

传感器网络与协同控制

克里斯特斯·g. 卡珊德拉和李玮

信息与系统工程中心制造工程系,
美国马萨诸塞州布鲁克林波士顿大学, 邮编 02446

本文从系统和控制理论的角度对传感器网络进行了教程式的概述。我们确定了关键的传感器网络设计和运行控制问题, 并提出了迄今为止已经提出的解决方案。这些包括部署、路由、调度和电源控制。在移动节点的情况下, 传感器网络被要求执行“任务”。我们提出了两类任务的求解方法, 包括随机任务空间和合作控制: 奖励最大化任务和覆盖控制任务。最后, 我们概述了与传感器网络以及通信、计算和控制的融合相关的一些基础研究问题。

关键词: 协同控制; DistributedControl 优化; 传感器网络

1. 正式介绍

传感器网络由一组(可能是移动的)传感设备组成, 这些设备可以通过无线通信协调它们的行动, 并旨在执行任务, 例如在特定区域(通常称为“任务空间”)内进行探测、监视或监控和跟踪“目标点”。收集的数

电子邮件: wli@bu.edu

致: C.G. 卡珊德拉. 电子邮件: cgc@bu.edu

寻求最大化适当定义的“网络寿命”的控制机制。这种机制的一部分可能涉及开关节点以便保存它们的能量

据随后被进一步处理, 通常支持更高级别的决策过程。

这种网络中的节点通常是不均匀的, 它们具有有限的机载资源(例如。功率和计算能力), 并且它们可能受到通信限制。应当指出, 传感器网络在许多关键方面不同于传统的通信网络。首先, 它们允许我们与物理世界互动, 而不仅仅是计算机、数据库或人类生成的数据。通过在这样的网络中插入决策和控制功能, 人们可以设想关闭远程进程上的循环, 否则将无法访问。因此, 传感器网络有望实现通信、计算和控制的长期融合[6]。第二, 这样的网络中至少一些节点是“活动的”, 例如, 它们执行感知过程或者它们是可移动的; 因此, 它们具有动态特性, 使得传感器网络作为一个整体成为一个具有挑战性的动态系统。此外, 节点通常较小且不昂贵, 通常在不利的随机环境中以有限的资源运行。这意味着优化传感器网络的设计和运行是一种真正的需求, 而不仅仅是一种奢侈。此外, 节点有限的计算能力通常使得分布式控制或优化方法不可或缺。最后, 在测量传感器网络的性能时, 度量标准可能与标准通信网络中使用的标准非常不同, 这就产生了新类型的问题。例如, 由于能量有限, 我们认识到节点的生命是有限的, 我们经常

2005年5月1日收到; 2005年6月15日接受。推荐人: E.F. Camacho, r. 节奏, s. 尤尔科维奇, P.J. 弗莱明。

或者寻找方法来周期性地补充它们的能量供应。当节点移动时, 还需要机制来确定节点在任务空间上的期

望轨迹, 并且合作控制开始发挥作用, 以满足特定的任务目标。

本文的主要目的是从系统和控制理论的角度对传感器网络进行全面的概述。为此, 我们确定了关键的传感器网络设计和运行控制问题, 并讨论了一些迄今为止已提出的解决方法。重点放在严格制定有趣的优化和合作控制问题, 描述解决方法的要点, 而不涉及技术细节, 只要这些可以在引用的参考文献中找到。论文的最后一部分讨论了“覆盖控制”问题, 我们之所以集中讨论这个问题, 是因为提出的公式和解决方案构成了一个新的贡献, 并且因为这个特殊的问题抓住了传感器网络中遇到的主要特征和控制挑战: 以非常规的方式定义网络性能的需要、协作控制的参与、需要分布式控制解决方案的节点的计算限制以及通信约束的作用。本文给出了仿真结果, 以说明各种控制方案和算法, 并指出有待进一步研究的问题。

论文的其余部分组织如下。第 2 节描述了传感器网络的基本结构, 并对其进行了分类, 以区分(1)具有固定、已知数据源和非移动节点的传感器网络, 以及(2)数据源未知且节点移动的传感器网络。在第 3 节中, 我们讨论了与第一种网络类型相关的主要问题, 包括部署、功率控制、路由和调度。在第 4 节中, 我们考虑带有移动节点的网络, 这些移动节点被要求执行“任务”。不同类型的任务导致定义不同类型的问题, 我们提出了两个这样的问题, 都涉及随机任务空间和合作控制: 奖励最大化任务和覆盖控制任务。在第 5 节中, 我们总结了一些与传感器网络以及通信、计算和控制的融合相关的基础研究问题, 以及一些迄今为止的发展所带来的更具体的问题。

2. 传感器网络结构

在其最基本的形式中, 传感器网络的主要目标是从由 R 表示的观察区域(“任务空间”)收集现场数据, 并将其路由到由 B 表示的基站(也称为“数据收集点”或“接收器”)。在任何时刻, R 中可能存在多个数据源(也称为“目标点”或简称为“目标”)。传感器网络中的节点相互协作, 以确保每一个源都被感知到, 并且收集到的数据被成功地转发到基站。在合作过程中, 传感器节点可能会陷入以下状态之一: (1) 传感: 传感节点使用集成传感器监控源, 将信息数字化, 对其进行处理, 并将数据存储在板载缓冲器中。这些数据最终将被发送回基站。(2) 中继: 中继节点从其他节点接收数据, 并将其转发到目的地。(3) 睡眠: 对于睡眠节点, 大多数设备功能要么关闭, 要么在低功耗模式下工作。休眠节点不参与感知或中继。然而, 它不时地“醒来”, 并监听通信信道, 以便回

答来自其他节点的请求。在接收到请求时, 可以发生到“感测”或“中继”的状态转换。(4) 死: 传感器网络不再有死节点。它要么耗尽了能量, 要么遭受了重大损失。一旦节点失效, 它就不能再进入任何其他状态。

一些传感器网络不再采用平面结构, 而是采用更具层次性的结构。在这种情况下, 除了传感器和基站, 还有节点充当簇头。这些节点通常具有更强大的数据处理和路由能力, 但代价是尺寸和成本。每个簇头负责一簇传感器节点, 这是通过对网络进行空间或逻辑划分而获得的。通过聚集从传感器节点发送的数据, 簇头改进了对簇区域的观察。然后, 它可能会产生一些后处理数据, 并将它们发送到基站。为了支持高速数据传输, 连接簇头和基站的链路可能具有更高的数据速率。图 1 展示了这个三层结构, 包括传感器节点、路由器和基站。

在当前的大多数应用中, 传感器网络具有低数据速率。按照 IEEE 802.15.4 标准, 每个通道的传输容量为 20kb/s (868 MHz 通道)、40kb/s (915MHz) 或 250kb/s (2.4GHz)。感测所需的数据速率可能更低, 这允许传感器节点同时作为数据源和中继。当前的

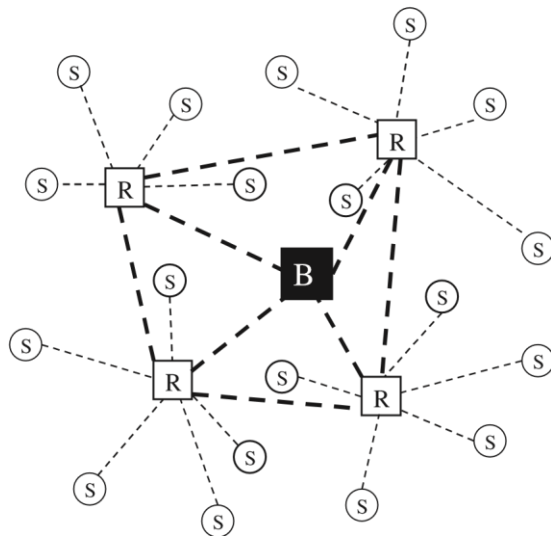


图. 1. 三层传感器网络结构。

传感器网络将节点之间的传输距离限制在 < 100 米 (通常为 5-10 米, 取决于环境), 任务空间内节点的空间密度在 0.1 和 20 节点/平方米之间变化。

让我们首先考虑传感器网络, 其中所有数据源都是固定的, 节点不是移动的。那么, 我们面临的第一个也是最基本的问题是部署问题, 即, 对节点进行定位, 以满足从源向基站成功传输数据的目标,

并最终优化某些网络性能指标。一旦完成，在不同的层(物理、数据链路、网络、传输等)会有许多操作控制问题。；有关全面概述，请参见参考文献[3, 10]。在这里，我们将把自己限制在一些最重要的问题上，在这些问题上，可以使用面向动态控制的方法，并且精确的数学公式是可能的。特别地，每个节点必须至少执行三个任务:(1)路由，即，。确定从某个节点 I 发送到基站的数据包的目的地节点；(2)调度，即，。确定用于传输可能不同类型的分组的精确定时机；以及(3)功率控制，即，。做出旨在以有利于整个网络的方式节约能源的决策。在下一节中，我们将更详细地讨论这些问题。

我们将考虑的第二类传感器网络是节点是移动的。这允许传感器网络执行任务，例如访问数据源目标(使得移动节点的数量可能比所寻找的目标的数量少得多)、搜索未知数据源以及跟踪可能也是移动的目标。在这种情况下，合作控制的作用变得至关重要，因为节点必须协作以在有限的资源和可能的关键时间限制下满足任务目标。在第 4.1 节中，我们将讨论一项基本任务，该任务要求移动节点在访问分配了不同(可能随时间变化)奖励的目标时进行合作；任务的目标是最大化可以收集的总回报。在第 4.2 节中，我们考虑了移动传感器网络的部署问题，在这种情况下，通常被称为覆盖控制问题。我们将详细介绍最近开发的解决这个问题的方案。

3. 具有固定数据源和无移动性的网络

3.1. 部署问题

传感器节点的部署可以是确定性的，也可以是随机的。后一种情况出现在侦察和探索等应用中，在这些应用中，传感器被随机投放到任务空间中，并且它们的确切位置不能被精确控制。在这种情况下，研究的重点是传感器密度和网络性能之间的关系。在参考文献中。[25]中，分析了在连接性约束条件下，临界通信功率如何随网络规模而变化，从而表明了网络寿命和密度之间的关系。这项工作参考文献中得到了扩展。[63]通过还考虑传感器节点的故障率。最近的工作[49]研究了异质传感器的部署(即，。正常节点和簇头),并获得最佳节点密度和能量，以保证期望的寿命，同时确保连通性和覆盖范围。

当预先知道任务空间的特征时(例如。在建筑物监控中)。这种情况下的基本问题包括(1)需要多少传感器节点来满足整体系统目标？(2)对于具有一定数量传

感器节点的给定网络，我们如何精确地部署这些节点以优化网络性能？(3)当数据源发生变化或部分网络出现故障时，我们如何调整网络拓扑和传感器部署？这些问题可以通过类似于广泛研究的设施位置优化问题的离线方案来解决。一种常用的方法是将任务空间离散化，并沿网格点放置传感器节点。由此产生的最优部署问题可以表述为一个线性规划。由于必须考虑所有的网格点和互连链路，这导致了显著的组合复杂性。或者，可以制定一个非线性优化问题，并寻求利用传感器网络的结构，以开发分解方法来解决它。在接下来的内容中，我们将回顾参考文献中介绍的这种方法。[40]。

，用前面讨论的传感器网络的源/基站结构，我们考虑驻留在点 s_2, R_2, m_1, \dots 和一个基站 B ，位置为 $x_0 \in \mathbb{R}^2$ 。每个数据源都有一个固定的位置和一个给定的数据速率， \dots, m 。为了在每个数据源收集数据，必须在其位置部署一个传感器。此外，由于数据源可能远离基站，并且距离可能超过无线电通信的范围，我们还需要部署一定数量的传感器节点作为中继。假设有 N 个活动传感器节点，每个节点的位置为 $x_k \in \mathbb{R}^2, k = 1, \dots, N$ 。设 $W = V, E, c, E_0$ 是一个具有底层有向图 $G = V, E_0$ 的流网络，其中 V 是节点集， E 是链路集。容量向量 $c_1, \dots, c_j \in \mathbb{R}^+$ 和成本向量 $c_1, \dots, c_j \in \mathbb{R}^+$ 是在每个带有 c_j, p_j 的链接 $j \in E$ 上定义的。每个链路 j 从节点 $s(j)$ 开始，到节点 $t(j)$ 结束， c_j 表示每单位数据的一些成本度量，这通常取决于节点位置。在这个流网络上，我们可以通过在每个链路上控制来自每个源的传感器节点 x_m 的位置和数据速率来公式化一个使总成本最小化的优化问题 \dots, m :

$$\begin{aligned} & \min_{x, f} \sum_{m \in M} \sum_{j \in E} c_j x_{jm} \\ & \text{s.t. } x \text{ 共同执行活动 } f_{jm} \text{ rmdim } i \in V, m = 1, \dots, m \\ & \quad \sum_{j \in E} x_{jm} = f_{jm} \quad \forall m \in M \\ & \quad \sum_{j \in E} x_{jm} \leq c_j \quad \forall j \in E \end{aligned}$$

$m \in \{1, \dots, M\}$
 $f_m = [f_{m1}, \dots, f_{mM}]^T$

在(1)中, 决策变量是 f_m 和 x_i , 其中流向量 f_m 的分量 f_{mj} 表示源自源 m 的链路 j 上的数据速率, $m \in \{1, \dots, M\}$ 。在流量平衡方程(2)中, α 因子是图 G 的节点-链路关联矩阵, $\alpha_{ij} = 1$ 如果弧 j 离开节点 i , $\alpha_{ij} = -1$ 如果弧 j 进入节点 i , 否则 $\alpha_{ij} = 0$ [57]:

$$\alpha_{ij} = \begin{cases} 1 & \text{如果弧 } j \text{ 离开节点 } i \\ -1 & \text{如果弧 } j \text{ 进入节点 } i \\ 0 & \text{否则} \end{cases}$$

$$\sum_{j \in E} \alpha_{ij} f_{mj} = 0, \quad \forall i \in V, m \in \{1, \dots, M\}$$
 $\alpha_{ij} = 1$
 $\alpha_{ij} = -1$

f_m 是数据源 m 的流量平衡向量

$$f_m = [f_{m1}, \dots, f_{mM}]^T$$
 $\alpha_{ij} = 1$
 $\alpha_{ij} = -1$

剩下的三个方程表示链路容量约束、流量非负性和 M 个源的固定位置。

虽然这个公式是通用的, 但我们的目标是确定最小功率配置。在这种情况下, 链路成本 c_{ij} 表示每单位数据消耗的能量。可以基于能量模型来指定函数 $c_{ij} = \alpha_{ij} x_i$, 该能量模型的关键参数是感测一个比特(E_s)、接收一个比特(E_r)和在距离 d 上发送一个比特所需的能量。通常假设路径损耗为 $1/d^\alpha$ [4], 在这种情况下, 我们可以写

$$E_t = P_t / (1 - \alpha), \quad E_r = P_r / (1 - \alpha), \quad E_s = P_s / (1 - \alpha)$$

其中, P_t 是发射机电子设备消耗的能量/比特, P_r 是发射运算放大器消耗的能量, P_s 是接收机电子设备消耗的能量/比特, P_d 是检测比特的能量成本。因此, 充当中继的节点接收一个比特, 然后将它向前传送一段距离 d 所消耗的能量是

$$E_{t,r,s} = P_t / (1 - \alpha) + P_r / (1 - \alpha) + P_s / (1 - \alpha)$$

典型的, 对于当前的收音机公司

$$P_t = 180 \text{ nJ/bit}, P_r = 210 \text{ pJ/bit}, P_s = 2 \text{ nJ/bit}$$

或者

0.001 pJ/bit/m-n [29]。因此, 在(1)中, 我们有

$$c_{ij} = \alpha_{ij} x_i$$

$$1 \leq x_i \leq 10, \quad \forall i \in V$$

对于从节点 s 开始并在节点 t 结束的每个链路 $j \in E$, (8)中表述的 c_{ij} 的一个性质是, 它是 x_i 和 x_t 的凸函数。只要保持凸性, 部署问题的解决方案对于特定形式的 c_{ij} 实际上是健壮的。

在参考文献中, [4]提出了一种基于中间传感器节点数量不受限制的最小功率拓扑。在这种假设下, 数据源和接收器之间最节能的路径是多跳的直线, 通过为网络中的每个数据源构建这样的路径来构建最小功耗拓扑。这种拓扑消耗的功率最小, 因为每个数据流 r_m 都采用最短路径到达汇聚节点, 并且通过优化中间节点的数量, 最短路径上的功率消耗也被最小化。理论上的最佳跳数 K_{opt} , 包括在数据源 s 处充当传感器的节点, 由 $K_{opt} = \lceil \sqrt{D/d_{char}} \rceil$ 给出,

其中 $D = \|s - b\|$ 是 s 和基站 b 之间的距离, d_{char} 是 $d_{char} = \frac{P_t}{P_r} \frac{1}{1 - \alpha}$ 给出的“特征

距离”

$$K_{opt} = \lceil \sqrt{D/d_{char}} \rceil$$

其中 I, n 由节点能量模型(6)定义。某个源 s_m 和基站之间相应的功耗下限为

$$P_m = \frac{P_t}{1 - \alpha} + \frac{P_r}{1 - \alpha} + \frac{P_s}{1 - \alpha}$$

$$1/d_{char}$$

其中 $D = \|s_m - b\|$ 。然而, 在构建这种最小功率拓扑时, 需要大量的中继, 因为每个数据流都是独立的,

并且没有与其他数据流共享中继。当节点数量有限时,一个自然的想法是通过(1)使两个或多个数据流共享一些中继或(2)在一些路由上部署更少的中继来最小化功耗。这使我们回到最小功率传感器部署问题(1),它耦合了两个传统的优化问题:如果给定流量矢量 \mathbf{f}_m , 并且(1)仅在传感器 x_i 的位置上优化, (1)可以被视为设施位置问题;然而,如果给定传感器位置 x_i 和 \mathbf{f}_m 是唯一的决策变量,它可以被简化为最小成本流问题。成本函数的非线性以及这两个问题的耦合使得(1)难以解决。例如,我们发现使用标准拉格朗日松弛方法并不能显著降低复杂性,因为这种耦合是紧密的。

作为替代,参考文献中提出的解决方法。[40]使用分解方法,利用两个事实:(1)链路成本的凸性,以及(2)在传感器网络中,数据流量总是从源流向基站,这允许我们通过仅考虑在网络上形成树形结构的流矢量来减少 \mathbf{f}_m 的可行空间。此外,我们还放宽了容量限制(3);当前的传感器网络确实是在流量很小的情况下运行的,链路上的实际数据流不可能达到链路的容量。当没有达到这个容量时,也很容易看到,除了树结构中的链接之外,从来没有使用过其他链接[57](如果使用任何这样的链接,则到接收器的距离增加;因此,功耗也增加了)。

3.1.1. 问题分解

所提出的分解方法是由问题的特殊结构所驱动的。由于链路 j 的成本 c_j 是其端点 x_{s_j} 和 x_{t_j} 的位置的凸函数,并且由于(1)中的总成本是所有链路成本的加权和,这意味着对于给定的一组流量矢量 \mathbf{f}_m , 成本也将是传感器 x_i 位置的凸函数。这种凸性允许设计快速算法来找到最佳传感器位置 x_i 和相应的最小成本 f_1, \dots 。对于给定的一组流动矢量。更正式地说,

M

$\min_{\mathbf{f}_1, \dots, \mathbf{f}_M} \sum_{j \in E} c_j(x_j) f_{jm}, \quad x_j \in V, j \in E$

s.t. $x_m \in V, \quad m = 0, 1, \dots, M$

, g_1, \dots 。如上所述,记住网络树结构和消除(3),主要问题(1)变成

, 小 f_1, \dots, f_M (11b)

f

s.t. x 共同执行活动 f_{jm} $\forall j \in E, m = 1, \dots, M$

$M \leq E$

(11a)

m

$\sum_{j \in E} f_{jm} \leq C_j, \quad m = 1, \dots, M$

$f_{jm} \geq 0, \quad m = 1, \dots, M$

(11b)

f_{jm} 在哪里

一 如果弧 j 离开节点 i

0 否则

在这个公式中, (12)仍然是流量平衡方程,而约束条件(13)和(14)在每个数据源和基站之间建立了唯一的路径;从而保证网络的树形结构。

子问题(10)和(11)提出了一种迭代方法来解决原始问题。从一组可行的流动矢量 \mathbf{f}_1 开始, ..., \mathbf{f}_M , 第一步是求解(10), 它提供用于更新流向量的信息。参考文献中详细介绍了一种有效的基于梯度的方法(称为“内力法”)。[40]. 简而言之, 它认为网络是一个动态系统, 每个节点都有“内力”; 特别地, 由连杆 j 施加到节点 i 的力被定义为

M

$\mathbf{f}_{ij} = \frac{\partial c_j}{\partial x_i} f_{jm}, \quad j \in E, i \in V$

$\mathbf{f}_{ij} = -\frac{\partial c_j}{\partial x_i} f_{jm}, \quad j \in E, i \in V$

每一个这样的力都会导致节点 i 向最陡的下降方向移动, 从而导致它到达一个平衡点(唯一的, 由于凸性), 在那里施加在 i 上的所有力被平衡。

第二步是解决子问题(11), 即。在第一步得到的树形结构中找到从所有数据源到接收器的最佳路由。虽然这个想法很简单, 但仍然有一个困难阻碍它的实施。困难在于 g_1, \dots, g_M 是流动矢量 \mathbf{f}_1 的非凸非凹函数, ..., \mathbf{f}_M , 这通常意味着存在多个局部极小值。因此, 我们遵循不同的方法, 基于以下思想:(1)一次增加一个节点的数目, (2)确定新节点和相应流向量的最佳位置, 以及(3)重复该过程, 直到达到可用节点的数目 N 或者成本足够接近已知下限(9)。

3.1.2. 增量迭代方法

在增量部署中, 最初的步骤是从 M 个节点开始, 每个节点位于 M 个源之一, 并以基站为根构建相应的树形结构。相关的流动矢量 f_1, \dots, f_M 由(12)立即给出, a_{ij} 、 b_{ij} 由这个简单的初始树结构确定。

下一步是添加一个节点并确定其最佳位置, 同时保留网络的树形结构。不幸的是, 随着节点数量的增加, 可能的树结构的数量呈指数级增加, 构建一个高效的算法来找到最佳拓扑是一个关键问题。参考文献中提出的方法。[40]是基于局部拓扑调整的, 因此问题的规模是有限的; 代价是无法再保证全球最优。然而, 如上所述, 由于我们知道无限数量节点的最佳部署由每个数据源和基站之间的多跳直线路径组成, 因此我们可以利用我们的解决方案进行比较的下限(9)。如数值结果所示, 所提出的算法快速接近该下限, 并且节点的数量明显小于先前给出的相关联的 K_{opt} 数量。

添加一个节点并确定其最佳位置需要三个步骤。首先, 我们确定网络的哪个部分最需要一个新的中继。然后, 获得该区域周围所有可能的拓扑调整。最后, 将检查每种情况下的功率改善, 提供最大改善的情况将成为新配置。图 2 以图形方式总结了整个过程。

第一步。确定瓶颈节点。瓶颈节点是指一个新的中继节点和相应的新拓扑结构能够为整个网络的能量节约带来最大改善的节点。瓶颈节点是通过检查施加到节点的内力来确定的: 如前所述, 链路上的内力包含该链路上能量消耗的梯度信息。节点上的连杆施加的内力越大, 通过缩短该连杆节省的能量就越多。在添加新节点之前, 网络中的所有节点都已达到其平衡点。因此,

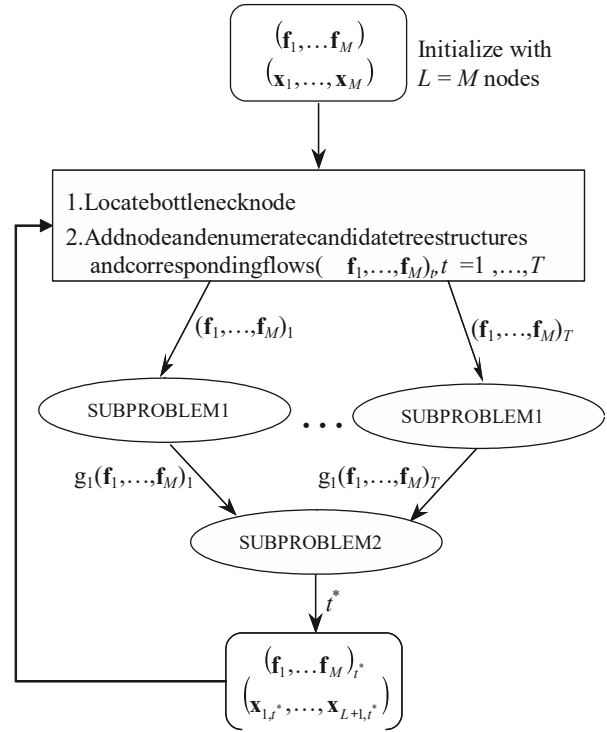


图 2. 增量迭代节点部署过程。

如果一个节点在几个相对较大的内力作用下保持平衡, 那么通过缩短其中一个环节, 这个环节的能量消耗将大大提高, 但其他环节的成本将超过这种提高。直观地说, 我们可以看到该传感器节点周围的区域更加稀疏, 并且在该区域更需要新的继电器。考虑到这些观察, 我们将节点 I 周围的稀疏性定义为

$$SP_i = \sum_{j \in \mathcal{N}} \frac{1}{d_{ij}^2} \quad (15)$$

根据(15)中给出的 F_{ij} , 瓶颈节点 k 被定义为具有最大稀疏性的节点。也就是,

$$k = \arg \max_i SP_i$$

$I=0, \dots, L$; 普通

显然, 不能保证新节点的最佳位置确实在上面定义的瓶颈节点附近, 因此这种方法隐含的解决方案通常是次优的。

第二步。枚举拓扑调整选项。瓶颈节点表示最需要新中继的区域。一旦确定, 必须确定新继电器的精确位置。由于我们正在研究一个树形结构, 插入一个新的中继也意味着添加一个新的链接。因此, 我们需要考虑目标区域中存在额外中继和链路时产

生的拓扑。一个例子将有助于理解拓扑枚举过程。在图. 3, x_2 是瓶颈节点。作为树结构中的一个节点, 它有三个子节点, x_3 、 x_4 和 x_5 , 它的父节点是 x_1 。图中的箭头表示这些关系, x_6 是我们添加的新节点。插入 x_6 后的几种(但不是全部)可能拓扑如图所示。3.如参考文献所示。[40]所有可能的新拓扑的数量是 $3 \cdot 2^{m-1}$, 其中 m 是瓶颈节点的子节点数量。例如, 在图 2 中。3 有 10 种可能的新拓扑。在图 1 的情况(a)中。3, 我们定义 x_3 为新节点 x_6 的子节点, x_2 为其父节点。作为替代, 如果我们将 x_1 定义为 x_6 的父代, 那么 x_2 和 x_1 的亲子关系可能改变也可能不改变: 如果它不改变, 则得到情况(d), 如果它改变, 则得到情况(c)。

第三步。获取新部署。当当前节点数为 $L < N$ 时, 步骤 2 的结果是多个可能的新网络树结构, 比如 T_1 , 每个网络树结构具有相关的流向量- $F_1, \dots, f_m, t_1, \dots, T_L$ 。对于每个这样结构测试, 子问题

,10)被求解(如前所述), 给出相应的最佳节点位置 $X_i; t, i=1, \dots, L$ 和成本 $g_{t \rightarrow f_1, \dots, f_m}$ 。FML。接下来, 问题(11)的解决方案简化为比较所有这样的成本并确定 $t \arg \min t_1; \dots, T_L g_{t \rightarrow f_1, \dots, f_m}$, 对应的节点位置 $X_i; t, i=1, \dots, L \geq 1$, 流量 f_1, \dots, f_m :

例子。如图 2 所示。4、解决了 $M=9$ 数据源传感器网络的部署问题。在这个网络中, $R_{M=1:0kb} = s-m_1, \dots, 9p$ 。数据源位于 3×3 格。水槽位于这个网格的左侧。两个相邻数据源之间的距离为 300 米。接收器和它最近的数据源之间的距离也是 300 米。在本例中, 我们假设 $n=4$ 和 $1180nJ=\text{bit}, 20:001pJ=\text{bit}=m_4$ 。

最初, 网络中只有九个传感器(图 1 中的星形节点。4), 每个数据源一个($N=M=9$)。为了将数据发送回基站, 已经建立了九条数据链路, 如图所示。接下来, 我们逐渐增加更多的传感器到网络中, 作为中继。图 4 展示了当存在 L 个继电器时获得的最小功率展开配置($L=5, 10, \dots, 45$)。随着更多节点的部署, 我们预计传输功耗会有所改善。图 5 展示了这一改进。在这个图中, L 是网络中中继的总数 ($L=N-M$)。Pself(实线)是网络的总传输功率(兆瓦), 假设继电器使用建议的增量自部署算法进行部署。Popt(显示为三角形)是基于对中继节点的数量没有约束的假设的网络的最小功耗(在这种情况下, Popt 16:122mW, 在 $L=59$ 处获得)。如图 2 所示。5, 对于

这九个数据源设置, 随着 L 的增加, 传输功率向下限 Popt 单调减小。图中还显示了传输功耗和网络中部署的中继数量之间的权衡。当 $L=28$ 时, 总传输功耗为 16.874 兆瓦, 比最小功耗 Popt 高出 $< 5\%$ 。然而, 这种部署方案使用的继电器不到构建最小功率网络所需的一半。

最后，我们应该指出，上面的增量部署方法是基于集中式方案的。它假设存在一个控制器，该控制器具有强大的计算能力、整个网络的完美信息以及对所有传感器节点的无限控制。在移动节点(将在第 4 节中讨论)但仍然是已知数据源的情况下，一个开放的问题

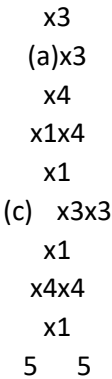
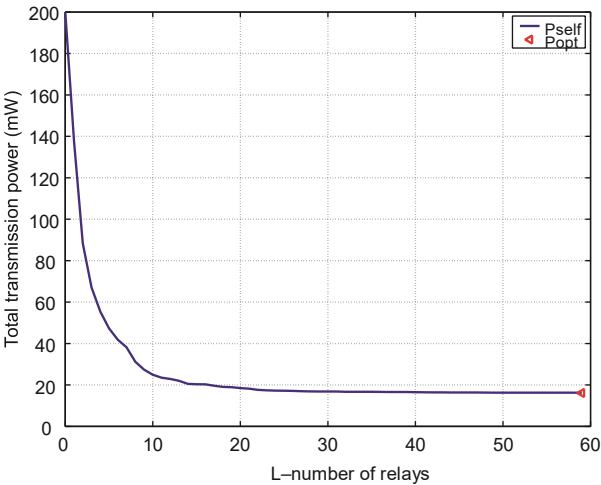


图 3。由于增加了节点 x6 而产生的一些可能的拓扑。⁶

是开发传感器节点部署的分布式算法，通过该算法，单个传感器节点可以基于其自身对整个系统的知识自主决定如何移动。

3.2. 路由和调度

传感器网络的无线特性使得路由和调度问题特别具有挑战性，尤其是当传感器节点的有限功率和计算资源、某些节点可能偶尔处于睡眠状态或干脆死亡的事实以及安全考虑等因素存在时。路由策略负责将数据包转发到最终目的地。与有线网络中依赖全局信息的传统路由协议不同，传感器网络中的路由协议通常采用本地成本信息(例如，两个节点之间的距离),以便做出路由决策。在接下来的内容中，我们将自己限制在对路由问题的最新贡献的简要概述中，以便提高读者对所涉及的问题的认识，因为其中一些问题在第 4 节中我们关于移动节点网络的讨论中起着至关重要的作用。的路由方法综述



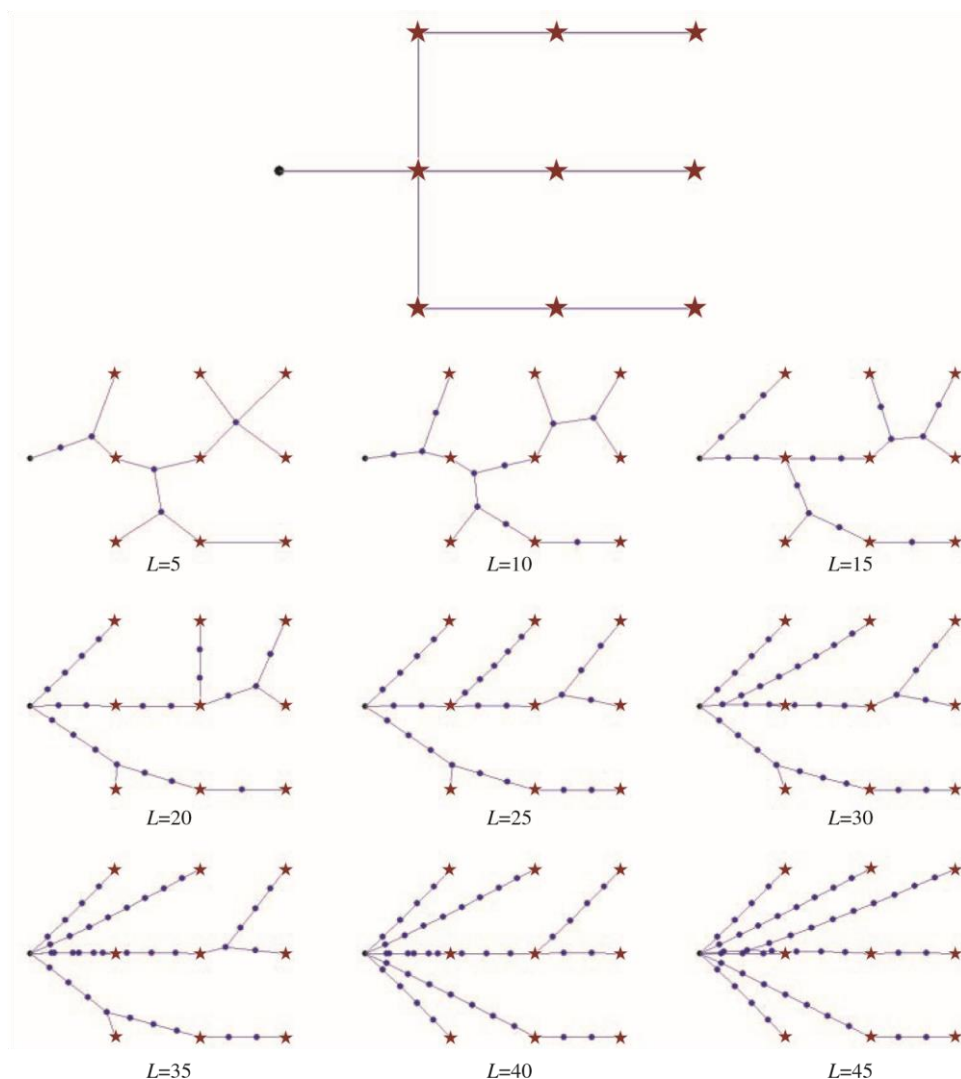


图 4. 九个数据源示例和增量节点部署。

图 5. 增量自部署算法性能是已部署节点的函数。

一般无线网络可以在参考文献中找到。[1]. 特别是对于传感器网络, 根据所采用的网络结构, 已经提出的路由算法可以分为两类, 如第 2 节: 平面和分层路由算法中所讨论的。平面算法的例子是“定向扩散”方法[32], 这是一种由信宿发起的反应路由技术, 其中当信宿请求时建立路由, 以及参考文献中的最小成本转发算法。[69]利用了这样一个事实, 即在传感器网络中, 数据流总是流向一个固定的接收器: 类似于自然重力场, 从山顶到地面驱动瀑布, 在每个点, 数据流量沿着最短路径从一个高点流向一个低点。为了让这个算法工作, 每个节点需要有从它自己到接收器的最小成本估计。分层算法的例子有: 低能量自适应分簇层次结构(LEACH)方案[28], 一种分层路由算法, 其中选出的簇头将聚集的数据直接传输到信宿, 以及传感器信息系统(PEGASIS)中的节能收集算法[43], 其

中节点仅与其最近的邻居通信, 并轮流作为传输到信宿的领导者。

仅限于平面网络结构, 公平地说, 大多数建议的路由方法是基于最短路径算法的[34, 51, 53]。这种算法通常要求每个节点维护一个全局代价(或状态)信息表, 这对资源受限的传感器网络来说是一个很大的负担。卡普和孔[36]提出了一种贪婪的周界无状态路由协议(GPSR), 该协议使用网络中节点的真实位置和数据包的目的地来做出路由决定。该协议的优点是每个节点只需要跟踪本地状态信息。为了处理节点故障的问题, Ganesan 等人。[24]提出了一种多路径路由算法, 这样就可以恢复主路径上的故障, 而无需启动全网络范围的泛洪过程来重新发现路径。由于泛洪会消耗大量能量, 因此这种路由方法可以在出现故障时延长网络的生命周期。然而, 找到多条路径并通过它们发送

数据包也会消耗能量, 因此如果没有故障, 会对网络的生命周期产生不利影响。

上面提到的路由策略可以间接减少传感器网络中的能量使用, 但是它们没有明确地使用能量消耗模型来解决路由策略相对于能量感知度量的最优性。近年来, 这种“能量感知”已经激发了许多最小能量路由算法, 这些算法通常寻找最小化每个分组消耗的能量(或最大化剩余节点能量)以到达目的地的路径[65]。然而, 正如参考文献中所指出的。[18], 寻求最小能量(或最大剩余能量)路径会迅速耗尽一些节点的能量, 并最终通过破坏其连通性来缩短整个网络的寿命。因此, 另一个性能指标是网络寿命。按照这些思路, Shah 和 Rabaey [62]提出了一种能量感知路由(EAR)策略, 该策略不试图使用单个最优路径, 而是使用多个次优路径, 这些路径是从概率上选择的, 目的是通过“传播”流量并迫使网络中的节点同时耗尽它们的能量来延长网络寿命。在EAR中, 每个节点构建一个成本信息表, 并将本地成本信息传播给其他节点。

成本由每个节点的剩余能量和它们之间的距离决定。每个节点还维护一个由本地开销信息决定的路由概率表。在参考文献中。[18]中, 以网络寿命最大化为目标的路由问题被表述为一个线性规划问题, 其中决策变量是源到目的地的路径流, 并提出了一种最短成本路径路由算法来有效地逼近其解; 链路成本被定义为结合每条链路末端节点的能量消耗和剩余能量。

让我们仔细看看传感器网络的能量感知最短路径路由算法的功能。回到传感器网络作为有向图的观点, 如前一节所述, 现在让我们用 $s_i, i \in V$ 表示节点位置, 用 $\pi, j \in E$ 表示链路, 代价 c_{ij} 由下式给出

$$c_{ij} = \frac{1}{E_i} \left(\frac{1}{\alpha} \left(\frac{1}{E_j} \right)^{\alpha} + \frac{1}{\beta} \left(\frac{1}{E_i} \right)^{\beta} \right) \quad (1)$$

其中 e 是 (i, j) 上的通信能耗, 如(8)所示。在 G 上, 路由协议生成一组“最短路径” $L = \{l_1, \dots\}$ 。每个传感器节点和基站 b 之间的 $l \in L$ 。在这里, 一条小路, $l \in L$ 。在 c_{ij} 的意义上, $k, 0 \in V$ 被称为节点 i 和 0 之间的“最短路径”; $l \in L$ 在 i 和 0 之间的所有可能路径上被最小化。众所周知, 最短路径集 L 形成树结构[57], 并且它可以由前向索引向量 h_1 表示, $h_1 = [h_1, \dots, h_n]$, 其中 $h_i \in \{0, 1, \dots, n\}$ 。 Ng 表示从 i 转发数据时下一跳节点的索引。特定的路由协议在每个传感器

节点 i 负责计算前向索引 h_i 和前向成本 c_i 。路由协议还向节点 i 提供上游向量 $U_i = [U_i^1, \dots, U_i^m]$ 。和累积流量系数 z_i , 定义为

普通

$$U_i^j = \begin{cases} 1 & \text{if } j \text{ is the upstream node of } i \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

z_i

其中, U_i^j 指示 j 是否是 i 的上游节点, z_i 记录从 i 到 0 收集的数据的总数据速率, $P_i = \sum_{j \in N} U_i^j z_j$ 是来自上游节点的总流量。

路由协议的一个例子是 Ref [62] 中提出的地理能量感知路由(GEAR)协议, 它动态更新最短路径集 L , 并在每个节点 i 以分布式方式生成前向索引 h_i 、前向成本 c_i 、上游向量 U_i 和累积流 z_i 。[70]. 根据该协议, 传感器网络中的节点通过重复应用学习步骤来动态更新下一跳目的地。假设在第 k 个学习步骤之后, 节点 i 存储了一组定义当前路由的状态变量 h_{ki} 、 c_{ki} 、 U_{ki} 和 z_{ki} 。在第 k 个学习步骤开始时, 节点 i 广播一个控制分组, 并从所有附近的节点请求转发成本 c_{kj} 和位置 s_j 。收到回复后, 我通过计算新的 h_{k+1} 和 c_{k+1} 来更新路由, 如下所示: $h_{k+1} = \arg \min_{j \in N} \{c_{kj} + \frac{1}{\alpha} \left(\frac{1}{E_j} \right)^{\alpha} + \frac{1}{\beta} \left(\frac{1}{E_i} \right)^{\beta}\}$

$$c_{k+1} = \frac{1}{E_i} \left(\frac{1}{\alpha} \left(\frac{1}{E_j} \right)^{\alpha} + \frac{1}{\beta} \left(\frac{1}{E_i} \right)^{\beta} \right)$$

其中 U_i 是响应节点 i 请求的一组相邻节点。在 h_{k+1} 的情况下, 节点 i 向节点 h_{k+1} 和 h_k 发送控制分组, 以便通知它们该路由改变。在接收到分组后, 节点 h_k 通过使用 h 更新其状态变量 U_{k+1} 和 z_{k+1} 来移除作为上游节点的 i

$$U_{k+1}^i = \begin{cases} 1 & \text{if } i \text{ is the upstream node of } h_k \\ 0 & \text{otherwise} \end{cases}$$

然而, 节点 h_{k+1} 添加 i 作为其上游节点, 因此

$$U_{k+1}^i = 1, \quad z_{k+1}^i = z_k^i + P_i$$

这样, 在一个学习步骤之后, 节点 I 更新它的本地路由和相应的状态变量。如参考文件所示。[70], 通过在每个传感器节点重复执行学习步骤, 得到的前向索引向量在有限时间内收敛到最短路径。从网络安全角度来看, 确定性路由策略(即, 源节点通过一条或多条固定路径发送数据的策略)极易受到节点泄露和伪造成本信息的攻击, 从而导致拒绝服务(DoS)攻击[68]。例如, “陷坑攻击”会危及一个节点, 并向相邻节点广播虚假的低成本, 从而诱使所有这样的节点将数据包路由到它。相邻节点反过来将受损节点的低成本广播给它们的邻居, 最终效果是该节点充当所有分组的接收孔, 同时也消耗网络节点的能量。为了减少这种攻击的影响, 概率路由是一个有趣的选择, 因为这使得攻击者很难确定一个“理想的”节点来接管。从这个意义上说, EAR 策略是有吸引力的, 因为它的概率路由结构, 即使它不试图为网络生命周期最大化提供最佳路由概率。此外, 考虑到传感器网络在拓扑变化、节点故障和能量消耗方面的动态行为, 可以预期最优路由概率是时变的。因此, 最优控制问题的公式是一个自然的设置。目标是最大限度地延长网络寿命。在参考文献中。[67]研究表明, 在固定拓扑情况下, 存在由固定路由概率组成的最优策略, 该策略可以通过求解一组相对简单的非线性规划问题来获得。

转到调度问题, 当前最具挑战性的问题再次与能量感知和基于系统状态控制节点传输速率的努力有关[66, 71]。在加性高斯白噪声(AWGN)信道中, 信道容量由香农定理[64]确定: $C = B \log_2(1 + \text{SNR})$, 其中 B 是信道带宽, 信噪比是信噪比。假设信道利用率为常数, $0 < \alpha < 1$, 数据速率可以用 $r = \alpha C$ 表示。可以看出, 线性增加数据速率将成倍增加传输功率。动态传输控制问题涉及控制传输速率, 传输速率又决定一组任务的传输时间, 目标是 최소화 总射频传输能量消耗。这个问题与动态电压缩放(DVS)问题有许多相似之处, 将在下一节讨论。

在某些情况下, 路由和调度问题被共同视为服从显式公平约束的效用最大化问题[55](例如, 在所有平均传输速率之间强加相等), 通过线性不等式建模。由节点 I 到节点 j 的业务类别 k 分组的传输由 (I, j, k) 表示。让 r_{ijk} 表示 (I, j, k) 或 (j, I, k) 传输的净流量, r 表示相应的传输速率矢量。然后, 参考文献中提出的这个问题。[55]是

最大值

s.t. $r \geq 0$ 转换器:

其中, R 是一组可行的传输速率向量(基于功率约束), S 是一组满足公平性约束和流量守恒的速率向量。还假设 $F(r)$ 在 $\text{Conv}(R)$ 中是连续的, 凹的, 有界的。解决这个问题一个明显的困难是需要描述 $\text{Conv}(R)$ 的特征。文献[1]提出了一种有效的分解算法。[55]其在有限次数的迭代中终止, 并产生在传感器网络通常运行的低传输功率水平下渐近最优的策略。产生的策略选择网络应该运行的多种模式(即, 确定谁向谁传输以及以什么功率传输)以及这些模式中的适当的分时调度。 s 。

3.3. 功率控制: 动态电压缩放问题

控制传感器节点功率的问题在我们到目前为止讨论的所有问题中都有体现。它出现在部署问题的目标函数中, 并且是已经为传感器网络提出的所有路由算法的中心部分。然而, 除了通过控制节点位置、链路数据速率和传输时间表来管理功率的间接方法之外, 还有功率控制的直接方法。一种方法是明确地关闭一个节点, 并周期性地进入“睡眠”状态, 如第 2 节所述。我们将在本节中讨论的另一种方法是, 在节点不处于“睡眠”状态。

如前所述, 传感器节点包含电池容量有限的小型廉价设备。延长电池寿命与网络的整体性能密切相关; 在传感器网络中, 一些节点的故障会导致显著的拓扑变化, 这需要相当大的额外功率来重新组织网络。据报道, 在低功耗系统中, 处理器占总功耗的 18-30%, 通常超过 50% [54]。控制电压和时钟频率提供了调节处理器功耗的手段, 从而产生了 DVS 技术[29, 52, 54, 60]。这些技术的核心是一个基本事实, 即器件处理器的能耗与工作电压 V 至 $E \propto V^2$ 有关, 其处理频率(时钟速度)由 $f \propto V$ 给出, 其中 C_1 、 C_2 是取决于器件物理特性的常数, V_t 是阈值电压, 因此 $V \propto V_t$ 。这些关系可能是近似的, 但是 V , E 和 f 的函数相互依赖性清楚地表明, 降低电压提供了降低能量的机会, 这是以更长的延迟为代价的。这反过来又会对性能产生不利影响, 在任务必须满足严格实时约束的系统中可能会产生灾难性后果[9]。因此, 管理这种折衷以利用通过 $E \propto V^2$ 的二次能量减少而不显著降低延迟性能成为基本的设计和动态控制问题。

在过去十年中, 已经提出了许多 DVS 算法。其中大多数是为实时系统的抢先调度而设计的, 如参考文献[2, 37]。非抢占调度在实践中通常是更好的选择, 尤其是对于资源非常有限的系统, 因为不受控制的抢占会导致大量上下文切换, 需要更大的堆栈大小和增加的能耗[33, 35]。参考文献回顾了为非

重复情况开发的 DVS 算法。[30]. 其中许多是为具有周期性任务的系统开发的, 如参考文献。[59]. 然而, 非周期性任务更可能发生在由异步操作组件组成的环境中, 例如传感器网络, 其中传感器节点异步地向处理节点(例如第 2 节中讨论的簇头)提供数据。

功率和延迟之间的基本权衡自然会产生各种优化问题, 这些问题取决于(1)任务定时的重要性, (2)与仅在任务之间控制电压相反的连续控制电压的能力, 以及(3)DVS 控制器可以操作的信息。

关于(1), 任务被分类为具有与其相关联的“软”或“硬”时间要求。在软需求的情况下, 优化问题的目标函数包含惩罚长任务完成时间的成本(一个“任务”可能涉及节点处的数据处理, 或者它可能涉及传输数据包)。在硬需求的情况下, 问题是用对任务完成时间的明确约束来表述的。

关于(2), DVS 最简单的形式是在任何新任务开始时改变电压。然而, 可替换地, 当重要的新事件发生时(典型地, 新任务到达节点), 或者甚至在任意时刻, 也可以在处理任务时应用控制。如果一个控制器在一个任务的整个处理过程中被限制保持不变, 我们称它为静态的, 否则它被称为动态的。但是, 请注意, 静态控制器仍然可以为每个新任务分配不同的电压值。

最后, (3)是关于控制者在被要求行动时所掌握的知识。通常, 我们可以假设一个任务的特性是已知的, 例如它所需的操作数(在数据处理的情况下)或数据包中的位数(在传输任务的情况下)。然而, 任务的到达时间可能是未知的, 除非处理器通过预先调度给定数量的任务来操作。当任务的到达时间已知时, 我们可以开发一个离线控制器; 否则, 我们称之为在线控制器, 它只知道已经在队列中的任务的到达时间, 并且可能知道一些有限的未来到达时间或它们的估计。

具有软定时要求的离线静态控制问题可以使用为离散事件和混合系统的最优控制开发的技术来解决。特别是, 这些类型的问题在参考文献中有所涉及。[11]其中表明, 可以使用有效的正向分解算法来避免这类问题经常带来的组合复杂性。FA 仍然需要求解 N 个凸优化问题(其中 N 是任务数), 这对于板载计算能力有限的在线应用来说一般是要求较高的。然而, 具有严格时序要求的离线静态控制问题具有结构特性, 允许我们进一步分解问题并解决它, 而不必依赖于任何优化, 并且独立于所涉及的设备的物理特性[50]。最近一个有趣的结果是, 对于具有硬时序要求的离线问题, 静态和动态版本给出了相

同的解决方案[44]。其结果是显著的, 因为它断言简单控制器的最优性, 在这种行为的成本高的环境中不需要任何数据收集或处理。此外, 静态控制器不需要额外的开销, 否则会涉及到连续的控制调整。

在接下来的内容中, 我们将只讨论离线静态控制问题。当没有任务到达时间的知识可用时, 在线 DVS 控制问题保持开放。如果这些到达时间具有概率特征, 这就产生了随机最优控制问题。另一方面, 如果一些有限的未来信息是可能的, 那么可以采用滚动时域控制方法, 其特征是最近在参考文献中显示的几个有吸引力的特性。[45].

3.3.1. 离线静态控制问题公式

我们考虑的传感器网络节点被建模为单阶段先到先服务排队系统。让 a_i 表示任务 I 的(已知)到达时间, x_i 表示它在完成处理后离开。该系统的动力学由 X_i $\max x_i$ $1, a_{iui}, i1, \dots, n$, 其中 u_i 是任务 I 的处理时间。设 I 为任务 I 的能耗函数, 依赖于 u_i 。

让 I 表示任务 I 所需的操作数(这可能取决于该任务的具体情况), 并使用关系 $E C1V2$ 和 $f-V vt = C2V$, 我们可以编写

2

$$iE iC1 \quad Vt_{ui} \quad \delta 17p \\ u_i iC2$$

然而, 我们强调, u_{iu} 的精确形式或常数的值并不重要; 重要的只是, 对于 $u_i > iC2$, u_i 是 u_i 的连续可微严格凸单调递减函数。请注意, 对电压的附加约束是由电压 V_{\max} 的要求所施加的, 其中 V_{\max} 是最大工作电压。这进而导致对控制变量的约束:

$$u_i u_{imin} \frac{iC2V_{\max}}{\quad} \\ \text{威玛} \quad \text{佛蒙特州}$$

对于软定时要求的情况, 我们现在可以公式化以下问题 P1:

$$1 \text{ 分钟} \quad (J X iu i x_i)N \\ u; \dots; \text{联合国} \quad i1$$

$$, : t: u_i u_{imin}, i1, \dots, n; x00; ;$$

, $i \max x_i 1$, $i u_i$, $i 1$, ... 普通 N

其中 $i X_i$ 是一个连续可微的严格凸函数, 用于惩罚出发时间。因此, 上面的成本函数捕捉了能量和任务时间之间的权衡。在硬时间限制的情况下, 每个任务也被分配了一个截止日期 d_i , 这个问题被称为 P2 问题

普通

$\min_1 \quad (J \times 1)$

$u; \dots$; 联合国

$i 1$

, $t: u_i u_{\min}, i 1, \dots, n; x_0 0; X_i \max x_i 1, u_i$

$d_i, i 1, \dots, n::$

3.3.2. 最优样本路径分解

, P1 和 P2 的两个问题中, 不借助于这些硬约束非线性优化问题的显式解而获得有效控制器的关键是存在一组指数 $M f m_1, \dots, m_1 < m_K N$ 的 $m M g$, 使得最佳采样路径 a_1, x_N 可以分解成区间 s

, $x m_1, a m_1, x m_2, \dots, a m_k, x m_{k p} 1, \dots, a m_M, x N x_N$

具有以下性质: $x m_k < a m_{k p} 1$ 和 P1 的最优成本 J 可以写成

M

$J X J_k$

$k 1$

其中, 为了保持一致性, $x m_0 a_1, x m_{M p} 1 x_N$ 和 J_k 是

$m k 1$

$u m k \dots; u m k i X M K \min i u i i X i$

, $t: u_i u_{\min}, i m_k, \dots, M K 1; ;$

, i , 我知道, $\dots, M K 1::$

P2 也是如此。可以看出[11]上面的每个子问题都是一个带有线性约束的凸优化问题。换句话说, 一个具有非线性、不可微约束的复杂非线性优化问题(因为涉及到最大算子)被一组简单得多的问题所代替。当然, 为了使这个分解实用, 我们需要一个有效的方法来识别 m_1 的值, \dots 。嗯。如参考文献所示。[11]对于 P1 来说, 这可以通过简单有效的算法来实现。P2 的情况

特别有趣, 因为 $d_i < a_{i p}$ 意味着 $i 2 M$ [50], 所以集合 M 可以先验地确定。更多-

通过一组事件进一步分解, 还可以显示[50]每个 $a m_{critical} t a s k s k, x m_{k p} 1$ 可以

, $r k, \dots, r r k g$ 。设置 $i u_i = i$, 这种分解的性质是所有最优控制

, 数; 特别是, 我, 我, \dots 。在每个 $x r j k$ 中, $x r j k$ 是 e

$x r j 1 j K 1 x r j K k, l r j K k 1, \dots, r j k k p$

p

我 $l r f j k k 1 l g$

, 指数 $R 1-k, \dots, r R k$ 也可以通过低复杂度算法来确定。

在参考文献中得到的关键任务分解算法(CTDA) [50]具有有限的计算复杂性, 不需要解决任何优化问题。它还具有较小的空间复杂性, 这使得它对涉及有限存储器设备的应用具有吸引力, 并且它独立于能量函数的细节, 这意味着不需要测量参数, 如(17)中的 C_1 或 C_2 。

4. 具有协作移动节点的网络

赋予传感器网络中的节点以移动性极大地扩展了网络的能力范围。此外, 假设每个移动节点拥有一定的决策自主权, 就会产生一个具有相当大的灵活性的动态系统, 这取决于节点能够合作以执行“任务”的程度。例如, 这种灵活性允许我们用更少的节点来处理大量的数据源目标(此后我们将简称为“目标”), 这些节点可以随着时间的推移移动和访问目标以执行各种任务。

自然, 移动性也意味着额外的复杂性。例如, 如果要保持通信连接, 我们必须确保每个节点都保持在至少一些其他节点的范围内。我们还必须考虑到, 移动性消耗了大量的能量, 这加大了对各种形式的功率控制的需求。移动性的另一个有趣的方面是, 节点的准确位置并不总是对其他节点或基站可用。在 GPS 跟踪不适用的情况下尤其如此, 例如定位建筑物中的人员或重要设备(这被称为“资产跟踪”)。

位置检测问题是一个特别具有挑战性的问题，虽然我们在本文中不讨论它。

从系统和控制理论的角度来看，移动网络为涉及其节点的实时协作控制提供了机会。合作控制的目标是协调节点的行为，以实现一个共同的目标，我们称之为网络的任务。迄今为止，它最受欢迎的应用是无人驾驶自动驾驶车辆网络[13, 15, 56]，特别是为了编队控制、避障或稳定目的的研究车辆轨迹生成[7,26,27,31,42]。

根据人们希望为网络定义的任务类型，可以在合作控制的环境中描述各种问题。在接下来的内容中，我们将仅限于适用于传感器网络的两种任务。首先，我们将讨论一个奖励最大化任务，其中任务空间包含由 i 索引的 N 个目标点， \dots 。目标 I 有一个相关的(可能时变的)奖励 R_i 。这里的任务是控制节点的移动和访问各种目标的过程，以便在给定的任务时间 T 内最大化收集的总回报。其次，我们将考虑覆盖控制任务，其中主要困难是目标未知，并且该任务涉及定位节点，以便最大化检测源自目标的数据的概率。

4.1. 奖励最大化任务

在这类任务中，我们考虑一组由 j 索引的 M 个移动节点， \dots 。和一组用 i 索引的目标， \dots 。在 2D 空间。与第 I 个目标相关联的是奖励 R_i 。任务的目标是在给定的任务时间 T 内，通过访问集合 T 中的点来获得最大的总奖励。这个问题因几个因素而变得复杂。(1)目标回报可能与时间相关，通常随时间而递减；因此，节点访问目标的顺序可能很关键。(2)不同的节点具有不同的能力，因此将特定的节点分配给特定的目标也很关键。(3)目标的确切位置可能并不总是事先知道。(4)任务空间可能存在障碍物，约束节点的可行轨迹。

这种设置产生了一个复杂的随机最优控制问题，即使对于相对简单的任务控制设置，其解在计算上也是难以处理的。因此，很自然地将其分解为不同级别的各种子问题——从详细的运动控制到更高级别的路径规划和节点到目标的分配。例如，参考文献[19, 23]解决了动态分配资源的问题，而[8]将合作路径规划问题公式化为混合整数线性规划(MILP)，该规划结合了任务定时约束和障碍物的存在。这种功能分解方法的替代方法是基于时间分解的方法。主要思想是解决一个优化问题，寻求在给定的时间范围内最大化网络累积的总期望回报，然后连续地向前扩展这个时间范围(周期性地或以纯粹的事件驱

动的方式)。这个想法，在参考文献中有介绍。[12]是在滚动时域(RH)方案的精神下提出的，它与模型预测控制相关联，用于解决反馈解极其困难或不可能获得的最优控制问题；参见参考文献[17, 48, 61]和最近的[20–22, 58]。由此产生的协同控制方案通过在规划范围内解决一系列优化问题并在更短的行动范围内执行它们来动态地确定节点轨迹。我们应该强调的是，所涉及的优化问题并不试图进行任何明确的节点到目标的分配，而只是为了确定在当前规划范围的末尾将节点放置在使总预期回报最大化的位置的目标。因此，这是一个相对简单的问题。然而，事实证明，节点轨迹实际上收敛到目标，尽管事实上这种方法从本质上来说，从未打算执行任何这样的离散节点到目标的分配。在下文中，我们概述了在其他地方详细描述的方法[38]。

第一个目标的位置用 y_{i2R2} 表示。然而，请注意，一些目标位置对于协作节点来说可能是未知的。让 x_{j2R2} 表示第 j 个节点在时间 t 的位置，初始位置由 x_{j0} , $j = 1, \dots$ 。米 (meter 的缩写))。为简单起见，我们假设节点在整个任务中以恒定速度行进，即

$$\cos u_{jt} \quad \dot{\theta}_j \quad \dot{\theta}_j \quad \dot{\theta}_j \\ \quad \quad \quad V_j, \quad x_{j0} \quad x_{j0} \quad 18 \sin u_{jt}$$

其中 u_{jt} 是节点 j 的可控航向， V_j 是相应的速度。我们注意到 M , N , y_i 可能会随时间变化。

为了区分目标在时间 t 的相对重要性，每个目标都有一个相关的奖励函数，用 $R_i T_t$ 表示，其中 R_i 是最大奖励， $i T_t$ 是描述奖励随时间变化的贴现函数。请注意，通过适当地选择 $i T_t$ ，可以捕获施加在目标点上的时序约束以及优先约束。通过允许某些 I 的 $R_i < 0$ 并适当选择 I ，我们也可以在任务空间中模拟一个障碍物(参见参考文献。[39])。一个简单的发现功能是：

$$\frac{1}{1 - \gamma} \sum_{t=0}^{T-1} \gamma^t R_i(t)$$

当一个截止日期与一个特定的目标点相关联时，我们可以使用

我

8 8

< 1 t 如果 t Di

Di 20p

:- 1 le-TDi 如果 t > Di

其中 D_i 是分配给目标点 I 的截止时间, 而 $I \in \{0, 1, \dots, n\}$ 是可以是特定于目标的参数, 并且被选择来反映不同的感兴趣的情况。

为了区分节点相对于目标 I 的有效性, 我们定义了节点能力因子 $p_{ij}(t)$, 它反映了节点 j 在时间 t 访问点 I 将完成其任务并获得奖励 $R_{ii}(t)$ 的概率。如果 $x_{jt}(y_i)$ 和 $p_{ij}(t) > 0$, 我们说车辆 j 在时间 t 访问目标点 I 。因此, $y_i > 0$ 可视为目标的大小。如果在访问期间, 节点成功地完成了它的任务, 它将获得相应的奖励, 同时, 目标 I 不再是任务感兴趣的, 它将从集合 T 中移除。由于 I 处的访问与节点 j 的资源消耗有关, 因此访问后能力因子 $p_{ij}(t)$ 可能会降低, 即, $p_{ij}(t) < p_{ij}(t_0)$ 。

4.1.1. 合作结构

节点通过动态划分任务空间并在它们之间隐式分配任务空间的区域来进行协作。给定任务空间中的任意点 $y \in \mathbb{R}^2$ (不一定是目标), 我们希望将该点分配给时间 t 处的飞行器, 以便将 y 分配给具有最高概率的最近飞行器。为了形式化这个想法, 我们首先定义一个

函数 $B_b(t)$, 其中, T_t 包括 $b/2$, 即 $1, \dots, n$ 。最接近的车辆。假设此时是 R_2

距离最近的第 l 辆车

$b_l(y, t) = \min_{k \in T_t} \|y - x_k(t)\|$,

$b_l(y, t) \leq b_{l+1}(y, t), \dots, b_l(y, t) \leq b_M(y, t)$

$l = 1, \dots, M$

, 以相对距离函数, $\dots, b_l(y, t), g_t$, 如下:。然后我们定义

$$q_{ij}(y, t) = \begin{cases} 1 & \text{if } b_l(y, t) \leq k/2 \\ 0 & \text{otherwise} \end{cases}$$

特别令人感兴趣的是这种情况 接下来, 我们定义

目标点 $q_{ij}(y, t)$ 我们写。此外, 当相对邻近函数 $q_{ij}(y, t)$ 与 T_2 重合时, 同样地, 我们设置 j ,

为的任何单调非递增函数, 如 $q_{ij}(y, t) = 1 - \frac{b_l(y, t)}{k/2}$ 。当 $b_l(y, t) = 0$ 为

如果 $j > 1$ 我们写。此外, 当相对邻近函数 $q_{ij}(y, t)$ 与 T_2 重合时, 同样地, 我们设置 j ,

为的任何单调非递增函数, 如 $q_{ij}(y, t) = 1 - \frac{b_l(y, t)}{k/2}$ 。当 $b_l(y, t) = 0$ 为

如果 $j > 1$ 我们写。此外, 当相对邻近函数 $q_{ij}(y, t)$ 与 T_2 重合时, 同样地, 我们设置 j ,

如果 $j > 1$ 我们写。此外, 当相对邻近函数 $q_{ij}(y, t)$ 与 T_2 重合时, 同样地, 我们设置 j ,

如果 $j > 1$ 我们写。此外, 当相对邻近函数 $q_{ij}(y, t)$ 与 T_2 重合时, 同样地, 我们设置 j ,

可解释为“捕获半径”: 如果 $target_where(2, 0, 1) = 2$ 是一个可调参数, 则

我碰巧离节点 j 很近, 所以我可以满足则在时间 t , 节点 j 被承诺访问目标 I 。同样, 当 y 与目标 y_i 重合时, 我们

$target_write(2, 0, 1)$ 的概率 $q_{ij}(y, t)$ 。我们现在可以查看我被分配到的节点 ij ASAT

时间 t , 基于 观察 $j_{qij} \delta ij1$ 。

4.1.2. 合作后退地平线(CRH)轨迹构建

任务的目标是在某个任务时间 T 结束时(或定义任务结束的某个特定事件)收集最大的总奖励。为了实现这个目标, 我们设计了一个协作控制器, 在 $0, T$ 期间为团队 A 中的每个节点生成一组轨迹。该在线控制器在由 $tk, k = 0, 1, \dots$ 。在任务期间。在 tk , 控制器通过求解一个优化问题 P_k 来运行, 该问题的解是控制向量 uk $u1 \delta tk \mathcal{P} uM \delta tk \mathcal{P}$ 。接下来, 我们解释 P_k 是如何制定的。

, 设节点被分配了标题 $u1 \delta, \dots$ 。时间 tk 时的 $uM \delta tk \mathcal{P}$, 旨在维持 H_k 表示的规划范围。然后, 在时间 tk 或 H_k , 节点的计划位置由下式给出

$$x_{jtk} H_k x_{jtk} _ jtk H_k \delta 23 \mathcal{P}$$

下定义

$$ij \delta uk, Tk Tk H_k jxj tk H_k yij = V_j$$

$\delta 24 \mathcal{P}$

注意, $ijuk, tk$ 是节点 j 能够到达目标 I 的最早时间, 条件是它从 tk 开始, 由 uk 控制, 然后直接从 x_{jtk} , H_k 前进到目标 I 。我们感兴趣的是, 如果节点 j 在时间 $ij \delta uk, TkK$ 到达目标, 它可以从目标 I 中提取的最大回报。显然, 这是 $R_{ii} ij \delta uk, TkK$ 给出的。为方便起见, 请定义

$$\sim ij \delta uk, Tkl ij uk, Tkl \quad \delta 25 \mathcal{P}$$

值得指出的是, 与 I 不同, I 依赖于 I 和 j 。同样清楚的是, 在时间 tk 评估的提取该奖励的概率是由 $p_{ij} ij \delta uk, tk$ 给出的。为了方便起见, 我们设置了

$$\sim ij \delta uk, Tk P_{ij} ij uk, Tk P_{ij} \quad \delta 26 \mathcal{P}$$

返回到函数 $q_{ij} j q_{ji}$, 我们对它在 $t tk H_k$ 的价值感兴趣, 并定义

$$\sim ij \delta uk, Tk q_{ij} ij \delta tk$$

$\delta 27 \mathcal{P}$

使用记号 k 和 T_k 来表示这些集合对 tk 的依赖性, 我们现在可以给出优化问题 P_k , 在时间 tk 时表述如下:

普通 M

$$\text{最大 } X_k X_{Ri} \sim ij \delta uk, Tk_j \sim ij uk, Tk_j \sim ij uk, Tk_j$$

英国

$i2T \quad j2Ak-28$

分别具有(25)、(26)和(27)中定义的 $\sim ij uk, TkE$ 、 $\sim ij uk, TkE$ 和 $\sim ij uk, TkE$ 。表达式 $R_i \sim ij \delta uk, tk_j \sim ij uk, Tk_j \sim ij uk, Tk_{in}(28)$ 可视为节点 j 从目标 I 收集的预期报酬, 在时间 T_k 使用计划范围 H_k 进行评估。

问题 P_k 由规划范围 H_k 参数化, 这对于获得该 CRH 控制器的理想特性至关重要。我们将特别设定

$$H_k \text{ 最小 } f_{kyi} XJ Tkk = V_{jg-29} J2ak; i2T k; p_{ij} \delta tk \mathcal{P} > 0$$

也就是说, 在时间 tk , 任何目标和任何有能力的节点之间的最小“距离”(以时间单位表示)。如参考文献所示。[38]这一选择对于确保 CRH 轨迹始终收敛于目标至关重要。此外, 从我们对节点“访问”目标的定义来看, $H_k > \min_j 2Aki2T k; p_{ij} \delta tk \mathcal{P} > 0 f_{si} = V_{jg}$, 其中 $s_i > 0$ 是 I 的大小。在访问时, 目标处的任务完成并且目标从集合 $T-K$ 中移除, 或者节点耗尽其资源(即, $p_{ij} Tk0$); 在这一点上, 也要重新评估规划范围。

基于 hk 和 tk 上所有可用的状态信息, 在获得(28)的最优 uk 后, 所有节点遵循行动地平线 H_k 的控制。然后在时间 $tk1, tk, hk = 0, 1, \dots$ 港元的价值由两个因素决定。首先, 如果一个意想不到的事件发生在某个时间点, 那么我们设置时间点。否则, 我们只需在预先指定的时间后更新控件。因此, 一般来说, f_{tkg} 是一个随机序列。当(1)所有目标奖励被收集, (2)所有节点被消除, (3)节点耗尽它们所有的资源或(4)任务时间到期时, CRH 控制器终止。

就计算而言，这种方法的复杂性依赖于数量级的神经元，比离散分配算法更有效，在离散分配算法中，每个节点 I 被分配给一系列目标，如参考文献中所讨论的。[38]. 然而，主要的问题是，CRH 轨迹是否能保证访问目标，因为节点从来没有明确地分配给它们。这个性质被称为轨迹平稳性。具体来说，

于 1 节点 N 目标情况，可以看出这种情况简化为简单的测试。对于两节点问题，仍有可能通过分析表明一个或两个目标满足该条件。除了这些情况之外，分析变得非常复杂，但是仿真结果说明了这种方法对于非常大的 N, m 值的有效性。最近，在参考文献中开发了 CRH 控制器的分布式版本。[39]还在实验室环境中

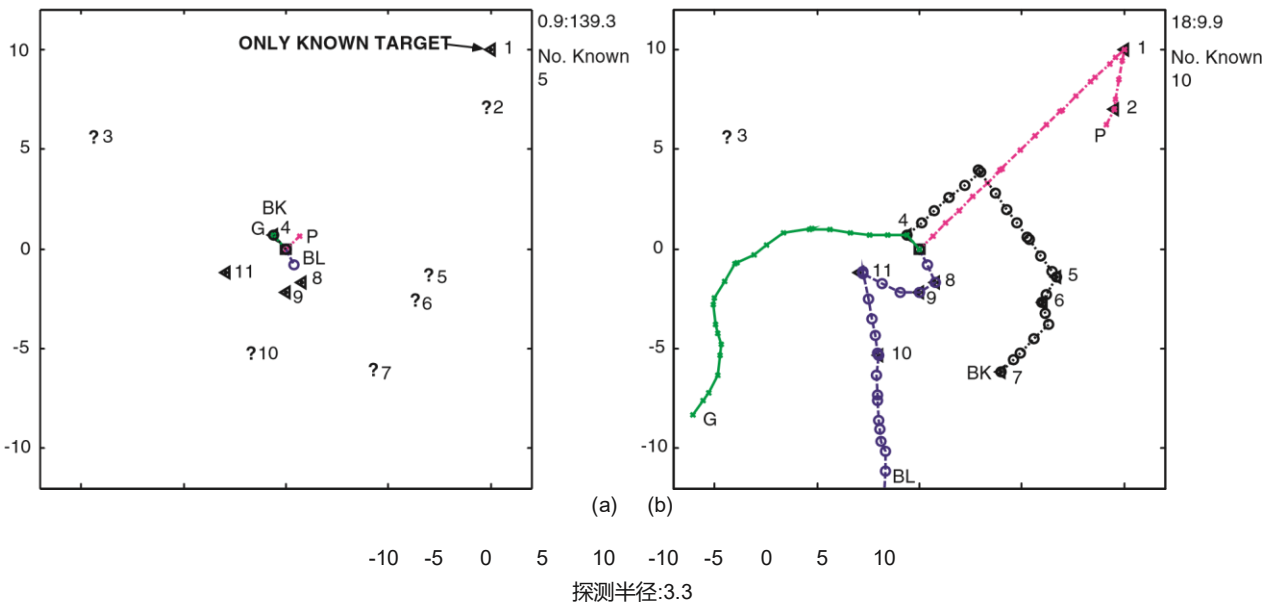


图 7. 目标不明的任务示例。

对于一个轨迹，如果存在一些 $tv < 1$ ，比如 $kxjtyvik$ $si, i \geq 2, j \geq 2$ A，那么 $x(t)$ 是一个稳定的轨迹，我们说

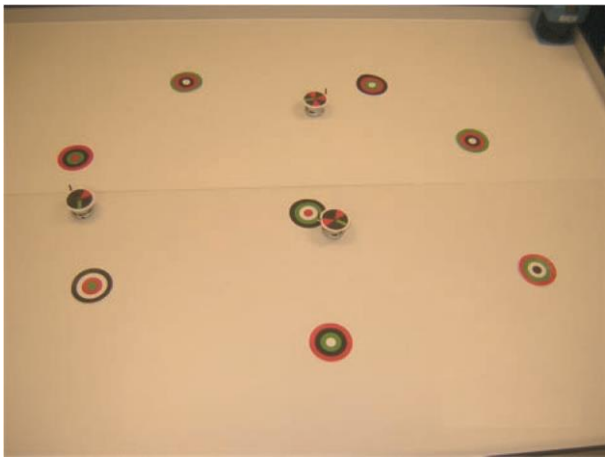


图. 6. 各种奖励最大化任务的实验室设置。

轨迹 x_{jt} 收敛到目标 I 。参考文献的主要贡献。[38]是为了证明 CRH 控制器在一定条件下确实产生稳定的轨迹，该条件表征了从(28)中的成本函数导出的势函数。然而，很难证明所有 N, m 都满足这个条件。对

成功实施，小型机器人扮演移动节点的角色。图 6 显示了这种设置的图片，三个无线机器人执行涉及八个目标的奖励最大化任务(颜色编码表示不同的奖励和截止日期)。

例子。图 7 显示了一个奖励最大化任务的例子，有四个节点(P, G, BK, BL, 最初都位于同一点)，其中最初只知道一个目标(标记为 1)。剩余的指示点对于网络来说是未知的，但是如果节点恰好位于“传感器检测区域”(在这种情况下，半径为 $10/3$ 单位的圆)。因此，任务的一个重要方面是寻找目标。控制器的实施需要分别选择计划和行动范围 hk 和 Hk 。在下文中， hk 由(29)给出， Hk 被选择为

$$12 Hk \text{ 如果 } Hk > r \quad 30 Hk$$

$$\text{香港} \quad \text{if } Hk \quad r$$

其中，对于任何目标点， r 是离它很小的距离(在续集中， r^2)。奖励折扣函数 iTt 由(20)qi 给出 ij 由(22)和给出 颈内 jyi ， Tt 由(21)和 $b \geq 2$ 定义。

在这种情况下, CRH 控制器用于分散四个节点, 以便它们向不同的方向前进。在图 7(a), 因为目标 4、8、9、11 位于由初始位置定义的检测半径内, 所以网络知道它们(假设节点可以在它们之间完全通信)。BK、G、BL 车向这些目标前进, P 车向目标 1 前进。7(b) 除一个目标点外, 所有目标点都已检测到。控制器的合作性质的一个例子可以在 BK 的行为中看到: 它最初伴随着 P 走向 1, 直到一个点, 在这个点上奇奇的效果 (28) 中的 $ijin$ 是这样的, 通过使节点 BK 被“排斥”, 期望的回报被最大化, 这相当于该节点搜索未知目标, 并且实际上检测到一些, 如图 2 所示。相对于图 7(b)。7(a)。沿着同样的路线, 节点 G 在访问目标 4 之后, 被引导远离其他三个节点正在覆盖的区域。尽管该节点无法探测到任何剩余的未知目标, 但在任务期间它将继续处于“搜索模式”。然而, 节点 BL 沿着它的轨迹访问它碰巧检测到的许多目标。

4.2. 覆盖控制任务

正如我们在第 3.1 节讨论的部署问题中已经看到的, 传感器网络的性能对其节点在任务空间中的位置很敏感。当节点是移动的并且数据源目标是未知的或者也是移动的时, 为了满足整个系统目标而部署传感器的问题被称为覆盖控制或主动感测问题[14, 46, 47]。特别是, 必须部署传感器, 以便最大限度地从任务空间提取信息, 同时保持可接受的通信和能源消耗水平。这个问题的静态版本涉及没有任何进一步移动性的定位传感器; 最佳位置可以通过类似于广泛研究的设施位置优化问题的离线方案来确定。动态版本允许传感器的协调运动, 这可以适应任务空间中不断变化的条件, 通常将它们部署到信息密度最高的地理区域。

由于覆盖控制与设施位置优化的相似性, 该问题通常在该框架中考虑。在参考文献中。[14] 作者开发了一种基于沃罗诺伊划分和劳埃德算法的分散覆盖控制算法。在参考文献中。[46] 在参考文献中, 提出了一种覆盖控制方案, 其目的是在某些监视应用中使目标暴露最大化。[72] 基于“虚拟力”的启发式算法被应用于增强传感器网络的覆盖。许多积极的文化参考。[47] 还集中于使用移动传感器跟踪特定目标的问题, 卡尔曼滤波器广泛用于处理观测结果和生成估计值。

已经提出的用于覆盖控制的一些方法假设均匀的感测质量和无限的感测范围。然而, 基于分区的部署方法往往忽略了这样一个事实, 即通过共享多个传感器所做的观察, 可以提高整体感测性能。也有依靠集中控制器来解决覆盖控制问题的努力。正如

我们已经讨论过的, 集中式方法不适合传感器网络的分布式通信和计算结构。此外, 该问题的组合复杂性限制了这种方案在限制传感器网络规模方面的应用。最后, 另一个似乎被忽视的问题是传感器重新定位的成本。传感器的移动不仅影响传感性能, 还会影响传感器网络的其他服务质量方面, 尤其是与无线通信相关的方面: 由于板载功率和计算能力有限, 传感器网络不仅需要传感, 还需要收集和传输数据。因此, 在控制传感器的部署时, 需要同时考虑传感质量和通信性能。

这激发了分布式覆盖控制合作感测的方法[41], 我们将在本节中介绍。现在将使用代表特定事件发生频率的密度函数来对任务空间进行建模(例如, 数据在某一点产生)。在两个极端情况下, 这允许我们对没有目标位置信息的任务空间(使用均匀密度函数)或具有已知位置的任务空间(使用概率质量函数)进行建模。我们假设移动传感器节点具有由概率模型定义的有限范围。在每个移动节点应用部署算法, 使得它最大化随机事件的联合检测概率。我们假设事件密度函数是固定且给定的; 然而, 在任务空间(或我们对任务空间的感知)随时间变化的情况下, 自适应重定位行为自然遵循最优覆盖公式。4.2.1. 任务空间和传感器模型

我们将任务空间建模为一个多面体 R^2 , 其上存在一个事件密度函数 $r(x)$, 它捕捉特定随机事件发生的频率或密度(单位为赫兹=平方米)。 $R(x)$ 满足所有 $x \in R^2$ 的 $r(x) \geq 0$ 和 $\int_{R^2} r(x) dx = 1$ 。

根据应用的不同, $R(x)$ 可能是某一类型的数据源出现在 x 的频率, 也可能是某一变量被检测到的概率(例如, 温度)超过特定阈值。在任务空间中, 有 N 个移动节点位于 s_1, \dots, s_N , R^2 , $i = 1, \dots, N$ 。名词 (noun 的缩写)。当一个事件发生在 x 点时, 它会发出一个信号, 这个信号由 s_i 位置的传感器节点观察到。接收信号强度通常随 $\|x - s_i\|$ (信号源和传感器之间的距离)衰减。类似于参考文献中的模型。[16] 中, 我们用单调递减的可微函数 ϕ_i 来表示这种退化, 它表示传感器 i 检测到事件发生在 x 的概率。

例如, 如果我们假设信号强度随距离多项式下降, 并考虑到环境噪声, s_i 接收的信号强度由 $E_i(x) = \frac{E}{\|x - s_i\|^n + \epsilon}$ 表示, 其中 E 是事件发生时发射的总能量, I 是噪声, n 是衰减系数(通常选择为 $2 \leq n \leq 5$)。如果传感器在 E_i 超过某个阈值时检测到事件, 则 $\phi_i(x)$ 可表示为

$\text{pixProb } k/$ $\text{kn } l > c$ 生命期
 望 大音阶第七
 音
 问题 $i > c k/$
 (=
 knot)
 海里
 生
 命期望
 大
 音阶第
 七音

对于给定的噪声概率分布(例如。加性高斯白噪声),这可以用作传感器模型。或者,具有更简单形式的传感器模型可以是

$\text{pixp0ieikxsik}\delta 31p$

其中检测概率随距离呈指数下降, $p0i$, I 由传感器的物理特性决定。

4.2.2. 最优覆盖问题公式和分布式控制

当将移动传感器节点部署到任务空间时,我们希望最大化检测到事件的概率。这推动了最优覆盖问题的形成。假设传感器独立地进行观测,当一个事件发生在 x 点且被传感器节点观测到时,检测到该事件的联合概率可以表示为

普通

$\text{px}, 1 Y 1 \text{pi}\delta \quad \delta 32p$

$i 1$

最优覆盖问题可以表述为传感器节点在任务空间上的预期事件检测频率的最大化:

应力 $z \text{RxPx}, \text{sdx}\delta 33p$

在这个优化问题中,可控变量是移动传感器在向量 s 中的位置。这个问题可以通过应用一个非线性优化器和一个可以数值计算积分的算法来解决。在这种情况下,需要具有大量计算能力的集中式控制器。在移动

传感器网络中,基站可能是这种控制器的候选。然而,这种解决方案仅适用于有限规模的网络。否则,优化问题的复杂性和通信开销都将使这种集中式方案不可行。

因此,我们将开发一种分布式控制方法来解决最优覆盖问题,而不是使用集中式方案。我们用下列等式表示(33)中的目标函数

$\text{FZ rxpx}, \text{sdx} \quad \delta 34p$

,取 si , $i 1$ 的偏导数时, ..。 n , 我们有 $@ \text{SFi} @ \text{pXi}$,

$\text{sdx}-35p$

rx

$@s$

如果这个偏导数可以由每个移动节点 I 进行局部评估,那么可以应用梯度方法,该方法将节点引向最大化 fSs 的位置。鉴于(32),偏导数

(35)可以改写为

————— $@ \text{Fi } Z \text{rxk } Y1N; K6 I 1$

$\text{pkxdpddi } xx \text{sdii } \text{xxdx}$

$@s$

$\delta 36p$

———其中 dixkx sik 。移动传感器节点很难直接计算(36),因为它需要全局信息,例如整个任务空间上的 $R(x)$ 值和所有其他节点的确切位置。此外,积分的计算仍然是传感器节点要执行的重要任务。为了解决这些困难,我们首先截断传感器模型,并通过应用感测半径来限制其感测能力。该近似值基于物理观察,即当 $\text{di}\delta$ 较大时,大多数传感设备的 $\text{pi}\delta$ 。对于所有 x s.t, 设 $\text{pi}\delta \text{ xP}$, $\text{ddi } xx$ 或 0。 $\text{dixd } DP$

$\delta 37p$

其中 D 表示感应半径。因此, (37)定义了节点 I 的覆盖区域,表示为

(36)byi I F0 xfor all:dixxx2 = Di, 我们可以使用。因为 xi 要替换 0, 所以 dpi Xin = DD x。使用(37)的另一个副产品是紧急情况

的概念和节点检测概率 k 6 的邻域, kx 的必要条件。在(36)中, 大于 0 的点为 2

dkδ xD。如图 2 所示。8, 当节点 I 和 k 之间的距离大于 2D 时, 每

点任何传感器 nodeBandi f l kx: pinkks IX isksatisfieskk 1 < 2 =对于所有 B2Di, kdk 6k x xi2 1 >都不会有影响, ...iD。如果我们定义一个集合; 因此, N, K6π, 那么, x0

(36)中的积分。

在应用(37)和使用铍之后, (36)减少到

$$\frac{1}{V} \int_{V_i} \rho(x) dx = \frac{1}{V} \sum_{j \in N_i} \rho(x_j) \quad (38)$$

δ38p

使(38)可计算的最后一步是将积分求值离散化。如图 2 所示。9, 在覆盖区域 I 上应用网格。因此,

Vi 用 $abD=c$ 表示, 其中 $2VDp$ 是 I 的分辨率 I, 笛卡尔坐标-2V 网格, 带网格。在每个节点的网格上定义 nate 系统, 其原点位于 si, 其轴平行于网格的设置, 单位长度为。在这个局部坐标系中, 让(u, v)表示 x 点的位置。然后, 将(u, v)映射到全局坐标的变换

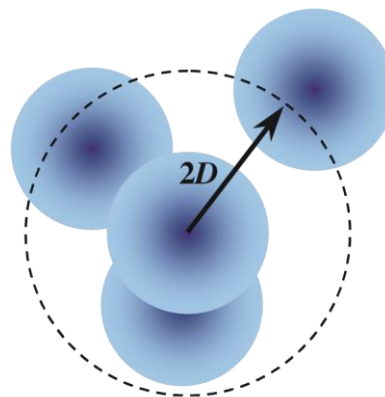


Fig. 8. Defining neighbor sets.

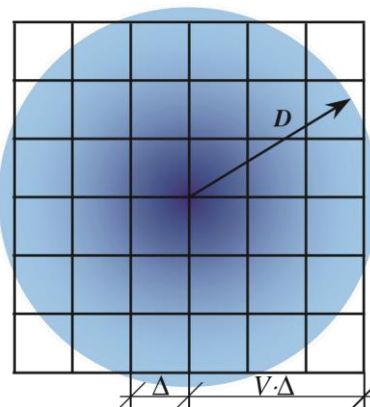


图. 9.使用网格进行离散化。

系统是 x_{si} 或 u_{vT} 。当切换到这个局部坐标系时, (38)中

的项变成

$$\rho(x) \approx \rho(u, v) \quad (39)$$

其中 $\sim I-u, v$ 表示节点 I 在任务空间事件密度上的局部感知(map)。在典型的动态部署应用中, 在部署开始时, 所有传感器节点都以估计的事件密度函数的相同副本开始。随着节点的部署和数据的收集, 单个节点可以通过将新的观察结果合并到其感知中, 并通过与附近的邻居交换信息来更新其本地地图。我们还将(38)中的产品术语改写为

$$\rho(x) \approx \rho(u, v)$$

k2Bi

_____ kYih1 ~ku sk1 si1, v sk2 si2i ~
iđ u, v

2B

其中, $u_{sk1 si1} = v$, $v_{sk2 si2} = v$ 是 x 在第 k 个节点的局部坐标系中的坐标。通过应用网格和坐标变换, (38)可以重写为

@F~ u, v ~

_____ @ _____ si1
2Ux VV Vx VV我pffiuiđ, Vv ~ 20i u, Vu

_____ @ _____
@ SFI 2 2uxV VxV ~ iu, Vp ~ ffiui2 u, vv2 ~
0i u, Vv

đ39p

这些导数可以由移动传感器节点仅使用可用的本地信息来容易地计算。切换到(39)中的局部坐标的一个优点是~ iđ, Vu 和~ 0iđ u, Vu 的值由(u, v)和传感器模型唯一确定。这促使在传感器节点的板上存储器中将~ iđ、Vu 和~ 0iđ u、Vu 存储为两个矩阵。通过获取关键传感器模型参数[例如,。(31)中的 $p0k$ 和 k]并适当地重新调整~ iu、Vu 和~ 0iđ u、Vu, 节点 I 也可以使用存储的矩阵轻松地计算~ iu、Vu。通过这样做, 在重复评估(39)中的计算工作量被显著减少。

上面的梯度信息为移动节点的移动提供了方向。使用该信息的精确方式取决于运动方案的选择。应用梯度法最常见的方法是确定第 I 个移动节点运动轨迹上的下一个航路点

sik 1 ski k___ @ @ SFKi

đ40p

其中 k 是迭代指数, 并且根据标准规则[5]选择步长 k , 以保证运动轨迹的收敛。

计算(39)中所示梯度的计算复杂度取决于网格的规模和相邻集 B_i 的大小。在最坏的情况下, 节点 I 有 N 1 个邻居, 计算 $@F=@_{si}$ 所需的操作数为 $NV2p$ 。最佳情况发生在节点 I 没有邻居的情况下, 相应的复杂度为 $o-v2$ 。在这两种情况下, 复杂度都是 v 的二次型。

4.2.3. 考虑通信成本的最优覆盖问题

除了传感和收集数据, 覆盖控制任务还包括向基站转发数据的任务。假设平坦的网络结构(即,。没有第 2 节中讨论的簇头), 通信成本来自无线传输的功耗。我们将再次使用第 3.1 节中的链路能量模型(6)和(7)。基站位置表示为 $s0 2 R2$, 并且

, 自第 I 个传感器节点的数据速率由 $R1$ 表示, ...。名词 (noun 的缩写)。请注意, ri 被定义为 si 的函数, 因为 I 转发的数据量由检测到的事件数量决定, 而后者取决于节点的位置。这里我们假设 ri 与检测到事件的频率成比例, 即

risi 3 Z rxpix xdxđ41p

其中 3 (比特/检测)是传感器节点检测到事件时转发的数据量。假设 ci 是网络为从节点基站传输一点数据而消耗到的总能量。然后, 最优覆盖问题可以通过结合感知覆盖和通信成本来修正如下:

普通

((

max sw1 RxPx, sdx w2 Xi 1 risi ci))

đ42p

其中 $w1$ 、 $w2$ 是加权因子。人们可以把 $w1$ 看作是探测到一个事件的奖励, $w2$ 看作是消耗一个单位能量的代价。

让我们用来表示通信成本

普通

Xi 第一中学đ43p

因此,

—— ——— ——— ——— ——— 普通 k
@ck26 i @eih i 许继j @ejhi j 375⁷5

r @s z @s z @s 048p
k 1 我 四 fjj 1g

通过组合(39)、(44)、(45)和(48), 传感器节点 I 可以局部导出 $@J=@si$ 。然后, 每个传感器使用梯度信息来指导运动控制, 如(40)所示, 用 $@J=@ski$ 代替 $@F=@ski$ 。通过适当选择步长, 移动传感器将最终收敛到最大值点。

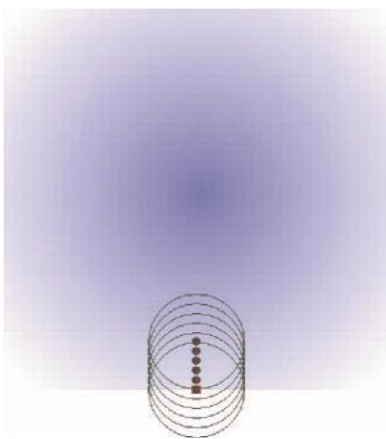


图. 10. 六个移动传感器节点的协作覆盖控制问题。

4.2.4. 模拟示例

上面的分布式部署算法已经在基于 Java 的仿真环境中实现。在图 1 所示的例子中, 10, 一个由六个移动传感器组成的小组正在等待被部署到 40 × 40 米的任务空间。事件密度函数 $R(x)$ 由下式给出:

$$r_{xR0} k_{x x0k} \quad 049p$$

其中 $R0 \ 3:0, \ 0:1, \ x0 \ 0, \ 20$ 。根据(49), 点 $x \ 2$ 的事件密度随着 x 和任务空间的中心点 $x0$ 之间的距离线性下降。

在时间 $t \ 0$, 移动传感器节点位于任务空间的原点附近。每个移动节点配备有传感器, 该传感器的检测概率由式(31)表示, 其中 $P01:0, \ i \ 1:0$ 表示所有 $i \ 1, \ \dots$ 。名词 (noun 的缩写)。传感半径为 $D \ 5:0$,

如图中黑色圆圈所示。10. 移动节点还具有无线传输器, 其能量消耗由(7)建模, 其中 $1 \ 0:01 \ nJ=bit, \ 2 \ 0:001 \ nJ=bit=m4$ 和 $n \ 4$ 。在任务空间中, 有一个无线电基站位于 $s0 \ 0, \ 0$ (由图中的红方标记)。10. 当传感器检测到一个事件时, 它会收集 32 位数据并将其转发回基站[因此 $3 \ 32 \ in \ (41)$]。

我们将通过观察两种不同的情况来给出这个覆盖控制问题的模拟结果。在第一种情况下, 不考虑通信成本, 这对应于最优覆盖公式(42)中的 $w1 > 0, \ w2 \ 0$ 。在第二种情况下, 传感覆盖和通信成本都包括在内($w1, \ w2 > 0$)。

图 11 展示了在第一种情况的部署过程中拍摄的几个快照。从图开始。11(a), 六个节点组成一个编队, 向任务空间中心移动。在其运动过程中, 编队不断进化, 使节点扩大了感知的整体面积, 同时共同覆盖了事件密度高的点。此外, 节点还与基站保持无线通信。这显示在图. 11 作为传感器节点和基站之间的链路。节点团队最终收敛到一个静止的编队, 如图. 11(d)。从这个对称的编队中可以看出, 所有六个节点都在共同感知事件密度最高的区域。我们通过在(42)中设置 $w2 \ 0:0008$ 和 $w1 \ 1 \ w2$, 将通信成本纳入最佳覆盖范围公式。相应的部署模拟结果如图所示。12. 与第一种情况相比, 在网络形成中可以观察到一个关键的差异: 节点不仅向事件密度高的区域移动, 而且保持到基站的经济的多跳路径。如图所示, 网络到达固定部署。12(d)。与第一种情况的最终形成相反[图. 11(d)], 只有四个节点聚集在任务空间的中心周围。另外两个节点作为中继站, 支持与基站的通信。

图 13 和图 14 展示了与前面显示的两种情况相关的传感覆盖范围和通信成本。图 13 描述了当节点向最优部署移动时, 感知覆盖范围的变化(通过事件检测的预期频率来测量)。一个直接的观察是, 在这两种情况下, 传感覆盖随着地层的演化单调增加。如果在部署期间不考虑通信成本, 感应覆盖在 91.47 赫兹达到最大值。然而, 在考虑通信成本的情况下, 当传感器达到最佳部署时, 每秒只能检测到 84.74 个事件, 这对应于 7.36% 的覆盖损失。这种覆盖损失是自然的, 因为最佳覆盖公式(42)实际上是用较低的通信成本来交换感知覆盖。这种权衡可以通过查看图 2 来进一步研究。14. 如果考虑通信成本, 最终功耗为 $8:01 \ 103 \ nW$ 。与第一种情况的通信成本相比($1:877105 \ nW$), 节电 95.73% 。在开发这种分布式协作覆盖控制方法时, 我们没有明确解决的一个问题是所涉及的基于梯度的算法的全局最优性。这仍然是一个正在进行的研究课

图 11. 不考虑通信成本的传感器节点部署。

题。特别是，人们期望全局最优性与事件密度函数和所采用的传感器模型的特性有内在联系。

5. Future Research Directions

We have attempted in this paper to categorize and describe major design and control problems related to sensor networks, as well as to cooperative control as it pertains to sensor networks with mobile nodes. For some of these problems, we have also presented selected solution approaches, some already published in the literature and some novel. Open research issues were mentioned or briefly hinted at throughout the previous sections and we summarize some of them here. First, we mentioned in Section 2 that a potentially better structure for sensor networks is a hierarchical one, making use of clusterheads acting as intermediate processing nodes between data sources and a base station. The presence of clusterheads implies different approaches for some of the problems we have discussed; for example, deployment may be quite different if a clusterhead can “aggregate” data from neighboring nodes and avoid the need for these nodes to use up energy for direct communication with the base station. Second, we also briefly mentioned that one form of power control is to switch the state of sensor nodes between “sleeping” and “sensing” or “relaying” (see Section 2). Formulating such a switching control problem and devising solution methods dependent on the information available to each node is an interesting direction for research. Third, the problem of location detection when nodes are mobile is also one that was only briefly mentioned (see Section 4) and it clearly deserves in-depth study. In the context of cooperative control, we saw that one can define different types of sensor network missions, two classes of which were discussed in Section 4. Some of

the most pressing technical concerns here are related to the properties of the associated optimization problems involved; particularly, the questions of local versus global optimality and the need for mechanisms consistent with the distributed nature of sensor networks. Although many of the open questions above are technically challenging in their own right, we would like to bring this section to a close by turning our attention to some more fundamental issues of much broader long-term impact where progress has been minimal. These issues are closely related to the convergence of communication, computing and control, which brings together three disciplines that often use different modeling paradigms and different ways of thinking. Naturally, bridging the gaps between them is a real challenge. One of these issues concerns the combination of asynchronous and synchronous modes of operation in a common system setting. Although the gathering of data is inherently asynchronous (owing to multiple sensor nodes operating in different temporal and spatial scales), the processes of data fusion and control are traditionally based on a synchronized time structure. This is one manifestation of the difference between time-driven and event-driven behavior; designing a system environment where both can coexist remains an open problem. The traditional setting of differential equation models and time-driven digital sampling provides a comfortable infrastructure for communication and control methodologies, but that is being challenged by computational models which rely on event-driven processes and by the simple intuitive observation that time-driven sampling is inherently wasteful. The limited resources of sensor network nodes emphasize the need to switch to a more efficient event-driven sampling approach, where data are collected only when “something interesting” happens. To do so, however, requires new sampling mechanisms and possibly new data collection hardware as well.

未来研究方向测覆盖范围的比较。

图 14。通信成本比较。

A second research issue of particular importance to control theory is the obvious shift from sensor-poor to data-rich control systems. Traditional feedback control systems have been designed under the premise that sensors are few and expensive and much of the “intelligence” in such systems is concentrated on compensating for limited state information. The sudden wealth of sensor data (subject, of course, to bandwidth and delay limitations) shifts the need for “intelligence” towards processing potentially huge amounts of data and combining model-based methodologies with increasingly data-driven ones. To date, there appears to be a significant gap between schools of thought advocating one versus the other approach. One would expect that a combination can enable us to exploit advantages of both.

在本文中,我们试图对与传感器网络相关的主要设计和控制问题进行分类和描述,以及与具有移动节点的传感器网络相关的协同控制问题。

对控制理论特别重要的第二个研究问题是从传感器贫乏到数据丰富的控制系统的明显转变。RO under grant DAAD19-01-0610, and by a grant from Honeywell Laboratories.

承认

1. 部分由 DMI-0330171 基金资助的国家科学基金会资助, GANTSF 49620-01-0056 AnD 49620-01-1-0348 基金资助的 AFOSR 资助, DAAD19-01-0610 基金资助的 RO 资助, 以及霍尼韦尔实验室的资助。引用 4; 11(6): 6–28
2. Al-Karaki JN, Kamal AE. 'D, Mejia-Alvarez P. 无线传感器网络路由技术综述。无线传输(to appear) 3. Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E. A survey on sensor networks. IEEE Commun Mag Commun 2004;11(6): 6–28
4. 艾登·赫、梅尔赫姆·雷、莫斯·德、梅希亚-阿尔瓦雷斯·普。周期性实时任务的功率感知调度。传感器网络综述。Idiz IF, Su W, Sankarasubramaniam Y, Cayirci E. 2002;40(8): 102–114 ear programming. Athena Bhardwaj M, Chandrakasan A, Garnett T.
5. Bertsekas DP. am S, Kumar PR. 非线性的编程。ons in the convergence of control with communication and computation. 雅典娜(智慧与技艺的女神)occeedings of the 44th IEEE conference on decision and control, 2005, submitted for publication
7. 《科学》, 1995 年 ard NE. Vehicle networks for gradient descent in a sampled environment. In: Proceedings of the 41st IEEE conference on decision and control. 2002, pp 112–117

8. 巴利加 G, 格雷厄姆 S, 库马尔 PR. ighanbari M, How JP. Cooperative path planning for multiple UAVs in dynamic and uncertain environments. 摘自: 2005 年第 44 届美国电气和电子工程师协会决策与控制会议记录, 已提交出版
9. 巴赫迈尔 R, 伦纳德 NE. 采样环境中梯度下降的车辆网络。ications. 摘自: 第 41 届美国电气和电子工程师学会决策与控制会议录。
10. 2002 年, 第 2816–2822 页, 怎么 JP. 动态不确定环境下多无人机协同路径规划。
11. Buttazzo GC. ndras CG, Pepyne DL. 硬实时计算系统: 可预测的调度算法和应用。克鲁瓦学术出版社, 马萨诸塞州诺维尔, 1997 年 13
12. 冲 c. s CG, Li W. 传感器网络: 演进、机遇和挑战。ntrol problems. Proc IEEE 2003;91: 1247–1256 IEEE conference on decision and control. 2002, pp 3760–3765
13. 赵·YC, 卡珊德拉·CG, 佩皮恩·DL. 一类混合系统最优控制的前向分解算法。国际鲁棒非线性控制 2001; 11(5): 497–513 Kluwer Academic Publishers, 2000, pp 35–53
14. 卡珊德拉 CG, 李 w. Karatas T, Bullo F. 求解某些合作控制问题的滚动时域方法。摘自: 第 41 届美国电气和电子工程师学会决策与控制会议录。
15. 克拉夫 BT. , Pachter M, Rasmussen S. In: Murphey R, Pardalos PM (编辑)。协同控制和优化。
16. 科尔特斯 J, 马丁内斯 S, 卡拉塔斯 T, 布洛克 f. n P, Saluja K. 移动传感网络的覆盖控制。on. IEEE Trans Robotics Autom 2004;20(2): 出现 p on wireless sensor networks and applications. Atlanta, GA, 2002, pp 42–48
17. 钱德勒公关, 帕切特 M, 拉斯姆森 s. 无人机协同控制。ic approaches for control strategies in a military operation. 2001 年美国控制会议记录。nd control. 2001 年, 第 50–55 页 33
18. Clouqueur T, Phipatanasuphorn V, Ramanathan P, Saluja K. 用于目标检测的传感器部署策略。In: 第一届 ACM 无线传感器网络与应用国际研讨会论文集。
19. 克鲁兹 JB, 思曼 MA, 加契奇 A, 刘 y. tive control for dynamic unreliable resource allocation. 军事行动中控制策略的横向移动理论方法。常, 塔索拉斯。ence on decision and control. 无线传感器网络中的最大寿命路由。
20. castan o=NdA, Wohletz JM. 动态不可靠资源分配的模式预测控制。在: 会议记录 n: Proceedings of the 43rd IEEE conference on decision and control. 2004, pp 1995–2002
21. 第 41 届 IEEE 决策与控制会议。2002 年, 第 3574–3579 页 tochastic time-varying environment. In: Proceedings of the 43rd IEEE conference on decision and control. 2004, pp 3357–3363
22. 邓巴 WB, 默里 RM. Polycarpou MM. 多飞行器编队的后退地平线控制: 分布式实现。eding-horizon approach. 摘自: 第 43 届美国电气和电子工程师学会决策与控制会议录。2004 年, 第 1995–2002 页
23. 弗拉索利 E, 布洛克 f. Sparks AG. 随机时变环境中车辆路径的分散算法。es. 摘自: 第 43 届美国电气和电子工程师学会决策与控制会议录。2004 年, 第 3357–3363 页

24. Franco E, Parisini T, Polycarpou MM, Lin D. 具有延迟信息交换的离散时间智能体的协同控制:一种滚动时域方法。摘自:第 43 届美国电气和电子工程师学会决策与控制会议录。
25. Finke J, Passino KM, Sparks AG. 基于任务负载平衡的网络化无人驾驶汽车协同控制。2003 年,第 31-36 页工程师学会决策与控制会议录。
26. 甘内桑 D, 戈文丹 R, 申克 S, 埃斯特林 D. 无线传感器网络中高效、节能的多路径路由。移动计算通信 2001 年版; 4(5): 11-25。sion and control. 2002, pp 2848-2853
27. 古普塔 P, 库马尔 P., Martinez S, Bullo F, Cortes J. 渐近连通性的临界功率。le problems and distributed algorithms. 摘自:第 37 届美国电气和电子工程师学会决策与控制会议录。005. submitted for publication
28. Gazi V, Passino KM, akasan A, Balakrishnan H. Energy-efficient communication protocols for wireless microsensor networks. 群居觅食群的稳定性分析:引诱/驱避剖面的综合效应。
29. 甘古利 A, 苏斯卡 S, 马丁内斯 S, 布洛克 F, 科尔特斯 j. 传感器网络中的肿瘤选择性运动:样本问题和分布式算法。摘自:2005 年第 44 届美国电气和电子工程师协会决策与控制会议录。提交出版
30. 无线微传感器网络的节能通信协议。
海因策尔曼 w. 无线网络专用协议体系结构。麻省理工学院博士论文, 2000 年 8(12): 1702-1714
31. 行 我, 基洛夫斯基 d, 夸脱 (quart 的缩写) g, 魔芋 m, dance and formation switching on Riemannian manifolds. In: Proceedings of the 40th IEEE conference on decision and control. 2001, pp 1071-1076
32. Srivastava MB, C, Govindan R, Estrin D. 变压堆芯系统的功率优化。digm for sensor networks. 集成电路系统 1999 年计算机辅助设计; 18(12): 1702-1714
33. 胡 J, 萨斯特里 s. H, Shin KG. 黎曼流形上的最优避碰和信息转换。ectures. 摘自:第 40 届美国电气与电子工程师学会决策与控制会议录。-time systems, vol 1586. Springer Verlag, 1999, pp 363-374
34. 在华盛顿特区, 在华盛顿特区, 在华盛顿特区。摘自:2000 年美国电气和电子工程师协会/美国计算机学会移动通信会议录
约翰松、容恩、申克。
35. 约翰逊数据库, 马尔茨数据库。artel CU. 无线自组织网络中的动态源路由。c tasks. 在 Imielinski, Korth(编辑)。real-time systems symposium. 移动计算, 第 353 卷。克鲁瓦学院
36. 出版社, 1996 年 GPSR: greedy perimeter stateless routing for wireless networks. In: Proceedings of the 6th annual international conference on mobile computing and networking 2000, pp 243-254
37. Jeffay K, Stanat KF, Martel CU. n SL. 周期性和零星任务的非抢占式调度。for hard real-time systems. 摘自:《美国电气和电子工程师协会实时系统研讨会论文集》。
38. 卡 普 B, 功 HT. A Cooperative Receding Horizon Controller for Multi-Vehicle Uncertain Environments. GPSR:贪婪的周边国家无线网络路由。
金威, 申德, 云 HS, 金杰, 闵 SL。
39. Li W, Cassandras CG. 用于多车辆不确定环境的操作员接收区域控制器。美国电气和电子工程师协会运输自动控制 2004; 提交给 5), to appear
40. 出版 ndras CG. A minimum-power wireless sensor network self-deployment scheme. In: Proceedings of the IEEE wireless communications and networking conference, 2005
41. 李 W, 卡珊德拉 CG. 自主飞行器任务的集中式和分布式协同后退视界控制。《数学计算模型》(2005), 即将出版 5, submitted for publication
42. 李 W, 卡珊德拉 CG. 一种最小功耗无线传感器网络自部署方案。f multi-vehicle systems. 摘自:2005 年美国电气和电子工程师协会无线通信和网络会议论文集 3769
43. 李 W, 卡珊德拉 CG. a CS. 传感器网络的分布式协作覆盖控制。ms. 2005 年, 提交出版
44. 丽安·弗, 默里·r. CG. 多车系统协同路径规划的实时轨迹生成。摘自:第 41 届美国电气和电子工程师学会决策与控制会议录。
45. 林赛 S, Raghavendra CS. 传感器信息系统中的能效收集。al-time constraints. In: 航空航天会议记录。rence on decision and control and European Control Conference, Seville, Spain, 2005, submitted
46. 苗 L, 卡珊德拉 CG. nfar F, Potkonjak M, Srivastava MB. 某些离散事件系统静态控制策略的最优性。出现了电气和电子工程师协会的自动变速器控制。387
47. 苗 L, 卡珊德拉 CG., Bruyninckx H, Gadeyne K. Active sensing for robotics-a survey. 一类具有实时约束的离散事件系统的滚动时域控制。Borovets, Bulgaria, 2002, pp 316-324
48. 梅格迪西恩 S, 库尚法尔 F, 波特魔芋 M, 斯里瓦斯塔瓦 MB. 无线自组织传感器网络中的覆盖问题。摘自:《电气和电子工程师协会信息通信会议录》。
49. 《机器人的主动感测-一项调查》。摘自《第五届国际数值方法及应用会议论文集》, 保加利亚博罗维特, 2002 年, 第 316-324 页
50. 迈尔·DQ. dras CG. 非线性系统的滚动时域控制。d systems with realtime constraints. 1990 年美国电气和电子工程师学会汽车控制; AC-35(7):814-824。ecision and control, December 2004, pp 1472-1477
51. Mhatre 副总裁, 罗森伯格 C, 考夫曼 D, 马祖姆达尔 R, 什洛夫 n. 具有生命周期约束的最小成本异构传感器网络。IEEE 跨移动计算 2005; 4(1): 4-15
52. 毛 JF, 赵 QC, 卡珊德拉 CG. 实时约束下功率受限系统的最优动态电压调节。摘自:《第 43 届美国电气与电子工程师学会决策与控制会议论文集》, 2004 年 12 月, 第 1472-1477 页
53. 珀金斯 CE, Bhagwat P. 移动计算机的高度动态目标排序距离矢量(DSDV)路由。in: ACM SIGCOMM 会议录。1994 年, 第 234-244 页
54. 伯林, 伯德, 布罗德森。H. 动态电压缩放和低功率微处理器系统的设计。In: 功率驱动微体系结构研讨会论文集(ISCA98), 1998, pp 251-259
55. 公园 VD, Corson MS. i W, Starobinski D. 一种适用于移动无线网络的高度自适应分布式计算算法。licit fairness constraints. 摘自:《电气和电子工程师协会信息通信会

- 议录》。erence on decision and control. 1997 年, 第 1405-1413 页
56. Pouwelse J, Langendoen K, Sips H, Pachter M, Liu Y, Yang Y, Flint M, Baum M. 2001 年, 第 251-259 页 In: 第七届移动计算和网络国际年会论文集. 00, pp 233-269
 57. Paschalidis I Ch, Lai W, Starobinski D. 摘自: 第 43 届美国电气和电子工程师学会决策与控制会议录。
 58. 帕西诺·K, 波利卡普·M, 雅克·德, 帕切特·M, 刘毅, 杨毅, 弗林特·M, 鲍姆·M. 自主飞行器的协同控制。在墨菲, 帕尔多斯总理(编辑)。协同控制和优化。
 59. Papadimitriou CH, Steiglitz K. 算法和复杂性. em. 多佛出版社, 1998 年 EEE real-time systems symposium, work-in-progress session, 2000
 60. 理查兹 A, 怎么 j. s N, Min R, Sinha A, Wang A, Chandrakasan A. 协同无人机的分散模型预测控制。rgy-efficient wireless sensor networks. 摘自: 第 43 届美国电气和电子工程师学会决策与控制会议录。
 61. Swaminathan V, Chakrabarty K. 能量感知嵌入式系统的实时任务调度。摘自: 2000 年美国电气和电子工程师协会实时系统研讨会论文集, 进行中会议
 62. 2001 年美国计算机学会会议录. 2001 年, 第 272-287 页
 63. 辛格 L, 富勒 j. nt R, Shroff N. 利用非线性最大功率控制生成无人机在复杂地形中的轨迹。2001 年, 第 2301-2308 页
 64. Shah R, Rabaey J. W. 低能耗自组织传感器网络的能量感知路由。
 65. Shakkottai S, Srikant R, Shroff N. 不可靠的传感器网络: 覆盖范围、连通性和直径。摘自: 《电气和电子工程师协会信息通信会议录》。2003
 66. 香农 CE, 韦弗 w. bhakar B, Gamal AE. 通信的数学理论 r a wireless link. IEEE/ACM Trans Networking 2002; 10: 487-499
 67. Singh S, Woo M, Raghavendra CS. 移动自组网络中的功率唤醒。or networks. 在: 美国电气和电子工程师协会/美国计算机学会会议录. ce on decision and control. 1998 年, 第 181-190 页
 68. 云萨尔-比伊科勒, 普拉巴卡尔, 加马。2002 年美国电气和电子工程师协会/美国计算机学会传输网络; 10: 487 - 499
 69. 吴 X, 卡珊德拉 CG. Zhang L. 传感器网络路由的最大时间最优控制方法。摘自: 第 44 届美国电气与电子工程师学会决策与控制会议录。munications and networks. 2005 年, 已提交 9
 70. 伍德 AD, 斯坦科维奇 JA. n D. 传感器网络中的拒绝服务。a recursive data dissemination protocol for wireless sensor networks. 计算机 2002; 35(10) A Computer Science Department Technical Report UCLA/CSD-TR-01-0023, 2001. Available at: <http://citeseer.nj.nec.com/yu01geographical.html>
 71. 叶 F, 陈 A, 刘 S, 张 l. na VK. 大型传感器网络中最小开销转发的可扩展解决方案。In: 第十届国际计算机通信与网络会议录。
 72. 于洋, 葛文丹 R, 埃斯特林 d. 地理和能量软件路由: 无线传感器网络的递归数据传播协议。3-1303