2，且每个重复序列在每条染色体上都以不同方向出现两次。最后，我们证明，即使每个重复序列在每条输入染色体中的重复数最多为 2，优化问题 SMSR 仍然是 NP 困难。

在 NP 困难性证明中，我们建立了本问题与圆图上的最小斯坦纳树问题之间的联系。Johnson 在 1985 [Joh85] 中指出，圆图上的最小斯坦纳树问题被认为属于 P。最近，Figueiredo等人重新审视了 Johnson表，仍将该问题标记为属于 P [FMS22], , 同时将参考文献标为“进行中”。在此，我们澄清，圆上的

图事实上是 NP 困难的，从而解决了这个悬而未决 38 年的问题。

本文结构安排如下。第 2 节给出一些定义。随后，我们在第 3 节提出一种算法，用于在一种特殊情形下求解 SSR：输入染色体的重复数为 2。第 4 节针对特殊的 2-平衡情形，提出一种求解 SMSR 的多项式算法。第5节提出一种求解一般情形下 SSR 的算法。第 6 节借助关于圆图上的最小斯坦纳树问题的新 NP 困难结果，证明当染色体的重复数为 2 时，SMSR 是 NP 困难的。最后，第 7 节给出结论。

2 预备知识

在基因组重排相关文献中，我们通常有一组整数$\Sigma_1 = \{1, \cdots, g\}$，其中每个整数代表一段较长的 DNA 序列（共线区块或基因）。为简便起见，以下均使用“基因”一词。由于我们将研究对称反转，因此将 $\Sigma_2 = \{r_0, r_1, r_2, \cdots, r_t\}$ 定义为一组符号，其中每个符号称为重复，表示相较于基因而言较短的 DNA 序列。接着，我们将 $\Sigma = \Sigma_1 \cup \Sigma_2$ 设为整个染色体的字母表。

由于逆转操作在染色体内部进行，因此，为了便于研究，可以将基因组视为一条染色体，即每个基因组都是单例，并且只包含一条染色体。这里假设每个基因在一条染色体上恰好出现一次；另一方面，顾名思义，一个重复序列可能出现多次。基因/重复序列 x 在一条染色体上可能以两种不同的方向出现，即可以是 $+x$ 或 $-x$。因此，我们所关注的每条染色体都表示为一个带符号整数/符号序列。

一个基因/重复序列 x 以两种方向出现的次数称为 x 在染色体 π上的重复数，记为 $dp[x, \pi]$。染色体 π的重复数记为 $dp[\pi]$，是其中各重复序列的最大重复数。例如，染色体 $\pi = [+r_0, +1, -r, +2, +r, -r_0]$，$dp[1, \pi] = dp[2, \pi] = 1$、$dp[r_0, \pi] = dp[r, \pi] = 2$和 $dp[\pi] = 2$。如果两条染色体 π_1 和 π_2 的所有基因和重复序列的重复数都相同，则称它们相关。设 $|x| \in \Sigma$ 为一个整数或符号，$+|x|$ 和 $-|x|$为 $|x|$的两次出现，且 $+|x|$ 和 $-|x|$ 的方向不同。由 n 个基因/重复序列组成的染色体记为 $\pi = [x_1, x_2, \ldots, x_{n-1}, x_n]$。线性染色体有两个末端，可以从任一端读到另一端，因此，染色体 $\pi = [x_1, x_2, \ldots, x_{n-1}, x_n]$ 也可以表示为 $[-x_n, -x_{n-1}, \ldots, -x_2, -x_1]$，这称为 π的反转并取负形式。

〈style id='1'〉反转 〈/style〉是一种操作，用于反转染色体上的一段连续整数（或符号）。〈style id='3'〉对称反转 〈/style〉是一种反转，被反转的片段两侧由方向不同的一对相同重复序列所围绕，即对于某个 $r \in \Sigma_2$，形式为 ($+r$〈style id='6'〉, 〈/style〉 $\cdots$ 〈style id='9'〉, 〈/style〉 $-r$) 或 ($-r$〈style id='13'〉, 〈/style〉 $\cdots$ 〈style id='16'〉, 〈/style〉 $+r$)。换句话说，设 $\pi =[x_1, x_2, \ldots, x_n]$ 为一条染色体。反转 ρ(i〈style id='24'〉, 〈/style〉 j) ($1 \le i < j \le n$) 会反转片段 $[x_i, x_{i+1}, \ldots, x_{j-1}, x_j]$，并得到 $\pi' = [x_1, x_2, \ldots, x_{i-1}, -x_j, -x_{j-1},$

〈style id='1'〉. .. , 〈/style〉 $-x_{i+1}$ 〈style id='4'〉 , 〈/style〉 $-x_i$ 〈style id='7'〉 , 〈/style〉 x_{j+1} 〈style id='10'〉 , . .. , 〈/style〉 $x_n]$。如果 $x_i = -x_j$，我们就称 $\rho(i$ 〈style id='16'〉 , 〈/style〉 $j)$ 是 $|x_i|$ 上的一次对称反转。反转整条染色体不会改变整数的相对顺序，只会改变它们的符号，因此我们假设每条染色体两侧分别由 $+r_0$ 和 $-r_0$ 夹着；随后，在 $+r_0$ 和 $-r_0$ 之间执行一次对称反转，就会将染色体变为其反转取负形式。

再次以一个简单的例子说明，设 $\pi = [+r_0, +1, -r_1, +2, +r_2, +r_1, +r_2, -r_0]$，则对 r_1 进行一次对称反转可得到 $\pi' = [+r_0, +1, -r_1, -r_2, -2, +r_1, +r_2, -r_0]$。

现在，我们正式定义本文将要研究的问题 .

定义 2.1 对称倒位排序，简称***SSR*** 。

实例： 两个相关染色体 π 和 τ，满足 $dp[\pi] = dp[\tau] \geq 2$。问题： 是否存在一系列对称反转，将 π 转换为τ?

定义 2.2 最少对称倒位排序，简称***SMSR*** 。

实例： 两个相关染色体 π 和 τ，满足 $dp[\pi] = dp[\tau] \geq 2$，以及一个整数 m。

问题： 是否存在一系列对称反转 $\rho_1, \rho_2, \ldots, \rho_m$，将 π 变换为 τ，并使 m 最小?

将带符号基因/重复序列转换为无符号形式有一种标准方法。设 $\pi = [x_0, x_1, \ldots, x_{n+1}]$ 为一条染色体，其中每个基因/重复序列 π 的出现位置，例如x_i ($0 \leq i \leq n+1$)，都用一对有序节点表示，即 $l(x_i)$ 和$r(x_i)$。如果 x_i 的符号为 $+$，那么 $l(x_i) = |x_i|^h$ 且 $r(x_i) = |x_i|^t$；否则，$l(x_i) = |x_i|^t$ 且 $r(x_i) = |x_i|^h$。注意，如果 x_i 和 x_j $(i \neq j)$ 是同一重复序列的不同出现位置，即 $|x_i| = |x_j|$，那么 $l(x_i)$、$l(x_j)$、 $r(x_i)$ 和 $r(x_j)$ 仅对应两个节点 $|x_i|^h$ 和 $|x_i|^t$ 。因此， π 也可表示为 $[l(x_0)\ r(x_0)\ l(x_1)\ r(x_1) \ldots, l(x_{n+1})\ r(x_{n+1})]$。我们称 $r(x_i)$ 和 $l(x_{i+1})$（其中 $0 \leq i \leq n$）构成一个邻接，记作 $\langle r(x_i)\ l(x_{i+1})\rangle$。（注意，在染色体 π的带符号表示中，我们只需说$\langle x_i, x_{i+1}\rangle$ 构成一个邻接；此外，$\langle x_i, x_{i+1}\rangle = \langle -x_{i+1}, -x_i\rangle$。）另外，我们称邻接 $\langle r(x_i)\ l(x_{i+1})\rangle$ 与 x_i 和 x_{i+1}相关联。令$\mathcal{A}[\pi]$ 表示 π的邻接多重集。我们以染色体$\pi = [+r_0, +1, -r_1, +2, +r_1, -r_0]$ 为例来说明上述记法。邻接多重集为 $\mathcal{A}[\pi] = \{\langle r_0^t, 1^h\rangle, \langle 1^t, r_1^t\rangle, \langle r_1^h, 2^h\rangle, \langle 2^t, r_1^h\rangle, \langle r_1^t, r_0^t\rangle\}$， π也可视为 $[r_0^h, r_0^t, 1^h, 1^t, r_1^t, r_1^h, 2^h, 2^t, r_1^h, r_1^t, r_0^t, r_0^h]$。

引理 2.1 设 π 为一个染色体， π' 由 π 经过一次对称反转得到。那么 $\mathcal{A}[\pi] = \mathcal{A}[\pi']$。

证明。显然，在 $+r_0$ 和$-r_0$ 之间执行对称反转不会改变 $\mathcal{A}[\pi]$。假设在染色体 $\pi = [x_0, x_1, \ldots, x_{n+1}]$ 上执行对称反转 $\rho(i, j)$，使得 $x_i = -x_j$，其中 $1 \leq i < j \leq n$，并得到 $\pi' = \pi \bullet \rho(i, j) = [x_0, x_1, \ldots, x_{i-1}, -x_j, -x_{j-1}, \ldots, -x_{i+1}, -x_i, x_{j+1}, \ldots, x_{n+1}]$。那么， $\rho(i, j)$会打断两个邻接关系 $\langle r(x_{i-1})\ l(x_i)\rangle$ 和 $\langle r(x_j)\ l(x_{j+1})\rangle$，并建立两个新的邻接关系 $\langle r(x_{i-1})\ l(-x_j)\rangle$ 和$\langle r(-x_i)\ l(x_{j+1})\rangle$。由于 $x_i = -x_j$, $l(x_i) = l(-x_j)$ 且 $r(x_j) = r(-x_i)$，因此

$\langle r(x_{i-1})\ l(x_i)\rangle = \langle r(x_{i-1})\ l(-x_j)\rangle$ 以及 $\langle r(x_j)\ l(x_{j+1})\rangle = \langle r(-x_i)\ l(x_{j+1})\rangle$。因此，$\mathcal{A}[\pi] = \mathcal{A}[\pi']$。□

事实上，引理 2.1 给出了回答 **SSR** 判定问题的一个必要条件。

定理 2.1 若 $\mathcal{A}[\pi] \neq \mathcal{A}[\tau]$，则无法通过一系列对称反转将染色体 π 转变为 τ。

一个简单的反例是 $\pi = [+r_0, +r_1, -2, +r_1, -1, -r_0]$ 和 $\tau = [+r_0, -r_1, +2, -r_1, +1, -r_0]$。很容易验证 $\mathcal{A}[\pi] \neq \mathcal{A}[\tau]$，这意味着无法通过对称反转将 π 转变为 τ。在下一节中，作为热身，我们先解决每个重复在 π 和 τ中至多出现两次的情形。尽管该方法并不特别复杂，但我们希望本文的讲解和一些概念能帮助读者更好地理解第 5 节一般情形的细节。

3 一个 $O(n^2)$ 重复数为 2 的 SSR 算法

本节考虑一种特殊情况：两个相关染色体 π 和 τ 的重复数均为 2。也就是说，$\mathcal{A}[\pi] = \mathcal{A}[\tau]$ 和 $dp[\pi] = dp[\tau] = 2$。我们将设计一个运行时间为 $O(n^2)$ 的算法，用于判断是否存在一系列对称反转，能将 π 转换为 τ。

注意，$\mathcal{A}[\pi]$ 是一个多重集，其中某个邻接可能出现不止一次。当染色体中每个重复的重复数至多为 2 时，同一个邻接在 $\mathcal{A}[\pi]$ 中至多出现两次。

设 $\pi = [x_0, x_1, \ldots, x_{n+1}]$ 是一个染色体。设 x_i 和 x_j 是重复 x 的两次出现，x_{i+1} 和 x_{j+1} 是 π 中另一个重复 x' 的两次出现。如果 $r(x_i) = r(x_j)$ 且 $l(x_{i+1}) = l(x_{j+1})$（或 $r(x_i) = l(x_j)$ 且 $l(x_{i+1}) = r(x_{j-1})$），我们就称 $|x_i|$ 和 $|x_{i+1}|$ 是冗余的。在这种情况下，邻接 $\langle r(x_i)\ l(x_{i+1})\rangle$ 出现了两次。事实上，这是邻接能够出现两次的唯一情况。例如：$\pi = [+r_0, +r_1, -r_2, +1, +r_2, -r_1, -r_0]$，其中邻接 $\langle +r_1, -r_2\rangle$ 出现了两次（第二次为负号），因此 r_1 和 r_2 是冗余的。下面的引理说明，如果 x_i 和 x_j 冗余，我们只需使用其中一个进行反转，另一个则可以从染色体中删除，从而使每个邻接只出现一次。

引理 3.1 给定两个染色体 $\pi = [x_0, x_1, \ldots, x_{n+1}]$ 和 τ，使得$\mathcal{A}[\pi] = \mathcal{A}[\tau]$。设 $|x_i|$ 和 $|x_{i+1}|$ 是 π 中的两个冗余重复序列。分别从 π 和 τ中删除 $|x_{i+1}|$ 的两个出现位置后，得到染色体π' 和 τ'。当且仅当 π' 可以通过一系列对称反转变换为 τ' 时，π 可以通过一系列对称反转变换为 τ。

证明。不失一般性，我们假设 $r(x_i) = r(x_j) = |x_i|^a$ 且$l(x_{i+1}) = l(x_{j+1}) = |x_{i+1}|^b$，其中 a，$b \in \{h, t\}$。另一种情况的证明类似。

(⇒) 假设存在一个对称反转序列， ρ_1， ρ_2， ...， ρ_m，将 π 变换为 τ，具体而言，对于每个 $1 \le k \le m$，都有 $\pi_0 = \pi, \pi_k = \pi_{k-1} \bullet \rho_k$，并且 $\pi_m = \tau$。假设存在一个对称反转 $\rho_k(1 \le k \le m)$，它位于 $|x_{i+1}|$上。引理 2.1 保证 $\langle |x_i|^a, |x_{i+1}|^b \rangle$ 仍在 $\mathcal{A}[\pi_{k-1}]$中出现两次。由于 $|x_{i+1}|$ 的两次出现位置在 π_{k-1}中的符号不同，因此两个 $|x_i|^a$分别位于两个 $|x_{i+1}|^b$的不同侧，这意味着某个 $|x_i|^a$ 是 $|x_i|$ 某次出现的左节点，而另一个$|x_i|^a$ 是 $|x_i|$另一次出现的右节点。因此， $|x_i|$ 的两次出现位置在 π_{k-1}中的符号也不同。显然，对 $|x_i|$ 执行对称反转也会将 π_{k-1} 变换为 π_k。

(⇐) 假设存在一个对称反转序列， ρ'_1， ρ'_2， ....., $\rho'_{m'}$，它将 π' 转换为 τ'。设对应的染色体为 $\pi'_0 = \pi'$、π'_1、.....、 $\pi'_{m'} = \tau'$，其中对于每个 $1 \le k' \le m'$，$\pi_{k'} = \pi_{k'-1} \bullet \rho'_{k'}$。通过在 $\pi'_{k'}$ 中将 x_i 替换为 $[x_i, x_{i+1}]$，并将 $-x_i$ 替换为 $[-x_{i+1}, -x_i]$，即可得到 $\pi'_{k'}$。显然，$\pi'_0 = \pi$ 且 $\pi'_{m'} = \tau$。对于每个 $1 \le k' \le m'$，$\rho'_{k'}$ 可作用于 $\pi'_{k'-1}$，因为 $\pi'_{k'-1}$ 上的所有元素与 $\pi'_{k'-1}$ 上的元素符号相同；此外，$\pi'_{k'} = \pi'_{k'-1} \bullet \rho'_{k'}$，因为形式为 $\langle |x_i|^a, |x_{i+1}|^b \rangle$ 的两个邻接关系不会被 $\rho'_{k'}$ 改变。□

关于前面的例子， $\pi = [+r_0, +r_1, -r_2, +1, +r_2, -r_1, -r_0]$，其中 r_1 和 r_2 是冗余的，根据上述引理，可以得到$\pi' = [+r_0, +r_1, +1, -r_1, -r_0]$。这实际上等价于将邻接 $\langle +r_1, -r_2 \rangle$ 替换为 r_1，并将 $\langle +r_2, -r_1 \rangle$ 替换为 $-r_1$。

如果染色体 π 中的每个邻接只出现一次，则称其为简单 染色体。根据引理 3.1，我们可以从染色体中移除冗余重复的两个出现位置。因此，如果 $dp[\pi] = dp[\tau] = 2$，我们总可以假设 π 和 τ 都是简单的。由此，两个对应邻接集 $\mathcal{A}[\pi]$ 和 $\mathcal{A}[\tau]$之间存在唯一的双射。我们称任意一对相同邻接彼此匹配。

对于每个带有 $dp[\pi, x] = dp[\tau, x] = 2$ 的重复序列 x，设 x_i、 x_j 是 x 在 π 中的两次出现位置， $y_{i'}$、 $y_{j'}$ 是 x 在 τ 中的两次出现位置，则 π 中与 x_i 和 x_j 相关联的邻接关系有四种： $\langle r(x_{i-1})\ l(x_i) \rangle$、 $\langle r(x_i)\ l(x_{i+1}) \rangle$、$\langle r(x_{j-1})\ l(x_j) \rangle$、 $\langle r(x_j)\ l(x_{j+1}) \rangle$。类似地， τ 中与 $y_{i'}$ 和 $y_{j'}$ 相关联的邻接关系也有四种。如果 $\langle r(x_{i-1})\ l(x_i) \rangle$ 和 $\langle r(x_i)\ l(x_{i+1}) \rangle$ 分别匹配到与 $y_{i'}$ 相关联的两个邻接关系，或分别匹配到与 $y_{j'}$ 相关联的两个邻接关系，我们就称 x 是邻居一致重复。也就是说，两个染色体中 x_i 的左右邻居相同。注意，如果 x 是邻居一致的，那么 $\mathcal{A}[\pi] = \mathcal{A}[\tau]$ 还意味着，其他出现位置 x_j 的左右邻居在这两个染色体中也相同。如果 $\langle r(x_{i-1})\ l(x_i) \rangle$ 和$\langle r(x_i)\ l(x_{i+1}) \rangle$ 分别匹配到两个邻接关系，其中一个与$y_{i'}$ 相关联，另一个与 $y_{j'}$ 相关联，那么 x 就是邻居不一致重复。在 π 中只出现一次的基因和重复序列，也定义为邻居一致。（示例见图 2。）根据定义以及 $\mathcal{A}[\pi] = \mathcal{A}[\tau]$ 这一事实，我们有

命题 3.1 对重复执行对称反转，会使该重复从邻居一致变为邻居不一致，或反之。（见图-

图 2。）

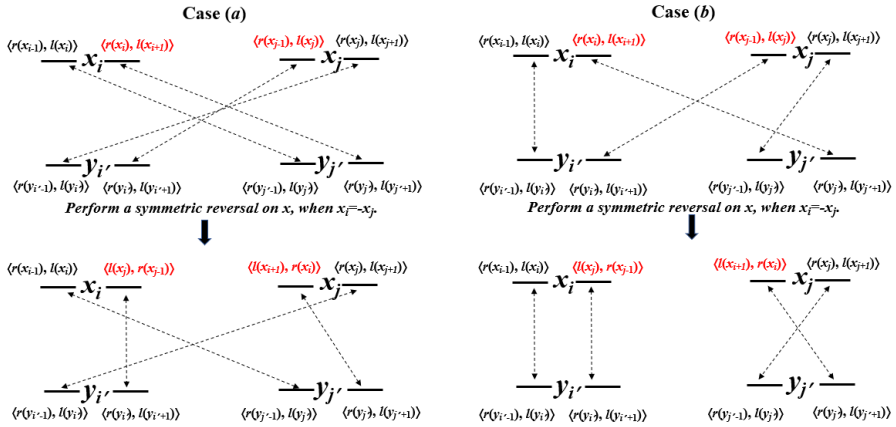

图 2： x_i、 x_j 是 π 中 x 的两个出现位置，且具有 $x_i = -x_j$； $y_{i'}$、 $y_{j'}$是 τ中 x 的两个出现位置。情况（a）： x 是邻居一致的，通过对其自身进行反转将变为邻居不一致。情况（b）： x 是邻居不一致的，通过对其自身进行反转将变为邻居一致。

定理 3.1 给定两个简单相关染色体 π^* 和 τ，当且仅当 $\mathcal{A}[\pi^*] = \mathcal{A}[\tau]$，且每个重复都是邻居一致的，其中 $dp[\pi^*] = dp[\tau] = 2$、 $\pi^* = \tau$。

证明。 假设 $\pi^* = [x_0, x_1, \ldots, x_n, x_{n+1}]$ 和 $\tau = [y_0, y_1, \ldots, y_n, y_{n+1}]$，其中 $x_0 = y_0 = +r_0$ 和 $x_{n+1} = y_{n+1} = -r_0$。

充分性部分显然为真 ，因为我们有 $x_i = y_i$ 对于 $0 \le i \le n+1$ 。现在我们通过归纳法证明必要性。我们的归纳假设是，对于 $0 \le i \le n+1$, $x_i = y_i$。初始时，由于 $\langle r_0^t, l(x_1) \rangle = \langle r_0^t, l(y_1) \rangle$，我们有 $x_0 = y_0 = +r_0$ 和 $x_1 = y_1$。在归纳步骤中，考虑 x_{i+1}。由于 $\mathcal{A}[\pi^*] = \mathcal{A}[\tau]$，邻接关系 $\langle r(x_i)\ l(x_{i+1}) \rangle$ 同时出现在 $\mathcal{A}[\pi^*]$和 $\mathcal{A}[\tau]$中。由于 $|x_i|$ 是偶数，两个邻接关系 $\langle r(x_{i-1})\ l(x_i) \rangle$ 和 $\langle r(x_i)\ l(x_{i+1}) \rangle$ 必须与 τ 中关联于 $|x_i|$的某个单次出现的两个邻接关系相匹配。根据归纳假设， $x_{i-1} = y_{i-1}$ 和 $x_i = y_i$， $\langle r(x_{i-1})\ l(x_i) \rangle$已与 $\langle r(y_{i-1})\ l(y_i) \rangle$匹配。因此， $\langle r(x_i)\ l(x_{i+1}) \rangle$ 必须与$\langle r(y_i)\ l(y_{i+1}) \rangle$匹配；再结合 $x_i = y_i$，我们得到 $l(x_{i+1}) = l(y_{i+1})$ 以及 $x_{i+1} = y_{i+1}$。□

根据命题 3.1 和定理 3.1，要将 π 转换为 τ，只需对每个邻居不一致重复执行奇数次（至少 1 次）对称反转，并对每个邻居一致重复执行偶数次（可以为 0 次）对称反转。此后，我们也将邻居一致重复（分别将邻居不一致重复）称为 偶数 （分别称为 奇数）重复。

在 π 和 τ 之间寻找一系列对称反转，主要难点在于每次都要选择一个“正确的”对称反转。注意，对于重复 x 的一对出现位置（$x_i,,x_j$），其方向当前可能相同，但经过若干次反转后，x_i 和 x_j 的方向可能不同。我们只能对方向不同的重复出现位置对执行反转。因此，在恰当的时机选择“正确的”对称反转至关重要。下面我们将使用“交集”图来处理这一问题。

假设给定两个简单相关染色体 π 和 τ，其中有$dp[\pi] = dp[\tau] = 2$和 $\mathcal{A}[\pi] = \mathcal{A}[\tau]$。在这种情况下，染色体中的每个重复都表示一个区间，该区间由该重复的两次出现位置确定。因此，我们可以构造交图 $IG(\pi,\tau) = (V[\pi], E[\pi])$。对于每个满足 $dp[\pi, x] = 2$ 的重复x，构造一个顶点 $x \in V_\pi$，并设置其权重：若x 为偶数，则为 $\omega(x) = 2$；若 x 为奇数，则为 $\omega(x) = 1$；若 x 在 π 中的两次出现位置的符号不同，则将 x 的颜色设为黑色，否则设为白色。当且仅当 x 和 y 的出现位置在 π中交替出现时，在两个顶点 x 和 y 之间连一条边。也就是说，设 x在 π 中的两次出现位置为 x_i 和 x_j $(i < j)$，设 y 在 π中的两次出现位置为 x_k和 x_l $(k < l)$，当且仅当 $i < k < j < l$ 或 $k < i < l < j$时，顶点 x 和 y 之间有一条边。$V[\pi]$中有三类顶点：黑色顶点（记作 $V_b[\pi]$）、权重为 1 的白色顶点（记作 $V_w^1[\pi]$）以及权重为 2 的白色顶点（记作 $V_w^2[\pi]$）。因此，$V[\pi] = V_b[\pi] \cup V_w^1[\pi] \cup V_w^2[\pi]$。事实上，忽略所有顶点的权重和颜色时，该交图是一个圆图。

引理 3.2 在 $IG(\pi,\tau)$ 中，权重为 *1* 的单个白色顶点不能构成一个连通分量。

证明。我们用反证法证明。假设 x 是一个权重为 1 的白色顶点，且构成 $IG(\pi, \tau)$ 的一个连通分量。设 x 的两次出现分别为 x_i 和 x_j $(i < j)$，位于 π 中，以及 y_k 和 y_l，位于 τ 中。由于 x 是奇数，不失一般性，假设 $\langle r(x_i)\ l(x_{i+1})\rangle$ 和 $\langle r(x_{j-1})\ l(x_j)\rangle$ 与两个都关联于 y_k 的邻接关系匹配。由于 x 是 $IG(\pi, \tau)$ 中的孤立顶点，$|x_{i+1}|$、.....、$|x_{j-1}|$ 的所有其他出现位置也都必须位于 π 中的 x_i 和 x_j 之间。

注意，在 $\mathcal{A}[\pi]$和 $\mathcal{A}[\tau]$中，每个邻接关系都是唯一的。如果 y_{k+1} 是 $|x_{i+1}|$的一次出现，那么邻接关系 $\langle r(y_{k+1})\ l(y_{k+2})\rangle$ 必须与位于 π 中 x_i 和 x_j 之间的某个邻接关系匹配。因此，$|y_{k+2}|$的一次出现也位于 π 中 x_i 和 x_j 之间，另一处出现（如果存在）也一样。因此，与 $|y_{k+2}|$的各次出现相关联的所有邻接关系都位于 π 中 x_i 和 x_j 之间，这意味着 y_{k+3} 存在。只有当某个 y_{k+t} 是 $|x_j|$的一次出现，同时邻接关系 $\langle r(y_{k+t})\ l(y_{k+t+1})\rangle = \langle r(x_{j-1})\ l(x_j)\rangle$ 也成立时，递归才会终止。这与$\langle r(y_{k-1})\ l(y_k)\rangle = \langle r(x_{i-1})\ l(x_i)\rangle$矛盾。

当 y_{k+1} 是 $|x_{j-1}|$ 的一次出现时，论证过程类似。 □

对于 $IG(\pi, \tau)$ 中的每个顶点 x，令 $N(x)$ 表示与 x 相邻的顶点集。对于 $IG(\pi, \tau)$ 中的黑色顶点 x，在 π 中对 x 执行对称反转，可得到 π'，其中交图 $IG(\pi') = (V[\pi'], E[\pi'])$ 可根据以下三条规则由 $IG(\pi, \tau)$ 推导得到：

• 规则 I：对于 $IG(\pi,\ \tau)$ 中的每个顶点 $v \in N(x)$，将其颜色从黑色改为白色，反之亦然。

• 规则 II：对于 $IG(\pi,\ \tau)$ 中的每一对顶点 $u,\ v \in N(x)$，若 $(u,\ v) \in E[\pi]$，则 $E[\pi'] = E[\pi] - \{(u,\ v)\}$；若 $(u,\ v) \notin E[\pi]$，则 $E[\pi'] = E[\pi] \cup \{(u,\ v)\}$。

• 规则 III：将 x 的权重减一；若 $\omega(x) > 0$，则 $V[\pi'] = V[\pi]$；若 $\omega(x) = 0$，则 $V[\pi'] = V[\pi] - \{x\}$。

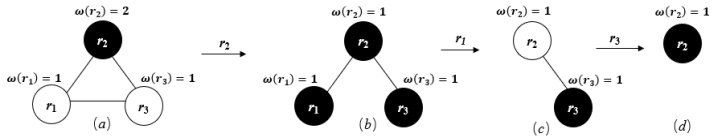

图3：$\pi = [+r_0, +r_1, +1, +r_2, +r_3, +r_1, -r_2, -2, +r_3, -r_0]$，以及 $\tau = [+r_0, +r_1, -r_2, -2, +r_3, +r_1, +1, +r_2, +r_3, -r_0]$。$\mathcal{A}[\pi] = \mathcal{A}[\tau] = \{\langle r_0^t, r_1^h\rangle, \langle r_1^t, 1^h\rangle, \langle 1^t, r_2^h\rangle, \langle r_2^t, r_3^h\rangle, \langle r_3^t, r_1^h\rangle, \langle r_1^t, r_2^t\rangle, \langle r_2^h, 2^t\rangle, \langle 2^h, r_3^h\rangle, \langle r_3^t, r_0^t\rangle\}$。重复 r_1 和 r_3 为奇数，而重复 r_2 为偶数。(*a*) 交集图$IG(\pi,\ \tau)$，对重复 r_2 执行对称反转会将 π 转换为$\pi' = [+r_0,\ +r_1,\ +1,\ +r_2,\ -r_1,\ -r_3,\ -r_2,\ -2,\ +r_3,\ -r_0]$，(*b*) 交集图 $IG(\pi',\ \tau)$。(*c*) 对重复 r_1执行对称反转后的交集图。(*d*) 对重复 r_3 执行对称反转后的交集图。

如果 x 是 $IG(\pi,\ \tau)$ 中的黑色顶点，且 $\omega(x) = 1$，那么对 π 中的 x 执行对称反转后得到 π'。设在 $IG(\pi',\ \tau)$ 中删除 x 后产生的连通分量为 C_1，C_2、.....、C_m，下面介绍执行该对称反转的一些性质。

引理 3.3 在每个 C_i $(1 \le i \le m)$ 中，至少存在一个顶点 z_i，使得 $z_i \in N(x)$ 位于 $IG(\pi、\ \tau)$ 中。

证明。考虑删除 x之前 $IG(\pi',\ \tau)$ 的情形，此时所有连通分量 C_1、C_2、.....、C_m 都处于同一个连通分量中，该连通分量还包含 x。但删除 x后，它们立即变成 m 个彼此分离的连通分量。□

引理 *3.4* 设 x' 是黑色顶点，$\omega(x) = \omega(x') = 1$，且 $x' \in N(x)$ 位于 $IG(\pi、\ \tau)$ 中。对 π中的 x执行对称反转后，设 y是连通分量 C_i中的一个顶点，且 x' 位于连通分量 C_j、$i \ne j$中。设对 π中的 x' 执行对称反转后得到的染色体为 π''，那么 y在 $IG(\pi'、\ \tau)$ 和 $IG(\pi''、\ \tau)$ 中的颜色相同。

证明。我们通过考虑以下两种情况来进行证明：（1）$y \in N(x)$；（2）$y \notin N(x)$ 位于 $IG(\pi, \tau)$ 中。在情况（1）中，$(y, x') \in E[\pi]$，那么

y 的颜色会因在π中对 x 或 x' 执行对称反转而改变。在情况（2）中，(y, x') $\notin E[\pi]$，那么对 π中的 x 或 x' 执行对称反转，不会改变 y 的颜色。□

引理 *3.5* 设 x' 为一个黑色顶点，$\omega(x) = \omega(x') = 1$，且 $x' \in N(x)$ 属于 $IG(\pi, \tau)$。在 π 中对 x 执行对称反转后，设 y、z 是连通分量 C_i 中的两个顶点，且 x' 属于连通分量C_j，$i \neq j$。设对 π 中的 x' 执行对称反转后得到染色体 π''。若 $(y, z) \in E[\pi']$，则 $(y, z) \in E[\pi'']$。

证明。 我们通过考察以下三种情况来证明：(1) y和 $z \in N(x)$ 都满足，(2) 只有一个满足，设为 y，属于 $N(x)$，以及 (3) 两者都不属于 $IG(\pi, \tau)$ 中的$N(x)$。图 4 展示了这三种情况。

(1) 两者均为 y，$z \in N(x)$。于是 $(y, z) \notin E[\pi]$，因为 y和 x'位于不同的连通分量中，所以 $(y, x') \in E[\pi]$ 且 $(z, x') \in E[\pi]$，因此 $(y, z) \in E[\pi'']$。

(2) $y \in N(x)$，$z \notin N(x)$。于是 $(y, z) \in E[\pi]$，因为 y和 x'位于不同的连通分量中，所以 $(y, x') \in E[\pi]$ 且 $(z, x') \notin E[\pi]$，因此 $(y, z) \in E[\pi'']$。

(3) $y \notin N(x)$，$z \notin N(x)$。于是 $(y, z) \in E_\pi$，因为 y和 x'位于不同的连通分量中，所以 $(y, x') \notin E[\pi]$ 且 $(z, x') \notin E[\pi]$，因此 $(y, z) \in E[\pi'']$。 □

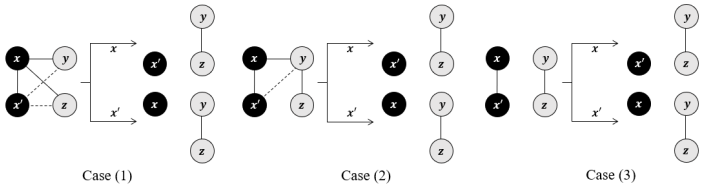

图 4：引理 3.5 证明中的三种情形。

定理 3.2 如果 $IG(\pi, \tau)$ 的某个连通分量至少包含一个黑色顶点，那么就存在一种对称反转，执行该反转后，任何新产生的包含权重为 *1* 的白色顶点的连通分量也包含一个黑色顶点。

证明。 显然，对权重为 2 的黑色顶点 x执行对称反转不会引入任何新的连通分量，而且顶点 x 仍然是黑色的。假设该连通分量中没有权重为 2 的黑色顶点。对于每个黑色顶点 x，令 $\Delta(x)$ 表示执行对称反转 x 后引入的、包含权重为 1 的白色顶点但不包含任何黑色顶点的连通分量中的白色顶点数量。接下来，我们证明必定存在某个顶点 x，使得 $\Delta(x) = 0$。

相反，令 x 为具有最小 $\Delta(x) > 0$ 的黑色顶点。对 x 执行对称反转后，引入的连通分量为 C_1、C_2、.....、C_m。不可能存在由权重为 2 的白色顶点组成的连通分量；否则，根据引理 3.3，该连通分量中与 x 相邻的顶点在执行

对 x 执行对称反转之前是黑色的，这与我们的假设矛盾。不失一般性，假设每个 C_1、C_2、.....、C_i $(1 \leq i \leq m)$ 都包含一个权重为 1 的白色顶点，但不含黑色顶点；而每个 C_{i+1}、C_{i+2}、.....、C_m 至少包含一个黑色顶点。

设 x' 是 C_1 中 x 的邻居。那么，在对 x 执行对称反转之前，x' 是权重为 1 的黑色顶点。只需证明 $\Delta(x') < \Delta(x)$，从而与 $\Delta(x)$ 取最小值相矛盾。$\Delta(x)$ 只统计 C_1、C_2、.....、C_i 中的顶点。根据引理 3.4 和引理 3.5，对 x' 执行对称反转后，C_{i+1}、C_{i+2}、.....、C_m 中所有顶点的颜色以及 C_{i+1}、C_{i+2}、.....、C_m 中所有的边都保持不变。因此，$\Delta(x')$ 也不会统计 C_{i+1}、C_{i+2}、.....、C_m 中的任何顶点，这意味着 $\Delta(x') \leq \Delta(x)$。

根据引理 3.2，x' 必须在 C_1中有一个邻居。设 x'' 是 x'在 C_1中的一个邻居。在 $IG(\pi, \tau)$ 中，若 $(x, x'') \in E_\pi$，则 $(x', x'') \notin E_\pi$，而 x''在对 x'执行对称反转前后都是黑色顶点。若 $(x, x'') \notin E_\pi$，则 $(x', x'') \in E_\pi$，而 x'' 在对 x执行对称反转前后都是白色顶点，但在对 x'执行对称反转后会变成黑色顶点。无论哪种情况，$\Delta(x')$都不会计入 x''，因此，$\Delta(x') <\Delta(x)$。证明见图 5。□

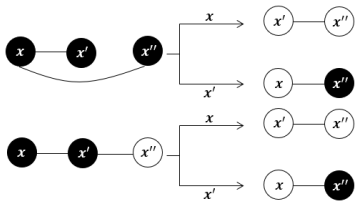

图 5：无论 x, x'' 是否如此，执行 x' 的对称反转后，x'' 总会变成黑色，因此 $\Delta(x') < \Delta(x)$。

本节的主要贡献是以下定理。

定理 3.3 一个染色体 π 能够转变为另一个染色体τ，当且仅当 *(I)* $\mathcal{A}[\pi] = \mathcal{A}[\tau]$, 且 *(II)* 每个权重为 *1* 的白色顶点都属于 $IG(\pi, \tau)$ 中包含一个黑色顶点的连通分量。

证明。$(\Rightarrow)$ 若 $IG(\pi, \tau)$ 存在一个连通分量，例如 C，它由白色顶点构成，其中包含一个权重为 1 的白色顶点 x，且 C 中的所有顶点都无法进行任何对称反转。此外，x 在 π 中为奇数。因此，不可能找到一系列对称反转使 x 变成黑色，进而使 x 变成偶数。根据定理 3.1，π 无法转变为 τ。

$(\Leftarrow)$ 由于 π 中的每个奇重复都对应一个权重为 1 的顶点，而 π 中的每个偶重复都对应一个权重为 2 的顶点。定理 3.2 保证，每个权重为 1 的顶点都会被反转一次，而每个权重为 2 的顶点要么不被反转，要么被反转两次。最后，根据定理 3.1，所有重复序列都会变为偶重复，因此 π 已转变为 τ。□

上述定理意味着，对 $IG[\pi,\tau]$ 进行广度优先搜索，就能判定 π 能否转变为 τ；该搜索需要 $O(n^2)$ 时间，因为$IG(\pi,\ \tau)$至多包含 n 个顶点和 n^2 条边。我们将在第5节详细介绍该算法，因为它也可作为一般情形的判定算法。

在下一节中，我们将展示，当输入基因组满足某些约束时，优化版本也能以令人意外的方式在多项式时间内求解。

4 SMSR 的 2-平衡情形算法

本节考虑 SMSR 的一个特殊情形，称为 *2*-平衡：两个简单相关染色体 π 和τ 的重复数均为 2，并且对于每个重复 $r \in \Sigma_2$，τ 都同时包含 $+r$ 和 $-r$ 。这里给出的算法将求出一个最短的对称反转序列，将 π 变换为 τ。

与两个相关染色体 π和 τ的情形一样，$\mathcal{A}[\pi]$ 和 $A[\tau]$的相同邻接之间存在双射。对于每一对相同邻接 $\langle r(x_i)\ l(x_{i+1})\rangle \in \mathrm{A}[\pi]$ 和 $\langle r(y_j)\ l(y_{j+1})\rangle \in \mathrm{A}[\tau]$，若 $r(x_i) = r(y_j) \neq l(x_{i+1}) = l(y_{j+1})$，则定义$\langle r(x_i)\ l(x_{i+1})\rangle$ 的符号为正；若 $r(x_i) = l(y_{j+1}) \neq l(x_{i+1}) = r(y_j)$，则定义其符号为负。我们总可以假设 $\langle r(x_0)\ l(x_1)\rangle$和 $\langle r(x_n)\ l(x_{n+1})\rangle$ 为正，因为否则我们可以整体反转 π 。注意，如果 $r(x_i) = l(x_{i+1}) = r(y_j) = l(y_{j+1})$，而 $\langle r(x_i)\ l(x_{i+1})\rangle$ 可以为正也可以为负，则称其为纠缠邻接。

I π 中的片段，如果其中的所有邻接关系都具有正方向，则称为正片段；如果都具有负方向，则称为负片段。若片段不是 π 中除 I之外任何其他正片段（或负片段）的子片段，则称其为极大的。极大正片段（或负片段）分别缩写为MPS （或MNS）。（示例见图 6。）

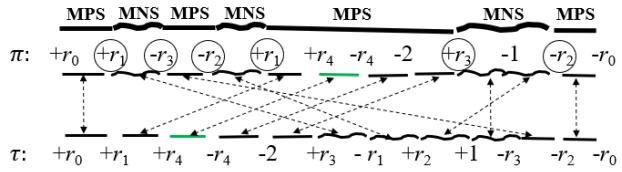

图 6：*MPS* 和*MNS*的示例。正邻接关系和*MPS* 用直线标记，负邻接关系和 *MNS* 用曲线标记，边界用圆圈标出，纠缠邻接 $\langle r_4, -r_4\rangle$以绿色标记。

现在，我们为纠缠邻接赋予适当的方向，使 π 的 *MNS* 数量最小。设 $\langle r(x_i)\ l(x_{i+1})\rangle$ 是一个纠缠邻接，则 $r(x_i) = l(x_{i+1}) = r(y_j) = l(y_{j+1})$，相应地，$l(x_i) = r(x_{i+1}) = l(y_j) = r(y_{j+1})$。由于 π 是简单的，$r(x_{i-1}) \neq l(x_{i+2})$；因此，若 $r(x_{i-1}) = r(y_{j-1})$

且 $l(x_{i+2}) = l(y_{j+2})$，则 $\langle r(x_{i-1})\ l(x_i)\rangle$ 和 $\langle r(x_{i+1})\ l(x_{i+2})\rangle$ 均为正；若 $r(x_{i-1}) = l(y_{j+2})$ 且 $l(x_{i+2}) = r(y_{j-1})$，则 $\langle r(x_{i-1})\ l(x_i)\rangle$ 和 $\langle r(x_{i+1})\ l(x_{i+2})\rangle$ 均为负。因此，我们得到：

命题 4.1 任何纠缠邻接的两个邻居总是具有相同的方向。若每个纠缠邻接的方向都与其邻居相同，则 *MNS* 的数量最少。

任何纠缠邻接的两个邻居总是具有相同的方向。因此，若每个纠缠邻接的方向都与其邻居相同，则 MNS 的数量最少。

固定 π 的所有邻接关系的方向后，正片段和负片段会在 π上交替出现，特别是，最左端和最右端的片段都是正片段。设 π上的 *MPS* 和 *MNS* 的数量分别为 $N_{MPS}[\pi]$ 和 $N_{MNS}[\pi]$，则 $N_{MPS}[\pi]-N_{MNS}[\pi] = 1$。因此，

定理 4.1 $\pi = \tau$ 当且仅当 $N_{\mathrm{MNS}}[\pi] = 0$。

证明。 充分性显然为真。

我们通过归纳法证明效率部分。归纳假设是，对于 $0 \le i \le n+1$，$x_i = y_i$。初始时，我们有 $x_0 = y_0 = +0$。在归纳步骤中，设 y_j 是 $|y_i|$的另一个出现，因为 y_i 和 y_j 在 τ中具有不同符号， $r(y_i) = l(y_j)$，再结合归纳假设 $x_i = y_i$，可得，$r(x_i) = r(y_i) = l(y_j)$。由于邻接 $\langle r(x_i)\ l(x_{i+1})\rangle$ 为正，因此必定存在某个 y_k，属于 τ，使得 $r(y_k) = r(x_i)$ 且 $l(y_{k+1}) = l(x_{i+1})$。由于 y_i 是满足 $r(y_i) = r(x_i)$ 的唯一出现，因此 $k = i$，进而有，$l(y_{i+1}) = l(x_{i+1})$ 且 $x_{i+1} = y_{i+1}$。归纳证明至此完成。□

$x_i\ (1 \le i \le n)$ 的一次出现，如果其左侧邻接 $\langle r(x_{i-1})\ l(x_i)\rangle$ 和右侧邻接 $\langle r(x_i)\ l(x_{i+1})\rangle$ 的方向不同，则称其为 **边界**。因此，任何边界都由一个 *MPS* 和一个 *MNS*共享。（示例见图 6。）设 x_i 和 $x_j\ (i < j)$ 是 π 中 $|x_i|$ 的一对符号不同的出现位置，对 $|x_i|$ 执行对称反转 $\rho(i, j)$ 后得到 π'。我们有，

引理 4.1 $N_{\mathrm{MNS}}[\pi] - N_{\mathrm{MNS}}[\pi'] \le 1.$

证明。 证明通过枚举 x_i 和 x_j 可能成为边界、位于 *MPS* 或 *MNS* 内部的所有情况来进行。收益为 1 时，唯一的情况是 x_i 和 x_j 都是边界，且子串 $[x_i, \ldots, x_j]$ 的开头和结尾均为 *MNS*，或均为 *MPS*。□

引理 4.1 表明，只要每次对称反转都使 *MNS* 的数量减少 1，所得方案就是最优的。幸运的是，我们总能找到这样的对称反转，直到 π 被转换为 τ。

引理*4.2* 设 x_i 和 x_j （$i < j$）是 π中重复项 x 的两个出现位置，则（*I*）它们要么都是边界，要么都不是边界。（*II*）如果 x_i和 x_j 都是边界，则邻接关系 $\langle r\ (x_i)\ \ l\ (x_{i+1})\ \rangle$ 和 $\langle r\ (x_{j-1})\ \ l\ (x_j)\ \rangle$具有相同的方向，并且邻接关系 $\langle r\ (x_{i-1})\ \ l\ (x_i)\ \rangle$ 和 $\langle r\ (x_j)\ \ l\ (x_{j+1})\ \rangle$具有相同的方向。（*III*）此外，如果 $x_i = x_j$，则 $\langle r\ (x_i)\ \ l\ (x_{i+1})\ \rangle$ 以及

$\langle r(x_{j-1})\ \ l(x_j)\rangle$ 与 τ 中的两个连续邻接关系相匹配，而 $\langle r(x_{i-1})\ \ l(x_i)\rangle$ 和 $\langle r(x_j)\ \ l(x_{j+1})\rangle$ 与 τ 中的两个连续邻接关系相匹配。

证明。(I) 不失一般性，假设 x_i 是一个边界。令 y_k 和 y_l 为 $|x_i|$ 在 τ 中具有不同符号的出现位置，因此 $l(y_k) = r(y_l) \neq r(y_k) = l(y_l)$。我们将与 y_k 和 y_l 相关联的邻接关系划分为两组：$\langle r(y_{k-1})\ l(y_k)\rangle$ 和 $\langle r(y_{l-1})\ l(y_l)\rangle$ 构成第(I)组，而 $\langle r(y_k)\ l(y_{k+1})\rangle$ 和 $\langle r(y_l)\ l(y_{l+1})\rangle$ 构成第（II）组。该分组保证 $l(y_k)$ 和 $r(y_l)$ 属于不同的组，且 $r(y_k)$ 和 $l(y_l)$ 也属于不同的组。由于两个邻接关系 $\langle r(x_{i-1})\ l(x_i)\rangle$ 和 $\langle r(x_i)\ l(x_{i+1})\rangle$ 具有不同方向，因此它们要么分别匹配第(I)组，要么分别匹配第（II）组；相应地，两个邻接关系 $\langle r(x_{j-1})\ l(x_j)\rangle$ 和 $\langle r(x_j)\ l(x_{j+1})\rangle$ 必须匹配另一组。无论哪种情况，$\langle r(x_{j-1})\ l(x_j)\rangle$ 和 $\langle r(x_j)\ l(x_{j+1})\rangle$ 都具有不同方向。因此，x_j 是一个边界。

(II) 若 $\langle r(x_{i-1})\ l(x_i)\rangle$ 和 $\langle r(x_i)\ l(x_{i+1})\rangle$ 匹配到第(I)组，则 $\langle r(x_i)\ l(x_{i+1})\rangle$ 为负；$l(x_j)$必须匹配到 $r(y_k)$或 $r(y_l)$，这意味着 $\langle r(x_{j-1})\ l(x_j)\rangle$ 也为负。若 $\langle r(x_{i-1})\ l(x_i)\rangle$ 和 $\langle r(x_i)\ l(x_{i+1})\rangle$匹配到第（II）组，则 $\langle r(x_i)\ l(x_{i+1})\rangle$ 为正；$l(x_j)$必须匹配到 $l(y_k)$或 $l(y_l)$，这意味着 $\langle r(x_{j-1})\ l(x_j)\rangle$ 也为正。由于 $\langle r(x_{i-1})\ l(x_i)\rangle$ 的方向与 $\langle r(x_i)\ l(x_{i+1})\rangle$不同，且 $\langle r(x_j)\ l(x_{j+1})\rangle$ 的方向与 $\langle r(x_{j-1})\ l(x_j)\rangle$不同，因此 $\langle r(x_{i-1})\ l(x_i)\rangle$ 和$\langle r(x_j)\ l(x_{j+1})\rangle$ 的方向相同。

(III) 当 $x_i = x_j$ 时，正邻接关系对必须与符号为“+”的出现所关联的两个邻接关系匹配，而负邻接关系对必须与符号为“-”的出现所关联的两个邻接关系匹配，因此它们在 τ 中相邻。□

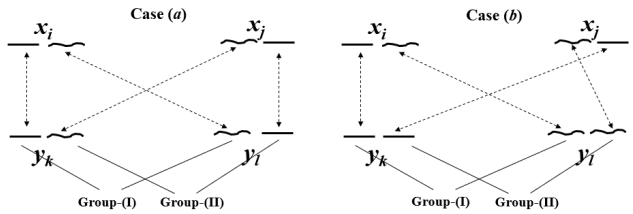

图7：引理4.2的示意图。假设 $x_i = y_k = -y_b\langle r(x_{j-1})\ l(x_j)\rangle$ 和 $\langle r(x_j)\ l(x_{j+1})\rangle$ 与第(I)组匹配。在情形 (a) 中，x_i 和 x_j 的符号不同；在情形 (b) 中，x_i 和 x_j 的符号相同。

引理4.3 π 中的每个 MPS 在 τ 中都有一个相同的片段，而 π 中的每个 MNS 在 τ 中都有一个反转且取反的片段。

证明。设 $I = [x_i, x_{i+1}, \ldots, x_{i+k}]$ 是 π 中的一个 MPS。必定存在一条邻接边$\langle r(y_j)\ l(y_{j+1})\rangle$，使得 $r(x_i) = r(y_j)$ 且 $l(x_{i+1}) = l(y_{j+1})$。我们进行归纳证明。归纳假设为，对所有 $0 \leq t \leq k$，$x_{i+t} = y_{j+t}$ 成立。初始时，我们有 $x_i = y_j$。在归纳步骤中，设 $y_{j'}$ 是另一个出现

算法1用于 SMSR 的 2-平衡情形的算法**输入：**：两个染色体 π和 τ，具有 $\mathcal{A}[\pi] = \mathcal{A}[\tau]$和 $dp[\pi] = dp[\tau] = 2$，且每个重复序列在 τ上的出现位置方向不同。**输出：**：将 π 转换为 τ所需的最少数量的对称反转序列。1: 当 $dp[\pi] = 2$时，从 π 和 τ 中删除冗余重复序列。（引理 3.1）2: 建立 $\mathcal{A}[\pi]$ 和 $\mathcal{A}[\tau]$之间的双射。3: 根据该双射确定 π 中每个邻接（纠缠邻接除外）的方向。4: 将每个纠缠邻接的方向设为与其两个邻居相同。（见命题 4.1，其中两个邻居的方向始终相同。）5: 识别 π中的所有MPS、MNS 和边界。6: **当**存在MNS **时，执行**7: 找出同一重复序列中方向不同的一对边界。8: 对这对反向重复序列执行一次对称反转。9: **结束循环**10: **返回**所有对称反转。

$|y_{j+t}|$ 在 τ 中的出现。由于 $y_{j'}$ 和 y_{j+t} 符号不同，因此 $r(y_{j+t}) = l(y_{j'})$；结合归纳假设，可得 $r(x_{i+t}) = r(y_{j+t}) = l(y_{j'})$，这意味着 $\langle r(x_{i+t})\ l(x_{i+t+1})\rangle$ 必须与 $\langle r(x_{j+t})\ l(y_{i+t+1})\rangle$配对。因此$l(x_{i+t+1}) = l(y_{i+t+1})$ 且 $x_{i+t+1} = y_{i+t+1}$。归纳证明至此完成。

对于 π 中的MNS，证明类似，故略去。 □

定理4.2 如果 π 中存在 *MNS*，那么就存在一对边界，它们是同一重复片段在不同方向上的出现位置。

证明。 反设每一对作为同一重复片段的重复副本的边界都具有相同的符号。由引理4.2和4.3可知，每一对绝对值相等的边界都位于两个不同的MPS上，而它们对应的片段在 τ 中相邻。由于每个MPS 都与两个边界相关联，因此在 π 中，沿着所有MPS和边界形成一条路径；所有MPS所对应的片段构成 τ 的一个子串。注意到$\langle r(x_0)\ l(x_1)\rangle$ 和 $\langle r(x_n)\ l(x_{n+1})\rangle$ 都为正，因此该子串同时包含 y_0 和 y_{n+1}，于是它就成为整个 τ。这便产生了矛盾，因为τ 还包含MNS所对应的片段。□

现在，我们将该算法正式呈现为算法1，用于计算将 π 转换为 τ 所需的最少对称反转次数。

定理4.3 算法*1*给出了将 π 转换为 τ 的最少对称反转序列，其运行时间为 $O(n^2)$ 。

证明。 定理4.2保证，只要存在一个MNS，就存在两个带有 $x_i = -x_j$的边界。根据引理4.2-(II)，执行对称

反转 $|x_i|$ 会使MNS 的数量减少一个。因此，算法$Balanced\text{-}2\ SMSR$ 执行的对称反转次数等于 π中的MNS 数量；根据定理 4.1，这是最优的。

至于时间复杂度，构建 $\mathcal{A}[\pi]$ 与 $\mathcal{A}[\tau]$ 之间的双射需要 $O(n^2)$ 时间。显然，步骤 2–4 需要线性时间。**while** 循环最多运行 $n/2$ 轮，每轮中，寻找一对适当边界需要线性时间。总的时间复杂度为 $O(n^2)$。□

我们指出，本节的结果或许已经是我们所能期望的最佳结果，因为我们将在第 6 节证明：当重复数为 2，但不要求 τ 对每个重复 $|x| \in \Sigma_2$ 都同时包含 $+|x|$ 和 $-|x|$ 时，$SMSR$ 问题将变为 NP 困难。更一般地，当 π 和 τ 的重复数不受限制时，情况截然不同。尽管如此，我们将在下一节着手处理这一一般情形。

5 一种适用于一般情形的 $O(n^2)$ 判定算法

对于一般情形，即两个相关输入基因组的重复数任意时，第 3 节中算法的扩展并非易事，因为不可能将基因组简化。我们的总体思路是固定输入基因组的（相同）邻接关系之间的任意双射 f，并构建相应的交替环图。该交替环图会随着相应的对称反转而变化；我们证明，当图中仅包含 1-环时，就能达到目标 τ。由于交替环图具有变化性，我们构建一个蓝色边相交图来捕捉这些变化。然而，这还不够，因为由交替环图构建的蓝色边相交图可能是不连通的；我们需要通过添加附加顶点使其连通，从而使所得的对称反转序列与原始输入基因组一致，并且能够在新的交图中找到（该交图称为 IG，它基于输入基因组 π 和 τ 以及 f）。下面将介绍具体细节。

假设给定两个相关染色体 $\pi = [x_0, x_1, \ldots, x_{n+1}]$和 $\tau = [y_0, y_1, \ldots, y_{n+1}]$，满足 $x_0 = y_0 = +r_0$和 $x_{n+1} = y_{n+1} = -r_0$。定理 2.1 表明，$\mathcal{A}[\pi] = \mathcal{A}[\tau]$ 是必要条件，因此在 $\mathcal{A}[\pi]$ 和 $\mathcal{A}[\tau]$ 中的相同邻接之间存在一个双射 f，如图 8 所示。基于双射 f，我们按如下方式构造交替环图 $ACG(\pi, \tau, f)$。对于 π 中的每个 x_i，构造一对有序节点，记为 $l(x_i)$ 和 $r(x_i)$，并用一条红边连接。对于 τ 中的每个 y_k，假设在双射 f 中，$\langle r(y_{k-1})\ l(y_k)\rangle$ 与 $\langle r(x_{i-1})\ l(x_i)\rangle$ 匹配，而 $\langle r(y_k)\ l(y_{k+1})\rangle$ 与 $\langle r(x_{j-1})\ l(x_j)\rangle$ 匹配。共有四种情况：

1. $l(y_k) = l(x_i)$ 和 $r(y_k) = r(x_{j-1})$，然后用一条蓝色边连接 $l(x_i)$ 和 $r(x_{j-1})$，

2. $l(y_k) = r(x_{i-1})$ 和 $r(y_k) = r(x_{j-1})$，然后用一条蓝色边连接 $r(x_{i-1})$ 和 $r(x_{j-1})$，3. $l(y_k) = l(x_i)$ 和 $r(y_k) = l(x_j)$，然后用一条蓝色边连接 $l(x_i)$ 和 $l(x_j)$，4. $l(y_k) = r(x_{i-1})$ 和 $r(y_k) = l(x_j)$，然后用一条蓝色边连接 $r(x_{i-1})$ 和 $l(x_j)$。

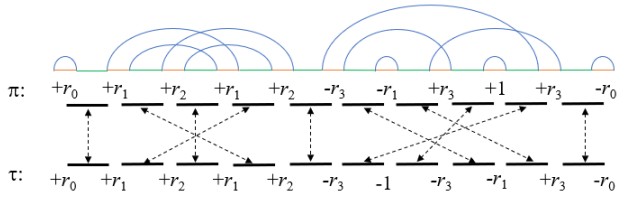

图 8： $\mathcal{A}[\pi]$ 和 $\mathcal{A}[\tau]$中相同邻接关系与相应交替环图之间的双射。

实际上，由红边连接的两个节点意味着它们来自 π中某个重复序列/基因的同一次出现，因此π 中的每次重复序列/基因出现都对应一条红边；类似地，由蓝色边连接的两个节点意味着它们来自 τ中某个重复序列/基因的同一次出现，因此 τ 中的每次重复序列/基因出现都对应一条蓝色边。请注意，每个节点都关联一条红边和一条蓝色边，因此 $ACG(\pi, \tau, f)$由边不相交的环组成，其中红边和蓝色边交替出现。由 c 条蓝色边和 c 条红边组成的环称为 c-圈；当 $c \geq 2$ 时，称为长环。

定理 5.1 给定两条染色体 π^* 和 τ， $\pi^* = \tau$ 当且仅当 $\mathcal{A}[\pi^*] = \mathcal{A}[\tau]$，并且存在一个双射 f，将 $\mathcal{A}[\pi^*]$ 和$\mathcal{A}[\tau]$中的相同邻接关系对应起来，使得所得交替环图 $ACG(\pi^*, \tau, f)$ 中的所有环都是 *1*-圈。

证明。 假设 $\pi^* = [z_0, z_1, \ldots, z_n, z_{n+1}]$ 和 $\tau = [y_0, y_1, \ldots, y_n, y_{n+1}]$，其中 $z_0 = y_0 = +r_0$ 和 $z_{n+1} = y_{n+1} = -r_0$。

充分性部分显然为真，因为我们有 $z_i = z_i$，双射 f 是将 $\langle r(z_{i-1})\ l(z_i)\rangle$ 与 $\langle r(y_{i-1})\ l(y_i)\rangle$匹配起来。

现在我们用归纳法证明必要性部分。我们的归纳假设是，对于 $0 \leq i \leq n+1$，$z_i = y_i$。初始时，我们有 $l(z_0) = l(y_0) = r_0^h$ 以及$r(z_0) = r(y_0) = r_0^t$，根据 $\mathcal{A}[\pi^*] = \mathcal{A}[\tau]$，我们始终假设 $z_1 = y_1$，否则我们可以反转整个染色体 τ。在归纳步骤中，由于$l(z_i)$ 和 $r(z_i)$ 构成一个 1-圈，因此邻接 $\langle r(z_i)\ l(z_{i+1})\rangle$ 与$\langle r(y_i)\ l(y_{i+1})\rangle$匹配；根据归纳假设， $r(z_i) = r(y_i)$，我们有，$l(z_{i+1}) = l(y_{i+1})$，因此，$z_{i+1} = y_{i+1}$。 □

上述定理为我们的算法给出了一个终止条件：设π 和 τ 为输入染色体，我们的算法持续更新交替环图，直到其中所有的圈都变为 1-环。不幸的是，我们在下文中发现，有些圈无法直接通过对称反转操作处理，因此我们将这些彼此相交的圈视为一个连通分量。但这仍然不够，因为还可能存在一些不允许进行任何对称反转的连通分量；我们通过将同一重复的所有圈合并为一个整体连通分量来处理这种情况。

引理 5.1 在交替环图中，每个圈都对应一个唯一的重复，并且圈中的每条边（红色和蓝色）都对应于该唯一重复的一次出现。

证明。不失一般性，假设 $l(x_i)$ 通过一条蓝色边与 $l(x_j)$ 相连。根据交替环图的构造，在τ中必定存在一次出现，比如 y_k，使得 $\{l(x_i)\ l(x_j\} = \{l(y_k)\ r(y_k)\}$，因此，$|x_i| = |x_j| = |y_k|$，而蓝色边（$l(x_i)\ l(x_j)$）对应于重复 $|y_k|$ 的出现 y_k。 □

由于每个基因在 π中只出现一次，引理 5.1 表明，每个基因在 $ACG(\pi,\ \tau,\ f)$ 中都有一个 1-圈；在整个算法过程中，这些 1-圈都不会被触及。

引理 5.2 在交替环图中，如果我们为每对节点 $r(x_i)$ 和 $l(x_{i+1})$ 添加一条绿色边（对所有 $0 \le i \le n$ 均如此），那么所有蓝边和绿色边共同构成一条（蓝绿交替）路径。（见图 6。）

证明。事实上，连接 $r(x_i)$ 和 $l(x_{i+1})$ $(0 \le i \le n)$ 的绿色边是 $\mathcal{A}[\pi]$ 的邻接关系 $\langle r(x_i)\ l(x_{i+1})\rangle$，根据 $\mathcal{A}[\pi]$ 和 $\mathcal{A}[\tau]$ 的相同邻接之间的双射，该邻接关系与 $\mathcal{A}[\tau]$ 的某个邻接关系 $\langle r(y_j)\ l(y_{j+1})\rangle$ 相同。因此，y_j 和 y_{j+1} 在 τ 中相邻出现。根据 $ACG(\pi, \tau, f)$ 的构造以及引理 5.1，它们对应于两条蓝边，其中一条与 $r(x_i)$ 相关联，另一条与 $l(x_{i+1})$ 在 $ACG(\pi, \tau, f)$ 中相关联。因此，这两条蓝边通过绿色边（$r(x_i)\ l(x_{i+1})$）连接起来。上述论证适用于每条绿色边，因此所有蓝边和绿色边构成一条路径。我们在图 8 中展示了一个示例。□

令 $x \in \Sigma$ 为重复。令 x_i 和 x_j 为 π 中 x 的两个出现位置，其中 $i \neq j$。若一条蓝色边连接 $l(x_i)$ 和 $l(x_j)$，或连接 $r(x_i)$ 和 $r(x_j)$，则该蓝色边为反向。若一条蓝色边连接 $l(x_i)$ 和 $r(x_j)$，或连接 $r(x_i)$ 和 $l(x_j)$，则该蓝色边为非反向。

特别地，任何 1-圈（由一条蓝色边和一条红边组成）上的蓝色边都是非反向的。若一个圈至少包含一条反向蓝边，则该圈为反向。

引理 *5.3* 设 x_i 和 x_j 是 π 中重复 x 的两个出现位置。在替代环图 $ACG(\pi, \tau, f)$ 中，如果 $l(x_i)$ 和 $l(x_j)$（或 $r(x_i)$ 和 $r(x_j)$）由一条反向边连接，则 x_i 和 x_j 的方向不同；如果 $l(x_i)$ 和 $r(x_j)$（或 $r(x_i)$ 和 $l(x_j)$）由一条非反向边连接，则 x_i 和 x_j 的方向相同。

证明。我们依据替代环图 $ACG(\pi, \tau, f)$ 的构造进行证明。

如果 $l(x_i)$ 和 $l(x_j)$ 由一条反向边连接，那么 τ 中必然存在 x 的一次出现，记作 y_k，使得 $\{l(x_i)\ l(x_j)\} = \{l(y_k)\ r(y_k)\}$。如果 $l(x_i) = l(y_k)$ 且 $l(x_j) = r(y_k)$，那么 x_i 和 y_k 的方向相同，而 x_j 和 y_k 的方向不同，因此 x_i 和 x_j 的方向不同。如果 $l(x_i) = r(y_k)$ 且 $l(x_j) = l(y_k)$，那么 x_i 和 y_k 的方向不同，而 x_j 和 y_k 的方向相同，因此 x_i 和 x_j 的方向也不同。当 $r(x_i)$ 和 $r(x_j)$ 由一条反向边连接时，情况类似。

如果 $l(x_i)$ 和 $r(x_j)$ 由一条非反向边连接，那么 τ 中必然存在 x 的一次出现，记作 y_k，使得 $\{l(x_i)\ r(x_j\} = \{l(y_k)\ r(y_k)\}$。如果 $l(x_i) = l(y_k)$ 且 $r(x_j) = r(y_k)$，那么 x_i 和 x_j 的方向都与 y_k 相同，因此 x_i 和 x_j 的方向相同。如果 $l(x_i) = r(y_k)$ 且 $r(x_j) = l(y_k)$，那么 x_i 和 x_j 的方向都与 y_k 不同，因此 x_i 和 x_j 的方向相同。当 $r(x_i)$ 和 $l(x_j)$ 由一条反向边连接时，情况类似。□

命题 5.1 给定 x 的一个 k-圈 C，对由反向蓝边连接的 x 的两个出现位置执行对称反转，将把 C 分解为一个 $(k-1)$-圈和一个 *1*-圈。给定 x 的一个 k_1-圈 C_1 和一个 k_2-圈 C_2，对 $x_i \in C_1$ 和 $x_j \in C_2$ 的两个出现位置执行对称反转，将把 C_1 和 C_2 合并为一个 $(k_1 + k_2)$-圈。

现在，我们根据 $ACG(\pi, \tau, f)$ 构造蓝色边相交图 $BG(\pi, \tau, f)$，将每条蓝色边视为其连接的两个节点所构成的区间。对于每个区间，在 BV_π 中构造一个原始顶点，并将其权重设为 1；如果蓝色边是反向蓝边，则将其颜色设为黑色，否则设为白色。当且仅当两个顶点所对应的区间相交但互不重叠时，BE_π 中才连接这两个顶点。蓝色边相交图的一个示例见图 9-（b）。

注意到，对每个连通分量 Pin $BG(\pi, \tau, f)$，$BG(\pi, \tau, f)$ 的每个连通分量都会在 π 上形成一个区间，我们用 P 表示其在 π 上对应的区间。

引理 5.4 设 P 是 $BG(\pi, \tau, f)$ 的某个连通分量，则 P 的最左端点必定是某个 x_i 的左节点，即 $l(x_i)$；而 P 的最右端点必定是某个 x_j 的右节点，即 $r(x_j)$，其中 $i < j$。

证明。由于每条蓝色边都连接区间 P 中的两个节点，P 中节点的数量必定为偶数。因此，一个区间的边界节点不可能同为左节点，也不可能同为右节点。

反设最左端点为 P r（x_i），最右端点为 P l（x_j）。P中总共有 2 个（$j - i$）节点。若用绿色边连接每对节点 r（x_k）和 l（x_{k+1}）（对所有 $i \le k \le j$均如此），那么 P的 2 个（$j - i$）节点之间有 $j - i$ 条绿色边以及 $j - i$ 条蓝边。由于每个节点都关联一条绿色边

和一条蓝色边，这些绿色边和蓝色边构成环，这与引理 5.2 矛盾。□

引理 5.5 在 $ACG(\pi, \tau, f)$ 中，同一长环上与蓝色边对应的 $BG(\pi, \tau, f)$ 中的所有顶点，都属于 $BG(\pi, \tau, f)$的同一个连通分量。

证明。反设存在一些连通分量，每个连通分量都包含某个圈的一部分蓝边。令 P 为这样一种连通分量，使得 P 不与其他此类连通分量的任何区间重叠。令 $e = (x_i^t, x_j^h)$ 和 $e' = (x_{i'}^t, x_{j'}^h)$ 为圈 C上的两条蓝色边，使得它们对应的顶点 $v \in P$，但 $v' \notin P$。根据我们选择 P的方式，在 C 中由 e' 连接的两个节点 $x_{i'}^t$ 和 $x_{j'}^h$ 不可能同时位于 P的区间内。因此，它们必定同时位于 P的区间外。此外，在 C 中由 e 连接的两个节点 x_i^t 和 x_j^h 不可能同时处于 P区间的边界上，否则， e 将不会与对应于 P中某个顶点的任何其他蓝色边相交。不妨设 x_i^t 出现在 P的区间内。由于 e 和 e' 位于同一个圈 C上，除 e外，必然存在一条从 x_i^t 到 $x_{j'}^h$ 、位于 C上的交替路径。根据引理 5.4，不可能沿着一条位于区间范围外的交替路径经过红边，因此，从 x_i^t 到 $x_{j'}^h$ 的交替路径必须包含一条蓝色边，该边将 P 内的节点与 P外的节点连接起来，这与 P 是一个连通分量相矛盾。□ — —

由于非相反 2-循环的两条蓝色边互不相交，因此我们有：

推论 5.1 非相反 *2*-循环不能构成$BG(\pi, \tau, f)$的一个连通分量。

对于每个重复 x，假设它在 $ACG(\pi, \tau, f)$ 中构成 k 个环。设 $x_{i_1}, x_{i_2}, \ldots, x_{i_k}$ 是 x 在 $ACG(\pi, \tau, f)$ 中分别属于不同环的 k 个出现位置，其中 $1 \le i_1 < i_2 < \cdots < i_k \le n$。对于每个这样的顶点，我们构造 $k-1$ 个额外顶点，分别对应区间 $[r(x_{i_j}) - \epsilon, l(x_{i_{j+1}}) + \epsilon]$ 至 $BV_\pi (1 \le j \le k-1)$，将其权重设为 1；若 x_{i_j} 和 $x_{i_{j+1}}$ 的符号不同，则将其颜色设为黑色，否则设为白色。示例见图9-(*c*) 中标记为 10 的顶点。此外， BV_π 中的两个顶点之间有一条边，当且仅当它们对应的区间相交，但彼此均不包含对方。所得图称为 π 的交集图，记为 $IG(\pi, \tau, f) = (V[\pi], E[\pi])$。示例见图9-(*c*)。设 $V_\pi^w \subseteq V[\pi]$ 为 $ACG(\pi, \tau, f)$ 中长环上非相反蓝边所对应顶点的子集。

根据引理 5.4 以及 π 交图的构造，同一重复对应的所有蓝色边所对应的顶点都位于同一个连通分量中。注意，当 $\mathcal{A}[\pi]$ 和 $\mathcal{A}[\tau]$ 的相同邻接之间的双射不同时， π 的交图可能不同。尽管如此，我们有：

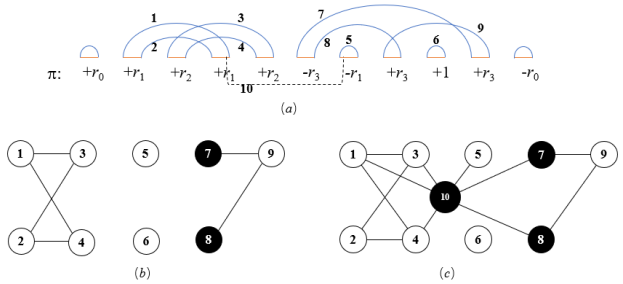

图9：(*a*) 交替环图 $ACG(\pi,\ \tau,\ f)$，其中每条蓝色边都标有一个数字。(*b*) 蓝色边相交图 $BG(\pi,\ \tau,\ f)$。(*c*) 带有附加顶点的交图 $IG(\pi,\ \tau,\ f)$，其中每个数字代表一个区间。

引理 5.6 设 π 和 τ 是两个相关染色体，且满足 $\mathcal{A}[\pi]=\mathcal{A}[\tau]$。设 x_i 和 x_j $(i<j)$ 是 π 中 x 的两个出现位置，$x_{i'}$ 和 $x_{j'}$ $(i'<j')$ 是 π 中 x' 的两个出现位置。如果满足 $i<i'<j<j'$ 或 $i'<i<j'<j$，那么基于 $\mathcal{A}[\pi]=\mathcal{A}[\tau]$ 之间的任意双射 f，在交图 $IG(\pi、\ \tau、\ f)$ 中，x 和 x' 的所有区间所对应的顶点都位于同一个连通分量中。

证明。 假设 $i<i'<j<j'$，因为另一种情况类似。根据引理 5.4 和交图的构造，在 π的任意交图中，与 x_i 和x_j 相关联的区间所对应的顶点位于同一个连通分量 P中；同样，与 $x_{i'}$ 和 $x_{j'}$ 相关联的区间所对应的顶点位于同一个连通分量P'中。我们证明 P与 P'相交，从而得到 $P=P'$。

不可能P和 P' 不相交，因为区间 $[l(x_i)\ldots,r(x_j)]$是 P的一部分，而区间 $[l(x_{i'})\ldots,r(x_{j'})]$ 是 P'的一部分，但 $[l(x_i)\ldots,r(x_j)]$与 $[l(x_{i'})\ldots,r(x_{j'})]$相交。

如果 P与 P'重叠，那么在交图 $IG(\pi,\ \tau,\ f)$ 中，存在一条路径，从某个与 x_j 相关联的区间通向某个与 x_i相关联的区间。在这条路径上对应各顶点的区间中，必有一个与 P'相交。

如果 P'与 P重叠，那么在交图 $IG(\pi,\ \tau,\ f)$ 中，存在一条路径，从某个与 $x_{i'}$ 相关联的区间通向某个与 $x_{j'}$相关联的区间。在这条路径上对应各顶点的区间中，必有一个与P相交。□

实际上，交图的连通分量将 π 上的重复序列划分为若干组。根据引理 5.4 和引理 5.6，重复序列的分组与$\mathcal{A}[\pi]$ 和 $\mathcal{A}[\tau]$的相同邻接关系之间的双射无关。换言之，一旦给定 π 和 τ，分组就固定了。

与重复数为 2 的染色体交图类似，重复数不受限制的染色体交图

也允许在对 π 执行对称反转时使用第 3 节中的规则 I、规则 II 和规则 III。

定理 5.2 如果 $IG(\pi, \tau, f)$ 的某个连通分量包含一个黑色顶点，那么存在一种对称反转；执行该反转后，我们得到 π'：任何新产生的、包含白色顶点的连通分量（该白色顶点对应于 $ACG(\pi', \tau, f)$中某个非对向长圈上的蓝色边）也包含一个黑色顶点。

证明。 $IG(\pi, \tau, f) = (V[\pi], E[\pi])$中的每个顶点都对应一个区间，该区间两侧是同一重复片段的两次出现。如果这两次出现的符号不同，该顶点就是黑色顶点，因此可以进行对称反转。对于每个黑色顶点 x，设在对 x执行对称反转后新产生的连通分量为 $C_1, C_2, \ldots, C_m$；其中， $C_1, C_2, \ldots, C_i$ 各自包含一个白色顶点，该顶点对应于 $ACG(\pi', \tau, f)$ 中某个非对向长圈上的蓝色边，但不包含黑色顶点； $C_{i+1}, C_{i+2}, \ldots, C_j$各自包含附加顶点以及 $ACG(\pi', \tau, f)$ 中 1-圈上的蓝色边所对应的顶点，但不包含黑色顶点； $C_{j+1}, C_{j+2}, \ldots, C_m$ 各自包含一个黑色顶点， $0 \leq i \leq j \leq m$。令 $\Delta(x)$ 表示 $C_1, C_2, \ldots, C_j$中的白色顶点数。

接下来我们证明：如果存在一个连通分量，其中包含一个对应于$ACG(\pi', \tau, f)$中某个非对向长圈上的蓝色边的白色顶点，但不包含黑色顶点，那么就存在一个黑色顶点 x'，使得 $\Delta(x') < \Delta(x)$。

反设 x 是满足 $\Delta(x) > 0$的黑色顶点，且其 $\Delta(x)$取最小值。令 x' 为 C_1中 x 的邻居。那么，在对 x执行对称反转之前， x' 是一个黑色顶点。只需证明 $\Delta(x') < \Delta(x)$，从而与 $\Delta(x)$取最小值矛盾。根据引理 3.4 和引理 3.5，对 x'执行对称反转后， C_{j+1}， C_{j+2}、.....、 C_m 中所有顶点的颜色以及 C_{j+1}， C_{j+2}、.....、 C_m 中的所有边都保持不变。因此， $\Delta(x')$也不会计入C_{j+1}， C_{j+2}、.....、C_m中的任何顶点，因此 $\Delta(x') \leq \Delta(x)$。

由于 C_1 包含一个白色顶点，该顶点对应于一个非对向长环上的蓝色边，根据引理 5.5 和推论 5.1， C_1 至少包含 3 个顶点。设 $N_1(x)$ 为 C_1中 x 的邻集，且 $x' \in N_1(x)$。必定存在 C_1中 x' 的一个邻居，设为 x''。若 $x'' \in N_1(x)$，则 $(x', x'') \notin E[\pi]$，并且在对 x执行对称反转之前，x'' 是黑色的。因此，改为对 x' 执行对称反转而非对 x 执行，将使 x'' 保持黑色，这意味着$\Delta(x') < \Delta(x)$。另一方面，若 $x'' \notin N_1(x)$，则 $(x', x'') \in E[\pi]$，并且在对 x执行对称反转之前，x''是白色的。因此，改为对 x' 执行对称反转而非对 x 执行，将使 x'' 变为黑色，这也意味着$\Delta(x') < \Delta(x)$。两种情况都与 $\Delta(x)$为最小值的假设相矛盾。□

定理 5.3 染色体 π 能通过对称反转变换为另一条染色体τ，当且仅当 ***(I)*** $\mathcal{A}[\pi] = \mathcal{A}[\tau]$，且 ***(II)*** V_π^w中的每个白色顶点都属于 $IG(\pi, \tau, f)$ 中包含黑色顶点的一个连通分量。

证明。 (⇒) 如果 $IG(\pi, \tau, f)$中存在一个连通分量（设为 C），它包含 V_π^w中的一个白色顶点，却不包含黑色顶点，那么 C 中的所有顶点都是白色的，且无法进行任何对称反转。此外，必定存在一个白色顶点（设为 x），它对应于 $ACG(\pi, \tau, f)$中一个非对向长圈的蓝色边。因此，不可能找到一系列对称反转将这个长环变为 1-环。根据定理 5.1， π 无法变换为 τ。

(⇐) 定理 5.2 保证，每个原始顶点要么可以通过对称反转处理，要么其对应的蓝色边已经处于一个 1-环中。最后，所有与原始顶点对应的蓝色边都处于 1-环中；根据定理 5.1， π 已被转换为 τ。□

现在，我们可以正式介绍基于定理 5.3 的判定算法，该算法适用于一般情况以及算法2 中重复数为 2 的情况。我们只需直接检验定理 5.3 中的条件 (I) 和 (II)。注意， $IG(\pi, \tau, f)$ 中的每个连通分量可能包含多个黑色顶点。通过在第 11 行设置 $Q = V_\pi^b$，可以保证在广度优先搜索过程中， $IG(\pi, \tau, f)$ 中的每个连通分量只被探索一次，从而可以控制 $O(n^2)$ 的运行时间。

算法2的运行时间：下面分析算法2的时间复杂度。验证是否满足 $\mathcal{A}[\pi] = \mathcal{A}[\tau]$ 可在 $O(n^2)$时间内完成。构造 $\mathcal{A}[\pi]$ 与 $\mathcal{A}[\tau]$之间的双射，并构造环图 $ACG(\pi, \tau, f)$以及相应的交图 $IG(\pi, \tau, f)$，需要$O(n^2)$时间。接下来分析 $IG(\pi, \tau, f)$的规模。对于每个重复，例如 x， $IG[\pi]$中有 $dp[x, \pi]$ 个原始顶点和 $c[x] - 1$ 个附加顶点，其中$c[x]$是 $ACG(\pi, \tau, f]$中的 x 的环数。注意， $c[x] \leq dp[x, \pi]$且 $\sum_{x \in \sum} dp[x, \pi] = n$。因此， $IG(\pi, \tau, f)$中的顶点总数至多为$\sum_{x \in \sum}($ $dp[x, \pi] + c[x] - 1) \leq 2 \sum_{x \in \sum} dp[x, \pi] - 1 = 2n - 1$，于是 $IG[\pi]$ 中的边数至多为 $4n^2$。整个广度优先搜索过程需要 $O(n^2)$时间，因为 $IG(\pi, \tau, f)$中至多有 $2n - 1$ 个顶点和 $4n^2$ 条边。因此，算法2的运行时间为 $O(n^2)$。

6 SMSR 的困难性结果

本节将证明优化问题 $SMSR$ 是 NP 困难的。最初研究输入基因组的重复数为 2 这一特殊情形下的 SMSR 时，我们发现它在某种程度上与在交图中寻找 Steiner 集有关。交图是一种顶点带有颜色和权重的圆图。下面回顾圆图的定义。

定义 6.1 如果一个图是圆上若干弦的交图，则称其为圆图 。

定义 6.2 如果一个图中的每个顶点都对应实数轴上的一个区间，并且当且仅当两个顶点对应的区间相交但互不包含时，这两个顶点之间有一条边，则该图称为重叠图。

算法2 SSR 的判定算法 输入：两个相关染色体 π 和 τ。输出： $YES\backslash NO$1：如果 $\mathcal{A}[\pi] \neq \mathcal{A}[\tau]$，则 2：返回 NO。3：结束 if 4：当 $dp[\pi] = 2$时，从 π 和 τ 中删除冗余重复序列。（引理 3.1）5：当 $dp[\pi] > 2$ 时，构造 f，使 $\mathcal{A}[\pi]$ 与 $\mathcal{A}[\tau]$ 之间建立双射：将相同的邻接对应起来，并在同一邻接出现多次时任取一个双射。6：当 $dp[\pi] >$时，基于 f构造交替环图 $ACG(\pi, \tau, f)$ 2。7：构造相应的交图 $IG(\pi, \tau, f) = (V_\pi, E_\pi)$（当 $dp[\pi] > 2$时，基于 $ACG(\pi, \tau, f)$ 构造；若 $dp[\pi] = 2$，则直接根据 π和 τ构造）。8：如果 $dp[\pi] = 2$，则 9：将 V_π^w设为权重为 1 的白色顶点集合。10：结束 if 11：如果 $dp[\pi] > 2$，则 12：将 V_π^w设为长环上与反向蓝边对应的白色顶点集合。13：结束 if 14：令 V_π^b为 $IG(\pi, \tau, f)$ 中黑色顶点的集合。15：设置队列 $Q = V_\pi^b$。16：以 Q 为初始值执行广度优先搜索，标记每个 $x \in V_\pi^b$ 中同一连通分量的所有顶点。17：如果存在未标记的 $u \in V_\pi^w$，则返回 NO。18：结束 if 19：返回 YES。

如果将实数轴的两个端点连接成一个圆，重叠图的图类就等价于圆图。圆图可以在多项式时间内识别 [GHS89]。

定义 6.3 最小斯坦纳树问题：

实例： 一个连通图 G，以及一个非空子集 $X \subseteq V(G)$，称为终端集。

任务： 求一个大小最小的子集 $S \subseteq V(G)\backslash X$，使得诱导子图 $G[S \cup X]$ 是连通的。

三十五年多以前，Johnson 曾指出， P [Joh85] 中的最小斯坦纳树问题属于 P 类，并将一份私人通信列为参考文献。然而，迄今为止，尚无已发表的多项式时间算法能够解决该问题。Figueiredo 等人最近重新审视了 Johnson表，尽管他们仍将该问题标记为属于 P，但参考文献

仍标注为“进行中” [FMS22]。在此，我们还顺带澄清，圆图上的最小斯坦纳树问题是 NP 困难的。该归约来自$MAX\text{-}(3,B2)\text{-}SAT$，其中每个子句恰好含有 3 个文字，且每个变量在正文字形式和负文字形式中各出现两次。

定理6.1 MAX-(3,B2)-SAT是 *NP* 困难 [BKS03]。

接下来，我们将$MAX\text{-}(3,B2)\text{-}SAT$归约到圆图上的最小斯坦纳树 问题。

定理6.2 圆图上的最小斯坦纳树问题是 *NP* 困难的。

证明。给定一个$MAX\text{-}(3,B2)\text{-}SAT$实例，其中包含 n 个变量 $\{x_1, x_2, ..., x_n\}$和 m 个子句 $\{c_1, c_2, ..., c_m\}$，我们按如下方式构造一个圆图 ，记作 $OLG(3,\ B2)$。

- 变量区间。对于每个变量 x_i，令 $q_i = 300(i-1)$。构造四组梯形区间，其中 $B[i]_a^b = [q_i + 50(a-1) + 4(b-1) + 1, q_i + 50(a-1) + 4(b-1) + 7]$，其中 $a = 1$ 取 2 、 3 、 4， $b = 1$ 取 2 、 3 、 4 、 5 、 6。令 $b[i] = \{B[i]_a^b | a = 1, 2, 3, 4,$ 和 $b = 2$、5$\}$以及 $B[i] = \{B[i]_a^b | a = 1, 2, 3, 4,$和 $b = 1$、3、4、6$\}$。然后构造四个区间: $P[i]_1 = [q_i+12, q_i+125]$，它与 $B[i]_1^3$和 $B[i]_3^6$均相交； $P[i]_2 = [q_i + 62, q_i + 175]$，它与 $B[i]_2^3$和 $B[i]_4^6$均相交； $N[i]_1 = [q_i+3, q_i+53]$，它与 $B[i]_1^1$和 $B[i]_2^1$均相交；以及 $N[i]_2 = [q_i + 116, q_i + 166]$，它与$B[i]_3^4$和$B[i]_4^4$均相交。令 $P[i] = \{P[i]_1, P[i]_2\}$，并令$N[i] = \{N[i]_1, N[i]_2\}$。

- 子句区间。对于每个子句 c_a $(1 \le a \le m)$，构造一个区间 $c_a = [300n + 3an, 300n + 3an + 2n + 1]$。我们仍用 $\mathbb{C} = \{c_1, c_2, \ldots, c_m\}$表示由这些子句构造的区间集合。

- 正文字区间。如果 x_i 在 c_j 和 $c_k(j < k)$ 中作为正文字出现，则构造两个区间 $f[i]^j = [q_i + 200, q_i + 210]$和 $f[i]^k = [q_i + 220, q_i + 230]$，它们与所有先前的区间都相互独立；并构造 $G[i]^j = [q_i+209, 300n+3jn+i]$，它与 $f[i]^j$和 c_j 相交；以及 $G[i]^k = [q_i + 229, 300n + 3kn + i]$，它与 $f[i]^k$和c_k相交。构造 $D[i]^j = [q_i + 25, q_i + 201]$，它与 $f[i]^j$和$B[i]_1^6$相交；以及 $D[i]^k = [q_i+75, q_i+221]$，它与 $f[i]^k$和 $B[i]_2^6$相交。

- 负文字区间。如果 x_i 在 $c_{j'}$ 和 $c_{k'}(j' < k')$ 中以负文字的形式出现，则构造两个区间 $f[i]^{j'} = [q_i + 240, q_i + 250]$和 $f[i]^{k'} = [q_i + 260, q_i + 270]$，它们与所有先前的区间都互不相交；并构造 $G[i]^{j'} = [q_i + 249, 300n + 3j'n + i]$，它与 $f[i]^{j'}$和 $c_{j'}$相交，以及 $G[i]^{k'} = [q_i + 269, 300n + 3k'n + i]$，它与 $f[i]^{k'}$和 $c_{k'}$相交。构造 $D[i]^{j'} = [q_i+103, q_i+241]$，它与 $f[i]^{j'}$和 $B[i]_1^6$相交，以及 $D[i]^{k'} = [q_i + 153, q_i + 261]$，它与 $f[i]^{k'}$ $B[i]_2^6$ $f[i] = \{f[i]^j, f[i]^k, f[i]^{j'}, f[i]^{k'}\}$ $D[i] =$和 相交。 $\{D[i]^j, D[i]^k, D[i]^{j'}, D[i]^{k'}\}$ 令 ，并令 。

- 子树区间。对于每个变量 x_i，构造四个区间 $t[i] = \{t[i]_1, t[i]_2, t[i]_3, t[i]_4\}$，其中 $t[i]_a = [-(4i + a) + 4,\ q_i + 50a - 1]$ 分别对应 $a = 1, 2, 3, 4$。最后，构造两个区间 $r = \{r_1 = [-4n - 1, 301n], r_2 = [-4n - 2, 0]\}$。

令 $b = \cup_1^n b[i]$, $f = \cup_1^n f[i]$, $t = \cup_1^n t[i]$, $\mathbb{B} = \cup_1^n B[i]$, $\mathbb{P} = \cup_1^n P[i]$, $\mathbb{N} = \cup_1^n N[i]$, $\mathbb{D} = \cup_1^n D[i]$, $\mathbb{G} = \cup_1^n G[i]$, $t \cup_1^n t[i]$。令 $OLG(3, B2)$ 为所有构造出的区间所对应的圆图，将输入的 斯坦纳树问题 终端集定义为与 $b \cup f \cup c \cup t \cup r$ 中的区间对应的顶点，而与 $\mathbb{B} \cup \mathbb{P} \cup \mathbb{N} \cup \mathbb{D} \cup \mathbb{G}$ 中的区间对应的顶点则是候选斯坦纳顶点。

不难发现，与 $t \cup r$ 中的区间对应的顶点已经构成一棵子树，并且所有候选斯坦纳顶点都与这棵子树相连，因此它们是否彼此相连并不重要。与 $b \cup f \cup c$ 中的区间对应的终端顶点彼此独立，因此它们必须通过斯坦纳顶点连接到这棵子树。

我们证明，$MAX\text{-}(3,B2)\text{-}SAT$ 实例可满足，当且仅当 $OLG(3, B2)$ 上的最小斯坦纳树 问题存在一个包含 $14n$ 个顶点的最优解。

对于每个变量 x_i，假设 c_j 和 c_k 是 x_i 作为正文字出现的两个子句，而 $c_{j'}$ 和 $c_{k'}$ 是 x_i 作为负文字出现的两个子句。

($\Rightarrow$) 假设存在一个针对$MAX\text{-}(3,B2)\text{-}SAT$ 实例的真值赋值，我们可以按如下方式为最小斯坦纳树问题构造一个包含 $14n$ 个区间的解。

如果将 x_i 赋值为真，通过选取与区间 $\{P[i]_1, P[i]_2, D[i]_{j'}, D[i]_{k'}, G[i]_j, G[i]_k\}$ 对应的 6 个顶点，以及与梯形区间 $\{B[i]_1^3, B[i]_2^3, B[i]_3^6, B[i]_4^6, B[i]_1^4, B[i]_2^4, B[i]_3^1, B[i]_4^1\}$ 对应的 8 个顶点，那么与 $b \cup f$ 中区间对应的全部 12 个终端顶点，以及与 c_j 和 c_k 对应的终端，都将连接到子树上。

如果将 x_i 赋值为假，通过选择对应于区间 $\{N[i]_1, N[i]_2, D[i]_j, D[i]_k, G[i]_{j'}, G[i]_{k'}\}$的 6 个顶点，以及对应于梯形区间 $\{B[i]_1^1, B[i]_2^1, B[i]_3^4, B[i]_4^4, B[i]_1^6, B[i]_2^6, B[i]_3^3, B[i]_4^3\}$的 8 个顶点， $b \cup f$中对应区间的全部 12 个终端顶点，以及对应于 $c_{j'}$ 和 $c_{k'}$ 的终端，都将连接到该子树。因此，我们得到一个大小为 $14n$的 Steiner 集。

($\Leftarrow$) 假设 OLG上的最小 *Steiner* 树问题（3， $B2$）具有一个包含 $14n$ 个顶点的 Steiner 集，我们将证明存在一个针对$MAX\text{-}(3,B2)\text{-}SAT$ 实例的真值赋值。

首先，对于每个变量 x_i，我们有约束 (I)：为了将 $b[i]$ 中对应区间的 8 个顶点连接到子树，Steiner 集至少包含 $B[i]$ 中对应区间的 8 个顶点；以及约束 (II)：为了将 $f[i]$ 中对应区间的 4 个顶点连接到子树，Steiner 集必须包含 $D[i] \cup G[i]$ 中对应区间的至少 4 个顶点。但任何满足上述约束的 12 个顶点

都不足以完成连接，因为对应于 $\{B[i]_1^2, B[i]_2^2, B[i]_3^5, B[i]_4^5\}$ 中区间的四个终端仍未连接到子树。对于其中的每一个，例如 $B[i]_1^2$，它必须通过 $B[i]_1^3$ 和 $P[i]_1$、$B[i]_1^1$ 和 $N[i]_1$，或 $B[i]_1^3$ 和 $B[i]_1^4$ 连接到子树。如果它选择 $B[i]_1^3$ 和 $B[i]_1^4$，那么对应于 $B[i]_1^5$ 的终端就必须通过 $B[i]_1^6$ 和 $D[i]^j$ 连接到子树，这意味着对应于 $B[i]_1^5$ 的顶点的两个邻居都被选入 Steiner 集。对 $\{B[i]_1^2, B[i]_2^2, B[i]_3^5, B[i]_4^5\}$ 中的每个元素也有类似的论证。因此，要将对应于 $\{B[i]_1^2, B[i]_2^2, B[i]_3^5, B[i]_4^5\}$ 中区间的四个终端连接到子树，除了每个顶点的一个邻居之外，它们还各自需要一个附加顶点，不过其中两个可以共用一个附加顶点。因此，Steiner 集必须至少再包含两个顶点，它们要么是 $P[i]_1$ 和 $P[i]_2$，要么是 $N[i]_1$ 和 $N[i]_2$。因此，由于 Steiner 集的大小为 $14n$，对于每个变量 x_i（$1 \leq i \leq n$），它恰好包含 $B[i] \cup P[i] \cup N[i] \cup D[i] \cup G[i]$ 中对应区间的 14 个顶点；此外，要么选中了对应于 $P[i]$ 区间的顶点，要么选中了对应于 $N[i]$ 区间的顶点。

如果 Steiner 集包含 $P[i]_1$ 和 $P[i]_2$，为了将分别对应于 $B[i]_3^2$ 和 $B[i]_4^2$ 的两个顶点连接到子树，它还包含 $D[i]^{j'}$ 和 $D[i]^{k'}$；那么，为了将分别对应于 $f[i]^j$ 和 $f[i]^k$ 的两个顶点连接起来，Steiner 集可以包含 $D[i]^j$ 和 $G[i]^j$ 中的一个，以及 $D[i]^k$ 和 $G[i]^k$ 中的一个。我们可以调整 Steiner 集，使其包含 $G[i]^j$ 和 $G[i]^k$；随后，分别对应于区间 c_j 和 c_k 的顶点便连接到了子树。在这种情况下，我们赋值 x_i 真，以满足 c_j 和 c_k。

如果 Steiner 集包含 $N[i]_1$ 和 $N[i]_2$，为了将分别对应于 $B[i]_1^5$ 和 $B[i]_2^5$ 的两个顶点连接到子树，它还包含 $D[i]^j$ 和 $D[i]^k$；那么，为了将分别对应于 $f[i]^{j'}$ 和 $f[i]^{k'}$ 的两个顶点连接起来，Steiner 集可以包含 $D[i]^{j'}$ 和 $G[i]^{j'}$ 中的一个，以及 $D[i]^{k'}$ 和 $G[i]^{k'}$ 中的一个。因此，我们可以调整 Steiner 集，使其包含 $G[i]^{j'}$ 和 $G[i]^{k'}$；随后，分别对应于区间 $c_{j'}$ 和 $c_{k'}$ 的顶点便连接到了子树。在这种情况下，我们赋值 x_i 假，以满足 $c_{j'}$ 和 $c_{k'}$。□

我们以*MAX-(3,B2)-SAT* 的一个实例为例，其中 $\{x_1, x_2, x_3\}$ 是变量集合，$\{c_1 = \{x_1, x_2, x_3\}, c_2 = \{x_1, \bar{x}_2, \bar{x}_3\}, c_3 = \{\bar{x}_1, \bar{x}_2, x_3\}, c_4 = \{\bar{x}_1, x_2, \bar{x}_3\}\}$ 是子句集合。构造出的实例及其对应的圆图分别见图 10 和图 11。

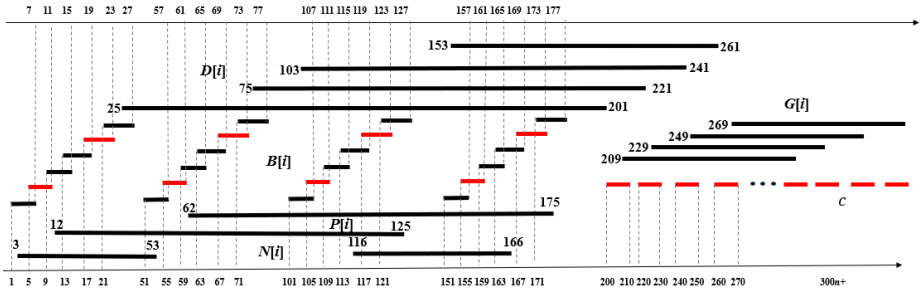

图 10：基于 x_1的区间。与终端顶点对应的区间为红色，与斯坦纳顶点对应的区间为黑色。

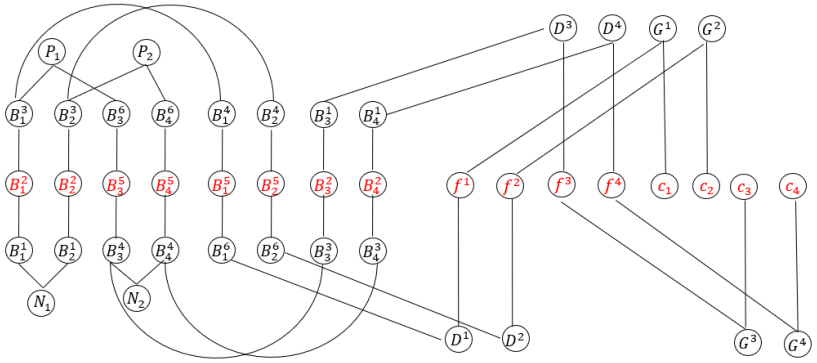

图 11：根据图 10 中的区间得到的对应圆图。

最后，即使转换后染色体的重复数为 2，我们仍将圆图上的最小斯坦纳树问题归约到 SMSR 问题。

引理6.1 给定两个相关的简单染色体 π 和 τ，二者的重复数均为 *2*，令 f为 $\mathcal{A}[\pi]$与 $\mathcal{A}[\tau]$之间唯一的双射。 $ACG(\pi,\tau,f)$由 *1*-环和 *2*-循环组成，每个偶重复的出现对应一个 *1*-圈，而每个奇重复的两次出现对应一个 *2*-循环。

证明。 由于 π 和 τ 的重复数为 2，根据引理 5.1， $ACG(\pi,\tau,f)$的每个圈至多包含 2 条红边。

设 x 是一个偶重复， x_i、 x_j 是它在 π 中的两次出现。由于 x 邻居一致，因此 x 必有两次出现，分别记为 y_k 和 y_l ，且位于 τ 中，使得 $\langle r(x_{i-1})\ l(x_i)\rangle$ 和 $\langle r(x_i)\ l(x_{i+1})\rangle$ 分别与 $\langle r(x_{k-1})\ l(x_k)\rangle$ 和 $\langle r(x_k)\ l(x_{k+1})\rangle$ 匹配，而 $\langle r(x_{j-1})\ l(x_j)\rangle$ 和 $\langle r(x_j)\ l(x_{j+1})\rangle$ 则与

$\langle r(x_{l-1})\ l(x_l)\rangle$ 和 $\langle r(x_l)\ l(x_{l+1})\rangle$ 在双射 f 中匹配。根据 $ACG(\pi,\ \tau,\ f)$ 的构造，$l(x_i)$ 和 $r(x_i)$ 由一条蓝色边相连，$l(x_j)$ 和 $r(x_j)$ 也由一条蓝色边相连，这意味着 x_i 和 x_j 分别对应于 1-环。

令 x 为一个奇重复，x_i、x_j 是它在 π 中的两次出现。由于 x 邻居不一致，因此 τ 中必定有 x 的两个出现位置，分别记为 y_k 和 y_l，使得 $\langle r(x_{i-1})\ l(x_i)\rangle$ 与涉及 y_k 的一条邻接相匹配，而 $\langle r(x_i)\ l(x_{i+1})\rangle$ 与涉及 y_l 的一条邻接相匹配。因此，$l(x_i)$ 和 $r(x_i)$ 之间没有蓝色边相连，这意味着 x_i 和 x_j 位于一个 2- 循环中。 □

定理 6.3 即使输入基因组的重复数为 *2*，*SMSR* 问题仍然是 *NP* 困难的。

证明。给定一个圆图 $G=(V,\ E)$，其中 $X\subseteq V$ 是终端顶点的集合。对于每个顶点 $v\in V$，令 $[l(v)\ r(v)]$ 为其在实数轴上对应的区间。假设 $l(v_{min})$ 是最小值，$r(v_{max})$ 是最大值。

我们首先构造两个 1-环：$(l(v_{min})-1-\delta,\ l(v_{min})-1+\delta)$，它对应于“+0”；以及 $(r(v_{max})+1-\delta,\ r(v_{max})+1+\delta)$，它对应于“-0”。

对于每个终端顶点 $t\in X$，我们构造两个相交的非相反 2-循环：一个是 $(l(t)-\delta, l(t)+\delta, r(t)-\delta, r(t)+\delta, l(t)-\delta)$，其中 $(l(t)-\delta, l(t)+\delta)$ 和 $(r(t)-\delta, r(t)+\delta)$ 是红边，而 $(l(t)-\delta, r(t)+\delta)$ 和 $(l(t)+\delta, r(t)-\delta)$ 是蓝边。令 t_1 为与此圈对应的重复项，这两个出现位置在 π 中都带有“+”符号；另一个是 $(r(t)-3\delta, r(t)-2\delta, r(t)+2\delta, r(t)+3\delta, r(t)-3\delta)$，其中 $(r(t)-3\delta, r(t)-2\delta)$ 和 $(r(t)+2\delta, r(t)+3\delta)$ 是红边，而 $(r(t)-3\delta, r(t)+3\delta)$ 和 $(r(t)-2\delta, r(t)+2\delta)$ 是蓝边。令t_2 为与此圈对应的重复项，这两个出现位置在 π 中都带有 “+” 符号。接着，我们将 $l(t)-\delta$ 和 $r(t)-\delta$ 记作 t_1^h, $l(t)+\delta$ 和 $r(t)+\delta$ 记作 t_1^t, $r(t)-3\delta$ 和 $r(t)+2\delta$ 记作 t_2^h, $r(t)-2\delta$ 和 $r(t)+3\delta$ 记作 t_2^t。

任取一个终端顶点 $x\in X$，我们构造一个相反的 2-圈：$(r(x)-5\delta, r(x)-4\delta, r(x)+5\delta, r(x)+4\delta, r(x)-5\delta)$，其中 $(r(x)-5\delta, r(x)-4\delta)$ 和 $(r(x)+4\delta, r(x)+5\delta)$ 是红边，而 $(r(x)-5\delta, r(x)+4\delta)$ 和 $(r(x)-4\delta, r(x)+5\delta)$ 是蓝边。设 x_3 为与此圈对应的重复，则与红边 $(r(x)-5\delta, r(x)-4\delta)$ 对应的出现带有“+”符号，而与红边 $(r(x)+4\delta, r(x)+5\delta)$ 对应的出现则在 π 中带有“-”符号。于是，我们用 x_3^h 表示 $r(x)-5\delta$ 和 $r(x)+\delta$，用 x_3^t 表示 $r(x)-4\delta$ 和 $r(x)+4\delta$ 。

对于每个候选斯坦纳顶点 s，我们构造两个 1-圈（$l(s)-\delta$，$l(s)+\delta$）和（$r(s)-\delta$，$r(s)+\delta$）。设 s_1 为与此圈对应的重复，则这两个出现位置在 π 中都带有“+”符号。然后，我们将 $l(s)-\delta$ 和 $r(s)-\delta$ 记为 s_1^h，将 $l(s)+\delta$ 和 $r(s)+\delta$ 记为 s_1^t。

将上述构造的图记为 $ACG(G,\ X)$。接下来，我们证明 $ACG(G,\ X)$ 是一个定义良好的交替环图，即存在两个相关的简单染色体 π 和 τ，满足 $\mathcal{A}[\pi]=\mathcal{A}[\tau]$（因此存在一个双射$f$），使得 $ACG(\pi,\ \tau,\ f)=ACG(G,\ X)$。

在 $ACG(G,\ X)$ 中，每条红边都对应于一个重复的出现位置，因此显然，染色体 π 是这些出现位置组成的序列。接下来我们可以将

红边之间的空白视为邻接关系。根据上述构造，每条蓝色边在对应于 v_i 的环中连接节点 v_i^h 和节点 v_i^t，因此，一条蓝色边对应于 τ 中 v_i 的一个出现位置。因此，根据引理 5.2，只需证明所有空白和蓝色边构成一条路径。

将 V 的顶点按 $l(v)$ 的递增顺序排列，再按 $ACG(G, X)$ 排列，可通过迭代添加一个 1-圈、两个相交的非相反 2-循环以及一个相反 2-环得到。我们归纳证明空白和蓝色边构成一条路径。起初，圈 $(0^h, 0^t)$ 和 $(0^t, 0^h)$ 的两条蓝色边必定构成一条路径。假设我们有一条由空白和蓝色边构成的路径，$\mathbb{P}$ $(0^h, n_1, \ldots, n_k, 0^h)$。

(1) 若在 n_i 和 n_j 的空白处添加一个 1-圈 (s_1^h, s_1^t)，如果 $j = i+1$，那么 $\mathbb{P}' = (0^h, \ldots, n_i, s_1^h, s_1^t, n_j, \ldots, n_k)$ 是一条路径；如果 $j = i-1$，那么 $\mathbb{P}' = (n_0, \ldots, n_j, s_1^t, s_1^h, n_i, \ldots, n_k, 0^h)$ 也是一条路径。

(2) 如果将两个相交的非相反 2-循环 $(t_1^h, t_1^t, t_1^h, t_1^t, t_1^h)$ 和 $(t_2^h, t_2^t, t_2^h, t_2^t, t_2^h)$ 插入 n_i, n_j 和 $n_{i'}, n_{j'}$ 之间的两个空隙中。我们只展示 $i+1 = j < i' = j' - 1$ 的情况，因为其他情况都类似。$\mathbb{P}' = (0^h, \ldots, n_i, t_1^h, t_1^t, t_2^h, t_2^t, t_1^h, t_1^t, n_j, \ldots, n_{i'}, t_2^h, t_2^t, n_{j'}, \ldots, n_k, 0^h)$ 也是一条路径。特别地，如果将两个相交的非相反 2-循环插入一个空隙中，则可以通过从 $\mathbb{P}'$ 中删除片段 $(n_j, \ldots, n_{i'})$ 得到新路径。

(3) 如果在 n_i, n_j 和 $n_{i'}, n_{j'}$ 这两处空白之间加入一个相反 2-环（$x_3^h, x_3^t, x_3^h, x_3^t, x_3^h$），我们只展示 $i+1 = j < i' = j' - 1$ 的情况，因为其他情况都类似。$\mathbb{P}' = (0^h, \ldots, n_i, x_3^h, x_3^t, n_{i'}, \ldots, n_j, x_3^t, x_3^h, n_{j'}, \ldots, n_k, 0^h)$ 也是一条路径。

因此，在 $ACG(G, X)$ 中，所有空白和蓝色边构成一条路径。我们可以沿着这条路径得到 τ：如果路径经过一条从 v_a^h 到 v_a^t 的蓝色边，那么它在 τ 中带有“+”符号；如果路径经过一条从 v_a^t 到 v_a^h 的蓝色边，那么它在 τ 中带有“-”符号。由于这些空白代表 π 和 τ 中的邻接关系，因此 $\mathcal{A}[\pi] = \mathcal{A}[\tau]$。为了使 π 和 τ，我们在每个邻接关系之间插入不同的基因。

最后，我们通过证明 G 存在大小为 k 的 Steiner 集，当且仅当 π 可以通过 $2|X|+2k+1$ 次对称反转变换为 τ，从而完成证明。

($\Rightarrow$) 假设 G 存在大小为 k 的 Steiner 集 S，这意味着 $G(X \cup S)$ 构成一棵树。因此，在交图 $IG(\pi, \tau)$ 中，对应于 $X \cup S$ 的重复序列位于同一个连通分量中，该连通分量包含一个黑色顶点 x_3。根据我们的构造，X 的每个顶点还对应两个奇顶点，而 S 的每个顶点对应一个偶顶点，因此 π 可以通过 $2|X|+2k+1$ 次对称反转变换为 τ。

($\Leftarrow$) 假设 π 可以通过 $2|X|+2k+1$ 次对称反转变换为 τ。由于交图 $IG(\pi, \tau)$ 中有 $2|X|+1$ 个奇数顶点，但只有一个黑色顶点，因此，解必须在这些奇数顶点上执行至少 $2|X|+1$ 次对称反转，并且在 k 个偶数顶点上至多执行 $2k$ 次对称反转。对一个顶点执行对称反转，当且仅当该顶点位于包含黑色顶点的连通分量中时才可行。因此，$2|X|+1$ 个奇数顶点和 k 个偶数顶点位于同一个连通分量中，这

意味着 G 有一个大小为 k 的 Steiner 集。 □

我们在图 12 中给出了上述归约的一个例子。

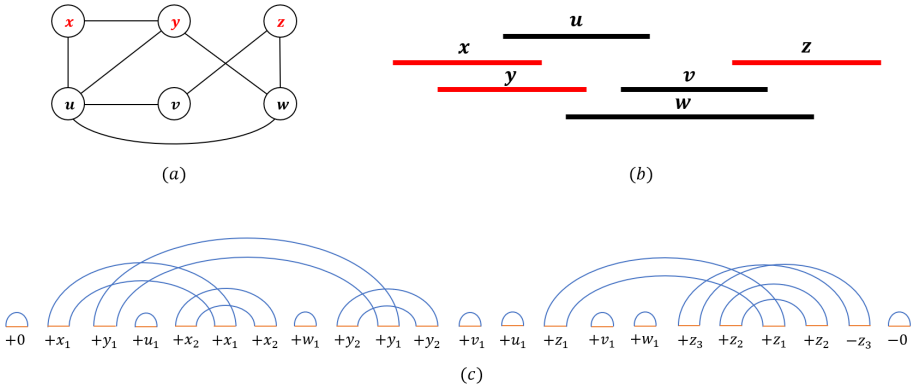

图 12：(*a*)。一个圆图 G，其中 $X = \{x, y, z\}$ 是终端集合。(*b*)。 G对应的区间。(*c*)。构造出的替代环图 $ACG(G, X)$，以及 $\pi = [+0, +x_1, +y_1, +u_1, +x_2, +x_1, +x_2, +w_1, +y_2, +y_1, +y_2, +v_1, +u_1, +z_1, +v_1, +w_1, +z_3, +z_2, +z_1, +z_2, -z_3, -0]$, $\tau = [+0, +x_1, +x_2, +x_1, +y_1, +y_2, +y_1, +u_1, +x_2, +w_1, +y_2, +v_1, +u_1, +z_1, +z_2, +z_1, +v_1, +w_1, +z_3, -z_2, -z_3, -0]$。

7 结论

本文研究了一种名为对称倒位排序的新型基因组重排模型。该模型基于基因组比较领域近期的新发现。判定一个染色体能否通过对称反转变换为另一个染色体的判定问题，可在多项式时间内求解。但在两个染色体之间的变换过程中，求出所需对称反转最少次数的优化问题是 NP 困难的。为这一优化问题设计一些近似算法颇有意义。此外，针对更符合现实的特殊情形，设计可在多项式时间内求解该优化问题的算法或许也很有意义。

致谢

本研究得到国家自然科学基金委员会项目（编号 61872427 和 61732009）的资助。

参考文献

[APC03] L.Armengol, M.A. Pujana, J. Cheung, S.W. Scherer, and X. Estivill. 人类与小鼠基因组之间共线性断裂区域中节段性重复的富集，表明它们参与了进化重排。

化重排。《人类分子遗传学》，12(17):2201-2208，2003年。[BBK04] J.A. Bailey, R. Baertsch, W.J. Kent, D. Haussler, and E.E. Eichler。哺乳动物染色体进化的热点。《基因组生物学》，5(4):R23，2004年。[BHK02] P. Berman, S. Hannenhalli, and M. Karpinski。用于反转排序的1.375近似算法。第*10*届欧洲算法研讨会（*ESA'02*）论文集，第200–210页，2002年。[BK99] P. Berman and M. Karpinski。关于一些更紧的不可近似性结果（扩展摘要）。第*26*届自动机、语言与编程国际研讨会（*ICALP'99*）论文集，第200–209页，1999年。[BKS03] P. Berman, M. Karpinski, and A.D. Scott。最大3可满足性短对称实例的近似困难性。计算复杂性电子研讨会，报告第49号，2003年。[BMD05]J.L. Bennetzen, J. Ma, and K.M. Devos。开花植物近期基因组大小变异的机制。《植物学年鉴》，95:127–32，2005年。[Cap99] A. Caprara。通过反转对排列排序与欧拉回路分解。《*SIAM*离散数学杂志》，12(1):91–110，1999年。[Chr98] D.A. Christie。用于反转排序的3/2近似算法。第*9*届*ACM-SIAM*离散算法年度研讨会（*SODA'98*）论文集，第244–252页，1998年。[FLR09] G. Fertin, A. Labarre, I. Rusu, S. Vialette, and E. Tannier。基因组重排组合学。MIT出版社，2009年。[FMS22] C.M.H. de Figueiredo, A.A. de Melo, D. Sasaki, and A. Silva。面向21世纪重新审视Johnson表。《离散应用数学》，323:284-300，2022年。[GHS89] C.P. Gabor, W.L. Hsu, and K.J. Supowit。在多项式时间内识别圆图。《*ACM*杂志》，36:435–474，1989年。[HP99] S. Hannenhalli and P. Pevzner。将卷心菜变成芜菁：通过反转对带符号排列排序的多项式时间算法。《*ACM*杂志》，46(1):1–27，1999年。[HP94] S.B. Hoot and J.D. Palmer。结构重排，包括银莲花属及相关属叶绿体基因组中的平行倒位。《分子进化杂志》，38: 274–281，1994年。[Joh85]D.S. Johnson。NP完全性专栏：持续更新的指南。《算法杂志》，6:434–451，1985年。

[KST00] H. Kaplan、R. Shamir 和 R.E. Tarjan。用于通过反转对带符号排列排序的更快、更简洁的算法。《*SIAM*计算杂志》，29(3):880–892，2000。[KS95] J. Kececioglu 和 D. Sankoff。反转排序的精确算法和近似算法及其在基因组重排中的应用。*Algorith-mica*，13(1):180–210，1995。[LCG09] M.S. Longo、D.M. Carone、E.D. Green、M.L. O'Neill 和 R.J. O'Neill。不同的反转座子类别界定了标示进化新颖性位点的进化断点。*BMC* 基因组学，10(1):334，2009。[OCM92] P. C. Orban、D. Chui、J.D. Marth。转基因小鼠中的组织和位点特异性重组。*Proc. Nat. Acad. Sci. USA*，89(15): 6861–6865，1992。[PH86] J.D. Palmer 和 L.A. Herbon。芸薹属和萝卜属的三环线粒体基因组：通过倒位改变重复序列构型。《核酸研究》，14: 9755–9764，1986。[PT03] P. Pevzner 和 Glenn Tesler。人类和小鼠基因组序列揭示了哺乳动物进化过程中广泛的断点复用。*Proc. Nat. Acad.Sci. USA*，100(13): 7672-7677，2003。[San09] D. Sankoff。进化断点何在，为何存在？《*J. Biology*》，8:66，2009。[SLA92] D. Sankoff、G. Leduc、N. Antoine、B. Paquin、B. F. Lang 和 R. Cedergran。用于系统发育推断的基因顺序比较：线粒体基因组的进化。*Proc. Nat. Acad. Sci. USA*，89: 6575–6579，1992。[Sau87] B. Sauer。Cre-Lox位点特异性重组系统在酿酒酵母中的功能性表达。*Mol Cell Biol.*，7(6): 2087–2096，1987。[SH88] B. Sauer 和 N. Henderson。P1噬菌体Cre重组酶在哺乳动物细胞中介导的位点特异性DNA重组。*Proc. Nat. Acad.Sci. USA*，85(14): 5166–5170，1988。[SIW97] K. Small、J. Iber 和 S.T. Warren。Emerin缺失揭示了一种由反向重复序列介导的常见X染色体倒位。《自然·遗传学》，16:96–99，1997。[TBS07] E. Tannier、A. Bergeron 和 M-F. Sagot。反转排序的进展。《离散应用数学》，155(6-7):881–888，2007。[TVO11] A. Thomas、J-S. Varr 和 A. Ouangraoua。通过双切断与连接和反转实现基因组倍性还原。*BMC* 生物信息学，12(9):S20，2011。

[WW18] D. Wang 和 L. Wang。GRSR：一种从多个单染色体基因组序列推导基因组重排情景的工具。*BMC* 生物信息学，19(9):11–19，2018。[WLG17] D. Wang、S. Li、F. Guo 和 L. Wang。核心基因组支架比较揭示了倒位事件常与成对反向重复序列相关。*BMC* 基因组学，18:268，2017。[WEH82] G.A. Watterson、W.J. Ewens、T.E. Hall 和 A. Morgan。染色体倒位问题。《理论生物学杂志》，99(1):1–7，1982。[WPR19]A.M. Wenger、P. Peluso、W.J. Rowell、P.C. Chang 和 M.W. Hunkapiller。准确的环状一致性长读长测序可改进人类基因组的变异检测和组装。《自然·生物技术》，37(11):1155-1162，2019。