

西北工业大学

硕士学位论文

无线传感器网络分布式节点定位算法研究

姓名：王建刚

申请学位级别：硕士

专业：计算机应用技术

指导教师：王福豹

20060301

## 摘 要

集成了传感器、嵌入式计算、网络和无线通讯技术而成的无线传感器网络是一种全新的信息获取和处理技术,无线传感器网络可以自组织网络拓扑结构,节点间通过协同工作能够实时监测,感知和采集各种环境或监测对象的信息,并对其进行处理。无线传感器网络在国防、环境监测、空间探索、医疗卫生、精细农业、交通管理、制造业、反恐抗灾等领域具有广阔的应用前景。

无线传感器网络节点定位问题,是无线传感器网络重要的共性支撑技术之一,对其研究具有非常重要的意义。本文对已有的无线传感器网络分布式节点定位算法进行了深入的研究,跟踪了国内外在该领域的研究进展,对已有的几种典型的分布式节点定位算法做出了深入分析,定量地比较了算法的性能。并在此基础上提出了一种新的分布式节点定位算法。该算法主要在两个方面改进了定位性能:一是从提高定位精度方面,结合了相关数学原理,引入了加权最小二乘估计的方法,建立了加权定位算法的方案,以抑制迭代定位过程中累积误差的影响;二是从提高定位覆盖率方面,建立了利用两个锚节点和一个连通限制条件定位的算法模型,与三边测量定位(trilateration)、多边测量定位(multilateration)以及 Euclidean 算法相结合,确立了一种高覆盖率的分布式节点定位算法。通过在 NS-2 仿真平台上对算法的仿真实验,统计分析了各种因素对算法定位性能的影响,仿真结果表明,该算法具有良好的性能。

在实用化的定位系统研究方面,在实际的原型实验平台上实现了 DV-hop 定位算法,做出了实现定位系统的初步探索,奠定了进一步研究的良好基础。文章的最后对全文做了总结并对将来的研究方向做出了展望。

关键字:无线传感器网络;分布式节点定位算法;定位系统

## Abstract

WSN(Wireless Sensor Network), which is made by the convergence of sensor, embedded computation, networks and wireless communication technologies, is a novel technology about acquiring and processing information. WSN have the ability to collaboratively monitoring, sensing, collecting and processing information of the physical world through ad-hoc deployment of numerous tiny, intelligent, wirelessly networked sensor nodes. Potential applications of such sensor networks span many domains: military surveillance; environmental monitoring (air, water, soil, chemistry); smart spaces; physiological monitoring; precision agriculture; transportation; factory instrumentation.

Node localization provided the foundation for many application of WSN. In this paper, we first discussed several have been proposed distributed localization algorithms and quantitatively analyzed the performance. Secondly, a new distributed localization algorithms being proposed. Our approach improved the localization performance from two aspect: first, to restrain error accumulation during iterated localization process, we applying WLSE (Weighted Least Square Estimates) to node localization algorithm and choose proper weight coefficient to reduce location error; second, to improve localization coverage rate, we established a coordinate compute model based on two anchor and a connection constrain condition and integrated it with trilateration /multilateration and Euclidean algorithm.

After that, we implemented DV-hop localization algorithm on our test-beds and evaluate and demonstrate the effectiveness of the localization system. Finally, we sum up this paper and the directions of research in this area are introduced.

**Key words:** Wireless sensor networks; Distributed Node Localization algorithm; Localization system

## 第一章 引言

### 1.1 研究背景

无线传感器网络 (Wireless Sensor Network, WSN) 是指由大量成本低廉的, 具有感知能力、计算能力、无线通讯能力的传感器节点组成的网络。WSN 一般被密集部署在特定的监控区域, 节点间以无线、多跳的通信方式, 自组织网络拓扑结构, 节点间通过协同工作完成单个节点不能完成的全局任务<sup>[1-3]</sup>。

以前的监控系统经常是将少量的大功率的, 具有很强数据分析能力的传感装置安放在离目标有一定距离的观测点进行监控, 这样获得的数据很不准确, 需要传感装置具有较强的处理分析能力, 从混杂的信号中, 分析出有用的信息。现在, 微机电系统 (MEMS, Micro-Electro-Mechanism System)、片上系统 (SOC, System on Chip) 和无线通信技术高度集成使我们可以生产出体积小、耗能少、造价低, 功能较强的传感器节点, 将这种传感器节点高密度的分布在观测区域, 我们就可以获得全面的、精确度相当高的数据。这从根本上改变了传统监控系统的模式。

无线传感器网络与传统的无线网络 (如 WLAN 和蜂窝移动电话网络) 有着不同的设计目标, 后者在高度移动的环境中通过优化路由和资源管理策略最大化带宽的利用率, 同时为用户提供一定的服务质量保证。而无线传感器网络中除少数节点需要移动外, 大部分节点都是静止的。因为它们通常运行在人无法接近的恶劣、甚至危险的远程环境中, 能源无法替代, 设计有效的策略延长网络的生命周期成了无线传感器网络的核心问题。在研究初期, 人们曾经一度认为成熟的 Internet 技术加上 Ad hoc 路由机制对无线传感器网络的设计是足够充分的, 但深入的研究表明: 无线传感器网络与传统无线网络有着明显不同的技术要求, 前者以数据为中心, 后者以传输数据为目的。一些为自组织的 Ad hoc 网络设计的协议和算法并不适合传感器网络的特点和应用的要求。节点标识 (如地址等) 的作用在传感器网络中就显得不是十分重要, 因为应用程序不怎么关心单节点上的信息; 中间节点上与具体应用相关的数据处理、融合和缓存倒显得非常必要。在密集性的无线传感器网络中, 相邻节点间的距离非常短, 低功耗的多跳通信模式节省功耗, 同时增加了通讯的隐蔽性, 也避免了长距离的无线通讯易受外界噪声干扰的影响。这些独特的要求和制约因素为无线传感器网

络的研究提出了新的技术问题。

无线传感器网络与无线自组织网的主要区别有以下几点:

(1) 无线传感器网络虽然是无线自组织网的一种特殊形式,但由于传感网络的高密度,大规模以及传感节点的高度自治性使它与无线自组织网存在很多区别。其中的主要区别包括:无线传感网络为了节能主要采用以数据为中心(data-centric)的路由协议,数据可以在中间节点被处理(比如压缩或聚合)以减少传递的跳数;而自组织网主要采用传统的以地址为中心(address-centric)的路由算法,中间节点只负责数据分组的转发。

(2) 无线传感器网络的拓扑相对固定,即传感节点一旦被部署之后位置一般不发生变化,这与自组织网节点的自由移动性存在差别。

(3) 传感器网络中节点的分布相对于自组织网中的节点更加密集,数量也更多,因此传感网络的通信协议必须具备比自组织网更高的可扩展性。此外,由于相邻节点间的距离非常短,低功耗的多跳通信方式节省功耗,同时增加了通信的隐蔽性,也避免了长距离的无线通信易受外界噪声干扰的影响。

(4) 传感器网络节点依靠电池供电且通常工作在无人环境下,补充电能非常困难,因此传感器节点对节能的要求非常高。

由于存在这些区别使得原有的无线自组织网上的很多研究成果不能照搬到传感器网络中来,必须要针对传感器网络的特点进行专门的研发。目前国内外对这项技术的研究开展的十分活跃。

## 1.2 无线传感器网络概述

无线传感器网络被认为是 21 世纪最重要的技术之一,它将会对人类未来的生活方式产生深远影响。2003 年 2 月份的美国《技术评论》杂志<sup>[4]</sup>评出对人类未来生活产生深远影响的十大新兴技术,无线传感器网络即被列为第一。

简单的讲,无线传感器网络就是由大量的能与物理环境进行交互,同时具有数据处理功能和无线通信功能的新型传感器节点构成的互连的系统,该系统可以完成复杂的监控任务。这里的传感器节点,并不是传统意义上的单纯的对物理信号进行感知并转化为数字信号的传感器,它是将感应模块,数据处理模块和无线通信模块集成在了一块很小的物理单元上,功能比传统的传感器大了许多,不仅能够对环境信息进行感知,而且具有数据处理及无线通信的功能。我们将这类传感器节点高密度的分散在某个环境里,通过无线通信进行连接,它们就从整体上构成了一种特殊的网络。每个节点都有自己控制的一个区域,通过感知设备,如声学光学设备,化学分析装置,电磁感应装置等,来监测热、

红外、声纳、雷达和地震波信号，从而探测包括温度、湿度、噪声、光强度、压力、化学成分、移动物体的大小、速度和方向等众多部署者感兴趣的物质现象。

无线传感器网络可以完成在突发灾难性环境如核反应堆的监控，火警及地震等灾难事件区的监测。尤其是对于军事占领区战况分析，无人居住勘测等方面更是具有明显的应用价值。近年来无线传感器网络在很多商业领域如智能家居，城市交通拥堵的实时监控，沙尘环境的监控，水源污染监控，森林火灾监控，城市车辆监控和跟踪中也有着越来越多的应用。以军事应用为例，无线传感器网络将会成为 C4ISRT (Command, Control, Communication, Computing, Intelligence, Surveillance, Reconnaissance and Targeting) 系统不可或缺的一部分。C4ISRT 系统的目标是利用先进的高科技技术，为未来的现代化战争设计一个集命令、控制、通信、计算、智能、监视、侦察和定位于一体的战场指挥系统，受到了军事发达国家的普遍重视。因为无线传感器网络是由密集型、低成本、随机分布的节点组成的，自组织性和容错能力使其不会因为某些节点在恶意攻击中的损坏而导致整个系统的崩溃，这一点是传统的传感器技术所无法比拟的，也正是这一点，使无线传感器网络非常适合应用在恶劣的战场环境中，包括监控我军兵力、装备和物资；监视冲突区；侦察敌方地形和布防；定位攻击目标；评估损失；侦察和探测核、生物和化学攻击。在战场，指挥员往往需要及时准确地了解部队、武器装备和军用物资供给的情况，敷设的传感器将采集相应的信息，并通过汇聚节点将数据送至指挥所，再转发到指挥部，最后融合来自各战场的的数据形成我军完备的战区态势图。在战争中，对冲突区和军事要地的监视也是至关重要的，通过铺设传感器网络，以更隐蔽的方式近距离的观察敌方的布防；当然，也可以直接将传感器节点撒向敌方阵地，在敌方还未来得及反应的时间内迅速收集利于作战的信息。

无线传感器网络为野外随机性研究数据获取提供了方便，例如，跟踪候鸟和昆虫的迁移，研究环境变化对农作物的影响，监测海洋、大气和土壤成分等。ALERT<sup>[5]</sup>系统中就有多种传感器来监测雨量、河水水位和土壤水分，并以此预测山洪爆发的可能性<sup>[6]</sup>。类似地，无线传感器网络对预防森林火灾，即时准确地报告灾情也有很大地帮助。此外，在农业中，无线传感器网络可以监测农作物中的害虫、土壤的酸碱度、施肥状况等。

在医疗健康领域，如果在病人的身上安装特殊的传感器节点，如心率和血压监测设备，利用无线传感器网络，医生就可以随时了解被监护病人的病情，进行及时处理<sup>[7]</sup>。还可以利用无线传感器网络长时间地收集人体的生理数据，这些数据在医疗研究、新药品研制的过程中有非常巨大的作用，而安装在被监测对象身上的微型传感器节点也不会给人的正常生活带来太多的不便。无线传

传感器网络为未来的远程医疗提供了更加方便、快捷的技术手段。

在空间探索领域,借助于航天器布撒的传感器网络节点实现对星球表面长时间的监测,应该是一种经济可行的方案。NASA 的 JPL(Jet Propulsion Laboratory)实验室研制的 Sensor Webs<sup>[8]</sup>就是为将来的火星探测进行技术准备的,已在佛罗里达宇航中心周围的环境监测项目中进行测试和完善。

在其他的商业领域,无线传感器网络也有广阔的应用前景,例如,嵌入家具和家电中的传感器组成的传感器网络与 Internet 连接在一起将会为我们提供更加舒适、方便和具有人性化的智能家居环境;在城市车辆监测和跟踪系统中已经成功应用了无线传感器网络<sup>[9]</sup>;在仓储管理、工厂自动化生产线等众多领域,无线传感器网络都会孕育出全新的设计和应用模式<sup>[11]</sup>。

由于无线传感器网络巨大的科学意义和应用价值,它已经引起了世界发达国家的学术界、军事部门和工业界的极大关注。从 2001 年开始,DARPA(美国国防部高级研究计划署)已把智能传感器网络作为一项最高优先发展的研究计划,出资近 7 亿美元,在众多大学和研究机构展开传感器网络的基础研究,以求获得五角大楼想要的所谓战区“超视觉”数据<sup>[10]</sup>。2002 年 8 月,NSF(美国国家科学基金会)一期资助 4000 万美元在 UCLA 成立了传感器网络研究中心,联合周边大学(包括 UCB、USC 和 USR 等)展开“嵌入式智能传感器”的研究项目<sup>[11]</sup>,以求利用传感器网络对我们生活的物理世界实现全方位的测试与控制,这也是美国国情咨文中有关 NGI 最主要的远景规划之一。比较著名的研究项目计划包括:加州大学伯克利分校(UCB)的 SmartDust 计划;佐治亚理工大学 SensorNet 项目;加州大学洛杉矶分校(UCLA)的 WINS(Wireless integrated network sensors)项目;麻省理工学院的 mAMPS 项目等。其中加州大学伯克利分校研制的无线传感器系统 Mica、Mica2、Mica2Dot 等系列产品已被广泛地用于低功耗无线传感器网络的研究和开发。

无线传感器网络技术进展飞速,硬件方面的条件逐渐成熟,成本不断降低,微型化程度不断提高。美国 Dust 公司已经开始设计最终能够悬浮于空气中的“灵巧微尘”传感器,直径只有 5mm 左右,计划在近期内最终设计出体积小于 1 立方毫米的产品<sup>[12]</sup>。具有无线通信和识别能力且价格十分低廉的射频标签可以看作是传感器网络技术的一个特例,它能够对物品进行无接触的快速标示和识别,大大提高物流管理的精确度和效率。

无线传感器网络是一个刚刚兴起的研究领域,它的技术和应用还不十分成熟,要达到希望的要求还会有很长一段发展历程,有待继续研究开发。无线传感器网络业务和应用也需要不断开发,形成新兴的应用领域。

### 1.3 无线传感器网络节点定位问题

在传感器网络中, 位置信息对传感器网络的监测活动至关重要, 事件发生的位置或获取信息的节点位置是传感器节点监测消息中所必须包含的重要信息, 因为没有位置信息的监测消息往往毫无意义。因此, 确定事件发生的位置或确定获取消息的节点位置是传感器网络最基本的功能之一, 对传感器网络应用的有效性起着关键的作用。

在传感器网络的各种应用中, 监测到事件之后关心的一个重要问题就是该事件发生的位置。如在环境监测应用中需要知道采集的环境信息所对应的具体区域位置; 对于突发事件, 如需要知道森林火灾的现场位置, 战场上敌方车辆运动的区域, 天然气管道泄漏的具体地点等。对于这些问题, 传感器节点必须首先知道自身的地理位置信息, 这是进一步采取措施和做出决策的基础。

由于被监测区域通常是人员无法到达的区域, 所以在部署无线传感器节点时通常具有不可控制性, 例如用飞机将传感器节点布撒到指定的区域中。随机部署的传感器节点无法事先知道自身的位置, 因此传感器节点必须能够在部署后实时地进行定位。传感器节点的定位往往利用少量已知位置的节点, 通过节点间的协作, 按照某种定位机制确定需要定位节点的位置。

节点的位置信息除用来报告事件发生的地点外, 还具有下列用途: 目标跟踪, 实时监测目标的行动路线, 预测目标的前进轨迹<sup>[13]</sup>; 辅助路由, 如利用节点的位置信息进行数据传递的地理路由协议, 提高路由效率<sup>[14-15]</sup>; 进行网络管理, 利用节点传回的位置信息构建网络拓扑图, 并实时统计网络覆盖情况<sup>[16-17]</sup>, 以对节点密度低的区域采取必要的措施, 等等。因此, 在传感器网络中, 实现传感器节点的定位对各种应用有着及其重要的作用。

全球卫星定位系统 (Global Positioning System, GPS) 是目前应用最广泛最成熟的定位系统, GPS 系统利用精确的同步卫星时钟提供授时和测距以对用户节点进行定位, 具有定位精度高、实时性好、抗干扰能力强等优点, 但是 GPS 定位仅适应于无遮挡的室外环境, 其用户设备通常能耗高体积大, 成本也较高, 而传感器节点受到成本、能量和体积的限制, GPS 的成本和功耗是传感器节点所无法接受的, GPS 不适用于低成本的无线传感器网络。在机器人领域中, 机器人节点的移动性和自组织性等特性, 使其定位技术与传感器网络的定位技术有一定的相似性, 但是机器人节点通常携带充足的能量供应和精确的测距设备, 并且系统中机器人节点的数量很少, 机器人领域的定位算法往往没有考虑算法复杂度和能量消耗问题。所以这些机器人定位算法也不适用于传感器网络。在 Ad-hoc 网络中, 对节点定位的研究取得了丰富的成果。虽然无线传感器网络与



Ad-hoc 网络有一定的差异,例如在节点的移动性上,无线传感器网络的节点在部署完成之后,一般是静止不动的,而 Ad-hoc 网络的节点移动性较强;在网络的规模上,无线传感器网络比 Ad-hoc 网络的规模可能要大的多;从能耗的角度,无线传感器网络更注重节能的考虑;等等。但是无线传感器网络与 Ad-hoc 网络也有着更多的相似之处,Ad-hoc 网络中已有的节点定位算法和方案为无线传感器网络节点定位的研究提供了良好的基础和借鉴。事实上,已有许多 Ad-hoc 网络定位算法能够解决无线传感器网络节点定位问题。

在传感器网络中,传感器节点能量有限、可靠性差、网络规模大且节点随机布放、无线模块的通信距离有限,对定位算法和定位技术提出了很高的要求。传感器网络的定位算法通常需要具备以下特点:

自组织性:传感器网络的节点随机部署,不依赖于全局基础设施协助定位;

健壮性:传感器节点的硬件配置低,能量有限,可靠性较差,定位算法必须能够容忍节点失效和测距误差;

节能性:尽可能地减少算法中计算的算法复杂度,减少节点间的通讯开销,以尽量延长网络的生存周期。

传感器网络节点定位问题的研究在国内才刚刚起步,国外虽有一定研究,但偏重于定位算法的理论研究,距离实用化还有一定的差距。

## 1.4 研究内容与论文组织

作为无线传感器网络重要的共性支撑技术之一,无线传感器网络节点定位问题的研究具有非常重要的意义。本文对已有的无线传感器网络分布式节点定位算法进行了深入的研究,跟踪了国内外在该领域的研究进展,并在此基础上提出了一种新的分布式节点定位算法。对于在实际的原型实验平台上实现定位系统也做出了初步的探索。

论文组织如下:

第一章主要介绍了研究背景和研究意义以及具体的研究内容。

第二章对四种典型的分布式定位算法(Bounding box、DV-hop、Euclidean、Robust Position)做出了深入的分析,以此为后续的研究工作提供良好的基础和借鉴。

第三章对节点定位中最重要的指标——误差问题进行了深入分析,特别对节点定位过程中的累积误差问题提出了一种解决方法。

第四章详细介绍了一种新的分布式节点定位算法,并进行了仿真验证,对算法的性能进行了综合分析。

第五章介绍了在实际的原型实验平台上实现的定位系统，对定位算法和系统在实用性方面做出了初步的探索，并为将来实现不同定位算法提供了研究基础。

## 第二章 无线传感器网络节点定位算法的相关研究工作

### 2.1 无线传感器网络节点定位的基本原理

国外的大学和研究机构提出了许多专用于传感器网络的定位系统。在文献<sup>[18,19]</sup>中对这些算法的原理和特点做了较详尽的分析和描述,提出了定位系统和算法的性能评价标准和分类方法。

节点间距离或角度的测量是无线传感器网络节点定位的基础,其常用的技术有 RSSI、TOF、TDOA 和 AOA:

**RSSI(Received Signal Strength Indicator):** 已知发射功率,在接收节点测量接收功率,计算传播损耗,使用理论或经验的信号传播模型将传播损耗转化为距离,该技术主要使用 RF 信号。因传感器节点本身具有无线通信能力,故其是一种低功率、廉价的测距技术。

**TOF (Time of Flight):** 该技术通过测量信号传播时间来测量距离,因此需要节点间精确的时间同步。使用 TOF 技术最基本的定位系统是 GPS, GPS 系统需要昂贵、高能耗的电子设备来精确同步卫星时钟。因 WSN 节点硬件尺寸、价格和功耗限制,实际应用 TOF 技术的定位方案较少,但近期 UWB 技术的发展以及在无线传感器网络中的应用,使得利用 TOF 定位具有了广阔的前景。

**TDOA(Time Difference On Arrival):** TDOA 测距技术被广泛应用在 WSN 定位方案中。一般是在节点上安装超声波收发器和 RF 收发器。测距时,在发射端两种收发器同时发射信号,利用声波与电磁波在空气中传播速度的巨大差异在接收端通过记录两种不同信号到达时间差异,基于已知信号传播速度,直接把时间转化为距离。该技术的测距精度较 RSSI 高,可达到厘米级,但受限于超声波传播距离有限和 NLOS 问题对超声波信号的传播影响。

**AOA(Angle of arrival):** 这是一种估算邻居节点发送信号方向的技术,可通过天线阵列或多个接收器结合来实现,除定位外,还能提供节点的方向信息,如 MIT 的 The Cricket Compass 等项目<sup>[45]</sup>就利用多个接收器提出了基于 AOA 的硬件解决方案,其原型系统可在  $\pm 40^\circ$  角内以  $\pm 5^\circ$  的误差确定接收信号的方向。同样,AOA 技术也受外界环境影响,如噪声、NLOS 问题等都会对测量结果产生不同影响。同时,AOA 需要额外硬件,在硬件尺寸和功耗上可能无法用于传感器节点。

## 2.2 典型的分布式节点定位算法分析

无线传感器网络的定位算法可以分为集中式和分布式两种，集中式算法就是指把所需信息传送到某个中心节点，并在那里进行节点定位计算的方式，其优点在于可以从全局角度统筹规划，计算量和存储量几乎没有限制，可获得相对精确的位置估算。集中式算法的缺点在于中心节点位置较近的节点会因为通信量大而过早消耗完电能，导致整个网络与中心节点信息交流的中断，无法实时定位等。典型的集中式算法有：Convex Position<sup>[15]</sup>，质心定位算法<sup>[20]</sup>，MDS-MAP 算法<sup>[25]</sup>，APIT 算法<sup>[26]</sup>等；分布式算法是指依赖节点间的信息交换和协调，计算节点位置的工作在节点本地完成，分布式算法特别适用于大规模的传感器网络。

由于无线传感器网络节点电池能量有限，一旦电池耗尽，节点失效后就会在监测区域中形成监测盲区。这些失效节点往往难以更换，特别是当网络布置在无人区时。这样将造成网络的覆盖范围、网络寿命和实用性大大降低。因此，节点自身定位作为基础服务，应当最小化能量的消耗，而节点的能量消耗主要由计算和通讯开销组成。因此，仔细研究各种定位算法的计算开销和通讯开销，做出定量的比较是非常有必要的。

下面对四种典型的分布式定位算法（Bounding box、DV-hop、Euclidean、Robust Position）做出了深入的分析，这四种算法是完全分布式的，具有良好的可实现性，对这四种算法的深入分析具有一定的代表意义。

在详细讨论分布式定位之前，首先归纳出算法运行的环境：（1）传感器网络是自我组织的，这意味着放置节点时不能很好地控制，我们假定节点是随机部署。（2）出于简化和易于表示，我们限定环境在 2 维空间。（3）无线传感器网络中，少量的节点需要通过安装特殊的定位系统和采取人工部署来确定其位置。我们将已知位置的节点一般称为锚节点（anchor）。而其他未知位置的节点称为未知节点。

在算法分析中，几个重要参数的标识如下：

N---网络中节点的数量

A---网络中锚节点的数量

C---网络的平均连通度（平均邻居节点的数量）

K---参与一次多边测量定位的锚节点个数

2.2.1 Bounding Box 算法<sup>[21]</sup>

Bounding box 算法是由加州大学伯克利分校的 S.N.Semic 等人提出的。该算法假定所有的节点都位于区域 Q 内，Q 又进一步分为  $n^2$  个单元，与以通讯半径为 R 的圆作为通讯模型不同，算法采用的通讯模型命名为离散通讯模型，该模型使用的通讯半径为单元数  $\rho$ ，并以边长为  $2\rho$  的正方形作为通讯区域，如图 2-1 所示。

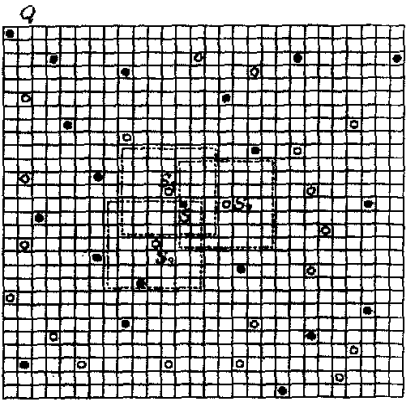


图2-1 离散通讯模型（空心圆表示锚节点，实心圆表示未知节点， $\rho = 3$ ）

这样，如果未知节点有  $k$  个邻居锚节点，则未知节点的位置估算可以取矩形区域的交集，如式（1）：

$$[\max(x_i - \rho), \max(y_i - \rho)] \times [\min(x_i + \rho), \min(y_i + \rho)] \quad i = 1 \dots k \tag{1}$$

通讯开销：锚节点发送广播消息，消息只传播到其一跳邻居，每个节点需要与其邻居节点通讯一次，所以整个网络发送的消息数量为  $N \cdot C$ 。

计算开销：该算法的计算包括求每个交集的两对最大值和最小值，对于每个交集，需要  $2C$  次比较运算和  $4C$  次加法运算，所以总的运算开销为  $6 \cdot C$  次 flop。

能量消耗：节点的能量消耗主要包括通讯和计算开销，综合上面的分析结果，并使用发送数据包的数量表示通讯的能量消耗，设单个 flop 的能量消耗为  $F$ ，算法的总能量开销见表 2-1：

表 2-1 Bounding box 算法的能量消耗		
算法	网络通讯开销	节点计算开销
Bounding box	$N \cdot C$	$6 \cdot C \cdot F$

评价：该算法需要非常少的计算和通讯开销，算法的覆盖速度很快，位置估计的精度随着锚节点数量的增加而提高，由于该算法是分布式的，所以也是可扩展的，每个节点的计算复杂度与网络的规模无关。算法的主要缺点是需要较

高的锚节点密度，否则定位精度和覆盖度将会很低。该算法适合于节点的计算能力非常有限的情况。

### 2.2.2 DV-hop 算法<sup>[22,23]</sup>

Niculescu 等人提出的 DV-hop 算法，其基本思想是将未知节点到锚节点之间的距离用网络平均每跳距离和两者之间的跳数乘积表示。首先使用距离矢量交换协议，使网络中所有节点获得距锚节点的跳数，然后锚节点计算网络平均每跳距离，并将其广播至网络中。当接收到校正值后，节点根据跳数计算与锚节点距离。

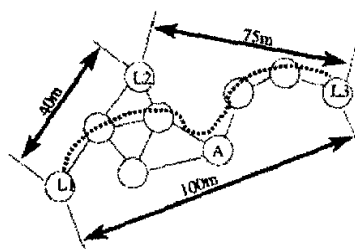


图2-2 DV-hop算法示意图

当未知节点获得与三个或三个以上锚节点的距离，则执行三边测量定位。设未知节点坐标为  $A(x, y)$ ，锚节点坐标为  $L1(x_1, y_1)$ 、 $\dots$ 、 $Lk(x_k, y_k)$ ，未知节点到锚节点的距离分别为  $r_1$ 、 $r_2$ 、 $\dots$ 、 $r_k$ ，则可建立线性方程组并表示为式(2-1)的形式。

$$Cx = b \quad (2-1)$$

$$\text{其中: } C = -2 \times \begin{bmatrix} (x_1 - x_k) & (y_1 - y_k) \\ (x_2 - x_k) & (y_2 - y_k) \\ \vdots & \vdots \\ (x_{k-1} - x_k) & (y_{k-1} - y_k) \end{bmatrix}, \quad x = \begin{bmatrix} x \\ y \end{bmatrix}$$

$$b = \begin{bmatrix} r_1^2 - r_k^2 - x_1^2 + x_k^2 - y_1^2 + y_k^2 \\ r_2^2 - r_k^2 - x_2^2 + x_k^2 - y_2^2 + y_k^2 \\ \vdots \\ r_{k-1}^2 - r_k^2 - x_{k-1}^2 + x_k^2 - y_{k-1}^2 + y_k^2 \end{bmatrix}$$

建立线性方程组之后，可以通过最小二乘法求解未知节点的位置估计：

$$\hat{x} = (C^T C)^{-1} C^T b \quad (2-2)$$

通讯开销：由于采用可控的泛洪在网络中传送消息，每个锚节点发送广播数据包，中间节点只发送未发送的数据包，由于要经过两次泛洪过程，所以网络中每个节点平均发送  $2A$  个数据包，算法的通讯开销为  $2 \cdot A \cdot N$ 。

计算开销：算法的计算开销包括距离计算和最小二乘算法。假定未知节点使用  $k$  个锚节点进行位置估计，每个节点执行  $k$  个乘法运算以计算  $k$  个距离，在计算距离的基础上，节点执行最小二乘运算来估计位置。位置估计所使用的最小二乘运算的计算复杂度与其矩阵的大小成比例。每一次最小二乘运算实际包含两个过程，数据首先要线性化，然后使用传统的最小二乘公式。线性化包括形成矩阵  $C$  和  $b$ ，形成  $C$  需要  $2(k-1)$  次减法运算，形成  $b$  需要  $5(k-1)$  次加法和  $6(k-1)$  次乘法。一旦数据线性化，就可以使用最小二乘估计位置。对于矩阵  $C \in R(k-1) \times 2$ ，如果采用 Normal Equations 方法求解最小二乘估计需要  $22(k-1)+23/3$  次 flop<sup>[24]</sup>。

综合以上分析结果，每个节点的计算开销为：

$$k + 2(k-1) + 5(k-1) + 6(k-1) + 2^2(k-1) + 2^3/3 = 18k - 17 + 2^3/3$$

能量消耗：算法的总能量开销见表 2-2：

表 2-2 DV-hop 算法的能量消耗

算法	网络通讯开销	节点计算开销
DV-hop	$2 \cdot A \cdot N$	$(18k - 17 + 23/3) \cdot F$

评价：算法只需要较少的锚节点，计算和通讯开销适中，不需要节点具备测距能力，是一个可扩展的算法，对于各向同性的密集网络，可以得到合理的平均每跳距离，从而能够达到适当的定位精度。但对于网络拓扑不规则的网络，定位精度迅速下降，不适合采用 DV-hop 算法。

### 2.2.3 Euclidean 算法<sup>[22,23]</sup>

Euclidean 算法基于围绕在锚节点周围的节点的局部几何拓扑，给出了计算与锚节点相隔两跳的未知节点位置的方法。如图 2-3，未知节点 B、C 与锚节点 L 两两相邻；节点 A 与 B、C 相邻。对于四边形 ABCL，所有边长和一个对角线 BC 已知，根据简单的几何原理可计算出 AL 的长度。但节点 A 有两个可能的位置 A 和 A'，假如 A 还有其他邻居节点 D 与锚节点 L 相邻，并与 B 或 C 之一相邻，那么可以使用 D 来替换 B 或 C 再次计算 AL 的距离，则 A 节点就能在两个可能的位置中择出正确的一个。使用这种方法，当未知节点获得与三个或更多锚节点距离后定位自身。

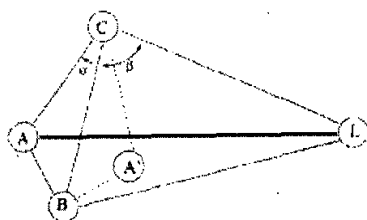


图2-3 Euclidean定位算法示意

$$\cos(\alpha) = \frac{AB^2 - AC^2 - BC^2}{2 \cdot AC \cdot BC}, \quad \cos(\beta) = \frac{BL^2 - BC^2 - CL^2}{2 \cdot CL \cdot BC}$$

$$AL^2 = AC^2 + CL^2 - 2AC \cdot CL \cdot \cos(\beta \pm \alpha)$$

通讯开销：每个锚节点发送 ttl 为 2 数据包，即数据包只转发 2 跳，锚节点的邻居节点转发一次数据包，整个网络发送的数据包数量为  $A+A \cdot C$ 。

计算开销：算法的计算开销包括距离计算和最小二乘算法。其中距离计算需要作  $44 \times 2 + 2 = 90$  次 flop 得到正确的距离  $AL$ ，多边测量定位需要  $17k-17+23/3$  次 flop。

能量消耗：算法的总能量开销见表 2-3：

表 2-3 Euclidean 算法的能量消耗

算法	网络通讯开销	节点计算开销
Euclidean	$A+A \cdot C$	$(17k+73+23/3) \cdot F$

评价：与 DV-hop 算法不同，Euclidean 算法依赖于局部几何拓扑，可以适用于网络拓扑不规则的网络，由于数据包只传送两跳，该算法通讯开销较小，同时具有适当的计算开销和定位精度。但该算法的覆盖度受到锚节点密度和局部几何拓扑的影响较大。

## 2.2.4 Robust position 算法<sup>[25]</sup>

该算法由两阶段组成：初始阶段和求精阶段。初始阶段使用 Hop-TERRAIN 算法（与 DV-hop 类似）提供每个节点的初始位置估计。在求精阶段，节点尝试提高位置估计精度。节点通过测量到所有一跳邻居的距离并依次更新自己的位置。该算法的所有位置计算都使用最小二乘估计三边测量定位。算法引入了置信度来提高求精阶段的性能，置信度被用来在三边定位中加权。所有未知节点的置信度初值为 0.1，锚节点为 1.0。当未知节点更新其位置估计时同样也更新其置信度为其邻居节点的置信度的平均值。这样，网络的平均置信度将随迭代而增加，提高了覆盖度和精度。

通讯开销：初始阶段与 DV-hop 类似，由每个锚节点发送广播数据包，中间



节点只发送未发送的数据包，所以网络中每个节点平均发送  $A$  个数据包。

在求精阶段，节点只广播其位置信息到其一跳邻居节点，所以每个节点发送  $s$ （迭代次数）个数据包。通过优化可以减少这个数量，例如只有当位置信息变化较大时才广播更新位置，这样就比估计的通讯开销要小。

计算开销：在初始阶段的 Hop-TERRAIN 算法过程中，与 DV-hop 类似，每个节点将执行  $17(k-1)+23/3$  次 flop。

求精的计算开销与 Hop-TERRAIN 类似。求精的每一次迭代，节点将做一次最小二乘估计并产生新的置信矩阵。如果算法共迭代  $s$  次，对于求精，每一次最小二乘估计将包含节点的一跳邻居。因此，每一次最小二乘运算将消耗  $17(C-1)+23/3$  次 flop。每一个置信度计算需要  $C$  次加法和 1 次除法。在整个求精过程中，每个节点需要  $s[17(C-1)+23/3+C+1]=s[18C-16+23/3]$  次 flop。

能量消耗：算法的总能量开销见表 2-4：

表 2-4 Robust Position 算法的能量消耗

算法	网络通讯开销	节点计算开销
初始阶段	$2 \cdot A \cdot N$	$[17(k-1)+23/3] \cdot F$
求精阶段	$s \cdot N$	$[18C-16+23/3] \cdot s \cdot F$
总计	$(2 \cdot A + s) \cdot N$	$[17(k-1)+(18 \cdot C - 16)s + 23/3(1+s)] \cdot F$

评价：算法能够达到较好的精度。该算法在网络连通度较高的情况下能较好的容忍距离误差。因为节点主要和其一跳邻居节点通信，该算法也是一个可扩展的算法。但是，由于迭代的过程，该算法是强计算的。如果初始位置估计非常不准确或误差是相关的，算法可能不能达到精确的估计。依赖于网络拓扑，算法可能需要很长的覆盖时间。

## 2.2.5 评价与比较

上述的 4 种定位算法都是完全分布式的，也是可扩展的，即每个节点的计算复杂度与网络的规模无关。Bounding box 算法具有最小的计算与通讯开销，而 Robust Position 算法的计算与通讯开销最高，DV-hop 与 Euclidean 的计算与通讯开销适中。

定位精度是评价算法性能的一个重要标准。Bounding box 算法只有当网络中锚节点密度较高时才能获得较好的定位精度，与之相比，DV-hop 与 Euclidean 算法在不同的网络环境下可以达到适中的定位精度，由于 Robust Position 算法采用了迭代求精的过程，能够达到较高的定位精度。

由于不使用距离测量进行定位计算，Bounding box 算法和 DV-hop 算法不受

测距误差影响，具有高的容错性。Robust Position 算法引入了迭代的求精过程，有效地抑止了测距误差的累积，从而达到了较好的容错性。

Bounding box 和 Euclidean 算法的覆盖度都与锚节点密度有关，但 Euclidean 算法使用了与锚节点两跳的距离估算，能够达到更高的覆盖度。此外，这两种算法都可以使用将已定位的节点升级为锚节点的方法来提高覆盖度。

对于无线传感器网络，成本是一个非常重要的限制因素，由于 Bounding box 算法需要较高的锚节点密度，增加了网络成本。而其他的算法没有这样的限制。上述比较的结果见表 2-5。

表 2-5 定位算法的比较

	Bounding box	DV-hop	Euclidean	Robust Position
计算开销	$6 \cdot C \cdot F$	$(18k-17+23/3) \cdot F$	$(17k+73+23/3) \cdot F$	$[17(k-1)+(18 \cdot C-16)s+23/3(1+s)] \cdot F$
通讯开销	$N \cdot C$	$2 \cdot A \cdot N$	$A+A \cdot C$	$(2 \cdot A+s) \cdot N$
定位精度	与锚节点密度相关	适中	适中	较高
容错性	不受测距误差影响	不受测距误差影响	中等	较高
覆盖度	与锚节点密度相关	较高	与锚节点密度相关	较高
覆盖速度	快	与网络规模相关	较快	较慢
网络成本	高	低	中	中

可见，在多种条件下衡量，没有一种算法是最优的。对于不同的设计目标，这四种算法中每一个算法在某一特定的条件下会优于其他的算法，而且，每一种算法还有一定优化与提升空间。

从上述分析中，可以看到 Robust Position 算法从定位精度、容错性、覆盖度、覆盖速度、网络成本几个方面都具有较强的优势，但该算法的计算复杂度较高，可以考虑在算法的初始阶段的位置计算时，引入 Bounding box 算法的思想，实质是求解二次方程组的问题简化为一次方程问题，从而避免了复杂的最小二乘解法，可以大大减少计算开销，这是一个有待进一步研究的一个方向。

2.3 无线传感器网络定位技术的发展

近年来，各种新技术的发展与应用以及定位算法的研究都取得了丰富的成

果,从提高定位精度和定位覆盖度、降低能耗和成本等不同方面不断地完善定位系统和算法,使得无线传感器网络定位系统的研究更加实用化。下面介绍基于 USB 的定位系统和 Calamari 定位系统的发展。

### 2.3.1 基于 UWB 的定位系统

超宽带(UWB)脉冲无线电技术具备了许多正弦载波通信技术无法比拟的优势,非常适于微小的传感器节点。特别是 UWB 脉冲的宽度为纳秒级,占用带宽在 1 GHz 以上,采用 TOA 技术测距,理论上可以达到厘米级的测距精度<sup>[36]</sup>,因为超宽带信号比窄带信号和宽带信号拥有更宽的信号带宽,它们可以穿透墙和其他障碍物,从而避免 NLOS 问题。UWB 的这种良好的性能为定位系统的设计提供了很好的支撑。

如前所述,使用 TOA 测距,需要节点间精确的时间同步,对于 UWB 系统的时间同步,已有了许多研究成果,利用精确的时间同步,节点可以获得较高精度测距值。文献<sup>[37]</sup>对使用 UWB 无线通讯的 TOA 测距技术做出了深入分析,提出了使用通用最大似然估计(generalized maximum-likelihood)的方法建模,快速测距的方案。

目前,基于 UWB 的定位系统已有一些研究成果。如 MIT 的 Damien B. Jourdan 等提出的 Monte Carlo<sup>[38]</sup>的定位系统,它利用 UWB 能克服多径问题(multipath fading)的良好特性,以及使用粒子滤波(Particle filter)解决非视距问题(NLOS propagation),为室内条件下的定位提供了良好的解决方案。芬兰奥鲁大学(University of Oulu)的 Kegen Yu 等利用非线性优化理论中的 Davidon-Fletcher-Powell(DFP)quasi-Newton 算法<sup>[39,40]</sup>验证了 UWB 应用于定位系统的性能。奥卢大学的 Jean Philippe Montillet 等也提出基于 TOA 和 AOA 两种技术建立成本函数(Cost Function)<sup>[41]</sup>,综合比较了 DFP 和 DM(Direct Method)两种算法。

### 2.3.2 Calamari 定位系统

加州大学伯克利分校开发的 Calamari 定位系统基于 Crossbow 公司的 Mica2 无线传感器网络节点平台和 TinyOS 嵌入式操作实时系统,Calamari 使用了 RSSI(RF received signal strength information)和 TOF(acoustic time of flight)相融合的一种方法来估计节点间的距离,这两种技术都是相对比较廉价的。

在 RSSI 距离估计方法中,一个节点发射频率在 916.5MHz 的射频载波信号,而另一个节点采样信号的接收强度。因为节点已经有无线电通信元件,所以这

是一种低功率、没有计算量消耗同时又不需额外硬件的距离估计方法。

而在 TOF 距离估计方法中,需要使用附加的硬件,并消耗更多的功率,但能够比 RSSI 产生更精确的距离估计。发射节点同时发送短的射频和声音脉冲,而接收节点对比两个脉冲到达的时间。因为无线电和声音的传播速度不同,则通过 TDOA (time difference on arrival) 则可计算出接收和发射节点间的距离。通过使用一种完全采用模拟装置的解决方法来取代数字信号处理。在 Calamari 中,一个大约 15ms 的声音脉冲和一个 25ms 的无线电信息包同时传送。当麦克风接收到声音脉冲,音频检测器的锁相回路 (PLL) 锁定这个信号。当 PLL 响应和麦克风响应的组合足够大时,处理器就发送一个带时间戳的脉冲。这个时间戳与无线电信息包的时间戳做比较,时间上的差异乘以声音的传播速度就得到了距离估计。

Calamari 提供了一种廉价的、低功率的解决方案,但是在节点间的距离估计上也产生了一定的不准确性,这就意味着需要高精度校准来获得合理的结果。例如,一个无线电波可能以两倍的功率于另外一个无线电波传播,这就可能导致 100% 的距离估计误差。发送节点的频率变化也可能影响 RSSI 的观测结果,这些变化在声学设备上也是同样存在的。因为这些硬件上的可变性会引起在两个发送/接收节点间距离估计的 300% 的差异。可以看到,传统的校准方法可以做成对的校准,但不能校准单个发送或接收器;因为系统只有一个输出结果,不能把发送器和接收器的影响分开。

Calamari 项目通过大量不同环境下的实验,获取实际数据,建立了称之为 Noisy Disk<sup>[42]</sup>的测距模型,其中环境因素包括节点无线通讯模块不同的发射功率;节点距离地面的高度;节点所处的室内、室外的环境中有无障碍物以及障碍物的高低等因素。研究了在不同网络密度和噪声条件等因素的影响下, Noisy Disk 模型对定位算法的影响。

Calamari 定位系统采用 TDOA 超声波测距,定位算法使用改进的 DV-distance 算法,部署 49 个节点于 144m<sup>2</sup> 的区域,平均定位误差达到 0.53m。而使用 RSSI 测距,将 49 个节点布置到半个足球场大小的区域,平均误差会达到 4.1m。

### 2.3.3 定位算法的硬件实现

近年来,无线传感器网络节点定位问题现有的研究基本集中于定位算法的设计、开发和仿真,但同样重要的算法实现却少有问题。针对这个问题,加州大学伯克利分校基于 Mica2 节点平台,利用 0.13  $\mu\text{m}$  CMOS 工艺研制了第一款无线传感器网络专用定位芯片<sup>[43]</sup>。定位模块仅占 0.55mm<sup>2</sup>,有效功率为 1.7mW。

由于在定位计算过程中求解最小二乘估计需要大量的浮点运算，其计算开销占据了定位计算的绝大部分，该芯片使用硬件实现了定位计算，采用了快速响应分解(QR decomposition)求解最小二乘算法。

快速相应求解最小二乘有两种算法：Householder reflection 和 Givens rotations。这两种算法都使用了迭代的方式。Givens rotations 每次迭代消去矩阵中的一个元素，而不是象 Householder reflection 那样每次消去一列，Givens rotations 需要  $4n2m-4/3n^2$  次浮点操作，这是 Householder reflection 算法的两倍，但是 Householder reflection 算法在每一次迭代中需要计算  $m$  维向量的标准差，这对于硬件实现来说是代价高昂的。因此选择了 Givens rotations 算法来实现快速响应分解。图 2-4 表示了使用 Givens rotations 进行快速相应分解的过程。

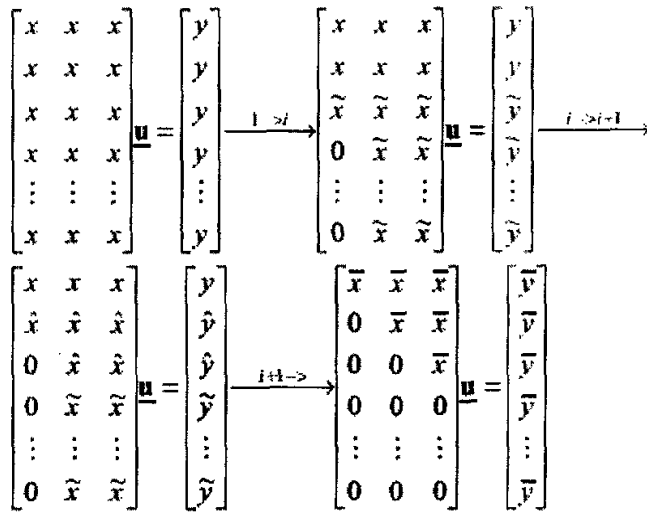


图 2-4 使用 Givens rotations 进行快速相应分解的过程

矩阵分解使用了锚节点链表中的数据计算。如果使用了 16 个锚节点，在最差的情况下需要约 825 个周期求解最小二乘估计。在 16MHz 的频率下工作，825 个周期需要花费  $52 \mu s$ 。

## 第四章 一种高覆盖率的节点定位算法

由于无线传感器网络的具有低功耗、低成本、高密度、大规模部署等特点，对节点定位算法具有特殊的要求。同时，不同的应用也对定位算法和系统有不同的要求。对于定位算法的评价从不同的方面有着不同的评价标准，但综合考虑，一个好的定位算法应当符合以下特点：

1、算法是分布式的：由于集中式算法需要把信息传送到某个中心节点，并在那里进行节点定位计算，这样靠近中心节点的节点会因为通信量大而能耗过大，可能造成能量过早耗尽而导致整个网络与中心节点信息交流的中断；此外，集中式算法也存在当网络拓扑或环境变化时无法实时定位等缺点。而分布式定位算法由于定位计算在节点本地完成，节点间的通讯较均匀的分布在整个网络中，分布式定位算法特别适合与大规模的无线传感器网络，具有良好的可扩展性。

2、具有较高的定位精度：定位精度是节点定位的首要评价指标之一。由于节点测距设备受成本、体积、功耗等方面的限制，不同的测距技术其精度都有局限性，所以必须从算法的角度尽可能的提高节点的定位精度。

3、具有较高的定位覆盖率：节点的定位覆盖率是指网络中能够定位节点占全部节点的比率。实现尽可能多的未知节点的定位，减少不能定位节点的数量，是良好的定位算法的重要指标。

4、具有较小的定位开销：定位开销包括定位过程中的计算开销和通讯开销以及存储开销等。良好的定位算法应当尽量减少这些开销，达到降低能量消耗的目的。

以上几点并不是孤立的，而是相互制约、相互影响的。例如为了提高定位精度引入迭代计算的过程就会增加定位的计算开销。因此，设计定位算法必须综合考虑各方面的因素，权衡各项指标，以使算法达到较好的性能，更具实用性。

基于以上几点，结合对现有定位算法的研究，综合了多种定位方法，提出了一种高覆盖率的无线传感器网络分布式节点定位算法。

## 4.1 节点的定位覆盖率问题

### 4.1.1 现有定位算法的覆盖率

节点的定位覆盖率，即网络中能够定位节点占全部节点的比率，是评价一个定位算法性能的重要指标之一。

对于三边测量定位或多边测量定位，至少需要 3 个锚节点到未知节点的 3 个距离值，才能实现未知节点的定位计算。由于无线传感器网络随机部署条件的限制，以及使用的定位算法的不同，未知节点常常不能获得 3 个或 3 个以上到锚节点的距离，从而无法实现定位，当网络连通度较低时，这种不能定位节点的比率将会比较大，即定位覆盖率较低。对于无线传感器网络的许多应用，这些不能定位节点监测到的信息或事件由于缺乏位置信息，这些信息可能会毫无意义，从而这些不能定位的节点也就成为了监测盲点，大大影响无线传感器网络的监测效能。

### 4.1.2 利用两个锚节点定位的算法模型

当只有两个锚节点与未知节点  $S$  相邻，可以计算出未知节点  $S$  的两个可能位置  $S$  与  $S'$ （图 4-1a）。此时，如果有第 3 个锚节点  $A_3$  与  $A_1$ 、 $A_2$  相邻，而  $A_3$  与  $S$  没有相邻意味着  $A_3$  与  $S$  的距离大于通讯半径，利用该限制条件可以判断  $S$  的真实位置(图 4-1b)。

对于 3 个锚节点共线的情况（图 4-1c），可以用同样的方法解决。

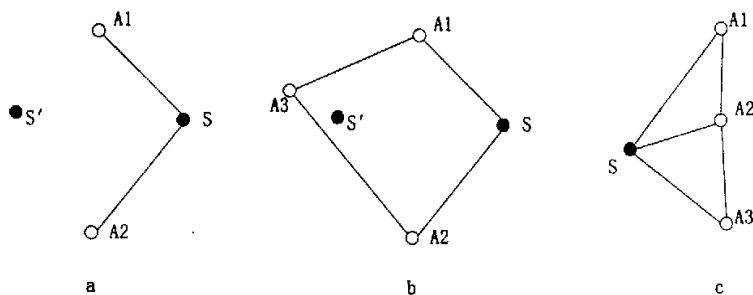


图 4-1 两个锚节点定位的条件

计算未知节点  $S$  坐标可参考图 4-2:

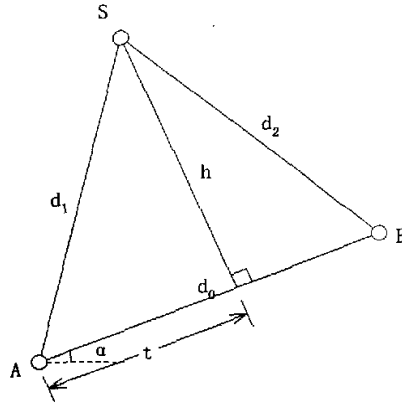


图 4-2 两个锚节点定位的算法

根据余弦定理,

$$d_2^2 = d_0^2 + d_1^2 - 2 \cdot d_0 \cdot d_1 \cdot \cos A$$

$$\cos A = \frac{1}{2 \cdot d_0 \cdot d_1} \cdot (d_1^2 + d_0^2 - d_2^2)$$

$$t = d_1 \cdot \cos A = \frac{1}{2 \cdot d_0} (d_1^2 + d_0^2 - d_2^2)$$

$$h = \pm \sqrt{d_1^2 - t^2}$$

$$\begin{cases} x_S = x_A + t \cdot \cos \alpha - h \cdot \sin \alpha \\ y_S = y_A + t \cdot \sin \alpha + h \cdot \cos \alpha \end{cases} \quad (4-1)$$

其中,  $h$  取正值, 则点  $S$  在  $AB$  的左侧;  $h$  取负值, 则点  $S$  在  $AB$  的右侧。因此可以计算出点  $S$  的两组坐标:  $(x_S, y_S), (x_{S'}, y_{S'})$ 。

取舍这两组坐标可以利用计算与第三个锚节点  $A_3$  的距离来判断:

$$\begin{cases} distance1 = \sqrt{(x_S - x_{A3})^2 + (y_S - y_{A3})^2} \\ distance2 = \sqrt{(x_{S'} - x_{A3})^2 + (y_{S'} - y_{A3})^2} \end{cases} \quad (4-2)$$

如果  $distance1$  小于通讯距离并且  $distance2$  大于通讯距离, 则取节点  $S$  的坐标为  $(x_S, y_S)$ ; 反之则取节点坐标为  $(x_{S'}, y_{S'})$ 。

## 4.2 节点定位算法的框架

算法由以下阶段组成:



1、信息收集阶段：在网络部署完成后，所有锚节点节点广播一个包含自身的 ID 以及自身的坐标信息的信息。该消息以泛洪的方式向网络中传播，为降低网络中的通讯流量，这种泛洪的采取可控的泛洪方式，即各个节点仅转发以前未收到的消息，如果该消息以前收到过则丢弃该消息。利用这个泛洪的过程，使每个节点建立邻居节点信息以及两跳内的锚节点信息，同时可以利用通过程进行 RSS 测距；

2、定位阶段：未知节点根据接收到的锚节点信息判断能否进行定位计算，分为 4 种情况：

(1) 节点与 3 个以上的锚节点相邻，则利用多边测量定位 (multilateration) 进行定位计算；

(2) 节点与 3 个锚节点相邻，则首先判断这 3 个锚节点是否共线，若不共线，则利用三边测量定位 (trilateration) 进行定位计算。若共线，则按第 3 种情况处理。

锚节点是否共线可按下面的方法判断，如图：

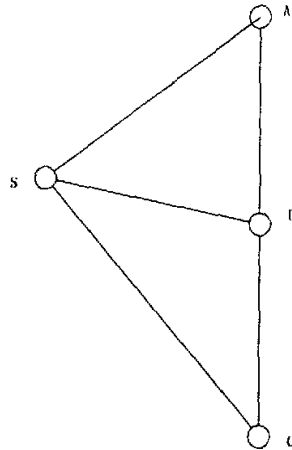


图 4-3 判断 3 个锚节点共线

$$\angle ASB = \cos^{-1} \left( \frac{|AB|^2 - |AS|^2 - |SB|^2}{2 \times |AS| \times |SB|} \right)$$

$$\angle BSC = \cos^{-1} \left( \frac{|BC|^2 - |BS|^2 - |SC|^2}{2 \times |BS| \times |SC|} \right)$$

$$\angle ASC = \angle ASB + \angle BSC$$

$$|AC| = \sqrt{2 \times |AS| \times |SC| \times \angle ASC + |AS|^2 + |SC|^2} \quad (4-3)$$

若  $|AC| = |AB| + |BC|$ ，则 A、B、C 三点共线。

(3) 节点与 2 个锚节点相邻，则判断是否符合第四章所述的两个锚节点定位的计算条件，如果符合则利用该算法模型进行定位计算；

(4) 如果以上条件均不满足，则利用 Euclidean 算法计算与锚节点两跳距离，若获得 3 个或 3 个以上的锚节点距离则利用三（多）边测量定位进行位置计算。

3、迭代阶段：在定位阶段已定位的节点升级为锚节点，并为该节点赋予合适的权值，用于在迭代定位过程中形成加权矩阵，从而利用加权最小二乘算法进行节点定位。

定位算法的流程如图 4-4 所示：

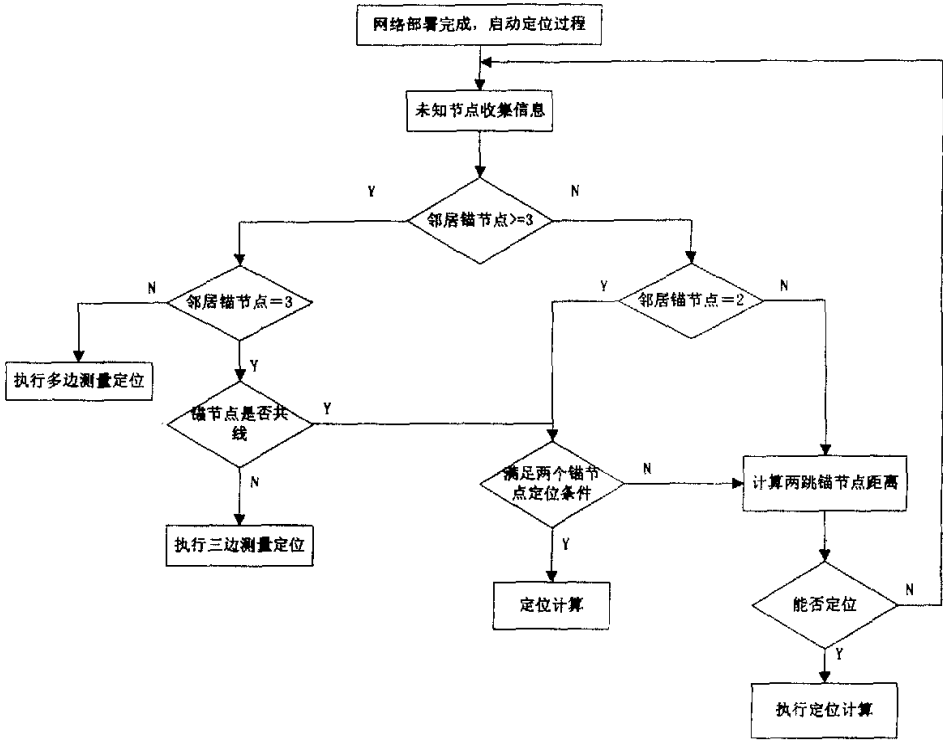


图 4-4 定位算法流程

4.3 仿真实验及性能评价

设计一个新型的节点定位算法，需要对该算法的性能进行合理的评价。可以进行这种评价的方法大致有以下三种：（1）数学分析的方法。这种方法需要有数学理论基础的支撑，因此可以得出精确而合理的结果。但是进行数学分析前需要首先建立一个合理的数学模型来对网络状况进行合理的描述。而这种模

型的建立, 往往是无法全面兼顾各种情况的。因此, 对定位算法的数学分析之余, 还需要有具体的实践来证实这样的分析的有效性。(2) 模拟仿真。这是一种将一些网络环境量化后, 建立模拟的拓扑结构, 利用仿真工具跟踪网络的行为进行网络分析与设计。(3) 原型实验平台。这种方式就是把需要的实验在实际的硬件上做, 在现实的环境中进行确实的实验。构建原型实验平台在实际的网络中测试, 一方面代价过高, 而且结果不可再现, 另一方面测试结构搭建之后, 拓扑难以改变, 因此对于大规模、不同环境下的测试比较困难, 一般原型实验平台作为一种验证辅助手段。现在对于网络的研究和分析主要采用模拟仿真结合数学分析的方法。

#### 4.3.1 网络部署

在仿真中, 为了使仿真结果更加接近真实, 根据锚节点密度和网络平均连通度在正方形区域中随机部署节点, 节点的位置生成利用了 NS-2 提供的随机变量生成, 服从平均分布。节点的拓扑生成可以用下面的代码实现。

```
set tempRNG [new RNG]
$tempRNG seed 0.0

set temp [new RandomVariable/Uniform]
$temp set min_ 0
$temp set max_ $val(x)
$temp use-rng $tempRNG

for {set i 0} {$i < $val(nm)} {incr i} {
    set node_($i) [$ns node]
    $node_($i) color green
    $node_($i) random-motion 0
    $tempRNG next-random
    set tempx [expr round([$temp value])]
    $node_($i) set X_ $tempx
    set tempy [expr round([$temp value])]
    $node_($i) set Y_ $tempy
    $node_($i) set Z_ 0.0
}
```

图 4-6 节点的拓扑生成的 Tcl 代码

图 4-7 是一个在 60m×60m 的正方形区域中随机部署 100 个节点的例子

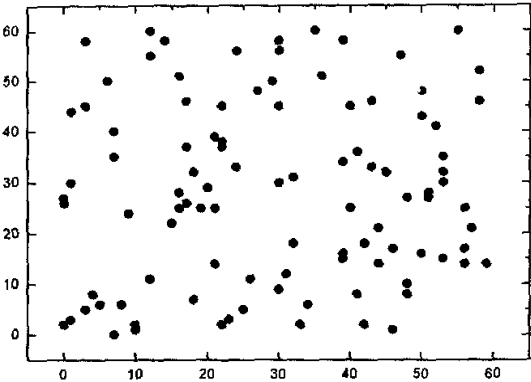


图 4-7 随机生成的网络拓扑（60m<sup>2</sup> 100 个节点）

为使仿真结果更加准确，对于每一个场景（不同的网络连通度、锚节点比率、测距误差）各仿真 50 次，最后对仿真结果进行统计，取平均值作为最后结果。

4.3.2 测距误差对节点定位的影响

提高测距准确度是提高节点定位准确度一个有效途径，难点是测距的实现要受节点能量、硬件复杂度及体积的限制。在实际应用中，测距准确度是不能无限提高的，这就需要评价不同测距误差条件下对定位精度的影响，为在具体应用中选择合适的测距技术提供参考。

为了评价测距误差对定位精度地影响，在仿真实验中利用高斯误差分布模型，在不同的测距误差条件下，做了一系列的实验。

图 4-8 给出了在网络连通度为 12.5，锚节点比例为 10%的条件下，定位误差随测距误差增大的情况。

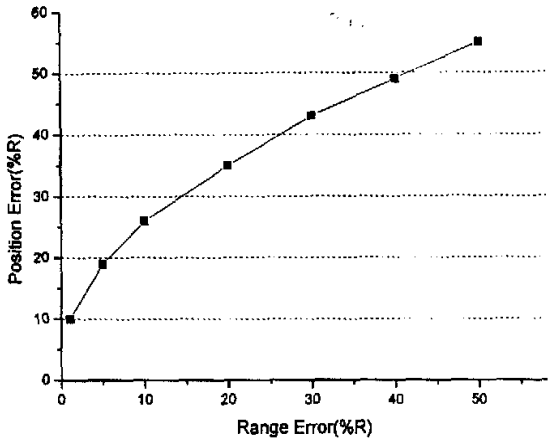


图 4-8 测距误差对定位精度的影响

从图中可以看到，当测距误差增大时，定位误差也随之显著地增大，当测距误差达到 50% 时，平均定位误差达到了 55%。

4.3.3 节点密度对节点定位的影响

节点密度可以使用网络的平均连通度来表示，节点的连通度就是可以于节点直接通讯的邻居节点的数量，而网络的平均连通度指网络中所有节点连通度的平均值。如果节点的无线通讯半径不变，当固定监测区域内节点的数量增加时，网络的平均连通度随之提高。

图 4-9 给出了当网络平均连通度提高时，节点定位误差随之下下降的趋势。图中所示的锚节点比率为 10%，测距误差为 10%。

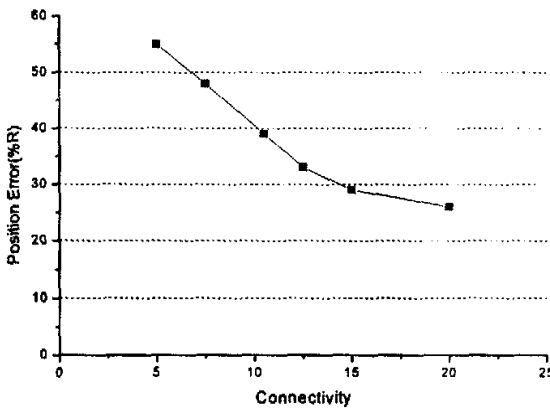


图 4-9 网络连通度对定位精度的影响

可以看到, 当网络平均连通度为 7.5 时, 平均定位误差为 47%R, 当网络平均连通度提高到 15 时, 平均定位误差降低到了 28%。说明节点密度对定位精度也有显著的影响。而且定位误差下降的趋势在开始的时候更为显著, 而当平均连通度高于 12.5 后, 下降的趋势变得较为平缓。

#### 4.3.4 锚节点比率对定位的影响

锚节点比率指网络中锚节点占网络中所有节点的比率。由于锚节点需要人工部署或配备 GPS 等定位设备, 锚节点数量的增加必然会增加网络的成本, 我们希望网络中所需要的锚节点数量尽可能的少。所以很有必要研究锚节点比率对节点定位的影响。图 4-10 给出了当网络平均连通度为 12.5, 测距误差为 10% 时, 增加锚节点密度, 定位误差的变化情况。

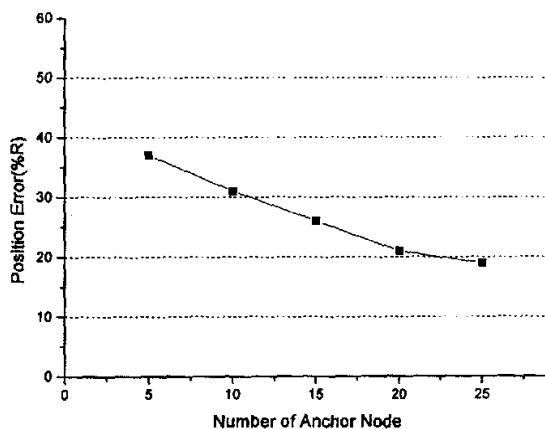


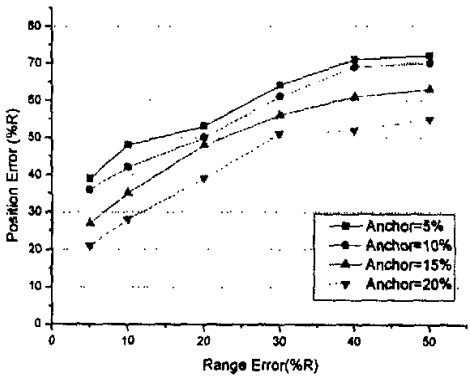
图 4-10 锚节点数量对定位精度的影响

从图中可见, 当锚节点数量增加时, 网络的平均定位误差随之下降, 当锚节点密度为 5% 时, 定位误差达到了 37%, 而当锚节点密度增加到 20% 时, 定位误差降低到了 21%。

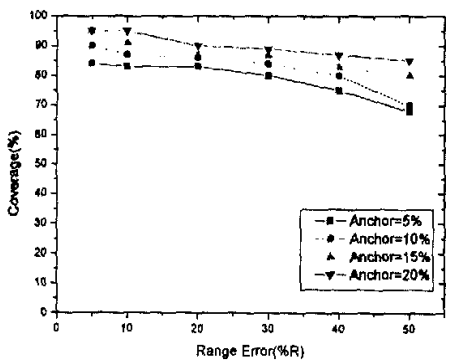
#### 4.3.5 算法性能的综合分析

节点定位算法能受到测距误差、网络平均连通度、锚节点比率等各项因素的影响, 为了综合评价算法的性能, 分别在  $60 \times 60\text{M}$ 、 $50 \times 50\text{M}$  和  $40 \times 40\text{M}$  的正方形二维区域随机部署 100 个节点(网络平均连通度分别为 7.5、10.5 和 12.5), 在 5%、10%、20%、30%、40%、50% 的测距误差条件和 5%、10%、15%、20% 的锚节点比率下, 分别对算法进行仿真运算多次, 检验算法的定位精度和定位

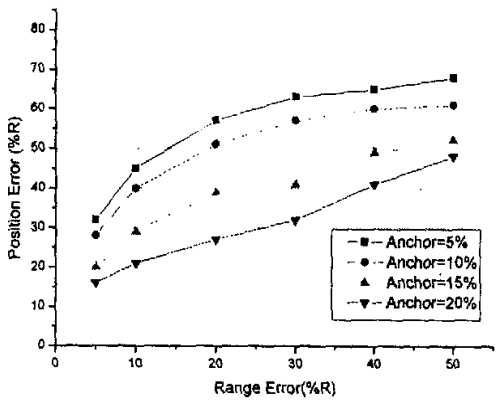
覆盖率。



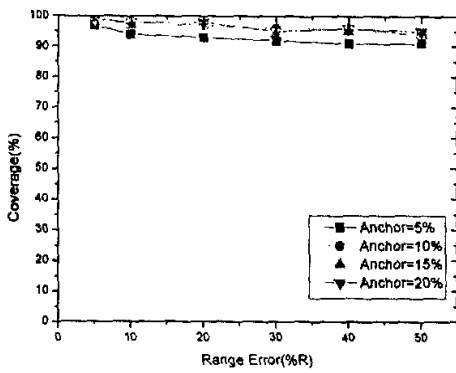
(a)Connection=7.5 测距误差对定位精度



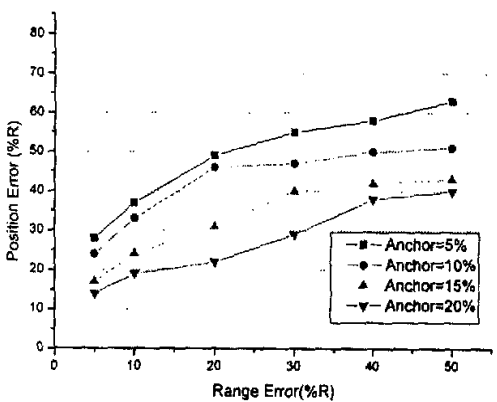
(b)Connection=7.5 测距误差对定位覆盖率



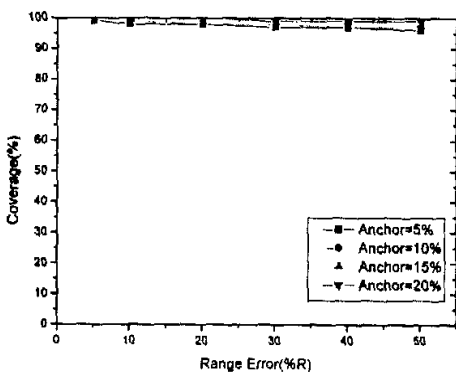
(c)Connection=10.5 测距误差对定位精度



(d)Connection=10.5 测距误差对定位覆盖率



(e)Connection=12.5 测距误差对定位精度



(f)Connection=12.5 测距误差对定位覆盖率

图 4-11 各种因素条件下算法的定位性能

如图 4-11, 当网络平均连通度为 7.5, 测距误差为 10%, 锚节点比率为 20% 时, 定位误差达到了 28%, 而定位覆盖率达到了 96%; 当网络平均连通度为 12.5, 测距误差为 10%, 锚节点比率为 20% 时, 定位误差达到了 19%, 而定位覆盖率接近了 100%。

仿真结果表明, 由于算法中采用了加权最小二乘估计, 成功地抑制了迭代定位过程中累计误差的影响, 降低了定位误差, 达到了较好的定位精度。而在三边测量定位、多边测量定位和 Euclidean 算法的基础上, 结合两个锚节点定位算法, 使得在一般情况下, 网络中绝大多数节点都能够进行定位, 大大提高了算法的定位覆盖率。



## 第五章 定位算法在硬件平台上的实现

### 5.1 节点硬件环境

无线传感器节点一般应满足以下要求:

**小尺寸和低功耗:**无线传感器网络具有大范围、高密度监测的特点,因此需要节点应该在体积上足够小,在某些场合甚至需要能够小到不容易让人察觉的程度,以完成一些特殊任务。尺寸和电源限制了节点基本设备的处理,存储和通讯能力,这对软件系统也提出了更高的要求。

**并发性和实时性:**传感器节点需要快速处理和传递消息,以保证信息被即时的捕获、处理,并通过网络传输,这就要求节点能够并发处理各种事件,并有一定的实时性要求。

**健壮性:**传感器节点一般应用于室外环境,需要在大部分时间中能够正常工作。在给定的温度、湿度、压力等外部条件下,无线传感器网络节点的处理器的、无线通信模块、电源模块要保证正常的功能,一方面系统在各种恶劣的气候下不会损坏,另一方面传感器部件能尽量接近检测环境以获得最真实的参数信息。

针对上述的关于传感器网络节点的基本特性,我们研制了基于 ATmega128L 处理器和 CC1000 射频芯片的传感器节点,并以此作为我们研究工作的原型实验系统。

ATmega128L 具有 128K 字节的系统内可编程 Flash、4K 字节的 EEPROM、4K 字节的 SRAM、53 个通用 I/O 口线、32 个通用工作寄存器、实时时钟 RTC、4 个灵活的具有比较模式和 PWM 功能的定时器/计数器(T/C)、两个 USART、面向字节的两线接口 TWI、8 通道 10 位 ADC(具有可选的可编程增益)、具有片内振荡器的可编程看门狗定时器、SPI 串行端口、与 IEEE 1149.1 规范兼容的 JTAG 测试接口(此接口同时还可以用于片上调试),以及六种可以通过软件选择的省电模式。空闲模式时 CPU 停止工作,而 SRAM、T/C、SPI 端口以及中断系统继续工作;掉电模式时晶体振荡器停止振荡,所有功能除了中断和硬件复位之外都停止工作,寄存器的内容则一直保持;省电模式时异步定时器继续运行,以允许用户维持时间基准,器件的其他部分则处于睡眠状态;ADC 噪声抑制模式时 CPU 和所有的 I/O 模块停止运行,而异步定时器和 ADC 继续工作,以减少 ADC 转换时的开关噪声; Standby 模式时振荡器工作而其他部

分睡眠,使得器件只消耗极少的电流,同时具有快速启动能力;扩展 Standby 模式则允许振荡器和异步定时器继续工作。器件是以 Atmel 的高密度非易失性内存技术生产的。片内 ISP Flash 可以通过 SPI 接口、通用编程器,或引导程序多次编程。引导程序可以使用任何接口来下载应用程序到应用 Flash 存储器。在更新应用 Flash 存储器时引导 Flash 区的程序继续运行,实现 RWW 操作。通过将 8 位 RISC CPU 与系统内可编程的 Flash 集成在一个芯片内, ATmega128L 为许多嵌入式控制应用提供了灵活而低成本方案。

节点的处理能力在 10MIPS 以上,内置微型化的实时嵌入式系统,具有独立的数据分析和处理能力,通信速率大于 50Kbps 的短距离无线通信能力。可扩展的传感器接口,支持 I2C 和 UART,为温度,湿度,加速度,地震波,噪声,光强度等各种传感器提供必要的物理接口。节点还具有 RS232,ISP,SPI 三种接口,为与 PC 机通信以及程序,硬件的调试增加了手段。

传感器板集成温度、声音、磁场、光强、震动、加速度等传感器,能够分别和集中采集温度、声音、磁场、光强、震动、加速度等相关数据。

## 5.2 操作系统与开发语言

### 5.2.1 TinyOS 操作系统

美国加州大学的伯利克分校针对无线传感器网络研发了 TinyOS (Tiny Micro threading Operating System) 操作系统,它是一种基于组件 (Comonent-Based) 的架构方式,在无线传感器网络的研究和应用领域得到了比较广泛的使用。

TinyOS 包含了经过特殊设计的组件模型,其目标是高效率的模块化和易于构造组建应用软件。TinyOS 的核心程序非常小 (仅 400Bytes 左右),非常适合传感器存储资源有限的限制,能够让操作系统有效的运行在无线传感器网络上并执行相应的管理工作。这种组件模型允许应用程序开发人员方便快捷地将独立的组件组合在一起,以完成特定系统的功能。TinyOS 中的组件通常可以分为三类:硬件抽象组件、合成组件、高层次软件组件。硬件抽象组件将物理硬件映射到 TinyOS 组件模型;合成硬件组件模拟高级硬件行为;高层次软件模块完成控制、路由以及数据传输等。

TinyOS 的应用运行环境一般有以下几部分实现:Main 组件 (调度程序),一个可选择的系统组件集合 (仅仅是应用需要的组件)以及为应用定义的组件。TinyOS 的这种体系结构使得用户可以快速便利地实现应用。用户不需要关心

HPL 层的具体实现细节和节点硬件所提供的功能, 只需要使用系统组件层提供的服务来满足具体的应用需求。硬件描述层的独立抽象, 增强了 TinyOS 的移植性。通过硬件抽象层对硬件平台合理的描述, 可使操作系统内核基本和具体的硬件无关, 从而容易地实现不同平台间的移植, 这样就简化了嵌入式 OS 内核的移植工作, 进而说明了 TinyOS 可以更好的满足传感器网络节点硬件的变化。TinyOS 的组件的层次结构如下图 5-1 所示。

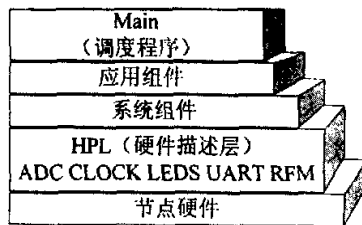


图 5-1 TinyOS 结构图

### 5.2.1 nesC 开发语言

nesC 是一门为网络嵌入式系统设计的编程语言。通过实现一个包含事件驱动执行、弹性并发型和面向组件程序设计等特征的编程模式, 来满足这个领域的程序设计的特定要求。nesC 的编译器进行数据竞争检测(提高可靠性)、积极的函数内联(降低资源消耗)等整体程序分析, 简化了应用程序的开发, 缩小了代码大小, 并且减少了许多潜在诱发错误的因素。

nesC 具有以下这些不同与其他语言的特点:

1. nesC 定义了一个支持事件驱动系统的组件模型: 这个模型提供的双向接口简化了事件流, 支持弹性的软硬件接口, 并且有效地避免了虚函数和动态组件构造。

2. nesC 定义了一个简单但是表现力很强的并发模型, 还有可扩充的编译期分析: nesC 的编译器会在编译期间检测绝大多数数据竞争。这个组合使得应用程序可以在资源有限的情况下实现高度并发操作。

3. 在提高可靠性、控制代码大小的精确程序分析和建立实际应用程序的表达能力之间, nesC 提供了一个独一无二的平衡点: 除了静态数据竞争检测之外, nesC 的编译器还执行静态组件实例化操作, 整体程序内联, 还有减少死代码。函数指针和动态内存分配等特征支持复杂的应用程序和牢固的用户通信, 但是它们会阻碍静态分析, 所以我们在 nesC 中禁止采用这些特征。

nesC 目前已经被用作 Berkeley 大学 TinyOS 的编程语言。一些重要的传感器网络应用程序, 如传感器网络查询引擎 TinyDB, 使得传感器网络能够快

速重新编程的小虚拟 Mate 等,也已经用 nesC 实现出来了。已经有很多传感器网络研究小组采用 nesC。经验和该语言的评估结果表明 nesC 能够高效地支持这个新类型的深度联网系统所需要的复杂、并发的编程风格。

nesC 的应用程序通过编写、组合组件(components)构造而成。一个组件提供和使用接口(interface)。组件的使用者只能通过这些接口访问组件。一个接口通常代表某项服务,并且会有一个接口类型说明。nesC 中的接口是双向的:它们包括命令(command)和事件(event),两者都是必要的函数。接口的提供者要实现接口的命令,而事件由接口的使用者来实现。将命令请求和事件响应放在同一个接口中,轻而易举就实现了分段操作。

在组件中,将接口类型的定义和接口的使用分离有助于定义标准接口让组件有更高的可重用性和更大的弹性。一个组件可以提供和使用同种接口类型,也可以多次提供同一个接口类型。组件同时是抽象软硬件之间的边界一个简明的方法。双向接口使得支持硬件中断变得非常简单。相反的,基于过程调用的单行接口不得不采用硬件轮询,或者采用两个独立的接口分别处理硬件操作和相应的中断。

nesC 中有两种组件:module 和 configuration。Module 类型的组件提供应用程序代码,实现一个或多个接口;configuration 类型的组件用于将别的组件绑定在一起,将组件使用的接口连接到其它组件提供的接口上。NesC 的每个应用程序都有一个顶层的 configuration 类型的组件,这个组件将该应用程序使用的组件绑定在一起。

## 5.3 定位算法在节点上的软件设计与实现

### 5.3.1 DV-hop 定位算法

美国路特葛斯大学(Rutgers University)的 Dragos Niculescu 等提出的分布式 DV-Hop 定位算法是一种简单易实现的算法,该算法无需节点具有测距能力,而是利用网络中部分锚节点的信息,通过节点间的信息交换和协作,估算网络中的平均每跳距离,从而根据最短路径计算未知节点到锚节点的距离,最后利用三边或多边测量定位计算未知节点的坐标。

DV-Hop 算法由三个阶段组成。首先使用典型的距离矢量交换协议,使网络中所有节点获得距锚节点的跳数(distance in hops)。第二阶段,在获得其他锚节点位置和相隔跳距后,锚节点计算网络平均每跳距离,然后将其作为一个校正值(correction)广播至网络中。校正值采用可控洪泛法在网络中传播,这意味

着一个节点仅接受获得的第一个校正值，而丢弃所有后来者，这个策略确保了绝大多数节点从最近的锚节点接收校正值。在大型网络中，可通过为数据包设置一个 TTL 域来减少通信量。当接收到校正值后，节点根据跳数计算与锚节点距离。当未知节点获得与三个或更多锚节点的距离，则在第三阶段执行三边测量定位。

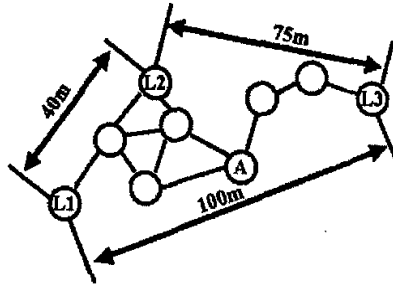


图 5-2 DV-hop 定位算法示意

如图 5-2，已知锚节点 L1 与 L2，L3 之间距离和跳数。L2 计算得到校正值（即平均每跳距离） $(40 + 75)/(2 + 5) = 16.42$ 。在上例中，假设 A 从 L2 获得校正值——则它与三个锚节点距离分别为 L1— $3 \times 16.42$ 、L2— $2 \times 16.42$ 、L3— $3 \times 16.42$ ，然后使用三边测量法确定节点 A 位置。

### 5.3.1 数据结构设计

定位程序使用的数据结构主要有 TOS\_Msg、MultihopMsg、CoordMsg:

#### (1) TOS\_Msg

TOS\_Msg 是通讯的基本数据结构，它包含的域包括目的地址，消息类型（即 AM 句柄号），长度，载荷等。最大载荷 TOSH\_DATA\_LENGTH 定义为 29 个字节。

```
typedef struct TOS_Msg
{
    uint16_t addr;
    uint8_t type;
    uint8_t group;
    uint8_t length;
    int8_t data[TOSH_DATA_LENGTH];
    uint16_t crc;

    uint16_t strength;
    uint8_t ack;
    uint16_t time;
}
```

```
uint8_t sendSecurityMode;
uint8_t receiveSecurityMode;
} TOS_Msg;
```

Addr	Type	Group	Length	Data	Crc					
2 字节	1 字节	1 字节	1 字节	29 字节	2 字节					

(2) MultihopMsg

MultihopMsg 包含了实现多跳协议的基本信息，包括源地址、起始地址、顺序号、多跳计数和最长 21 字节的载荷。MultihopMsg 封装在 TOS\_Msg 的 data 域中。

```
typedef struct MultihopMsg {
    uint16_t sourceaddr;
    uint16_t originaddr;
    int16_t seqno;
    uint8_t hopcount;
    uint8_t data[(TOSH_DATA_LENGTH - 7)];
} __attribute__((packed)) TOS_MHopMsg;
```

SourceAddr	Originaddr	Seqno	Hopcount	Data
2 字节	2 字节	2 字节	1 字节	21 字节

(3) CoordMsg

CoordMsg 中包含横坐标 x、纵坐标 y、节点号 src、跳数 distance\_metric 和平均每跳距离 correction。当广播坐标信息时 CoordMsg 直接封装在 TOS\_Msg 的 data 域中；当回传坐标信息时 CoordMsg 封装在 MultiHopMsg 的 data 域中，利用多跳路由协议回传坐标信息。

```
typedef struct CoordMsg {
    float x;
    float y;
    short src;
    float distance_metric;
    float correction;
} __attribute__((packed)) CoordMsg;
```

x	y	src	distance_metric	correction
4 字节	4 字节	2 字节	4 字节	4 字节

5.3.1 模块程序设计

节点程序主要分为两个部分：锚节点程序和未知节点程序。锚节点程序完

成广播自身坐标信息，并收集其他锚节点的信息，计算平均每跳距离，将计算结果封装到数据包中广播到其他节点，该模块的流程如图 5-3 所示。

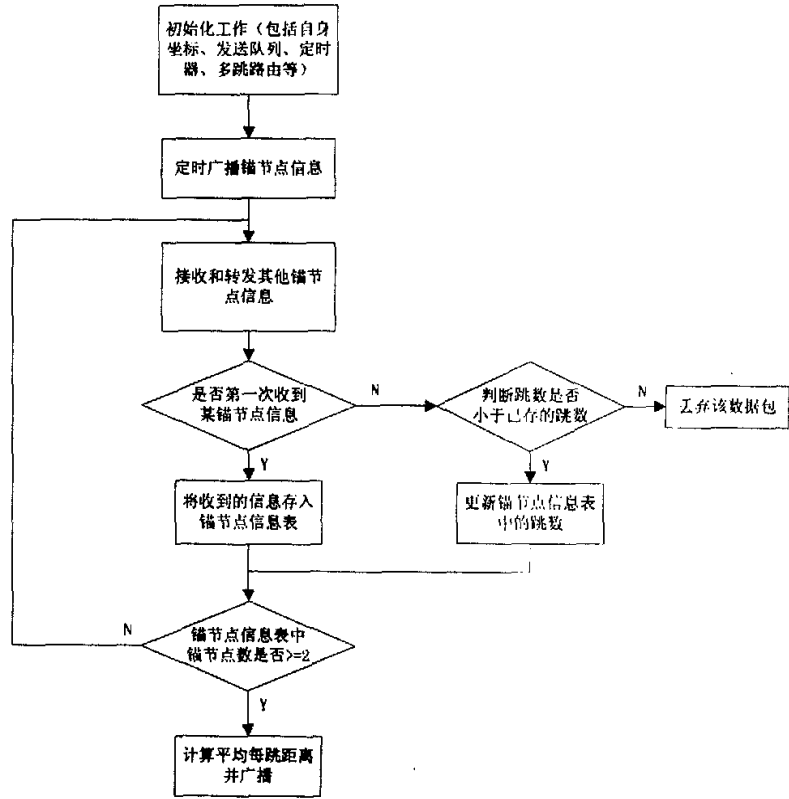


图 5-3 锚节点模块流程图

未知节点程序收集从锚节点传来的坐标及平均每跳距离信息，并以此利用最小二乘算法计算自身的坐标，将计算结果回传至网关或 PC 机，该模块流程如图 5-4 所示。

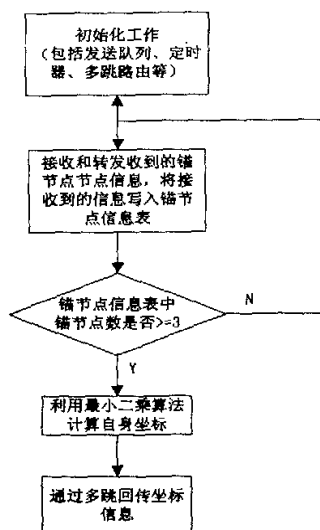


图 5-4 未知节点模块流程图

## 5.4 小结

目前对定位系统实现方面的研究还比较少, 还没有一套完整的定位系统评价体系。本章介绍了无线传感器网络节点的硬件环境以及相关的操作系统与开发平台, 并在此平台上实现了 DV-hop 定位算法, 该算法在理论上比较简单并且易于实现, 但其定位性能受网络拓扑的影响较大。该定位系统的开发为将来进一步实现自有的定位算法, 提供比对研究奠定了良好的基础。



## 第六章 总结与展望

### 6.1 总结

本文在阐述了无线传感器网络概念和特点的基础上,着重进行了无线传感器网络分布式节点定位算法的研究,通过对现有几种典型的分布式节点定位算法的深入分析研究,指出在多种条件下衡量,没有一种算法是最优的。对于不同的设计目标,每一个算法在某一特定的条件下会优于其他的算法,而且,每一种算法还有一定的优化与提升空间。

为了提高定位算法的性能,结合了相关数学原理,从提高定位精度方面,引入了加权最小二乘估计的方法,建立了加权定位算法的方案,以抑制在迭代定位过程中累积误差的影响;从提高定位覆盖率方面,建立了利用两个锚节点和一个连通限制条件定位的算法模型,与三边测量定位(trilateration)和多边测量定位(multilateration)相结合,确立了一种新的分布式节点定位算法的框架。通过在 NS-2 网络仿真平台下大量的仿真实验,仿真结果表明,该定位算法具有良好的定位性能,定位精度和定位覆盖度等各项指标达到了预期的设计目标。

在定位系统的实用性研究方面,我们在实际的硬件节点上实现了 DV-hop 定位算法,为将来进一步实现自有的定位算法,奠定了进一步研究的良好基础。

### 6.2 展望

近年来,无线传感器网络这一新兴的研究领域已成为一个巨大的研究热点,在节点定位方面的研究也取得了丰硕的成果。但总体来说,节点定位方面的研究大多还处于理论研究方面,在实用的定位系统的研究方面虽然有了一定的初步成果,但还是处于起步阶段。在未来的研究中,研究的重心将逐渐从单纯的理论研究和仿真实验向实用的定位系统实现方面转移。下面是一些可能的研究方向。

测距技术的研究,包括研究体积更小、功耗更低、测量精度更高的测距设备,满足无线传感器网络节点距离测量的要求,以及为特点的测距技术和设备建立实用的测距模型,研究校正方法等。

基于 UWB 技术的定位系统的研究也是一个非常有前景的研究方向, UWB 基于 UWB 技术的定位系统的研究也是一个非常有前景的研究方向, UWB

良好的抗干扰能力和对障碍物的穿透能力，特别是其精确的测距能力为定位系统的设计提供了良好的支撑，是一种极具潜力的技术。

基于组件技术的节点定位模块设计，提供标准的定位 API，可以实现不同的定位算法组件，为上层应用提供可选的、灵活的定位组件模块，同时可以灵活配置不同的测距模块，这也是一个需要解决的非常有挑战性的课题。

## 参考文献

- [1] Akyildiz I.F, Su W, Sankarasubramaniam Y, Cayirci E. Wireless sensor network: A survey. *Computer Networks*. 2002, vol.38, pp.393-422.
- [2] Chee-Yee Chong and Srikanta P.Kumar. Sensor Networks: Evolution, Opportunities and Challenges. *Proceeding of the IEEE*, Aug. 2003, vol.91, no.8.
- [3] D.Estrin, R.Govindan, J.Heidemann, and S.Kumar. Next century challenges: Scalable coordination in sensor networks. *Proc. Int. Conf. Mobile Computing and Networking (MOBICOM)*. 1999. pp.263-270.
- [4] Ten emerging technologies that will change the world. *Technology Review*. Feb.2003, vol.106, no.1, pp.22-49. <http://www.techreview.com/articles/emerging0203.asp>
- [5] ALERT. <http://www.alertsystem.org>.
- [6] Bonnet P, Gehrke J, Seshadri P. Querying th physical world. *IEEE Personal Communication*, 2000. 10-15
- [7] Noury N, Herve T, Rialle V, Virone G, Mereier E. Monitoring behavior in home using a smart fall sensor. In: *Proceedings fo the IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine and Biology*. 2000. 607-610.
- [8] Sensor Webs. <http://sensorwebs.jpl.nasa.gov/>.
- [9] Shih E., cho S, Ickes N, Min R, Sinha A, Wang A. Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks. In: *Proceedings of the ACM MobiCom 2001*. Rome: ACM Press, 2001. 272-286.
- [10] DARPA Program, <http://dtsn.darpa.mil/ixo/programs.asp>
- [11] Center for Embedded Networked Sensing, <http://www.cens.ucla.edu/>
- [12] <http://robotics.eecs.berkeley.edu/~pister/SmartDust/>
- [13] Chris Savarese, Jan M.Rabaey, Jan Beutel. Locationing in Distributed Ad-Hoc wireless sensor network. In: *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal (ICASSP '01)*, Salt Lake, USA: IEEE Computer Society, 2001.5, vol.4: 2037-2040
- [14] Srdjan Capkun, Maher Hamdi and Jean-pierre Hubaux. GPS-Free Positioning in Mobile Ad-Hoc Networks. In: *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, Maui, Hawaii, USA: IEEE Computer Society, 2001.1, 3481 -3490
- [15] Lance Doherty, Laurent El Ghaoui and Kristofer S.J. Pister. Convex Position Estimation in Wireless Sensor Networks. In: *Proceedings of Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2001)*, Anchorage, AK, USA: IEEE Computer and Communications Societies, 2001.4, vol.3: 1655 -1663
- [16] Seapahn Meguerdichian, Farinaz Koushanfar, Miodrag Potkonjak, and Mani B. Srivastava. Coverage problems in wireless ad-hoc sensor networks. In: *Proceedings of Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2001)*, Anchorage, Alaska, USA: IEEE Computer and Communications Societies, 2001.4, vol.3: 1380-1387
- [17] Nirupama Bulusu, John Heidemann, and Deborah Estrin. Adaptive beacon placement. In: *Proceedings of the 21st International Conference on Distributed Computing Systems*, Phoenix, AZ, USA: IEEE Computer Society, 2001.4, 489-498

- [18] 王福豹, 史龙, 任丰原. 无线传感器网络中的自身定位系统和算法[J]. 软件学报, 2005. 5, Vol. 16, 857-868.
- [19] 史龙, 王福豹, 段渭军, 任丰原. 无线传感器网络 Range-Free 自身定位机制与算法[J]. 计算机工程与应用, 2004. 23, 127-130.
- [20] Nirupama Bulusu, John Heidemann, Deborah Estrin. GPS-less Low Cost Outdoor Localization for Very Small Devices [J]. IEEE Personal Communications, 2000.10, Vol.7(5): 28-34
- [21] Simic SN, Sastry S, Distributed localization in wireless ad hoc networks [R]. Tech. report, UC Berkeley, 2002, Memorandum No. UCB/ERL M02/26.
- [22] Dragos Nicolescu and Badri Nath. Ad-Hoc Positioning Systems (APS) [C]. In: Proceedings of 2001 IEEE Global Telecommunications Conference (IEEE GLOBECOM'01), San Antonio, TX, USA: IEEE Communications Society, 2001.11, vol.5: 2926-2931
- [23] Dragos Niculescu, Badri Nath. DV based positioning in ad hoc networks [J]. Journal of Telecommunication Systems, 2003.1-4, Vol.22 (No.1/4), 267-280
- [24] Gene H Golub, Charles F Van Loan. Matrix Computations[M]. 3rd Edition, The Johns Hopkins University Press, 1996.
- [25] Savarese C, Rabay J, Langendoen K. Robust Positioning Algorithms for Distributed Ad-Hoc Wireless Sensor Networks [C]. In: Carla Schlatter Ellis ed. Proceedings of the USENIX Technical Annual Conference, Monterey, CA, USA: USENIX Press, 2002.6, 317-327
- [26] Yi Shang, Wheeler Ruml, Ying Zhang, Markus P J Fromherz. Localization from mere connectivity [C]. In: Proceedings of the fourth ACM international symposium on Mobile ad hoc networking & computing (MOBIHOC 2003), Annapolis, Maryland, USA: ACM Press, 2003.6, 201-212
- [27] Tian He, Chengdu Huang, Brian M Blum, John A Stankovic, Tarek Abdelzaher. Range-Free Localization Schemes in Large Scale Sensor Networks [C]. In: Proceedings of the 9th annual international conference on Mobile computing and networking (MobiCom), San Diego, California, USA: ACM Press, 2003.9, 81~95
- [28] Paramvir Bahl, Venkata N Padmanabhan. RADAR: An in-building RF-based user location and tracking system [C]. In: Proceedings of Infocom'2000, Tel Aviv, Israel: IEEE Computer and Communications Societies, 2000.3, vol.2: 775-584
- [29] Radhika Nagpal, Howard Shrobe, Jonathan Bachrach. Organizing a Global Coordinate System from Local Information on an Ad Hoc Sensor Network [C]. In: Zhao Feng, Guibas Leonidas eds. Proceedings of the 2nd International Workshop on Information Processing in Sensor Networks (IPSN '03), Palo Alto, CA, USA: Springer-Verlag Press, 2003.4, 333-348
- [30] Andreas Savvides, Chih-Chieh Han, Mani B Srivastava. Dynamic Fine-Grained Localization in Ad-Hoc Networks of Sensors [C]. In: Proceedings of Mobile Computing and Networking (MOBICOM'01), Rome, Italy: ACM Press, 2001.7, 166-179
- [31] Andreas Savvides, Heemin Park, Mani B Srivastava. The Bits and Flops of the N-Hop Multilateration Primitive for Node Localization Problems [C]. In: Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Application (WSNA'02), Atlanta, Georgia, USA: ACM Press, 2002.9, 112-121

- [32] Dragos Niculescu, Badri Nath. Ad Hoc Positioning System (APS) using AoA [C] . In: Proceedings of The 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003). San Francisco, CA, USA: IEEE Computer and Communications Societies. 2003.3. 1734 –1743
- [33] Seapahn Meguerdichian, Sasa Slijepcevic, Vahag Karayan, Miodrag Potkonjak. Localized Algorithms In Wireless Ad-Hoc Networks: Location Discovery And Sensor Exposure [C] . In: Proceedings of the 2001 ACM International Symposium on Mobile Ad Hoc Networking & Computing, Long Beach, USA: ACM Press, 2001.10, 106–116
- [34] Xiaoli Li, Hongchi Shi, Yi Shang. A Map-growing Localization Algorithm for Ad-hoc Wireless Sensor Networks [C] . In: Proceedings of the Tenth International Conference on Parallel and Distributed Systems(ICPADS'04)
- [35] Mihail L Sichitiu, Vaidyanathan Ramadurai. Localization of Wireless Sensor Networks with a Mobile Beacon [C] . In: IEEE International Conference on Mobile Ad-hoc and Sensor Systems. 2004, 174-183
- [36] WOO Cheol Chung, Dong Sam Ha. An Accurate Ultra-Wideband (UWB) Ranging for Precision Asset Location [C] . Proceedings of the IEEE Conference on Ultra Wideband Systems and Technologies Reston(VA, USA), 2003. Piscataway(NJ, USA): IEEE, 2003.389-393
- [37] J.Y. Lee, R.A. Scholtz. Ranging in a dense multipath environment using an UWB radio link [J] . IEEE Journal on Selected Areas in Communications, vol.20, no.9. 2002. 1677-1683
- [38] Damien B. Jourdan, John J. Deyst, Jr., Moe Z. Win. Monte Carlo Localization in Dense Multipath Environments Using UWB Ranging [C] . Proceedings of the IEEE International Conference on Ultra-Wideband (ICU 2005). Zurich, Switzerland, September 2005.
- [39] K. Yu, I. Oppermann. Performance of UWB position estimation based on TOA measurements [J] . Proc. Joint UWBST & IWUWBS,(Kyoto, Japan), 2004.
- [40] K. Yu , I. Oppermann. Uwb positioning for wireless embedded networks [C] . Proc. IEEE RAWCON conference, USA,Atlanta, 2004.
- [41] Jean-Philippe Montillet, Timo Braysy, Ian Oppermann.. Algorithm for nodes localization in wireless Ad-hoc networks based on cost function [C] . In: Proc. International Workshop on Wireless Ad-hoc Networks (IWWAN 2005), London, UK, May 23 - 26, 2005
- [42] Kamin Whitehouse, Chris Karlof, Alec Woo, Fred Jiang, David Culler. The Effects of Ranging Noise on Multihop Localization: an Empirical Study [C] . The Fourth International Conference on Information Processing in Sensor Networks (IPSN '05). Los Angeles, California. April 25-27, 2005.
- [43] Tufan C. Karalar, Shunzo Yama shita, Michael Sheets, Jan Rabaey. A Low Power Localization Architecture and System for Wireless Sensor Networks [C] . Signal Processing Systems, 2004. SIPS 2004. IEEE Workshop on 2004 Page(s):89 – 94
- [44] Jeffrey Hightower, Gaetano Borriello, Roy Want. SpotON: An indoor 3D Location Sensing Technology Based on RF Signal Strength. Technical Report UW CSE 00-02-02, University of Washington, Department of Computer Science and Engineering, Seattle, WA, USA 2000.2
- [45] Nissanka B.Priyantha, Anit Chakraborty, Hari Balakrishnan. The Cricket Location-Support System. In: Proceedings of 6th ACM International Conference on Mobile Computing and Networking (ACM MOBICOM), Boston, MA, USA: ACM Press, 2000.8, 32-43
- [46] 史龙. 无线传感器网络自身定位算法研究[D]. 硕士论文, 西北工业大学, 2005. 38-42.
- [47] 陈希儒, 王松柱. 线性模型中的最小二乘法[M]. 上海科学技术出版社, 2003. 3, 1-26.

- 
- [48] 沈凤麟, 叶中付, 钱玉美, 信号统计分析与处理[M]. 中国科学技术大学出版社, 2001. 3, 365-373.
  - [49] 徐雷鸣, 庞博, 赵耀. NS 与网络模拟[M]. 人民邮电出版社. 2003. 2-8.
  - [50] Wendi Beth Heinzelman, Application-Specific Protocol Architectures for Wireless Networks. MIT, 2000, 134-144
  - [51] Chalermek Intanagonwiwat, Directed Diffusion: An Application-Specific and Data-Centric Communication Paradigm for Wireless Sensor Networks. University of Southern California, 2002, 44-57
  - [52] Ian Downard, Simulating Sensor Networks in NS-2, Computers:Computer Programming and Software, May 2004 .

## 在校期间论文发表情况

1、王建刚，王福豹，段谓军. “无线传感器网络分布式节点定位算法研究”. 《计算机应用》，（国际标准刊号：ISSN1001-9081 国内统一刊号：CN51-1307/TP），已出版，2005 年第 25 卷第 11 期

2、王建刚，王福豹，段谓军. “加权最小二乘在无线传感器网络定位中的应用”. 《计算机应用研究》，（国际标准刊号：ISSN1001-3695 国内统一刊号：CN51-1196/TP），已录用（于 2006 年 10 月发表）

## 致 谢

本文是在导师王福豹教授的指导下完成的。王老师对论文的选题、开题及开发方案等方面都给予我很多参考意见和指导，并在百忙中对论文进行认真的审阅和修改。王老师坚实的专业知识、严谨的治学态度和孜孜不倦的工作精神，对我产生了深刻的影响，并将使我受益终身。在此，向精心培养与关心我的导师致以崇高的敬意和深深的谢意。

感谢段渭军老师和肖琳老师在学习和生活上对我的关心和帮助，以及网络与教育中心的其他各位老师和工作人员，他们为我们提供了良好的学习和科研环境，帮助我们解决了很多实际遇到的问题，使科研工作得以顺利开展。

感谢教研室所有的同学们，我们共同创造了和谐温暖的集体，在以后的日子里，我会常常怀念他们。

在此向曾经帮助过我的室友和所有的朋友表示诚挚的感谢。

最后特别感谢我的父母多年来含辛茹苦的养育之恩，感谢我的妻子在我攻读硕士学位期间所给予的理解、支持与付出。



# 西北工业大学

## 学位论文知识产权声明书

本人完全了解学校有关保护知识产权的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属于西北工业大学。学校有权保留并向国家有关部门或机构送交论文的复印件和电子版。本人允许论文被查阅和借阅。学校可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。同时本人保证，毕业后结合学位论文研究课题再撰写的文章一律注明作者单位为西北工业大学。

保密论文待解密后适用本声明。

学位论文作者签名：王建刚

指导教师签名：王

2006年3月1日

2006年3月1日

---

# 西北工业大学

## 学位论文原创性声明

秉承学校严谨的学风和优良的科学道德，本人郑重声明：所呈交的学位论文，是本人在导师的指导下进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容和致谢的地方外，本论文不包含任何其他个人或集体已经公开发表或撰写过的研究成果，不包含本人或其他已申请学位或其他用途使用过的成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式表明。

本人学位论文与资料若有不实，愿意承担一切相关的法律责任。

学位论文作者签名：王建刚

2006年3月1日