arXiv:2302.07600v1 [cs.DC] 15 Feb 2023

有限可见范围下圆周上有限记忆机器人的异步聚集*

Satakshi Ghosh[0000−0003−1747−4037]、Avisek Sharma[0000−0001−8940−392X]、
Pritam Goswami[0000−0002−0546−3894]和 Buddhadeb Sau[0000−0001−7008−6135]

贾达普尔大学，印度加尔各答 700032，Raja S.C. Mallick 路 188 号{
satakshighosh.math.rs,aviseks.math.rs,
pritamgoswami.math.rs,buddhadeb.sau}@jadavpuruniversity.in

摘要。考虑一组n 个移动实体，称为机器人，它们位于连续圆上并在其上运行，即所有机器人最初都处于圆上的不同位置。聚集问题旨在设计一种分布式算法，使机器人能够在圆上的某一点集合。机器人是匿名、同构且同质的。机器人在圆形路径内按照确定性的“观察—计算—移动”周期运行。机器人对顺时针方向达成一致。机器人的移动是刚性的，并且具有有限可见性π，即每个机器人只能看到圆上与自身角距离严格小于π 的点。

Di Luna等人. [DISC’2020] 为圆上的无记忆静默机器人提出了一种确定性聚集算法，采用半同步（SSync）调度器。Buchin等人. [IPDPS(W)’2021] 证明，在完全可见的条件下，采用 SSync 调度器的$\mathcal{OBLOT}$ 机器人模型与采用异步（ASync）调度器的$\mathcal{FSTA}$ 机器人模型（机器人为静默机器人，但具有有限持久记忆）不可比较。在有限可见性条件下，这种比较仍未得到解答。因此，本文将 Di Luna等人在 [DISC’2020] 中的工作扩展至 ASync 调度器下的$\mathcal{FSTA}$ 机器人模型。

关键词：聚集、异步、圆、有限可见性、机器人、有限内存。

1 引言

在群体机器人学中，研究重点是让能力有限的机器人完成某些任务。在过去二十年里，研究机器人协同问题的热度大幅上升。在实际应用中，使用能力强大的机器人并不总是容易，因为制造这类机器人并不划算。如果一群能力有限的机器人能够完成同样的任务，那么使用群体机器人就比使用功能繁多的机器人更有效，因为制造这群机器人所需的成本远低于制造功能繁多的机器人。此外，由于群体中的机器人能力有限，因此设计这类机器人也很容易。

* 前三位作者是贾达普尔大学的全职研究学者。

聚集问题一直受到研究人员的广泛关注，是分布式系统中自主机器人的基本任务之一。聚集是指，若有n 个机器人初始时任意分布，那么在有限时间内，所有机器人都会在一个初始位置未固定的点会合。当只有两个机器人时，这项任务称为会合。在分布式系统中，能力非常有限的机器人并不总是能轻易会合于一点。因此，设计一种分布式算法，让一些成本极低且能力有限的机器人实现聚集，是一项颇具挑战性的工作。

在理论研究中，机器人被假定为自主、同构且同质的。它们没有全局坐标系。根据能力的不同，通常有四种机器人模型，分别是 $\mathcal{OBLOT}$、$\mathcal{FSTA}$、$\mathcal{FCOM}$ 和 $\mathcal{LUMI}$。在这些模型中，机器人都被假定为自主的（即机器人不受集中控制）、同构的（即机器人在物理上无法区分）且同质的（即每个机器人运行相同的算法）。此外，在 $\mathcal{OBLOT}$ 模型中，机器人是静默且无记忆的（即机器人之间无法通信，且没有持久内存来记住先前的状态）；在 $\mathcal{FSTA}$ 模型中，机器人静默但并非无记忆；在 $\mathcal{FCOM}$ 模型中，机器人无记忆但并非静默；而在 $\mathcal{LUMI}$模型中，机器人既不静默也非无记忆。机器人被激活后，会执行观察-计算-移动（LCM）周期。在观察阶段，机器人从周围环境获取输入，然后在计算阶段根据该输入运行算法，并输出一个目标点。随后，机器人在移动阶段前往该目标点。调度器控制机器人的激活。文献中主要考虑三种调度器。在同步调度器中，时间被划分为全局轮次。在（FSync）调度器中，每个机器人在每一轮都会被激活，并同时执行 LCM 周期。在半同步调度器（SSync）中，并非每一轮都会激活所有机器人，但同一轮中被激活的机器人会同时执行 LCM 周期。最后，在异步调度器（ASync）中，没有共同的时间概念，机器人可以在任意时刻被激活，也不存在轮次的概念。因此，不对同步性作任何假设。

本文研究机器人在圆上的聚集。机器人位于圆的圆周上，只能沿着圆周移动。这里的机器人具有有限可视范围。如果机器人具有完全可见性，并且能够选举出唯一领导者，那么即使在 ASync 调度器下，机器人的聚集也不难实现。但是，在有限可见性条件下，由于唯一领导者选举并不容易，聚集也并非易事。本文研究在异步调度器下，$\mathcal{FSTA}$ 个具有有限可视范围的机器人在圆上的聚集问题。

2 相关工作

在分布式计算领域，在各种计算能力和通信方式下，移动机器人的聚集一直是深入研究的重点。聚集问题已在分布式计算领域得到广泛研究。这一问题在连续域和离散域中都得到了深入探讨。关于这一问题的综述不多 [4,17,8]。在连续情形下，自主移动机器人群体的聚集问题和会合问题都已得到研究，这些机器人在一维和二维空间中运行，需要会合于同一点或汇聚到同一点。

在一维和二维空间中运行，需要它们会合于同一点或汇聚到同一点。本文主要关注机器人可视范围有限时的机器人聚集问题。Flochhini [18] 在其论文中指出，在假设所有机器人对两个坐标轴的方向达成一致的情况下，机器人能够在有限可见性下通过 ASync 调度器在平面上实现聚集。在 [25] 中，作者提出了一种在有限可见性下使机器人在平面上聚集的最优算法。在另一项研究 [26] 中，作者解决了在 SSync 调度器下、仅对一个坐标轴达成一致时的聚集问题。在 [2] 中，作者证明，在没有手性且对局部坐标系没有任何一致约定的情况下，机器人能够在 FSync（SSync）调度器下实现聚集（类似地实现汇聚）。之后，他们在论文 [13] 中指出，时间复杂度取决于群体的数量。

在离散情形中，机器人分散在一个以图建模的网络中，需要聚集到同一个节点并终止。解决这一问题的主要难点在于对称性。然而，当网络节点是匿名的、网络具有对称性、移动智能体彼此同构且没有通信手段时，便无法通过确定性方法解决该问题。因此，研究重点是通过尽可能少的假设，使问题能够确定性地求解。在论文 [20],中，作者展示了如何在二分图中聚集异步无记忆机器人，并采用局部视野，这意味着机器人只能看到自己的直接邻居。此外，在 [9] 中，他们为任意图中的会合 问题提供了一种算法。研究分布式系统的另一种有趣方式，是考察存在故障机器人的分布式问题。在 [5]中，作者研究了存在拜占庭机器人的网络中的机器人聚集问题。此外，在 [1,12] 中，作者提出了一种能够应对崩溃故障和拜占庭故障的聚集算法。近年来，研究人员开始研究动态图 [7],，即拓扑变化不局限于某些区域，而是持续发生在不可预测的位置；这些变化是系统本质的一部分。在论文 [22]中，他们开始研究动态环中的聚集问题。在 [23,10]中，他们研究了多边形地形中的机器人聚集问题。在 [19]中，他们提出了三角网格上的机器人最优聚集算法。关于网格网络中的机器人聚集，已有大量研究成果 [11,3]。

在我们的论文中，机器人位于一个连续圆上，我们将解决所有机器人聚集到圆上某一点的问题。近期有一些关于这一模型的研究。在 [21,15] 中，作者展示了在一个环上速度不同的移动智能体的会合 。由于领导者选举（[14]）是解决分布式计算中许多问题的一项重要任务，因此在 [16] 中，作者开始研究由部署在连续环 C上的机器人集合R 来解决聚集和选举问题，他们主要关注选举。不过，在他们的模型中，机器人没有可视范围，也就是说，它们无法看到任何距离之外的东西。此后，Di Luna等人[24], 证明，在有限可见性 π 下，匿名无记忆机器人的聚集问题可以在 SSync 调度器中解决；但当连续圆 C上的可视范围为 $\frac{\pi}{2}$ 时，所有机器人不可能聚集。本文提出了一种算法，使n 个机器人能够在 ASync 调度器下有限时间内聚集；在 $\mathcal{FSTA}$ 机器人模型中，圆上的可视范围为 π 。

3 我们的贡献

在完全可见的情况下，如果机器人对全局手性方向达成一致，那么在异步调度器下让机器人在圆上聚集并不困难。但若从可视范围中排除一个点，聚集问题就会变得有些困难。Di Luna等人（[24]）展示了在 SSync 调度器下，可视范围为 π 时无记忆机器人在圆上的聚集问题。在本文中，我们将这项工作扩展到异步调度器。这里的机器人是匿名、同构且静默的。不过，为了获得异步调度器带来的灵活性，我们只放宽了机器人无记忆这一条件。我们假设每个机器人都有有限持久记忆。

在 [6]，中，作者证明，在完全可见的条件下，由无记忆、静默机器人组成的半同步调度器（简称 $\mathcal{OBLOT}^{S}$）与由静默、有限内存机器人组成的异步调度器（简称 $\mathcal{FSTA}^{A}$）无法相互比较。对于有限可视范围模型，目前尚未进行这种比较。该结果表明，引入有限内存并不会使该模型变得与 $\mathcal{OBLOT}^{S}$ 一样强大。与维持同步调度器相比，拥有有限内存并不是什么大事。

此外，我们并未考虑一般的异步调度器。具体而言，我们假设，在机器人的 LCM 环中，完成其观察与计算阶段需要非零时间（并非瞬时完成）。这一假设在实际场景中相当合理。

4 模型与定义

4.1 机器人模型

在这个问题中，我们考虑 $\mathcal{FSTA}$ 机器人模型。机器人是匿名且同构的，但并非无记忆。机器人具有有限持久记忆。机器人之间无法通信。机器人具有弱重数检测能力，即机器人可以检测到重数点，但无法确定重数点上有多少个机器人。所有机器人都位于一个圆上。机器人对手性有共同的约定。机器人按照观察-计算-移动周期运行。在每个周期中，机器人根据自身的局部坐标系对其他机器人的位置进行快照（观察）；根据该快照，执行确定性算法，以决定停留原地还是移动到圆上的另一个点（计算）；然后根据算法，机器人要么保持静止，要么移动到相邻点（移动）。我们假设机器人由完全异步对抗调度器（ASync）控制。机器人独立激活，每个机器人也独立执行其周期。这意味着机器人在观察、计算、移动和非激活状态中所花费的时间有限但无上界、不可预测，且不同机器人所花费的时间并不相同。我们假设，在机器人的一个观察-计算-移动周期中，观察与计算阶段合在一起并非瞬时完成。机器人没有共同的时间概念。所有机器人以相同速度移动，且移动是刚性的。这里的初始配置是非对称的。机器人具有有限可视范围，这意味着机器人无法看到整个圆。设a 和b 是圆 C上的两个点，则a 和b 之间的角距离是以 C 的中心为顶点、由较短

以a 和b为端点的圆弧所对应的角的度量。机器人的可视范围为 π，也就是说，它能看到所有与其角距离小于 π的其他机器人。

4.2 定义和预备知识

定义 1（配置）。给定一个圆，并在圆上放置有限个机器人。我们称其为圆上的机器人配置，或简称配置。

定义 2（旋转对称配置）。若一个配置中没有重数点，且存在一个以中心为基准的非平凡旋转，使配置保持不变，则称该配置为旋转对称配置。

定义 3（对跖机器人）。如果存在一个机器人 r'，它与机器人 r 的角距离为 π，则称机器人 r 为对跖机器人。在这种情况下，r 和 r' 互为对跖机器人。

注意，不是对跖机器人的机器人称为非对跖机器人。

定义 4。设 r' 和 r'' 是配置中的两个机器人，且位于不同位置。则 $cwAngle$（r'，r''）（$ccwAngle$（r'，r''））表示从 r' 到 r'' 的顺时针（逆时针）角距离。

定义 5（角序列）。设 r 是给定配置中的一个机器人，且该配置中不存在重合点；设 $r_1, r_2, \ldots\ldots, r_n$ 是圆上按顺时针顺序排列的其他机器人。则机器人 r 的角序列为序列

$$(cwAngle(r, r_1), cwAngle(r_1, r_2), cwAngle(r_2, r_3), \ldots, cwAngle(r_n, r)).$$

我们将该序列记为 $\mathcal{S}$（r）。进一步地，我们将子序列记为

$$(cwAngle(r, r_1), cwAngle(r_1, r_2), cwAngle(r_2, r_3), \ldots, cwAngle(r_{i-1}, r_i))$$

$\mathcal{S}$（r）中的 $\mathcal{S}$（r，r_i）称为 $\mathcal{S}$（r，r_i）。此外，我们将 $cwAngle$（r，r_1）称为 r 的前导角。

注意，由于配置最初是旋转非对称的，根据论文 [24] 中的结果，我们可以说所有机器人都具有不同的角序列。

定义 6（字典序）。设 $\tilde{a} = (a_1, \ldots, a_n)$ 和 $b = (b_1, \ldots, b_n)$ 是两个长度相同的有限实数序列。若 $\tilde{a}$ 严格按字典序小于序列 b，即 $a_1 < b_1$，或者存在 $1 < k < n$，使得对于所有 $i = 1, 2, \ldots, k$，$a_i = b_i$，且 $a_{k+1} < b_{k+1}$，则称 a 为严格按字典序更小的序列。若 $\tilde{a} = b$，或者 $\tilde{a}$ 严格按字典序更小，则称 a 为按字典序更小的序列。

定义 7（真正领导者）。在一个配置中，具有字典序最小角度序列的机器人称为真正领导者。

如果配置是旋转非对称的，且不包含重数点，则恰好存在一个机器人拥有严格最小的字典序角度序列。因此，这种配置只有一个真正领导者。由于圆上的机器人无法判断其对跖位置是否被另一机器人占据，因此机器人可以作出两种假设：1）对跖位置为空，称此配置为$C_0(r)$；2）对跖位置非空，称此配置为$C_1(r)$。因此，机器人r 可以形成两个角度序列：一个基于$C_0(r)$配置，另一个基于$C_1(r)$配置。接下来的两个定义是从机器人的视角出发的。如果真正领导者机器人能够确认自己是真正领导者，我们就称其为确定领导者。如果真正领导者或其他机器人会因$C_0(r)$或$C_1(r)$配置是否成立而无法确定自己是否为真正领导者，那么我们称其为困惑领导者。可能存在以下几种情况。

– 可能性 1：$C_0(r)$ 配置具有旋转对称性，因此 $C_1(r)$ 是唯一可能的配置。– 可能性 2：$C_1(r)$ 配置具有旋转对称性，因此 $C_0(r)$ 是唯一可能的配置。– 可能性 3：$C_0(r)$ 和 $C_1(r)$ 均不具有旋转对称性，因此 $C_0(r)$ 和$C_1(r)$ 都可能是配置。

定义 8（确定领导者）。在没有重数点的旋转非对称配置中，如果机器人 r 在任何可能的配置中都是真正领导者，则称其为确定领导者。

请注意，确定领导者必定是该配置的真正领导者。因此，在任何时刻，如果配置是非对称的且不包含重数点，则至多存在一个确定领导者。

定义 9（困惑领导者）。在没有重数点的旋转非对称配置中，如果机器人 r 的 $C_0(r)$ 和 $C_1(r)$ 都是可能的配置，且 r 在一种配置中是真正领导者、在另一种配置中却不是，则称其为困惑领导者。

定义 10（跟随机器人）。在没有重数点的非对称配置中，若某个机器人既不是确定领导者，也不是困惑领导者，则称其为跟随机器人。

定义 11（预期领导者）。在没有重数点的非对称配置中，若某个机器人不是跟随机器人，则称其为预期领导者。也就是说，预期领导者要么是确定领导者，要么是困惑领导者。

请注意，上述定义的设置方式使得确定领导者（或困惑领导者、跟随机器人）能够识别出自己是确定领导者（或困惑领导者、跟随机器人）。

定义 12. 对于位于圆上不同位置的两个机器人 r 和 $r's$，若 $cwAngle(r, r') = \theta$，则定义 $[r,r']$ 为圆上满足$0 \leq cwAngle(r, x) \leq \theta$ 的点 x 的集合；并将(r,r') 定义为圆上满足 $0 < cwAngle(r,x) < \theta$的点 x 的集合。

定义 13. 设配置中有一个机器人 r，若另一个机器人 r_1 s 满足 $cwAngle(r,r_1) > \pi$，则称其位于 r 的左侧；若满足 $cwAngle(r,r_1) < \pi$，则称其位于右侧。

请注意，构型的真正领导者可能变成困惑领导者，而非真正领导者的其他机器人也可能变成困惑领导者。接下来的结果将帮助我们找出预期领导者的最大可能数量。

命题 1. 设 $(a_1, \ldots, a_k)$ 是旋转非对称配置中真正领导者 r_0，的角度序列，且该配置没有重数点。那么，不存在另一个机器人 r' 同时满足以下性质。

1. r' 位于 r_0, 的左侧。*2.* $\mathcal{S}(r', r_0) = (a_1, a_2, \ldots, a_i)$；*3.* $\mathcal{S}(r_0)$ 的前 i 个角依次为 $a_1, a_2, \ldots, a_{i-1}$ 和 a_i。

证明。我们用矛盾法证明这个结果。假设存在这样一个机器人 r′。注意， $\mathcal{S}$(r$_0$) 是 ˜a = (a$_1$, . . . , a$_i$, a$_{i+1}$, . . . , a$_{k-i}$, a$_1$, . . . , a$_i$)，而 $\mathcal{S}$(r′) 是 ˜a$_1$ =(a$_1$, . . . , a$_i$, a$_1$, . . . , a$_i$, a$_{i+1}$, . . . , a$_{k-i}$)。由于 r$_0$ 具有严格最小的角度序列，因此 a$_1$ 是配置中的最小角，并且 a$_{i+1}$ ≤ a$_1$,，这导致 a$_{i+1}$ =a$_1$。接下来，我们证明 a$_2$ = a$_{i+2}$。由于 ˜a 严格来说是最小的角度序列，因此 a$_{i+2}$ ≤ a$_2$。若 a$_{i+2}$ < a$_2$,，则角度序列 (a$_{i+1}$, . . . , a$_{k-i}$, a$_1$, . . . , a$_i$, a$_1$, . . . , a$_i$) 小于 ˜a，这与矛盾。因此 a$_2$ = a$_{i+2}$。类似地，我们可以证明，对于 j = 3, . . . , i，a$_{i+j}$ =a j。现在，如果 2i =k，那么我们看到 ˜a = ˜a$_1$,，这与配置是旋转非对称的事实相矛盾。否则，按照类似的方法，我们可以证明，对于 j = 1, 2, . . . , i，a$_{2i+j}$ =a j。重复同样的论证，我们可以证明，对于 j = 1, 2, . . . , i，a$_{pi+j}$ =a j。由于角的数量有限，经过有限步后，我们必然会得到 k = ti，其中 t ≥ 2，且 ˜a = ˜a$_1$ = (a$_1$, . . . , a$_i$, a$_1$, . . . , a$_i$, . . . , a$_1$, . . . , a$_i$)，这又导致矛盾。

接下来，我们在命题 2 中陈述一个简单的观察结果。

命题 2. 假设存在一个没有重数点且具有真正领导者的旋转非对称配置，真正领导者为 r_0 （例如）。在圆上一个空位置加入机器人 r，且不引入任何旋转对称性后，新配置的真正领导者必定位于 $[r_0, r]$。

对于困惑领导者 r，有两种可能。第一种情况是，r 在$C_0(r)$ 配置中是真正领导者，但在 $C_1(r)$ 中不是。第二种情况是，r 在 $C_1(r)$ 配置中是真正领导者，但在$C_0(r)$ 中不是。我们证明第二种情况不可能发生。我们在下述命题中正式陈述这一结果。

命题 3. 若机器人 r 是一个没有重数点的非对称配置中的困惑领导者，则 r 是 $C_0(r)$ 配置中的真正领导者，但不是 $C_1(r)$ 配置中的真正领导者。

从命题 3 可以观察到，如果一个没有重数点的非对称配置中的真正领导者是困惑领导者，那么其对跖位置必定为空。此外，如果一个并非该配置真正领导者的机器人变成了困惑领导者，那么其对跖位置必定非空。我们将这些观察记录在以下推论中。

推论1. 在一个没有重数点的旋转非对称配置中，如果该配置的真正领导者是困惑领导者，那么其对跖位置必定为空。

推论2。 在没有重数点的旋转非对称配置中，如果除真正领导者以外的某个机器人变成困惑领导者，则其对跖位置必定非空。

命题4。 设 C 是一个没有重数点的旋转非对称配置，且 L 是该配置的真正领导者，则 C 的另一个预期领导者 r 必须满足 $cwAngle(L, r) \geq \pi$。

证明。 假设存在另一个预期领导者 r，使得 顺时针角 $(L, r) < \pi$。由于 r 不是该配置的领导者，因此 r 必定是困惑领导者。由命题 3 可知，在$C_0(r)$ 配置中，r 是真正领导者，而在$C_1(r)$ 配置中，L 是真正领导者。这与命题 2 矛盾。

命题5。 对于任意给定的没有重数点的旋转非对称配置，除真正领导者之外至多可以有一个困惑领导者。

证明。设 C 为给定的配置。若可能，设 r_1 和 r_2 是除真正领导者 L 之外的两个困惑领导者。根据命题4，首先有 $cwAngle(L, r_i) \geq \pi$，其中 $i = 1、2$。不失一般性地，假设 $cwAngle(L, r_1) < cwAngle(L, r_2)$。由于 r_i 是困惑领导者，根据命题 3，它们的对跖位置非空。对每个 i，设 r_i' 为 r_i 的对跖机器人。这里有两种穷尽的情形。第一种是 $cwAngle(L, r_1) = \pi$，第二种是 $cwAngle(L, r_1) > \pi$。

情形I：$cwAngle(L, r_1) = \pi$

情形IA：在这种情况下，设 r_1 和 L 的角度序列的第一个角不同。设 $\mathcal{S}(L)$ 和 $\mathcal{S}(r_1)$ 中的第一个角分别为 θ 和 θ_1，则 $\theta < \theta_1$。由于 r_2' 不可能是 L 的顺时针邻居，因此 r_2 可以看到 L 的前导角。因此，为了成为困惑领导者，r_2 的前导角必须为 θ。现在可能有两种情况。首先，r_2 的顺时针邻居可能是 L，也可能不是。如果 r_2 的顺时针邻居是 L，那么根据命题 2，L 不再是该配置的真正领导者，这就产生了矛盾。其次，如果 r_2 的顺时针邻居不是 L，那么 r_1 可以看到 r_2，的前导角，即 θ，它小于 r_1 的前导角。这说明 $C \setminus L$ 不可能让 r_1 成为真正领导者。这与 r_1 是困惑领导者这一事实相矛盾。

情形IB：设 $\mathcal{S}(L)$ 和 $\mathcal{S}(r_1)$ 的前 t 个角相同，而 $(t+1)^{th}$ 个角不同。设 L 的前 t 个顺时针邻居按顺时针顺序为 $x_1, x_2, \ldots\ldots, x_t$；$r_1$ 的前 t 个顺时针邻居按顺时针顺序为 x_1'、x_2'、……、x_t'。首先，我们证明 r_2 不是任何一个 x_i'。假设 $r_2 = x_i'$，其中 $i < t$。那么，r_2 可以看到 $\mathcal{S}(L)$ 和 $\mathcal{S}(r_1)$ 的前 $i-1$ 个角。由于 r_2 的第 i^{th} 个顺时针邻居位于 r_2' 的左侧，r_2 在原始构型中可以看到 $\mathcal{S}(r_2)$ 的前 i 个角。首先，我们注意到 r_2 和 L 的前 $i-1$ 个角相同。否则，r_2 就不是困惑领导者，或者 L 就不是构型的领导者。现在我们看到，$\mathcal{S}(L)$ 和 $\mathcal{S}(r_2)$ 的第 i^{th} 个角也相同。假设 $\mathcal{S}(L)$ 和 $\mathcal{S}(r_2)$ 的第 i^{th} 个角分别为 θ_i 和 α_i，且$\alpha_i > \theta_i$（注意，情形 $\alpha_i < \theta_i$ 可由 L 是真正领导者这一事实排除。

构型）。由于i^{th} 个角r_1 也为 θ_i，且r_2，可以看到这个角，因此r_2 就不会是困惑领导者。因此，$\mathcal{S}(r_1)$ 和 $\mathcal{S}(r_2)$ 的前i 个角相同。因此，根据命题 2，r_2 不可能是困惑领导者。因此，对每个$i < t$，$r_2 \neq x'_i$。现在很容易看出$r_2 \neq x_t$，否则，r_2 的前导角会严格大于L 的前导角，从而r_2 不是困惑领导者。

因此，对于每个 $1 \leq \mathrm{i} \leq \mathrm{t}$，$\mathrm{r}_2 \neq \mathrm{x}'_i$。因此请注意，$\mathrm{r}_2$ 可以看到 $\mathcal{S}(\mathrm{L})$ 的前 t 个角 $+1$，而要继续作为困惑领导者，$\mathcal{S}(\mathrm{r}_2)$ 的前 t 个角 $+1$ 必须与之匹配。现在有两种情况：要么 $(\mathrm{t}+1)^{th}$ 是 r_2 的顺时针邻居，要么不是。如果不是，那么 r_1 可以看到 $\mathcal{S}(\mathrm{r}_2)$ 的前 t 个角 $+1$。而且，$\mathcal{S}(\mathrm{r}_2)$ 的第 t 个角 $+1^{th}$ 小于 $\mathcal{S}(\mathrm{r}_1)$ 的相应角。因此，r_1 不会是困惑领导者。另一种情况是，如果 $(\mathrm{t}+1)^{th}$ 是 r_2 的顺时针邻居，那么根据命题 2，L 不再是构型的领导者。因此，在情形I中，如果除真正的领导者外还有多个困惑领导者，就会得出矛盾。

情形II：顺时针角$(L, r_1) > \pi$

情形*IIA*：在这种情形下，令r_1 和L 的角度序列中的第一个角不同。令 $\mathcal{S}(L)$和 $\mathcal{S}(r_1)$中的第一个角分别为 θ 和 θ_1，则 $\theta < \theta_1$。由于在这种情形下r'_2 不可能是L的顺时针邻居，因此r_2 可以看到L的前导角。因此，要成为困惑领导者，r_2 的前导角必须是 θ。现在r_1 可以看到r_2, 的前导角，即 θ，它小于r_1的前导角。这说明 $C \setminus r'_1$ 不可能以 r_1 为真正领导者。这与r_1 是困惑领导者这一事实矛盾。

情形IIB：令 $\mathcal{S}(\mathrm{L})$和 $\mathcal{S}(\mathrm{r}_1)$的前t个角相同，而第$(\mathrm{t}+1)^{th}$ 个角不同。令L的顺时针方向上前t个顺时针邻居依次为$\mathrm{x}_1, 、\mathrm{x}_2, \ldots\ldots \mathrm{x}_t$，$\mathrm{r}_1$ 的顺时针方向上前t个顺时针邻居依次为x'_1、$\mathrm{x}'_2 \ldots\ldots \mathrm{x}'_t$。由于$\mathrm{r}_1$ 与L不对跖，因此r'_2 不可能是L顺时针方向的第一个邻居。借鉴情形I中的论证，我们可以得出r_2 不是任何一个x'_i。在其他情况下，r_2 可以看到 $\mathcal{S}(\mathrm{L})$的前t个角；要成为困惑领导者，$\mathcal{S}(\mathrm{r}_2)$的前t个角必须与 $\mathcal{S}(\mathrm{L})$的前t个角相同。现在我们看到，r_1 可以看到$\mathcal{S}(\mathrm{r}_2)$的前t个角。否则，由于$\mathrm{r}'_1$ 位于L的右侧，L 至多是r_1的第t^{th} 个邻居。这意味着，根据命题 2，L不是该配置的真正领导者。因此，r_1 可以看到 $\mathcal{S}(\mathrm{r}_2)$的前t个角。由于 $\mathcal{S}(\mathrm{r}_2)$的第$(\mathrm{t}+1)^{th}$ 个角小于 $\mathcal{S}(\mathrm{r}_1)$中相应的角，因此$\mathrm{r}_1$ 不再是该配置的困惑领导者。这是一个矛盾。

因此可以证明，除了配置中的真正领导者之外，困惑领导者至多只有一个。

设 C 是一个没有重数点的旋转非对称配置。那么 C 将有一个真正领导者，根据上述命题 2，此外至多还有一个困惑领导者。因此，对于 C，我们可以有以下四种穷尽情形。

1. C 恰有一个预期领导者，且该领导者是确定领导者。 2. C 恰有一个预期领导者，且该领导者是困惑领导者。 3. C 恰有两个预期领导者，且二者都是困惑领导者。 4. C 恰有两个预期领导者。其中一个是确定领导者，另一个是困惑领导者。

图 1 和图 2 展示了上述四种情况均存在。对于第三种情况，我们在以下命题中观察到两个性质。

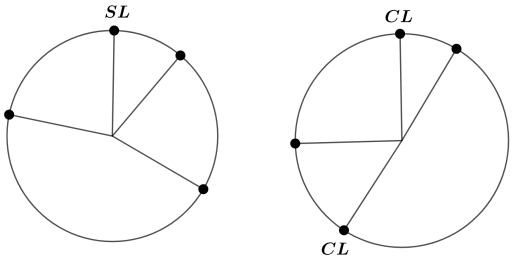

图1：左图的配置中只有一个确定领导者（SL），右图的配置中有两个困惑领导者（CL）。

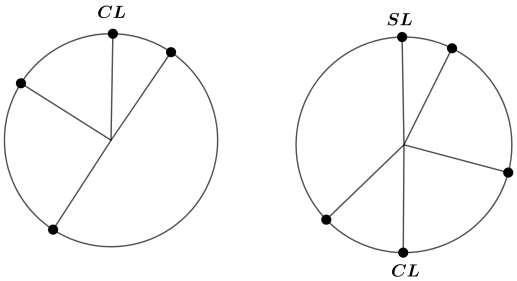

图2：左图的配置中只有一个困惑领导者（CL），右图的配置中有一个确定领导者（SL）和一个困惑领导者（CL）。

命题6。 对于没有重合点的旋转非对称配置，如果存在两个困惑领导者，那么他们不可能互为对跖点。

证明。 设p 和q 是一个旋转非对称且没有重数点的配置中的两个困惑领导者。此时p 和q 中必有一个是该配置的真正领导者。不失一般性，设p 是构型的领导者。根据推论1，p 的对跖位置必为空。因此，另一个困惑领导者q 不能位于p的对跖位置。

命题7。 若一个旋转非对称且没有重数点的配置中存在两个困惑领导者，则它们的顺时针邻居不能彼此对跖。

证明。 若可能，设两位困惑领导者的顺时针邻居彼此处于对跖位置。设L 为该配置的真正领导者，r 为另一位困惑领导者。由命题4和命题6可得顺时针角$(L,r) > \pi$。于是L 的前导角严格大于r 的前导角（图3）。这与L 是该配置的真正领导者这一事实相矛盾。

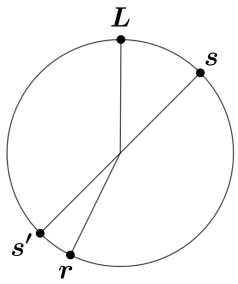

图3：与命题7相关的图像

5 所提算法

本节给出适用于具有有限内存的异步机器人的所提算法。初始配置是旋转非对称的，且机器人位于圆上的不同点。每个机器人都有 π 可视范围，这意味着机器人看不到其对跖位置。每个机器人都能以匀速沿圆弧移动。所有圆的移动速度都相同。所提算法见算法1。

算法讨论：机器人最初分散在圆上的不同点，且初始配置是旋转非对称的。机器人具有有限可视范围。机器人无法看到自己的对跖位置。由于初始配置是旋转非对称的，因此每个机器人都有不同的角度序列。机器人r被激活后，首先检查自己是否是其可视范围内唯一的机器人。如果是，则顺时针移动 $\pi/2$ 角度。否则，它会寻找重数点。如果可见配置中存在重数点，则有两种可能。第一种可能是r自身位于重数点上；第二种可能是r不位于重数点上。如果r 不位于重数点上，且某个重数点是r的顺时针或逆时针邻居，那么r 就移向距离较近的重数点。假设r 位于某个重数点上。在这种情况下，如果存在另一个重数点，且该点与r 所在的位置不同，并且该重数点的顺时针距离小于 π，那么r 就移向该重数点。接下来，如果可见配置中不存在重数点，它就会判断自己是预期领导者还是跟随机器人。如果r 是跟随机器人，则不执行任何操作。如果r 是确定领导者且其状态为关闭，则移向第一个顺时针邻居。如果r 是困惑领导者，则检查其状态。如果其状态为终止，则不执行任何操作。如果其状态为关闭，则有以下几种情况。情形I：如果r的第一个顺时针邻居是安全的，那么r 就移至其邻居所在的位置。情形II：如果r 的第一个顺时针邻居不安全，且C_0(r)配置中存在另一个困惑领导者（不是r），那么r不执行任何操作。情形III：如果情形I和情形II均不成立，那么r 将其状态改为半步移动，并顺时针移动 $\theta/2$ 的角距离。

接下来，假设r 被激活时处于半步移动状态。如果r的第一个顺时针邻居（设为r'）不是对跖的，那么r 将其状态改为终止，并且

不移动。否则，令s 为位于r' 的对跖位置上的机器人，且r的前导角为 $\theta/2$。接着，如果r能看到圆弧 $[s - \frac{\theta}{2}R、 s + \frac{\theta}{2}R)$内的某个机器人，那么$r$将其状态改为终止，并沿逆时针方向移动 $\theta/2$角距离。如果r' 处于对跖位置，且r看不到圆弧 $[s - \frac{\theta}{2}R、 s + \frac{\theta}{2}R)$内的任何机器人，那么$r$将状态改为多步移动，并沿顺时针方向移动 $\theta/4$角距离。

接下来，假设r 被激活时处于多步移动状态。如果r的第一个顺时针邻居（设为r'）不是对跖的，那么r 会将其状态改为终止，并且不移动。否则，设s 为位于r' 的对跖位置的机器人，且r的前导角为 $\theta/4$。此时，若圆弧 $[s - \frac{\theta}{4}R、 s + \frac{\theta}{4}R)$上存在$r$可见的机器人，则$r$会将其状态改为终止，并沿逆时针方向移动 $3\theta/4$角距离。若r' 是对跖的，且圆弧 $[s - \frac{3\theta}{4}R、 s + \frac{\theta}{4}R)$上不存在$r$ 可见的机器人，则r 会将状态改为关闭，并移动到其第一个顺时针邻居所在的位置。

请注意，根据确定领导者（困惑领导者、跟随机器人）的定义，机器人可以判断自身是否为确定领导者（困惑领导者、跟随机器人）。

在正式介绍所提算法之前，我们先定义安全顺时针邻居。

定义14（安全邻居）。设 r 是一个困惑的预期领导者，s 是 r 的顺时针方向的第一个邻居。如果配置 $C_1(r)$ 中真正领导者的顺时针方向的第一个邻居与 s 不对跖，则称机器人 s 是 r 的安全邻居。

接下来，我们正式介绍算法1中给出的算法。

接下来，我们将初始配置（即没有重合点的旋转非对称配置）分类如下：

– 配置-A：配置中只有预期领导者是真正领导者。如果预期领导者是困惑领导者，则其顺时针方向的第一个邻居是安全的。– 配置-B：配置中有两个预期领导者。

- 配置-BI：当不是真正领导者的那个困惑领导者发现其顺时针方向的第一个邻居是安全的。
- 配置-BII：当困惑领导者（并非真正领导者）发现其顺时针第一个邻居不安全时。

– 配置-C：一名预期领导者同时也是困惑领导者，它发现自己的顺时针第一个邻居不安全。

6 正确性

我们将证明：如果所有机器人最初位于一个旋转非对称配置中，且不存在重数点，那么在有限时间内执行算法1后，所有机器人最终会聚集于同一点，并在异步调度器下不再移动。首先，我们证明，从初始配置开始执行算法1有限次后，机器人将创建至少一个、至多两个重数点。然后，从具有一个或两个重数点的配置出发，机器人最终会聚集到其中一个重数点，并且不再继续移动。

算法1： 可视范围为 π 的聚集算法

通用机器人r 以关闭状态执行该算法；
输入： 机器人r可见的机器人占据的点集
输出： 机器人r的目标点
如果 存在可见的机器人**则**
  如果不存在重数点**则**
    如果如果机器人 r 的状态为关闭**则**
      如果机器人 r 是确定领导者**则**
        移动到其顺时针邻居的位置；
      否则，如果机器人 r 是困惑领导者**则**
        如果r 的顺时针方向的第一个邻居是安全的**则**
          移动到其顺时针方向的第一个邻居所在的位置
        否则如果 $C_0(r)$ 配置中没有另一个困惑领导者比 r **则**
          令 r 的前导角为 θ；
          将状态更改为半步移动，并沿 $\frac{\theta}{2}$ 角距离移动顺时针；
    否则，如果机器人 r 处于半步移动状态，**则**
      令r 的前导角为 $\frac{\theta}{2}$；
      令s 为r的对跖位置；
      如果r 的顺时针邻居不是对跖，**则**
        将状态更改为终止；
      否则，如果圆弧 $[s-\frac{\theta}{2}R,\ s+\frac{\theta}{2}R)$ 中没有机器人，**则**
        将状态更改为多步移动；
        顺时针移动 $\frac{\theta}{4}$ 角距离；
      否则，如果圆弧 $[s-\frac{\theta}{2}R,\ s+\frac{\theta}{2}R)$ 上有一台机器人**那么**
        将状态更改为终止；
        逆时针移动 $\frac{\theta}{2}$ 角距离；
    否则，如果机器人 r 的状态为多步移动**那么**
      令r 的前导角为 $\frac{\theta}{4}$；
      令s 为r的对跖位置；
      如果r 的顺时针邻居不是对跖位置**则**
        将状态更改为终止；
      否则，如果圆弧 $[s-\frac{3\theta}{4}R、\ s+\frac{\theta}{4}R)$ 内没有机器人**则**
        将状态更改为关闭；
        移动到其顺时针方向的第一个邻居所在的位置；
      否则，如果圆弧 $[s-\frac{3\theta}{4}R, s+\frac{\theta}{4}R)$ 上有机器人**则**
        将状态更改为终止；
        逆时针移动 $\frac{3\theta}{4}$ 角距离；
  否则，如果存在重数点，但机器人 r 不在任何重数点上**则**
    如果其顺时针或逆时针邻居是重数点**则**
      移动到较近的重数点
  否则如果唯一的可见位置是重数点**则**
    如果从重数点起的顺时针角距离为 $<\pi/2$ **则**
      移动到重数点
否则
  沿顺时针方向移动 $\frac{\pi}{2}$ 距离

引理1. 若初始配置为配置-*A*类型，则算法*1*执行有限时间后，至少会形成一个重数点。

引理2. 若初始配置为配置-*B*类型，则算法*1*执行有限时间后，至少会形成一个重数点。

引理3. 若初始配置为配置-*C*类型，则算法*1*执行有限时间后，至少会形成一个重数点。

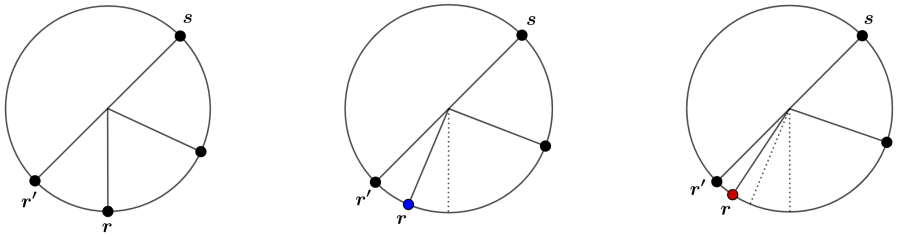

图4：在配置-C类型中，一个困惑领导者（r）的移动，其中r 的初始颜色为关闭，蓝色表示半步移动，红色表示多步移动

定理1. 设 C 是一个没有重数点的旋转非对称配置，则机器人在 C 上执行算法*1*有限时间后，至少会形成一个重数点。

证明。 设 C 是一个没有重数点的旋转非对称配置。那么，所有可能的配置可分为三类：配置-A、配置-B 和配置-C。

根据引理1、引理2和引理3，我们可以得出，在任何类型的配置中，至少会形成一个重数点。

引理4。 对于任何没有重数点的旋转非对称配置，通过有限时间执行算法*1*，机器人最多可以形成两个重数点，然后所有机器人聚集到圆上的某一点。

因此，我们可以得出以下定理。

定理 2。存在一种聚集算法，能够在异步调度器下，从任意初始旋转非对称配置出发，聚集具有有限内存且可视范围为 π 的任意一组机器人。

7 结论

本文提出了一种在异步调度器下，使具有有限内存的机器人在圆上聚集的算法，其可视范围为 π。机器人最初位于圆上互不相同的位置，构成任意旋转非对称配置。我们假设每个机器人都具有有限持久记忆。对于这一问题的后续研究，如果能在机器人无记忆或可视范围小于 π 时给出聚集算法，将会很有意义。

参考文献

1. Agmon, N., Peleg, D.：自主移动机器人容错聚集算法。SIAM J. Comput. **36**(1), 56–82 (2006) 2. Ando, H., Oasa, Y., Suzuki, I., Yamashita, M.：面向有限可视范围移动机器人的分布式无记忆点收敛算法。IEEE Transactions on Robotics and Au- tomation **15**795787 3 5 818 828 1999 10 1109 70 ( ), – ( ). https://doi.org/ . / . . Bhagat, S., Chakraborty, A., Das, B., Mukhopadhyaya, K.：无限网格上的会合节点聚集*。Fundam. Informaticae **187**222128 4 1 1 30 2022 10 3233 ( ), – ( ). https://doi.org/ . /FI- . Bhagat, S., Mukhopadhyaya, K., Mukhopadhyaya, S.：受限可视范围下的计算。载于：Flocchini, P., Prencipe, G., Santoro, N.（编），《移动实体的分布式计算》，《移动与计算前沿研究》，《计算机科学讲义》，第 11340, 134 183 2019 10 1007 978 3 030 11072 7 7 卷,- 第 – 5 页。施普林格（）。https://doi.org/ . / - - - - . Bouchard, S., Dieudonne, Y., Ducourthial, B.：网络中的拜占庭聚集。CoRR**abs/1504.01623**(2015) 6. Buchin, K., Flocchini, P., Kostitsyna, I., Peters, T., Santoro, N., Wada, K.：自主移动机器人：细化计算领域。载于：IEEE国际并行与分布式处理研讨会专题研讨会，IPDPS Workshops 2021，美国俄勒冈州波特兰，2021年6月17日至21日。第576–585页。IEEE（2021）。https://doi.org/10 00091 7 1109 52791 2021 . /IPDPSW . . . Casteigts, A., Flocchini, P., Quattrociocchi, W., Santoro, N.：时变图与动态网络。CoRR **abs/1012.0009** (2010) 8. Cieliebak, M., Flocchini, P., Prencipe, G., Santoro, N.：移动机器人分布式计算：聚集。SIAM J. Comput. **41**(4), 829–879 (2012) 9. Czyzowicz, J., Kosowski, A., Pelc, A.：遗忘时如何会合：任意图上的对数空间会合。Distributed Comput. **25**(2), 165–178 (2012) 10. Czyzowicz, J., Kosowski, A., Pelc, A.：多边形地形中异步有界记忆智能体的确定性会合。THEORY OF COMPUTING SYSTEMS **52**(2), 179– 199 (2013) 11. D'Angelo, G., Stefano, G.D., Klasing, R., Navarra, A.：在没有重数检测的情况下，在匿名网格和树上聚集机器人。Theor. Comput. Sci. **610**, 158–168 (2016)。https://doi.org/ 10 1016 2014 06 045 . /j.tcs. . . . Defago, X., Potop-Butucaru, M.G., Clement, J., Messika, S., Parv´edy, P.R.：容错及拜占庭容错自稳定移动机器人聚集——可行性研究——。CoRR**abs/1602.05546**(2016) 13. Degener, B., Kempkes, B., Langner, T., Meyer auf der Heide, F., Pietrzyk, P., Wattenhofer, R.：有限可视范围下自主机器人的同步聚集紧确运行时间界。载于：《第二十三届ACM算法与体系结构并行性研讨会论文集》。第139–148页。SPAA '11，美国计算机协会，纽约，纽约州，美国（2011）。https://doi.org/10.1145/1989493.1989515

14. Dieudonn´e, Y., Petit, F., Villain, V.：领导者选举问题与模式形成问题。CoRR **abs/0902.2851** (2009) 15. Feinerman, O., Korman, A., Kutten, S., Rodeh, Y.：不同速度智能体在环上的快速会合。载于：分布式计算与网络——第15届国际会议，ICDCN 2014，印度哥印拜陀，2014年1月4日至7日。会议论文集。第1–13页（2014）。https://doi.org/10 9 1 16 1007 978 3 642 45249 . / - - - - . Flocchini, P., Killick, R., Kranakis, E., Santoro, N., Yamashita, M.：移动机器人在连续环上的聚集与选举。载于：Lu, P., Zhang, G.（编），第30届算法与计算国际研讨会，ISAAC 2019，2019年12月8日至11日，上海财经大学，中国上海。LIPIcs，第149卷，第8:1–8:19页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik（2019）。https://doi.org/10 8 17 4230 2019 . /LIPIcs.ISAAC. . . Flocchini, P., Prencipe, G., Santoro, N.（编）：《移动实体的分布式计算》，《移动与计算前沿研究》，《计算机科学讲义》，第11340卷。施普林格（2019）18. Flocchini, P., Prencipe, G., Santoro, N., Widmayer, P.：有限可视范围下异步机器人的聚集。Theor. Comput. Sci. **337**(1-3), 147–168 (2005)。https://doi.org/10 001 19 1016 2005 01 . /j.tcs. . . . Goswami, P., Sharma, A., Ghosh, S., Sau, B.：无限三角网格上目光短浅机器人的时间最优聚集。载于：Devismes, S., Petit, F., Altisen, K., Luna, G.A.D., Anta, A.F.（编），《分布式系统的稳定化、安全与安保——第24届国际研讨会》，SSS 2022，法国克莱蒙费朗，2022年11月15日至17日，会议论文集。《计算机科学讲义》，第13751卷，第270–284页。施普林格（2022）。https://doi.org/10.1007/978- 3-031-21017-4 18 20. Guilbault, S., Pelc, A.：正则二分图中具有局部视野的异步无记忆智能体聚集。载于：Kosowski, A., Yamashita, M.（编），《结构信息与通信复杂性——第18届国际研讨会》，SIROCCO 2011，波兰格但斯克，2011年6月26日至29日。会议论文集。《计算机科学讲义》，第6796卷，第162–173页。施普林格（2011 2 15 21 10 1007 978 3 642 22212 ）。https://doi.org/ . / - - - - . Huus, E., Kranakis, E.：环上不同速度的多个智能体会合。载于：Papavassiliou, S., Ruehrup, S.（编），《自组织、移动与无线网络——第14届国际会议》，ADHOC-NOW 2015，希腊雅典，2015年6月29日至7月1日，会议论文集。《计算机科学讲义》，第9143卷，第195–209页。施普林格（2015）。https://doi.org/10 6 14 22 1007 978 3 319 19662 . / - - - - . Luna, G.A.D., Flocchini, P., Pagli, L., Prencipe, G., Santoro, N., Viglietta, G.：动态环中的聚集。CoRR **abs/1704.02427** (2017) 23. Luna, G.A.D., Flocchini, P., Santoro, N., Viglietta, G., Yamashita, M.：匿名无记忆机器人在多边形中的会合。Distributed Comput. **33**(5), 445–469 (2020) 24. Luna, G.A.D., Uehara, R., Viglietta, G., Yamauchi, Y.：匿名无记忆机器人在有限可视范围下于圆上聚集。载于：Attiya, H.（编），第34届分布式计算国际研讨会，DISC 2020，2020年10月12日至16日，虚拟会议。LIPIcs，第179卷，第12:1–12:17页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik（2020）。https://doi.org/10 12 25 4230 2020 . /LIPIcs.DISC. . . Poudel, P., Sharma, G.：有限可视范围下的普遍最优聚集。载于：Spirakis, P.G., Tsigas, P.（编），《分布式系统的稳定化、安全与安保——第19届国际研讨会》，SSS 2017，美国马萨诸塞州波士顿，2017年11月5日至8日，会议论文集。《计算机科学讲义》，第10616卷，第323–340页。施普林格（2017）。https://doi.org/10 1007 978 3 319 69084 1 23 . / - - - - . Souissi, S., Defago, X., Yamashita, M.：利用最终一致罗盘，在有限可视范围下聚集无记忆移动机器人。载于：Datta, A.K., Gradinariu, M.（编），《分布式系统的稳定化、安全与安保》，第8届国际研讨会，SSS

2006年，达拉斯，德克萨斯州，美国，2006年11月17日至19日，会议论文集。《计算机科学讲义》第4280卷，第484–500页。施普林格（2006）。https://doi.org/10.1007/978-3-540-49823- 0 34
–