

超宽频群范围

山锋*, 曾嘉欣†, 李增宝*, 罗俊洲*, 吴伟伟*

* 东南大学计算机科学与工程学院, 江苏南京 210096†东南大学-莫纳什联合研究生院, 江苏苏州 215123
‡网络科学与工程学院, 江苏南京 210096 邮箱:{shanfeng, jiaxinzeng, lizengbao, jl原因, weiweiwu}@seu.edu.cn

摘要:如今, 空中和地面机器人、可穿戴和便携式设备正变得更小、更轻、更便宜, 从而受到欢迎。现在可以利用成千上万的蜂群来完成复杂的协作任务, 如搜索、救援、测绘和战斗。蜂群通常包含大量的机器人或设备, 它们彼此之间的距离很短, 可以动态移动。因此本文对动态密集群进行了研究。超宽带(UWB)技术被提出作为组网和定位的基础技术, 因为超宽带对时间非常敏感, 可以通过发送和接收数据包的时间戳计算出准确的距离。本文设计了一种超宽带群测距协议, 该协议具有简单高效、自适应鲁棒、可扩展和支持等特点。这个蜂群测距协议被一部分一部分地介绍, 以揭示它对每一个这些特性的支持。它是在 crazyfly 2.1 无人机上实现的, 由 STM32 微控制器驱动的空机器人, 机载超宽带无线收发芯片 DW1000。在一个紧凑的区域内, 用 9 架疯狂苍蝇无人机进行了大量的真实实验来验证所提出的协议。

索引术语: 超宽带, 群, 测距, 协议设计, 网络, 双向测距

我的介绍。

随着电子制造业的快速发展, 越来越多的空中和地面机器人、可穿戴和便携式设备进入商用。例如, 比特热在 2019 年 2 月发布了微型无人机 crazyfly 2.1[1], 大疆在 2020 年 3 月发布了 RoboMaster EP[2], 波士顿动力在 2020 年 6 月推出了 SPOT 腿机器人[3], 各厂商的智能手机和智能手表都在不断发布和升级。

由于这些机器人和设备的体积越来越小, 重量越来越轻, 价格越来越便宜, 因此越来越受欢迎, 现在可以利用数以万计的机器人和设备组成一个机器人和设备群, 相互合作完成复杂的任务。例如, 一组室内无人机搜索给定的目标, 一群小型机器人探索未知的室内环境并绘制地图, 一群有腿的机器狗在森林深处战斗。与全功能的大型单个机器人相比, 一群小型机器人和设备具有容错更高、部署规模和数量更灵活、部署速度更快的优点。因此, 机器人和设备群是目前的研究趋势, 在《自然》和《科学》等顶级期刊[4], [5], [6], [7]上出现了文章。

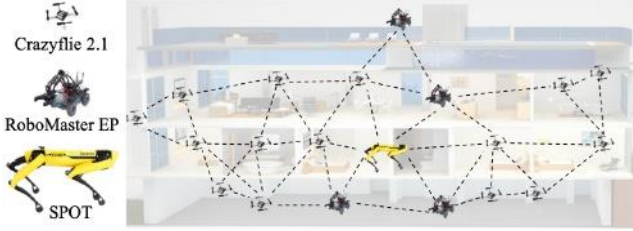


图 1: 机器人和设备群是动态密集的。

我们总结了机器人和设备在这样的群体中的三个重要特征: 数量多、机动性高和距离短。首先, 根据任务的复杂性, 可能部署数以万计的机器人和设备进行协作。第二, 微型无人机、轮式和腿式机器人、人类携带的可穿戴和便携设备, 都能够按照要求快速移动。第三, 由于这些机器人和设备体积小, 他们可以在短距离内合作, 以完成复杂的任务。综上所述, 在未来的应用中, 机器人和设备群将是动态的、密集的。

一个成功的动态和密集的群应用程序需要低延迟通信和实时本地化。在没有外部支持基础设施的情况下, 实现自组网和群内相对定位是至关重要的。

本文提出利用超宽带(UWB)无线电技术实现蜂群的组网和定位。超宽带对时间非常敏感, 可以通过发送和接收数据包的时间戳计算出准确的距离。换句话说, 无线通信和距离测距可以同时实现, 注意到远程距离是定位的基础。本文研究了一种针对动态密集群的超宽带测距协议。

我们总结这些挑战如下。

大量。一个蜂群包含大量的机器人和设备, 由于无线测距信息的广播性质, 它们访问相同的无线信道。不合适的协议可能会导致测距消息冲突, 从而影响测距频率, 而复杂的协议需要较高的计算量, 这对于低端机器人和设备来说是非常宝贵的。因此, 如何设计一种简单而高效的超宽带测距协议是一个挑战。

高流动性。高频测距要求高

流动性。但机器人和设备也可以根据要求保持静止。当两个机器人或设备相距较远且几乎不移动时，应降低测距频率，以节省宝贵的无线信道资源。此外，高移动性会导致无线信道不稳定，导致测距数据包丢失。如何设计一个自适应的、鲁棒的超宽带测距协议是另一个挑战。

短的距离。在一个群体中，两个机器人或设备之间的距离可能很短，因此一个机器人或设备可能有密集的邻居。但是，测距消息对于每个邻居携带的测距信息(时间戳)有大小限制。此外，已经有一些网络和本地化协议。一个好的测距协议应该支持和兼容它们。因此，如何设计一个可扩展的

和兼容的 UWB 测距协议是第三个挑战。因此，本文的目标是针对动态密集的机器人和设备群设计一种超宽带测距协议。本文的贡献总结如下。

- 1)据我们所知，我们率先将超宽带技术应用于密集动态的蜂群中，从而同时支持无线测距和数据通信。
- 2)我们设计了一个简单而高效的超宽带测距协议。该设计利用了无线测距报文的广播特性，并且只有一种易于实现的消息类型，因此在群中具有更高的效率。
- 3)设计了一种自适应、鲁棒的超宽带测距协议。两个测距对应物之间的测距频率随相对速度和距离的变化而变化。适当处理丢包和频率失配。
- 4)设计了一种可扩展、兼容的超宽带测距协议。设计了一种旋转方案来处理单个测距消息中携带过多邻居信息的情况。我们的协议与更高级别的网络协议(如 OLSR)兼容。
- 5)该协议在 crazyfly 2.1 无人机上实现，采用 STM32 微控制器为空中机器人提供动力，搭载 UWB 无线收发芯片 DW1000。在一个紧凑的区域内对 9 只 crazyfly 进行了大量的现实实验来验证该协议。

本文的其余部分组织如下。第二部分介绍了超宽带和测距协议的基础知识。第三部分给出了群测距协议的详细设计。第四节给出了实现细节。第五节对实验结果进行了讨论，第六节对相关工作进行了总结。第七节对本文进行了总结。

2 初步

A. 超宽带(UWB)技术

超宽带(UWB)是一种能在宽频范围内以高码率传输数据的无线通信技术。这些比特是用宽度为 1ns 的细脉冲传播的。

一个典型的超宽带无线电收发芯片跨越无线电频率波段从 3.5 GHz 到 6.5GHz，测距精度小于 10 cm[8]。最新的 iPhone 11(撰写本文时)已经配备了超宽带芯片[9]。

B. 双向测距(DS-TWR)协议

在 IEEE 标准 802.15.4-2011[10]中已经存在一个测距协议，后来由行业巨头 Decawave[11]，[8]改进。我们将改进后的测距协议称为双向测距(DS-TWR)协议，在此简单介绍。如图 2 所示，

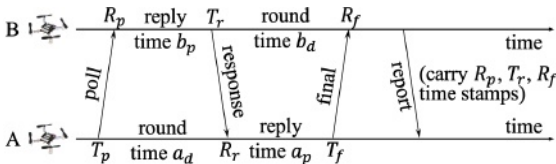


图 2:双向测距(DS-TWR)协议。

有四种类型的消息，即，*poll, response, final, report* 消息，在 A 和 b 双方之间交换，我们定义它们的发送和接收时间戳为 $T_p, R_p, T_r, R_r, T_f, R_f$ ，分别。我们对双方的回复和回合时长的定义如下

$$a_d = R_r - T_p, b_p = T_r - R_p, b_d = R_f - T_r, a_p = T_f - R_r, (1)$$

如图 2 所示。让 t_p 为飞行时间(ToF)，即无线电信号传播时间。那么，两轮时间持续时间可以写成 $a_d t = 2p + b_p b_d t = 2p + a_p$ 。所以 ToF 可以计算为[11]，[8]

$$t_p = \frac{a_d b_d - a_p b_p}{a_d + b_d + a_p + b_p} (2)$$

然而，双方都有由晶体偏移引起的时钟偏移。设晶体误差为 e_A 和 e_B 分别。因此，我们实际计算的 ToF 为

$$\hat{t}_p = \frac{a_d(1 + e_A) \times b_d(1 + e_B) - a_p(1 + e_A) \times b_p(1 + e_B)}{a_d(1 + e_A) + b_d(1 + e_B) + a_p(1 + e_A) + b_p(1 + e_B)}.$$

它与真实值的偏差是

$$\hat{t}_p - t_p = \frac{e_A + e_B + 2e_A e_B}{2 + e_A + e_B} t_p. (3)$$

可以看出，如果 e_A 和 e_B 都很小，偏差也很小。

值得注意的是，在大多数机器人和设备的操作系统中，传输时间戳只有在消息发送后才可用。因此，需要一个附加的报告消息来携带三个相关的时间戳，以便进行 ToF 计算。

C. 基于令牌环的测距

DS-TWR 协议是为一对一测距设计的。随着对多端测距需求的增加，提出了一种简单的扩展 DS-TWR 协议[12]。采用令牌环技术

控制测距过程。基本的思想是所有的边都形成一个环，并且只有一个令牌，所以在任何时候最多有一边持有令牌，允许它依次为所有邻居初始化 DS-TWR 进程，一个接一个。一旦持有令牌的一方完成了范围，令牌被传递给下一方。

虽然这个扩展很容易理解，但它有一些缺点。(1)每当双方交换消息时，由于无线通信的广播性质，这些消息可以被许多邻居听到。然而，他们被忽视，这是缺乏效率。(2)在形成令牌环之前，必须知道所有的邻居。这就带来了额外的邻居发现和过期机制，必须设计，这是缺乏可伸缩性。(3)无线通信不可靠，存在丢包的风险。当令牌丢失时，测距停止。因此，token 必须被监控并从丢失中恢复，这是一个复杂的实现过程。

这启发了我们从零开始设计一种新颖的蜂群测距协议。

D. 群体测距的基本思想

回想一下，DS-TWR 协议有四种类型的消息，与双方互动，答疑应战消息。对于新的蜂群测距协议，我们需要它尽可能简单，所以我们使用单一类型的消息，对测距消息命名。每一方定期传输一个范围消息，而不是回复任何收到的消息。但是，ToF 计算需要 6 个时间戳根据对于式 (1) 和 (2)，如何仅用范围 - 实现测距荷兰国际集团(ing)的消息吗?图 3 中的三面玩具示例提供了一种

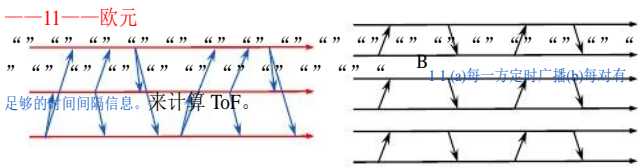


图 3:一个三方面的例子启发了群测距协议的设计。

图 3 中，A、B、C 三面轮流传输六条消息，即 A₁B₁C₁，一个 B₂C₂。每一个消息能被其他双方接收到是因为无线通信的广播性质。然后每消息生成三个时间戳，即，一次传送和两个接收时间戳，如图 3(a) 所示。我们可以请注意，每对都有两轮消息交换

如图 3(b) 所示。因此，有足够的时间戳来计算每对的 ToF。

每一方只传输两条消息，三对就可以进行远程通信，这比基于令牌环的扩展方式效率高得多。这个观察启发我们

设计了一种新的群测距协议。但是，动态密集的机器人和设备群必须面对数量大、机动性高、距离短的挑战。因此，必须回答以下问题。

Q1(简单而有效)如何设计测距消息，以便携带足够的时间戳?测距计算需要 6 个时间戳，每边 3 个。哪个时间戳应该包括在下一个范围消息广播?

Q2(简单而有效)放大的传输周期是否会影响精度?在 DS-TWR 中，消息是立即回复的，但在群播中，消息是周期性发送的。放大的应答时间会影响测距精度吗?

Q3(自适应)高机动性如何影响测距精度?如果测距信息不经常广播，但是机器人或设备移动得很快，测距精度会受到影响吗?

问 4(自适应)测距讯息应多久广播一次?当两个机器人或设备相距很远时

或者移动缓慢，测距频率可以低，但是当它们靠近或移动迅速时，那么它一定很高为了避免碰撞。但是标准是什么呢?

Q5(强壮的)如果消息丢失或测距怎么办使两侧频率不匹配广播消息数量不平衡?

Q6(可扩展)如何处理密集的邻居?在情况下邻居太多，一个测距消息不能吗携带所有的时间戳，一个好的选择必须作出。

Q7(兼容)群测距协议是否支持或兼容更高级别的协议，如 OLSR 路由协议和网络定位算法?

3 群测距协议的设计

A. 测距消息和主框架

回想一下图 3 中的三面例子，让我们关注一下 C 等消息₂。在接收 C₂，A 和 B 应该是能够分别计算出 to C 的 ToF。A 和 B 需要六个(不同的)时间戳来计算根据方程 (1) 和 (2) 所以 C₂ 必须包括传输时间戳 C₁ (仅在 C₁ 完成传输)，有些接收时间戳，例如 A₂ 和 B₂ 接收时间。现在我们来看看正式的定义。

定义 1(范围消息)。测距消息 X_i 是机器人或设备 x 广播的第 i 条消息定义为

消息 X_i X = (iTX_{i-1} RxM, v),

X_i 消息标识，例如，发送者和序列号;TX_{i-1}

即。最后发短信;RxM 是接收时间戳及其消息标识的集合，例如，RxM = {(A₂, RA₂), (B)₂, RB₂}};v 是 X 的当前速度。

在群体测距协议的主要框架中，有两个部分:，传输(TX)相关部分和

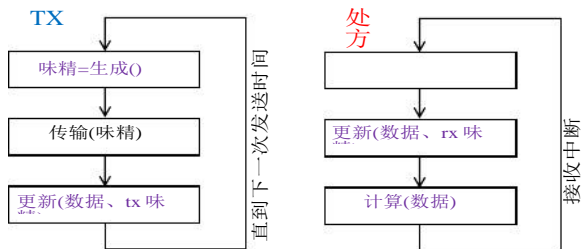


图 4 群测距协议的主要框架，分为发射(TX)和接收(RX)两部分。

接收(RX)相关部分，如图 4 所示。TX 部分每隔一段时间重复一次:过程 `Generate()`产生消息消息，然后由过程 `Transmit()`广播，然后调用 `Update()`来更新范围数据。每次接收到一个测距消息，RX 部分就会执行:过程 `Receive()`返回消息 `msg`，用于 `Update()`中更新测距数据，并由 `Compute()`根据更新的测距数据计算 `ToF`。

`Generate()`、`Update()`、`Compute()`和测距数据是设计蜂群测距协议的关键。

B. 在简单场景中生成消息和更新数据

让我们从一个简单的场景开始协议设计，在这个场景中，有许多机器人或设备，`a`、`B`、`C` 等，在很短的距离内。他们每个人传送的信息都能被其他所有人听到，他们以相同的速度广播信息。因此，在任意两个连续的消息传输之间，机器人或设备可以听到所有其他设备的消息。范围消息被设计为包括所有这些接收时间戳以及它们的消息标识。伪代码在过程 `generate()`中给出。

```

过程 1:生成(Ai, TAi-1, v)
1 RxM ← ∅;
2 对于每个接收到的消息 Yj 自从上次传送以来
3   RxM ← RxM ∪ (Yj, R.yj)
4 结束
5 返回消息(TAi-1 RxM, v)

```

让我们关注其中一对 `A` 和 `Y` 之间的消息交互，其中 `Y` 可以是 `B`、`C` 等，因为它们是相等的。前几个消息交互如图 5 所示，其中每个消息有一个发送和一个接收

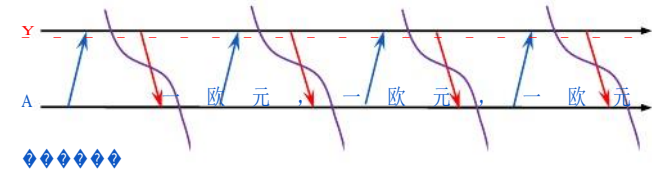


图 5·A 和 Y 之间的前几个消息交互(可以是 `B`、`C`，etc)。

时间戳。从 `A` 的角度来看，在接收到消息 `Y` 后 `i` 时间戳 `TAi`，`RAi`，`RYi` 根据生成信息可知。自从消息 `Y2` 携带它的最后一次传输时间 `TY1`，其最新接收时间 `RA2` 和其他接收时间戳不属于 `a2`，有 7 个时间戳(紫色第二行左侧)，前 6 个时间戳可以通过式(1)(2)计算 `ToF`。同理，收到 `A` 后可以计算新的 `ToF3` 和一个 4。

通常，只要接收到范围消息，就可以计算出一个新的 `ToF`，而计算只需要一小部分时间戳。因此，我们建议使用表 I 所示的范围数据结构来存储最新的 7 个时间戳。每次接收到一个范围消息，就计算 `ToF`

表一，测距数据结构为一个简单的例子。

Y 边 A 边	$R_p = RA_{i-1}$ $T_p =$	$T_r = TY_{j-1}$ $R_r =$	$R_f = RA_i$ $T_f =$	$R_e = RY_j$
---------	-----------------------------	-----------------------------	-------------------------	--------------

并对测距数据进行更新。假设一个 `i-1` 最后发送的消息是 `Y` 吗 `j-1` 是 `Y` 最后接收到的消息，那么在测距数据结构(表 I)中，`Tp`，`Rp` 和 `Rr` 最初以新一轮著称。下发消息 `A` 后 `iTr` 更新为 `TAi`。在接收 `Yj`，其接收时间戳 `RYj` 可以更新为 `Re`。同时，`Yj` 带来了两个 `Y` 侧时间戳 `TYj-1` 和类风湿性关节炎 `i`，即 `Tr` 和 `Rf` 可以更新。更新的正式步骤在 `UpdateS` 中。注意 `A` 可能有很多邻居，它就是这样

```

过程 2:更新(表, case, msg)
1 如果情况 == "tx"
2 为每个现有的表做
3 更新其 Tr。
4 结束
5 else if case == "rx" then
6 如果 msg 来自邻居 Y，那么
7 更新 Tr, Rf 和 Re 的表(AY);
8 结束
9 结束

```

要维护许多排列表，每个邻接一个。发送时间戳应该更新到每个表中，而接收时间戳应该更新到发送方的表中。

在时间戳更新之后，`ToF` 可以在 `ComputeS` 中计算。注意，第 3 行和第 4 行中的表更新是 `to`

```

步骤 3:计算(表(AY))
1 if Table(AY)不完整则返回∅;
2 利用表(AY)中的数据，通过公式(1)、(2)计算 ToF;
3 Rp ← Rf, Tp ← Tf, Rr ← Re;
4 Tr ← ∅, Rf ← ∅, Tf ← ∅, Re ← ∅;
5 返回 ToF

```

为下一轮 `ToF` 计算做准备。

至此，`Q1` 得到解答。我们现在看 `Q2`，即。，增大的传输周期会影响精度吗?在我们的群测距协议中，消息是按周期传输的，因此

应答延迟(接收时间戳和发送时间戳之间的时间间隔)大于 DS-TWR，即立即回复消息。但由式(3)可知，计算出的 ToF 的精度仅与两个器件晶体的偏移误差有关，与应答延迟无关。所以，问题得到了答案。

C. 自适应测距协议设计

一些重要的问题仍然是开放的:如何设置一个合适的测距信息传输周期的快速移动机器人或设备(Q3 和 Q4)?为简单起见，我们将测距周期 P 定义为两次连续测距消息传输之间的时间周期。很明显，测距周期短，测距频率高¹，这有助于快速移动的机器人和设备。但是，较短的测距周期可能会导致无线信道被过多的测距消息占用，从而导致消息冲突，降低同一信道上的数据通信吞吐量。当两个机器人或设备相距较远或移动较慢时，应设置较低的测距频率。

测距周期 P 应与速度和距离相适应。我们研究这种适应在高速但大范围周期的情况下。如图 6 所示，假设 A 为

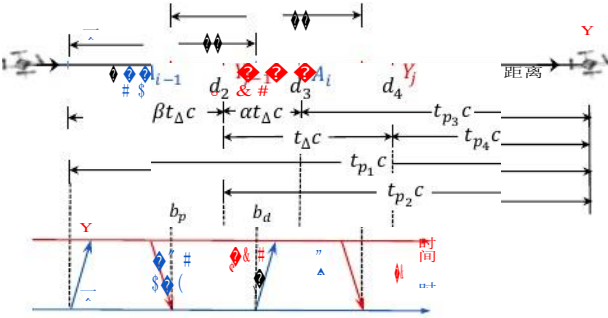


图 6:高速大测距周期场景。

以相对速度 v 向 Y 移动，设 P 为 A 和 Y 的测距周期，在此期间， A 移动副总统的距离。假设 A 传输 A_{i-1} ，得到 Y_{j-1} 、传输一个 i 和接收 Y_j 按顺序在维 $d_1 d_2 d_3$ 和 d_4 距离分别指向 Y 。所以 tp_1 ， tp_2 ， tp_3 和 tp_4 是持续时间为无线信号传输这些距离，分别。定义 t_Δ 是无线信号传播的时间 vP 距离，因此 $t_\Delta c = vP$ ，其中 c 是光速。在一个传输周期内，必须有一个测距消息收到从另一边。让接收时间分一次发送

周期变为 $\beta : \alpha$ 的比值，其中 $\alpha + \beta = 1$ 。图 6 详细介绍了符号及其关系。

因此，我们有 $tp_1 = tp_2 + \beta t_\Delta$ ， $tp_3 = tp_2 - \alpha t_\Delta$ ， $tp_4 = tp_2 - t_\Delta$ 。Therefore, $a_d = b_p + 2tp_2 + \beta t_\Delta$ and $b_d = a_p + 2tp_2 - \alpha t_\Delta$ 。因为处理时间远远大于

¹ 本文将测距和 ToF 计算互换使用。

信号传播时间， $a_d \approx b_p$ 因此 $b_p : a_p = \beta : \alpha$ 。结果，我们得到以下计算的 ToF

$$t_p^{computed} = \frac{a_d \times b_d - a_p \times b_p}{a_d + b_d + a_p + b_p} \\ = t_{p_2} + \frac{t_{p_2} t_\Delta (\beta - \alpha) + (\beta a_p - \alpha b_p) t_\Delta - \alpha \beta t_\Delta^2}{4t_{p_2} + 2a_p + 2b_p + t_\Delta (\beta - \alpha)} \approx t_{p_2},$$

注意最后的近似是因为 $a_p b_p \gg t_{p_2} t_\Delta$ 。因为计算发生在接收 Y 的时候 j 为实际的 ToF

$$t_p^{actual} = t_{p_4} = t_{p_2} - t_\Delta = t_p^{computed} - \frac{vP}{c}. \quad (4)$$

如果我们把误差限制在

$$\frac{|t_p^{actual} - t_p^{computed}|}{t_p^{actual}} \leq e_0,$$

我们 $t_\Delta \leq 1 - e_0 \cdot tp_2$ 因为 $vP = t_\Delta c$ 。我们有

$$P \leq \frac{e_0}{1 - e_0} \frac{d_2}{v}. \quad (5)$$

以公式(5)为准则，在给定速度 v 和计算距离 d 的情况下确定测距周期 P_2 。由这个方程可知，距离越小，测距周期越短;同时，速度越快，测距周期越短，解决了 Q3 和 Q4 的问题。

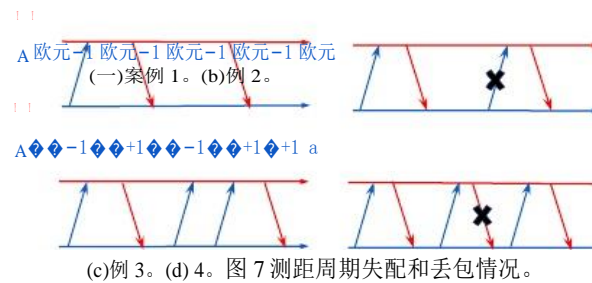
D. 处理测距周期不匹配、丢包问题

我们知道，对于机器人或设备 A ，它的测距周期可能因邻居而异，根据 Eq.(5)。因此我们选择最小值作为测距一个时期，

$$P_A = \min_{Y \in \text{neighbors}} \left\{ \frac{e_0}{1 - e_0} \frac{d_{AY}}{v_{AY}} \right\}, \quad (6)$$

在维 AY 和 v_{AY} 是距离和相对速度吗分别。

一个直接的问题随之而来:如果测距对应物有不同的测距周期或如果数据包丢失怎么办在传输 (Q5)? 这两个问题都与不平衡有关消息交换，我们总结为四种情况图 7 所示。在情况 1 中， A 接收到的比它发送的多。以防



2、 A 的一条消息丢失。在情况 3 中， A 发送的比接收的多。在情况 4 中，来自 Y 的一条消息丢失。我们的蜂群测距协议必须能处理所有案件，我们要逐一调查。

在情形 1 和情形 2 中，都是消息 Y_j 将不会携带 A 的任何接收时间戳，因为从 Y_{j-1} 传播。因此，在测距表中， R_f 将在接收更新后不存在，如表 2 中的灰色单元格。我们的解决方案丢弃了 A 的时间戳 $j-1$ 和 X_i (如果存在)，如表二下半部分所示，以便下一轮的范围可以继续。

表二。处理案例 1 和案例 2 的失配和丢失。

Y 侧 $R_p = R_{Ai} - 1 \quad T_r = T_{Yj} - 1 \quad R_f =$			
$T_p = T_{Ai} - 1 \quad R_r = R_{Yj} - 1 \quad T_f = T_{Ai} \quad R_e = R_{Yj}$			
一个 Y 的 一个			or $T_f =$
	$R_p = R_{Ai} - 1 \quad T_r =$	$R_f =$	
	$T_p = T_{Ai} - 1 \quad R_r = R_{Yj} \quad R_{Yj}$	$T_f =$	$R_e =$

情况 3 很容易处理。我们总是更新/覆盖传输时间戳 T_f 当消息被传输时，测距表的。

在情况 4 中，从消息 Y 开始 j lost，在 A_{i+1} 时，发送时间戳 T_f 被 $T_{A_{i+1}}$ 。当 A 接收到消息 Y 时 $j+1$ ，则存在

表 3。处理失配和丢失

Y 的	$R_p = R_{A_{i-1}}$	$T_r = T_{Y_j}$	$R_f = R_{A_{i+1}}$
	$T_p = T_{A_{i-1}}$	$R_r =$ DV_{i-1}	$T_f = T_{A_{i+1}} \quad R_e = R_{Y_{j+1}}$
Y 的	$R_p = R_{A_{i+1}}$	$T_r =$	$R_f =$
	$T_p = T_{A_{i+1}}$	$R_r = R_{Y_{j+1}}$	$T_f =$
$R_e =$			

不匹配。在 Y 的一边， Y_j 是最近传输的消息，那么时间戳 T_{Y_j} 在消息 Y 中携带 $j+1$ 。这个时间戳被更新到测距表中，这样 $T_r = T_{A_i}$ 。但是在 A 这边，因为 Y_{j-1} 是最后接收到的消息，所以测距表保持 $R_r = R_{Y_{j-1}}$ 。然后，消息索引不匹配，如表 III 中的黄色单元格所示。我们的解决方案丢弃了 A 的时间戳 $i-1$ Y_{j-1} 和 Y_j ，如表三下半部分所示，以便下一轮的测距可以继续进行。

我们通过 UpdateM 总结这些步骤。

过程 4:UpdateM(表, case, msg)

```
1 更新(表、案例、味精);
2 if case== ' rx '和 msg 是来自邻居 Y 然后
3 如果  $R_f$  则==表(A Y)中的 $\emptyset$ 
4   |  $R_r \leftarrow R_e, T_r \leftarrow \emptyset, T_f \leftarrow \emptyset, R_e \leftarrow \emptyset$  for table(A Y);
5   else if  $index(T_r) \neq index(R_r)$  in table(A Y) then
6  $r_p \leftarrow R_f \quad T_p \leftarrow T_f, R_r \leftarrow R_e$  对于表(A Y);
7    $T_r \leftarrow \emptyset, R_f \leftarrow \emptyset, T_f \leftarrow \emptyset, R_e \leftarrow \emptyset$  for table(A Y);
8 日结束
9 日结束
束
```

请注意，如果表 AY 不完整，Procedure computing 将不会根据第一行的 if 语句计算 ToF。至此，问题 5 得到了解答。

E. 处理密集和动态邻居

虽然上述设计的测距协议已经可以很好地适用于任何小型群组，但当群组数量增加时，它会遇到可伸缩性问题。让我们感受

难度来自下面的例子。假设 A、B 和 C 是静止的，相隔很远 (但在传输范围内)，而它们各自都有一个非常近且快速移动的邻居。因此，根据式(6)，A、B、C 的测距周期都很短，因此，静止且距离较远的 A、B、C 之间的测距频率非常高，这是不必要的。当人数和流动性增加时，这种情况将变得更加严重。最终，测距消息容量将不足以携带每个邻居的所有时间戳。回答问题 6 是至关重要的:如何选择时间戳以适应消息容量，以便可伸缩到密集的邻居?

我们的解决方案的高层基本思想体现在三个方面。(1).对于任何机器人或设备，我们允许它对不同的邻居有不同的测距周期，而不是对所有邻居设置一个很小的周期。因此，并不是所有邻居的时间戳都需要携带在每一个范围消息中，例如，接收时间戳到一个远的和不动的邻居需要更少的次数。(2).由于不同的邻居具有不同的测距周期，因此测距消息中携带的邻居时间戳的频率也不同。为了让每个邻居在要求的时间段内轮流携带时间戳，核心思想是为每个邻居维护下一个(预期的)交付时间。测距消息容量是根据所有邻居的下一个(预期)交付时间的时间顺序共享的。(3).当一个邻居在一段时间内没有被听到，我们将其设置为过期，不再将其视为邻居。

我们从改进的测距表数据结构开始详细设计，如 table 所示。四、有三个

表四改进的测距表数据结构。

Y 一边 R_p T_p 一面	T_r R_r	射 特 性	P tn 再保险
------------------------	----------------	-------------	-------------

与表一比较的新符号，即、 P 、 t_n 、 t_s ，分别表示 A 和 Y 的最新的范围周期，邻居 Y 的下一个(预期的)交付时间，邻居 Y 的过期时间。注意，对于每个邻居 Y, A 维护一个范围表(A Y)。

在计算 ToF 之后，就会更新测距周期，如 Compute 中所示。

步骤 5:计算(表(A Y))

```
1 ToF =计算(表(A Y));
2 对表(A Y)用式(5)更新 P;
3. 返回 ToF
```

下(预计)交货时间 t_n 一旦邻居 Y 的消息被传递，即 t_n 应该比当前时间晚一个时间段。而过期时间为 t_s 一旦收到来自邻居 Y 的消息，即 t_s 过期时间应该晚于当前时间。详细的伪代码可以在 Update 中找到。

改进并适当维护了测距表之后，我们现在将重点放在如何生成测距消息上

过程 6:更新(表, case, msg)

```
1 update(表、案例、味精);
2 if case== ' tx '然后
3 表示每一个接收时间戳在 do msg 中携带的 Y
4  $t_n \leftarrow t_{current} + P$  表示表(AY);
5 日结束
6 else if case== ' rx ' and msg is from neighbor Y then
7  $t_s \leftarrow t_{current} + T_{expiration}$  for table(AY);
8 日结束
```

根据下次(预计)交货时间 t_n 过期时间 t_s 从所有的排列的桌子。假设测距消息最多携带 m 个接收时间戳。如何选择这些时间戳是产生范围消息的关键。选择原则很简单:对于还没有过期的邻居,我们按照下一个(预期的)交货时间排序,使它们按照时间顺序排列,并选择 m 个最紧急的邻居。生成并发送一个携带这些邻居的接收时间戳的范围消息。然后,根据对应的测距周期,通过 Update 更新 m 个下一个交货时间。这个过程重复执行,在下一个(预期的)交付时间最近的时间生成下一个范围消息。Procedure Generate 显示详细信息。

步骤 7:生成()

```
1 为每个现有表(AY)做
2 如果  $t_{current} > t_s$  然后删除表(AY);
3 结束
按 t 对所有表排序n 以升序排序;
5  $rxm \leftarrow \emptyset$ ;
6 表示每一个邻接 Y 从 top m 表
7  $RxM \leftarrow RxM \cup (Y, R_Y)$ 
8 日结束
9 返回消息( $X_i T_{X_i-1} RxM, v$ )
```

在上述设计的帮助下,我们的协议现在处理密集邻居,这回答了 Q6。

支持更高级别的协议和算法

对于任何成功的群体应用,例如,一组室内无人机搜索给定目标,一群小型机器人探索未知的室内环境并绘制地图,一群有腿的机器狗在森林深处作战,两项基本技术是 1)低延迟自组织网络和 2)实时网络定位。结构如图 8 所示。

群测距同时支持自组网和网络定位。(1)在自组网中,一些高级路由协议需要维护一个活动邻居列表。我们的蜂群测距协议已经维护了这样一个列表(在过期时间 t_s),可以直接使用。此外,一些路由协议还会周期性地广播探测消息,例如 OLSR[13]中的 HELLO 消息,它可以与我们的测距消息相结合。(2)在



图 8:群测距支持自组织网络和网络定位,这是任何群应用的基础。

网络定位(Network localization),一个由节点组成的网络用于帮助本地化其成员,特别是使用彼此之间的距离信息[14]。我们的群测距协议为更高层次的定位算法提供了基本的距离信息。至此,第 7 个问题得到解答。

四、实现

提出的蜂群测距协议在 crazyfly 2.1 无人机上实现,该无人机由 STM32 微控制器驱动,内存 192KB,机载超宽带无线收发芯片 DW1000。所有无人机都使用同一个超宽带频道进行广播。我们将数据速率设置为 6.8 Mbps,并使用 128bit 的前置码。在我们的实验中,无人机的数据是通过一台笔记本电脑通过工作在 2.4GHz 频段的通讯设备 Crazyradio PA 收集的。设有流动甲板,用于自动飞行控制和高度测量。

在一些涉及距离地面真值的实验中,使用了 HTC Lighthouse 系统,该系统是一种光学室内定位系统,提供毫米级的位置数据作为地面真值。需要在无人机上安装一个额外的灯塔甲板来获取位置数据。

诉实验

本节通过实验从多个方面对群体测距协议进行评估。为了最小化测距消息碰撞概率,我们将周期 P 随机分成两部分,即: $p + w$ 。底 p 是给定的固定实数 $p < p$,随机 w 是服从均匀分布的随机数 $w \in U(0, w)$,我们称 w 为随机窗口大小。所以我们 have $\bar{P} = E(p + w) = p + W/2$ 。

A. 测距周期和精度

为了评估测距周期对测距精度的影响,我们将两架无人机停在视线清晰、距离任意一面墙 1 米以上的独立椅子上。我们禁用了范围周期适应,并设置 $W = 0$ 。研究了步长为 25ms 的不同测距周期 $P \sim 100ms$,步长为 0.5m 的 1.0 ~ 3.0m 不同距离 d 下的测距精度。在每个设置下,测距结果遵循一定的随机分布。其概率密度函数如图 9 所示。可以看出,在不同距离下,测距周期 P 不影响测距精度。

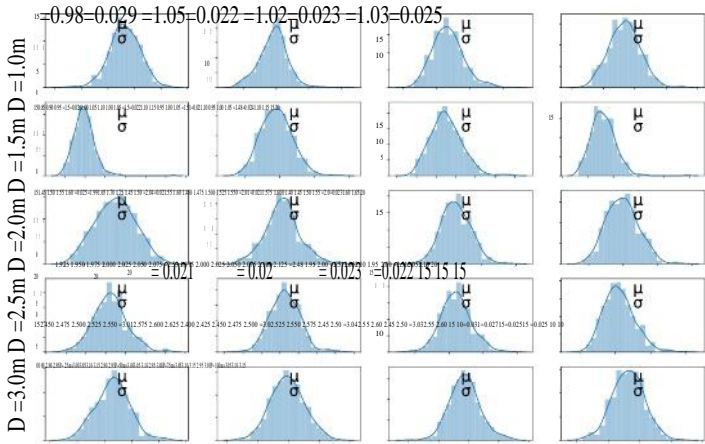


图 9:测距周期对测距精度的影响。

B. 消息接收比和测距比

本小节利用 4 架无人机组成一个小蜂群来评估信息接收比和测距比。A、B、C、D 4 架无人机静置近距离，保证良好的无线信道。我们为所有无人机设置了平均射程周期 $P=50ms$ ($P= 30ms, W=40ms$)。每架无人机总共传送 6000 条测距信息。

我们定义消息接收率为接收到的消息数与发送的消息数之比，测距率为测距计算成功的次数与发送的消息数之比。表 V 为 a 无人机记录的次数和比率，可以看出，平均消息接收率在 90% 左右，平均测距率在 60% 左右。测距比低的原因是测距周期是随机的，这导致了周期随机错配，即：，图 7 中的 Case 1 和 Case 3，从而减小了测距比。

表五、消息接收比和测距比。

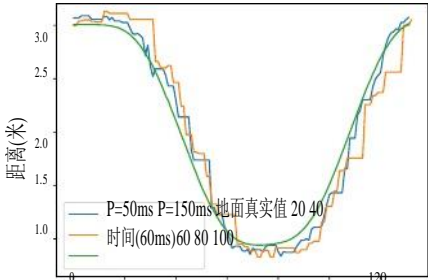
包括对	接收数	接待比率	包括数	不等比例
AB	5451	90.85%	3563	59.38%
交流	5371	89.52%	3540	59.00%
广告	5348	90.53%	3701	61.68%

C. 速度和测距精度

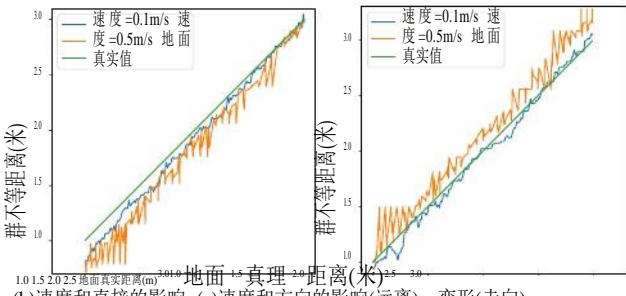
本节评估速度 v 对测距精度的影响。HTC Lighthouse 室内定位系统用于提供距离地面真实度。

由式(4)可知，测距周期 P 和无人机速度 v 都会影响测距精度。因此，我们分别研究它们。

首先，由两架无人机 A 和 B 评估测距周期的影响。无人机 A 在固定位置盘旋，无人机 B 从 3 米外的位置开始飞向它，两者高度相同，均为 100 厘米。B 在距离 A 1 米的地方停下来，然后飞回起始位置。B 的飞行速度设为 $v=0.5m/s$ 。我们重复两次



(a)测距周期的影响



(b)速度和直接的影响- (c)速度和方向的影响(远离)。变形(走向)。

图 10:不同速度和测距周期下的测距精度。

测距周期分别为 50ms 和 150ms。测距结果如图 10(a)所示。很明显，当 P 变大时，准确率下降。我们还观察到，测距结果与地面真实值相比有延迟，且 P 越大，延迟越大，这与我们在 Eq.(4)中的理论分析是一致的。

其次，在相同的设置下考察了飞行速度的影响。不同之处在于，我们现在将射程固定为 $P=50ms$ ($P= 45ms, W=10ms$)，但为无人机 B 设置了两种飞行速度，0.1m/s 和 0.5m/s。我们将两个飞行方向分开，分别朝向和远离，如图 10(b)(c)所示。飞行速度越快，采集的采样测距数据越少，因为飞行时间越短。为了获得足够的样本点，设置无人机 B 以 0.5m/s 的速度飞行 5 次。从图 10(b)(c)可以看出， $v=0.1m/s$ 时的测距结果比 0.5m/s 时更稳定。此外，(b)(c)也揭示了(a)的相同结论，测距有延迟。

D. 不匹配测距周期的性能

本次实验共有 4 架无人机，分别为 A、B、C、d，我们设置平均测距周期 $P= 30ms, 40ms$ ，

A、B、C、D 分别为 50ms 和 60ms，其中所有无人机的随机窗口大小设为 $W = 40ms$ 。图 11 为无人机 A 在 200s 持续时间内记录的测距计数。可以看出，我们的蜂群测距协议很好地处理了测距周期不匹配的问题，即邻居的测距周期越短，测距次数越多。在不受干扰的情况下，将该对移出蜂群，距离计数增加。

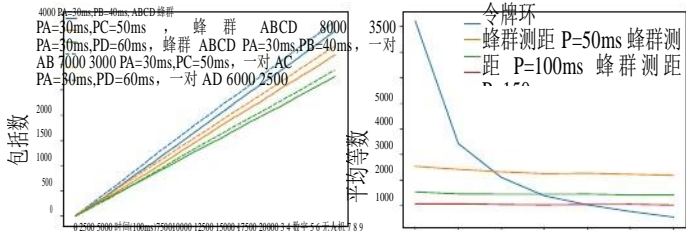


图 12:与测距匹配的测距周期的比较。基于令牌环。



图 13:疯狂飞行的无人机紧凑飞行时避免碰撞。

E. 与基于令牌环的测距比较

我们将我们的群测距协议与令牌环进行比较基于测距算法。我们改变平均测距周期 P ，从 50ms 到 150ms ，同时保持随机窗口尺寸 $W = 80\text{ms}$ 。我们改变无人机的数量，从 3 架到 9。图 12 为 A 无人机记录的距离计数 200 年代。

图 12 中的任何一点都是成功测距计数的平均值无人机 A 和它的邻居我们可以看到 **perfor-**

令牌环算法的 **Mance** 随着参与者的增长急剧下降，因为它是顺序执行的。而群测距的性能略有下降，因为随着数量的增加，消息冲突的概率会增加。当无人机数量超过 5 架时，我们的群测距协议性能优于令牌环算法。当 9 架无人机参与时，根据我们的协议，一架无人机成功与另一架无人机进行航程的平均数量约为令牌环算法的 5 倍。

f. 演示实验

为了测试实时测距精度，我们进行了避碰实验。可以在 <https://www.youtube.com/watch?v=hJ8yo2ReBdA> 上找到演示视频。

在这个实验中，8 架 crazyfly 无人机在一个小于 $3\text{m} \times 3\text{m}$ 的紧凑区域中，在 70cm 的高度悬停。第九架“疯狂苍蝇”无人机被手动控制飞进了这个区域。我们设置平均测距周期 $P=100\text{ms}$ ($P=60\text{ms}$, $W=80\text{ms}$) 和最大接收容量 $m=8$ 。如演示视频和图 13 所示，通过群体测距协议，无人机通过测距距离检测到飞来的无人机，当距离小于阈值 30cm 时，降低高度以避免碰撞。

六. 相关工作

超宽带被广泛用于室内定位[15], [16], [17], [18]。Tiemann 等[15]和 Corbalan 等[16]

设计了基于 TDOA 的测距算法时间同步。Xu 等人。[17] 引信视觉、惯性和航空群定位的超宽带信息。Cao 等只使用一个超宽带锚(基站)来定位机器人。

近年来，并行测距已成为一种新兴的趋势在超宽带研究界 [19], [20], [21], [22]。Corbalan 等人 [19], [20] 提出使用信道脉冲反应要区分个别到达的时间对同一请求的响应重叠，这被调用并发等。Heydariaan 等人随后扩展了思想要反射有弹性。而 Stocker 等人 [21] 扩展它支持无限数量的标签。

也有一些工作集中在大范围的超宽带测距数字 [16], [21] 或高机动性 [23]。Corbalan 等 *al.*[16] 和 Stocker 等人的[21] 聚焦定位无数的标签通过 TDOA 或并行测距。他们的工作是奉献的对于锚 - 标签模型，不适用于群体场景。Risset 等人研究了超宽带测距问题快速移动，只有两个超宽带标签。

超宽带也被用于 [24]、[25] 的数据通信。Mohammadmoradi 等人 [24] 提议同时测距以及超宽带网络中的通信。Vecchia 等研究了超宽带的并发传输问题。

然而，上述相关工作都没有集中在一个范围上针对动态密集蜂群的协议设计。

7 结论

提出了一种超宽带群测距协议

专为动态和密集的机器人或设备群集而设计。其基本思想是设计周期性广播的测距信息。时间戳由该消息携带，以便计算距离。我们的群测距协议简单而高效，因为只有一种类型的消息;由于测距周期与测距对的速度和距离相适应，并对丢包问题进行了适当的处理，因此具有自适应和鲁棒性;它具有可扩展性和兼容性，因为设计了一个旋转方案来处理密集邻居，并支持更高级别的网络和定位协议和算法。最后，该协议在 crazyfly 无人机上实现，该无人机由 STM32 微控制器驱动，只有 192KB 内存。得益于蜂群测距协议，9 架“疯狂苍蝇”无人机在狭小空间飞行时可以自动避免碰撞。

鸣谢

基金资助:国家重点研发计划项目 2018AAA0101200、2018YFB2100300，国家自然科学基金项目 62072101、61632008、61972086、62072102、61872079、62072099，江苏省网络与信息安全重点实验室项目 BM2003201，教育部计算机网络与信息集成重点实验室项目 93K- 9，软件创新技术与产业化协同创新中心。

参考文献

- [1] “疯狂苍蝇 2.1 -比特热商店”，<https://www.bitcraze.io/2019/02/the-crazyfly-2-1-is-here/>，于 2020 年 7 月 30 日访问。
- [2] “RoboMaster EP - DJI”<https://www.dji.com/robomaster-ep>，于 2020 年 7 月 30 日登陆。
- [3] “波士顿动力公司推出 SPOT 机器人的商业销售”，<https://www.bostondynamics.com/press-release-spot-commercial->发射，于 2020 年 7 月 30 日接入。
- 余杰，金丹，金凯福。陈强，王强，袁凯，张磊，“生物流体中微机器人群的主动生成与磁驱动”，《自然通讯》，第 10 卷，第 1 期。1, 2019 年 1 - 12 页。
- 杨洋和贝文，“胶体蜂群捕获和运输货物”，《科学进展》，第 6 卷，第 6 期。4, p. eaay7679, 2020。
- [6] K. McGuire, C. De Wagter, K. Tuyls, H. Kappen and G. De Croon, “一群微型飞行机器人探索未知环境的最小导航解决方案”，《科学机器人》，vol. 4, no. 35, p. eaaw9710, 2019。
- [7] Z. Zhakypov, K. Mori, K. Hosoda and J. Paik, “设计最小和可扩展的昆虫启发的多运动微型机器人”，《自然》，第 571 期。7765, pp. 381-386, 2019 年。
- [8] Decawave, DW1000 用户手册。Decawave 有限公司,2017 年版。
- [9] “iPhone11 -苹果”，<https://www.apple.com/iphone-11/specs>，访问 2020 年 8 月 11 日。
- 《局域网和城域网的 IEEE 标准-低速率无线个人区域网(lr - wpan)》，IEEE Std. 802.15.4- 2011, pp. i-294, 2011。
- [11] D. Neiryneck, E. Luk 和 M. McLaughlin, “一种替代的双面双向测距方法”，Proc. 2016 13th Work. 定位, Navig. Commun. WPNC 2016, pp. 16-19, 2017。
- [12] B. Broecker, K. Tuyls and J. Butterworth, “基于距离的口袋无人机多机器人协调”，2018 年 IEEE Int. 相依的机器人。奥特曼。IEEE, 2018 年 5 月，第 6389-6394 页。(在线)。可用:<https://ieeexplore.ieee.org/document/8461176/>
- 《优化的链路状态路由协议(OLSR)》，《互联网请求评论》，RFC 编辑器，RFC 362610, 2003。(在线)。可用:<https://tools.ietf.org/html/rfc3626>
- 戴伟，沈勇，温明志，“基于网络的高效定位与导航策略研究”，《计算机应用与发展》，第 1 卷第 1 期。7, pp. 1224-1254, 2018。
- [15] J. Tiemann, Y. Elmasry, L. Koring 和 C. Wietfeld, “ATLAS FaST:用于可靠的超宽带本地化的快速和简单的预定 TDOA”，载于 Proc. ICRA, 2019, 第 2554-2560 页。
- [16] P. Corbal'an, G. P. Picco 和 S. Palipana, “合唱:超宽带并发传输用于对无数目标进行类似 gps 的被动定位”，载于 Proc. IPSN, 2019, 第 133-144 页。
- 王磊，徐慧华，张勇，邱凯，“基于离散视觉-惯性-超宽带融合的航空蜂群相对状态估计”，中国航空航天大学报，2020,pp. 8776-8782。
- [18] Y. Cao, C. Yang, R. Li, a. Knoll, 和 G. Beltrame, “用单一超宽带锚精确位置跟踪”，在 Proc. ICRA, 2020。
- [19] P. Corbal'an 和 G. P. Picco, “超宽带无线电的并发范围:实验证据、挑战和机会。”在 EWSN, 2018 年,55 - 66 页。
- [20] P. Corbal'an 和 G. P. Picco, “超宽带并发测距”，ACM Trans. 参议员网络,2020 年。(在线)。可用:<http://arxiv.org/abs/2004.06324>
- [21] M. Stocker, B. Großwindhager, C. A. Boano, 和 K. R'omer, “Snaploc:一种基于超快超宽带的无限标签室内定位系统”，见 Proc. IPSN, 2019, 第 348-349 页。
- [22] M. Heydariaan, H. Mohammadmoradi, 和 O. Gnawali, “R3:用超宽带无线电反射弹性并行测距”，Proc. - 15 Annu. Int. 相依 Distrib. 第一版。参议员系统。dcos2019, 第 1-8 页, 2019。
- [23] T. Risset, C. Goursaud, X. Brun, K. Marquet 和 F. Meyer, “快速运动的超宽带范围”，在 IPIN 2018 -第九届国际会议上。相依室内的位置。室内 Navig. IEEE 2018。
- [24] H. Mohammadmoradi, M. Heydariaan 和 O. Gnawali, “SRAC: UWB 网络中的同步测距和通信”，Proc. -第 15 Annu. Int. 相依 Distrib. 第一版。参议员系统。dcos2019, 2019 年 9-16 页。
- [25] D. Vecchia, P. Corbalan, T. Istomin 和 G. P. Picco, “玩火:探索超宽带无线电的并发传输”，在 Proc. SECON, 2019 年。