

# 移动无线传感网络的分布式协作定位的研究

贺伟<sup>1</sup> 梁潘<sup>2</sup>

<sup>1</sup>(阿坝师范学院电子信息与自动化学院 四川 汶川 623002)

<sup>2</sup>(成都航空职业技术学院机电工程学院 四川 成都 610100)

**摘要** 针对移动无线传感网络,并基于到达时间 TOA (Time-of-Arrival) 的测距模型,分析利用传感节点的移动信息进行定位问题,提出基于二阶锥规划的分布式定位算法。推导最大似然 ML (Maximum Likelihood) 定位估计表达式;考虑到基于 ML 的协作定位的非凸性,选用二阶锥规划 SOCP (Second-order Cone Programming) 松弛技术求解。为了降低计算成本,采用分布式策略实施 SOCP 算法。实验数据表明,该算法减少了均方根误差,提高了定位精度。

**关键词** 移动无线传感网络 定位 到达时间 最大似然估计 二阶锥规划

中图分类号 TP393 文献标识码 A DOI:10.3969/j.issn.1000-386x.2019.04.025

## DISTRIBUTED COOPERATIVE LOCALIZATION FOR MOBILE WIRELESS SENSOR NETWORK

He Wei<sup>1</sup> Liang Pan<sup>2</sup>

<sup>1</sup>(College of Electric Information and Automation, Aba Teachers University, Wenchuan 623002, Sichuan, China)

<sup>2</sup>(Department of Aviation Manufacturing Engineering, Chengdu Aeronautic Polytechnic, Chengdu 610100, Sichuan, China)

**Abstract** For mobile wireless sensor network, we studied the problem of localization using mobile information of sensor nodes based on the time-of-Arrival (TOA) model. Then we proposed distributed localization algorithm based on second-order cone programming. The expression of maximum likelihood (ML) location estimation was derived. Considering the non-convexity of ML-based cooperative localization, we used the second-order cone programming (SOCP) relaxation technique to solve the problem. In order to reduce the computing cost, a distributed strategy was adopted to implement the SOCP algorithm. The experimental data show that the algorithm reduces the root mean square error and improves the localization accuracy.

**Keywords** Mobile wireless sensor network Localization Time of arrival Maximum likelihood estimation Second-order cone programming

## 0 引言

微型的、能量受限的具有数据感知能力的传感节点组成的无线传感网络 WSNs (Wireless Sensor Networks)<sup>[1-2]</sup>已在医疗、环境监测、军事等领域得到广泛应用。这些应用均以节点有效的收集数据为前提,并且节点所感知的数据需配备准确的位置才具有价值,换言之,无准确位置的感测数据是无价值的。因此,节

点定位已成为 WSNs 的研究焦点<sup>[3-4]</sup>。

考虑到 WSNs 的节点能量、尺寸和成本限制,常通过测量传感节点与锚节点间的测距信息估计传感节点位置,其中锚节点是指已知位置的传感节点<sup>[3]</sup>。通常,通过测量无线信号的物理参数获取测距数据。在众多的测量技术中,基于到达时间 TOA<sup>[4]</sup>受到广泛关注。

目前,尽管研究人员提出不同的定位算法,但是这些定位算法是针对静态传感网络,即传感节点和锚节点都是静态的。然而,在一些实际场景中,如水下传感

网络,小型的跟踪移动网络、车联网。这些场景中节点是非静态的。

现只有少数文献关注了移动锚节点协助定位问题<sup>[5-6]</sup>。文献[6]针对移动传感节点环境,研究了基于最大似然 ML(Maximum Likelihood)估计的节点定位问题。但是,文献[6]所提出的定位方案属非协作式<sup>[7]</sup>。在非协作式定位中,每个传感节点需要不断与锚节点通信。然而,如果传感节点不能直接与足够多的锚节点通信时,传感节点就无法定位。例如,在二维空间中,每个传感节点至少需要与三个锚节点连接,才能定位,否则传感节点将无法获取自己的位置。为了解决此问题,协作定位<sup>[7]</sup>得到关注。在协作定位算法中,每个节点与其通信范围内的节点进行通信,获取更多测量信息,进而提高定位精度。

为此,基于 TOA 测量,提出二阶锥规划的分布式定位算 SOCP-DL(Second-order Cone Programming-based distributed localization)算法。SOCP-DL 算法将复杂的 ML 问题分解为分布式子问题,然后再由传感节点局部求解。因此,提出的 SOCP-DL 算法适用于大型 WSNs 网络。

## 1 测距模型

假定 WSNs 内有  $m_s$  个传感节点和  $m_a$  个移动锚节点构成,这些移动锚节点能独自移动,并且锚节点位置已知。类似文献[6],假定每个移动锚节点能测量它的速度矢量,包括大小和方向,并将这些速度矢量用于定位阶段。

用  $\mathbf{s}_i^{(n)}$  矢量表示在观察时刻  $n$ ,第  $i$  个节点的实际坐标值,且  $i=1,2,\dots,m_s+m_a,n=1,2,\dots,N$ 。其中  $N$  是总的观察时间数。不失一般性,假定  $s_1^{(n)},\dots,s_{m_s}^{(n)}$  表示  $m_s$  个传感节点的位置,而  $s_{m_s+1}^{(n)},\dots,s_{m_s+m_a}^{(n)}$  表示移动锚节点位置。此外,用  $\mathbf{v}_i^{(n)}$  表示第  $i$  个节点在时刻  $n-1$  至  $n$  间的真实速度矢量,其中, $i=1,2,\dots,m_s,n=2,3,\dots,N$ 。假定节点在时刻  $n-1$  至  $n$  间,移动速度不变。

此外,引用式(1)的感测模型协作速度测量,进而完成定位<sup>[7]</sup>:

$$\mathbf{s}_i^{(n)} = \mathbf{s}_i^{(n-1)} + T_s (\mathbf{v}_i^{(n)} + \mathbf{e}_i^{(n)}) \quad (1)$$

式中: $T_s$ 表示抽样间隔; $\mathbf{e}_i^{(n)}$ 表示速度测量误差,其服从零均值的高斯白噪声,且方差为  $\sigma_e^2 \mathbf{I}_2$ 。

在时刻  $n$ ,第  $i$  个节点与第  $j$  个节点间的欧式距离为  $E_{ij}^{(n)}$ :

$$E_{ij}^{(n)} = \|\mathbf{s}_i^{(n)} - \mathbf{s}_j^{(n)}\| \quad (2)$$

只有当节点  $i,j$  在彼此的通信范围内,即  $\|\mathbf{s}_i^{(n)} - \mathbf{s}_j^{(n)}\| \leq R$ ,它们才能测量欧式距离。此外,用  $A$  表示链路集,其不包含锚节点与锚节点间的链路,定义如下:

$$A = \{(i,j,n) \mid \|\mathbf{s}_i^{(n)} - \mathbf{s}_j^{(n)}\| \leq R\} \quad (3)$$

其中, $1 \leq i \leq m_s, 1 \leq j \leq m_s + m_a$ ,且  $i \neq j, 1 \leq n \leq N$ 。

本文利用 TOA 算法测距<sup>[4]</sup>。在时刻  $n$ ,第  $i$  个节点与第  $j$  个节点间的测距值  $d_{ij}^{(n)}$ :

$$d_{ij}^{(n)} = \|\mathbf{s}_i^{(n)} - \mathbf{s}_j^{(n)}\| + \lambda_{ij}^{(n)} \quad \forall (i,j,n) \in A \quad (4)$$

式中: $\lambda_{ij}^{(n)}$ 表示 TOA 测距误差,其是零均值的白噪声,且方差  $\sigma_\lambda^2$ 。

## 2 算法设计

本文算法先利用测距值和测速值作为 ML 估计的观察数据,再建立 ML 估计表达式。考虑到 ML 估计为非凸、NP-hard 问题,利用 SOCP 松弛技术求解。然而,为了降低算法的复杂度,将集中式求解转化为分布式求解,整个 SOCP-DL 算法框架如图 1 所示。

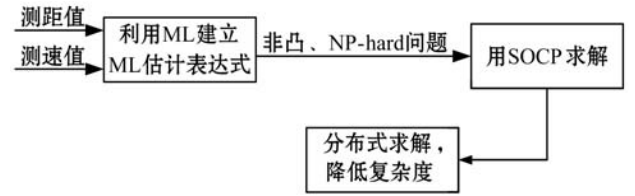


图1 本文算法框架

### 2.1 基于 ML 定位

第  $i$  个节点在时刻  $n$  的速度矢量  $\mathbf{v}_i^n$  的定义如下:

$$\mathbf{v}_i^n = \frac{1}{T_s} (\mathbf{s}_i^{(n)} - \mathbf{s}_i^{(n-1)}) - \mathbf{e}_i^{(n)} \quad (5)$$

依据式(5)可建立  $m_s$  个节点的速度测量值集  $V$ ,  $V = \{\mathbf{v}_i^{(n)} \mid i=1,2,\dots,m_s\}$ 。接下来,推导传感节点的位置的估计值,即  $S = \{\mathbf{s}_i^{(n)} \mid i=1,2,\dots,m_s,n=1,2,\dots,N\}$ 。

基于 TOA 场景,定义  $D$  集,其内元素表示所有的测距值,如式(6)所示:

$$D = \{d_{ij}^{(n)} = \|\mathbf{s}_i^{(n)} - \mathbf{s}_j^{(n)}\| + \lambda_{ij}^{(n)}\} \quad \forall (i,j,n) \in A \quad (6)$$

依据速度测量值  $V$  和测距值  $D$ ,再最大化条件概率分布函数  $f(D,V|S)$ ,便可获取 ML 估计。具体而言,由于速度和距离测量误差相互独立,可得  $f(D,V|S) = f(D|S)f(V|S)$ ,其定义如式(7)所示:

$$f(D,V|S) = \prod_{(i,j,n) \in A} \frac{1}{\sqrt{2\pi\sigma_\lambda^2}} \exp\left(-\frac{(\|\mathbf{s}_i^{(n)} - \mathbf{s}_j^{(n)}\| - d_{ij}^{(n)})^2}{2\sigma_\lambda^2}\right) \times \prod_{(i,n) \in V} \frac{1}{\sqrt{2\pi\sigma_e^2}} \exp\left(-\frac{\left\|\frac{1}{T_s} (\|\mathbf{s}_i^{(n)} - \mathbf{s}_i^{(n-1)}\|) - v_i^{(n)}\right\|^2}{2\sigma_e^2}\right) \quad (7)$$

再对式(7)两边取对数,便可得 ML 估计值:

$$\min_S \left[ \sum_{(i,j,n) \in A} \frac{1}{\sigma_\lambda^2} (\|s_i^{(n)} - s_i^{(n-1)}\| - d_{ij}^{(n)})^2 + \sum_{(i,n) \in V} \frac{1}{\sigma_e^2} \left\| \frac{1}{T_s} (s_i^{(n)} - s_i^{(n-1)}) - v_i^{(n)} \right\|^2 \right] \quad (8)$$

式(8)的第一项是测距值的似然估计,第二项为测速值的似然估计。测距值和测速值的不准确,均会导致位置估计值的误差。对比分析式(8)的第一项和第二项不难发现,当  $\sigma_e^2/\sigma_\lambda^2$  很小时,式(8)值主要取决于第二项,可忽略第一项;换言之,当  $\sigma_e^2/\sigma_\lambda^2$  很大时,式(8)值主要取决于第一项,第二项可忽略。

此外,式(8)是基于 ML 的协作定位估计,且其属于非凸的,通常认为是 NP-hard。针对非凸问题,常利用半定规划 SDP (Semi-definite Programming) 松弛技术求解。然而,SDP 技术并不适用于求解式(8),这主要是基于两点原因:1) 对于大型网络,利用 SDP 技术求解,计算复杂度很高;2) 由于它们的复杂结构<sup>[8]</sup>,利用 SDP 技术求解,需要集中计算。为此,SOCP-DL 算法利用 SOCP 松弛技术求解,降低计算时间,更适合于分布式算法。

## 2.2 基于 SOCP 定位

将式(8)转换成集中式凸 SOCP 问题。先建立约束问题,如式(9)所示:

$$\min_{S,B} \left[ \sum_{(i,j,n) \in A} \frac{1}{\sigma_\lambda^2} (b_{ij}^{(n)} - d_{ij}^{(n)})^2 + \sum_{(i,n) \in V} \frac{1}{\sigma_e^2} \left\| \frac{1}{T_s} (s_i^{(n)} - s_i^{(n-1)}) - v_i^{(n)} \right\|^2 \right] \quad (9)$$

s. t.  $b_{ij}^{(n)} = \|(s_i^{(n)} - s_i^{(n-1)})\| \quad \forall (i,j,n) \in A$

式中:  $B = \{b_{ij}^{(n)} \mid (i,j,n) \in A\}$ 。令  $C = \{c_{ij}^{(n)} \mid (i,j,n) \in A\}$ 、 $X = \{x_i^{(n)} \mid (i,n) \in V\}$ ,式(9)可重写为:

$$\min_{S,C,B,X} \sum_{(i,j,n) \in A} (c_{ij}^{(n)})^2 + \sum_{(i,n) \in V} (x_i^{(n)})^2 \quad (10)$$

s. t.  $x_i^{(n)} \geq \frac{1}{\sigma_e} \left\| \frac{1}{T_s} (s_i^{(n)} - s_i^{(n-1)}) - v_i^{(n)} \right\| \quad \forall (i,n) \in V$

$$c_{ij}^{(n)} \geq \frac{1}{\sigma_\lambda} \|(b_{ij}^{(n)} - d_{ij}^{(n)})\| \quad \forall (i,j,n) \in A$$

$$b_{ij}^{(n)} = \|s_i^{(n)} - s_i^{(n-1)}\| \quad \forall (i,j,n) \in A$$

令  $z = \{z_{ij}^{(n)} \mid (i,j,n) \in A, x_i^{(n)} \mid (i,n) \in V\}$ ,再引用 SOCP 松弛技术,进而得到标准的 SOCP 形式:

$$\min_{S,z,B,y} y \quad (11)$$

s. t.  $y \geq \|z\|^2$

$$x_i^{(n)} \geq \frac{1}{\sigma_e} \left\| \frac{1}{T_s} (s_i^{(n)} - s_i^{(n-1)}) - v_i^{(n)} \right\| \quad \forall (i,n) \in V$$

$$c_{ij}^{(n)} \geq \frac{1}{\sigma_\lambda} \|(b_{ij}^{(n)} - d_{ij}^{(n)})\| \quad \forall (i,j,n) \in A$$

$$b_{ij}^{(n)} = \|s_i^{(n)} - s_i^{(n-1)}\| \quad \forall (i,j,n) \in A$$

利用现成的凸优化软件包(CVX)便可有效地求解式(11)。CVX 是 MATLAB 软件自带的软件包,并提供 MATLAB 接口。通过 CVX 软件包可解决线性规划、最小二乘法等问题。

然而,式(11)需要集中方式求解,其要求将所有测距值传输至融合中心 FC,这将导致大量的功率消耗和高的计算成本。为此,以分布方式求解式(11)。

## 2.3 分布式实施 SOCP

分析如何利用分布方式实现 SOCP 算法。首先,将约束优化的式(9)问题进行松弛:

$$\min_{S,B} \left[ \sum_{(i,j,n) \in A} \frac{1}{\sigma_\lambda^2} (b_{ij}^{(n)} - d_{ij}^{(n)})^2 + \sum_{(i,n) \in V} \frac{1}{\sigma_e^2} \left\| \frac{1}{T_s} (s_i^{(n)} - s_i^{(n-1)}) - v_i^{(n)} \right\|^2 \right] \quad (12)$$

s. t.  $b_{ij}^{(n)} \geq \|(s_i^{(n)} - s_i^{(n-1)})\| \quad \forall (i,j,n) \in A$

然后,再利用对数级数函数  $f(\alpha) = -\frac{1}{\eta} \log(-\alpha)$  替代不等式约束条件。因此,式(12)可近似为:

$$\min_{S,B} \sum_{(i,j,n) \in A} \frac{1}{\sigma_\lambda^2} (b_{ij}^{(n)} - d_{ij}^{(n)})^2 + \sum_{(i,j,n) \in A} f(\|(s_i^{(n)} - s_i^{(n-1)})\|^2 - (b_{ij}^{(n)})^2) + \sum_{(i,n) \in V} \frac{1}{\sigma_e^2} \left\| \frac{1}{T_s} (s_i^{(n)} - s_i^{(n-1)}) - v_i^{(n)} \right\|^2 \quad (13)$$

观察到式(13)不难发现,对于每个传感节点  $i \in \{1, 2, \dots, m_s\}$ ,式(13)仅取决于位置和它的邻居节点所转发的测距信息和速度测量值。因此,可将式(13)分解为  $m_s$  个独立的子问题。

令  $\pi_i = \{(j,n) \mid (i,j,n) \in A\}$ ,例如,  $(j,n) \in \pi_i$  意味着在时刻  $n$  第  $j$  个传感节点或锚节点落在第  $i$  个节点的通信范围内。此外,假定  $B_i = \{b_{ij}^{(n)} \mid (j,n) \in \pi_i\}$ 、 $C_i = \{c_{ij}^{(n)} \mid (j,n) \in \pi_i\}$ 、 $X_i = \{x_i^{(n)} \mid n = 2, 3, \dots, N\}$ 、 $z_i = \{z_{ij}^{(n)} \mid (j,n) \in \pi_i, n = 2, 3, \dots, N\}$ 。因此,可得第  $i$  个传感节点需解决的式(14)的子问题:

$$\min_{S_i, C_i, B_i, X_i, z_i} y_i \quad (14)$$

s. t.  $y_i \geq \|z_i\|^2$

$$x_i^{(n)} \geq \frac{1}{\sigma_e} \left\| \frac{1}{T_s} (s_i^{(n)} - s_i^{(n-1)}) - v_i^{(n)} \right\| \quad 2 \leq n \leq N$$

$$c_{ij}^{(n)} \geq \frac{1}{\sigma_\lambda} \|(b_{ij}^{(n)} - d_{ij}^{(n)})\| \quad \forall (j,n) \in \pi_i$$

$$b_{ij}^{(n)} = \|s_i^{(n)} - s_i^{(n-1)}\| \quad \forall (j,n) \in \pi_i$$

其中  $i = 1, 2, \dots, m_s$ ,  $S_i = [s_i^{(1)}, s_i^{(2)}, \dots, s_i^{(n)}]$ 。利用式(14),每个传感节点仅通过它自己的观察独立求解。考虑了  $\pi_i$  是  $A$  的子集,因此,式(14)的算法复杂度远

低于式(11)。

### 3 实验

#### 3.1 仿真环境

利用 MATLAB 7.1 软件建立仿真平台,并通过实验分析 SOCP-DL 算法的性能。利用 CVX 编程工具实施凸规划<sup>[9]</sup>。所有传感节点和锚节点部署于  $10\text{ m} \times 10\text{ m}$  方形区域。最初( $n=1$ ),所有锚节点和传感节点随机分布于  $10\text{ m} \times 10\text{ m}$  区域,并且它们依据随机的点移动模型(Random-way point mobility)<sup>[10-11]</sup>进行移动。最大的移动速度为  $V_{\max}$ 。

在每个时刻,每个节点就随机地选择目的地,当它达到了目的地后,再随机地选择下一目的地,并以不同的速度移动。此外,引用均方根误差 RMSE(Root Mean Square Error)作为性能指标。

#### 3.2 实验数据分析

首先分析 TOA 测距噪声方差  $\sigma_\lambda^2$ 、 $V_{\max}$  对 RMSE 的影响,且  $\sigma_\lambda^2$  从 0.1 至 0.5 变化,传感节点数  $m_s = 10$ 、 $m_a = 3$ 、 $\sigma_e^2 = 0.1\text{ (m/s)}^2$ 、 $R = 2.5\text{ m}$ 。实验数据如图 2 所示。

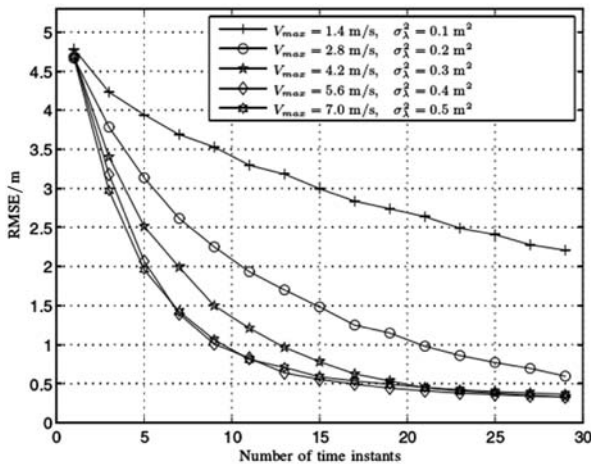


图 2 RMSE 随  $\sigma_\lambda^2$ 、 $V_{\max}$  的变化曲线

从图 2 可知,  $V_{\max}$  的增加,降低了 RMSE。原因在于节点移动速度越快,节点获取的与锚节点测距信息越多,越有利于测距。 $\sigma_\lambda^2$  为测距误差,  $\sigma_\lambda^2$  值越大,测距误差越大,必定增加 RMSE。此外,从图 2 可知,不论  $\sigma_\lambda^2$  的增加,大的  $V_{\max}$  仍能降低 RMSE。例如,当  $V_{\max} = 4.2\text{ m/s}$ 、 $\sigma_\lambda^2 = 0.3\text{ m}^2$  时的 RMSE 高于  $V_{\max} = 7.0\text{ m/s}$ 、 $\sigma_\lambda^2 = 0.5\text{ m}^2$  情况。

接下来,分析通信半径  $R$ 、 $V_{\max}$  对 RMSE 的影响,其中  $m_s = 10$ 、 $m_a = 3$ 、 $\sigma_\lambda^2 = 0.1\text{ m}^2$ 、 $\sigma_e^2 = 0.1\text{ (m/s)}^2$ ,通信半径  $R$  从 2 m 至 4 m 变化,实验数据如图 3 所示。

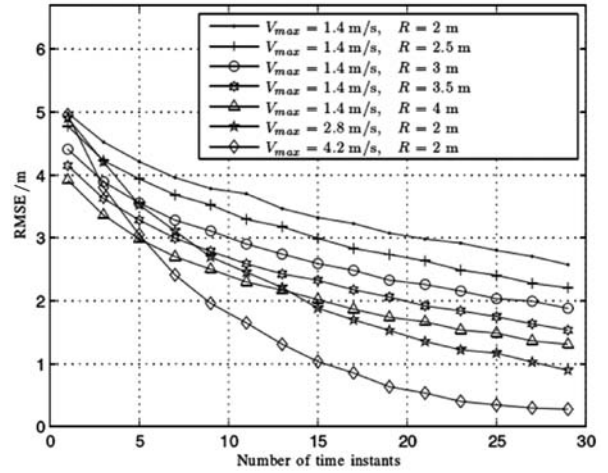


图 3 RMSE 随  $R$ 、 $V_{\max}$  变化曲线

从图 3 可知,通信半径  $R$  的增加,有利于降低 RMSE。这主要是因为:通信半径越大,节点的通信范围越宽,节点获取的信息就越多,越有利于定位。例如,当  $V_{\max} = 1.4\text{ m/s}$  时,  $R = 2.5\text{ m}$  增加至  $R = 5.0\text{ m}$ , RMSE 也随之下落。但是,  $V_{\max}$  从  $1.4\text{ m/s}$  增加至  $2.8\text{ m/s}$  比通信半径  $R$  从  $2\text{ m}$  增加至  $4\text{ m}$ , RMSE 降低得更多。

然后,选择同类算法进行对比分析,选择文献[7]的算法作为参照,用“Proposed Cooperative”表示本文提出的算法。 $\sigma_\lambda^2 = 0.1\text{ m}^2$ ,传感节点数  $m_s = 10$ 、 $m_a = 3$ 、 $\sigma_e^2 = 0.1\text{ (m/s)}^2$ ,  $R = 3\text{ m}$ 。

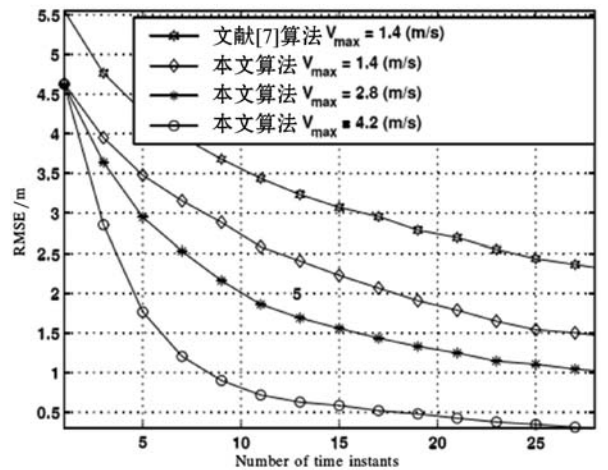


图 4 对比实验

从图 4 可知,与文献[7]的算法相比,提出的 SOCP-DL 算法的 RMSE 得到有效地降低。例如,在  $n=15$ ,  $V_{\max} = 1.4\text{ m/s}$  时,文献[7]算法的 RMSE 为  $3\text{ m}$ ,而提出 SOCP-DL 算法的 RMSE 为  $2.25\text{ m}$ 。这些数据表明,提出的 SOCP-DL 算法能够有效地降低 RMSE。

最后,分析算法的复杂度。执行算法的电脑参数为: Intel Core i5、2.4 GHz CPU。  $m_s = 10$ 、 $m_a = 3$ 、 $\sigma_e^2 = 0.1\text{ (m/s)}^2$ ,  $R = 2.5\text{ m}$ 。此外,为了更好地体现分布式

求解比集中式求解的成本低,将这两者进行对比,并仍选择文献[7]算法作为参照。表1显示算法的运行时间,其中,E-NIL-C表示采用集中式求解,而E-NIL表示采用分布式求解。

表1 算法的运行时间

算法	运行时间
E-NIL	2.34 s
文献[7]算法	3.01 s
E-NIL-C	3.32 s

E-NIL算法的计算时间为2.34 s,而文献[7]算法的计算时间为3.01 s。但是采用集中式求解的运行时间达到3.32 s,高于E-NIL。这也说明通过分布式实施SOCP能够有效地降低算法的复杂度。

## 4 结 语

本文针对移动WSNs的节点定位,提出基于二阶锥规划的分布式定位算法SOCP-DL。SOCP-DL算法先基于测距和速度测量值,建立ML估计表达式,然后再利用SOCP松弛技术求解,并对SOCP松弛技术分布式实施,进而降低算法的复杂度。实验数据表明,提出的SOCP-DL算法能够有效地提高定位精度。

## 参 考 文 献

- [1] Tomic S, Beko M, Dinis R. RSS-Based Localization in Wireless Sensor Networks Using Convex Relaxation: Noncooperative and Cooperative Schemes[J]. IEEE Transactions on Vehicular Technology, 2015, 64(5):2037-2050.
- [2] 江禹生,冯砚毫,管芳,等. 无线传感网非测距三维节点定位算法[J]. 西安电子科技大学学报(自然科学版), 2016,39(5):140-148.
- [3] Ghari P, Shahbazian R, Ghorashi S. Wireless Sensor Network Localization in Harsh Environments using SDP Relaxation[J]. IEEE Communications Letters, 2016, 20(1):137-140.
- [4] Naddafzadehshirazi G, Shenouda M B, Lampe L. Second order cone programming for sensor network localization with anchor position uncertainty[J]. IEEE Transactions on Wireless Communications, 2014, 13(2):749-763.
- [5] Han G J, Jiang J F, Zhang C Y, et al. A survey on mobile anchor node assisted localization in wireless sensor networks [J]. IEEE Communications Surveys & Tutorials, 2016, 18(3): 2220-2243.
- [6] Tomic S, Beko M, Dinis R. RSS-Based Localization in Wireless Sensor Networks Using Convex Relaxation: Noncooperative and Cooperative Schemes[J]. IEEE Transactions on

- Vehicular Technology, 2015, 64(5):2037-2050.
- [7] Salari S, Shahbazpanahi S, Ozdemir K. Mobility-Aided Wireless Sensor Network Localization via Semidefinite Programming[J]. IEEE Transactions on Wireless Communications, 2013, 12(12):5966-5978.
- [8] Shi Q, He C, Chen H, et al. Distributed Wireless Sensor Network Localization Via Sequential Greedy Optimization Algorithm[J]. IEEE Transactions on Signal Processing, 2010, 58(6):3328-3340.
- [9] Grant M, Boyd S. CVX: Matlab software for disciplined convex programming[OL]. version 1.21, Apr. 2011. <http://cvxr.com/cvx>.
- [10] Ssu K F, Ou C H, Jiau H C. Localization with mobile anchor points in wireless sensor networks[J]. IEEE Transactions on Vehicular Technology, 2005, 54(3):1187-1197.
- [11] Sheu J P, Hu W K, Lin J C. Distributed Localization Scheme for Mobile Sensor Networks[J]. IEEE Transactions on Mobile Computing, 2010, 9(4):516-526.

## (上接第154页)

- [2] 余贝,王红霞,谢杰荣. 基于可靠性的海底观测网传输网络拓扑结构研究[J]. 光纤与电缆及其应用技术, 2018(1):25-27.
- [3] Milevsky A, Walrod J. Development and test of IEEE 1588 Precision Timing Protocol for ocean observatory networks [C]//Oceans. IEEE, 2009.
- [4] Yokobiki T, Eiichiro E, Araki A, et al. Time-synchronization system for cabled observation systems[C]//The Seventeenth International Offshore and Polar Engineering Conference, 1-6 July, Lisbon, Portugal, 2007.
- [5] Kawaguchi K, Araki E, Kogure Y, et al. Development of DONET2-Off Kii channel observatory network[C]//Underwater Technology Symposium. IEEE, 2013.
- [6] Rio J D, Toma D M, Shariat-Panahi S, et al. Smart IEEE-1588 GPS clock emulator for cabled ocean sensors[J]. IEEE Journal of Oceanic Engineering, 2014, 39(2):269-275.
- [7] Lentz S, Lecroart A. Precision timing in the NEPTUNE Canada network[C]. Oceans. IEEE, 2009.
- [8] Li D, Wang J, Zhang Z, et al. Research and implementation of an IEEE 1588 PTP-based time synchronization system for Chinese experimental ocean observatory network[J]. Marine Technology Society Journal, 2015, 49(1):47-58.
- [9] 杨帆,王红霞,余贝,等. 水下信息传输网中 OTN 与 SDH 技术集成研究[J]. 光通信技术,2016,40(8):4-7.
- [10] 余贝. 缆系水下信息传输网络传输性能优化研究[D]. 武汉:海军工程大学,2017.
- [11] Puhm A, Kramer M, Moosbrugger P, et al. Problems and solutions for refitting a sensor network with IEEE1588 clock synchronisation[C]//Emerging Technology and Factory Automation. IEEE, 2015.