

基于网络编码的无线传感器网络数据广播算法

肖玲, 李仁发, 罗娟

(湖南大学 计算机与通信学院, 湖南 长沙 410082)

摘 要: 无线传感器网络中 sink 节点需要经常进行广播, 如代码更新、参数设置等。把流内随机网络编码应用在 sink 节点的广播中, 提出了 E-Code 算法。针对传感器节点的存储和计算能力有限的特点, 改变网络编码的系数向量选取方式, 以减少编解码的复杂性, 并设计了相应的转发机制来提高算法的有效性。分析和实验表明, E-Code 在节点收到相同数目的源数据包条件下, 所需的总传输次数更少, 从而达到节省节点能耗的目的。

关键词: 网络编码; 无线传感器网络; 广播; 能量有效

中图分类号: TP393

文献标识码: A

文章编号: 1000-436X(2010)9A-0090-07

Broadcast algorithm based on network coding in wireless sensor networks

XIAO Ling, LI Ren-fa, LUO Juan

(School of Computer and Communication, Hunan University, Changsha 410082, China)

Abstract: Broadcast from sink, such as code updates, was frequent and expensive in the wireless sensor network applications. E-code was presented, which was a reliable and efficient one-to-all broadcast protocol based on intra-flow network coding. The key idea is change the way of choosing coefficients of the linear combination to save the storage space of node and reduce the complexity of computing and develop a packet forwarding scheme for wireless sensor networks to improve the efficiency of the algorithm. E-Code performs better in terms of the average overhead of transmissions to achieves more power saving.

Key words: network coding; wireless sensor networks; broadcast; energy efficiency

1 引言

无线传感器网络已被广泛应用于军事、环境监测、医疗护理、建筑物状态监控等很多方面。在传感器网络中, sink 节点经常需要对所有节点进行广播, 比如应用程序开发与调试、代码更新、参数设置、任务分配等。本文所讨论的广播是 sink 节点将数据包传输给网络中所有节点的一种操作。该广播必须是可靠、快速, 并且尽可能地减少能量消耗。

网络编码理论^[1]融合了编码和路由技术, 允许

网络中间节点在传统数据转发的基础上参与编码, 具有提升网络吞吐量、改善网络负载平衡、提高带宽利用率、节省无线网络节点能量消耗等优点。

在无线传感器网络中, 无线链路的不可靠性和物理层的广播特性非常适合使用编码的方法。如图 1 所示, sink 节点广播 4 个包 (p_1, p_2, p_3, p_4) 到所有节点。各个不同节点接收到的数据包是相互独立的, 每个节点都有丢包的情况发生(各链路丢包率为均 0.5)。假设每个节点收到了如图 1 所示的数据包, 即节点 1 收到了 p_1 和 p_2 , 丢失了 p_3 和 p_4 , 节点 2

收稿日期: 2010-06-06

基金项目: 国家自然科学基金资助项目(60903019)

Foundation Item: The National Natural Science Foundation of China (60903019)

收到了 p_2 和 p_3 , 丢失了 p_1 和 p_4 , 节点 3 收到了 p_3 和 p_4 , 丢失了 p_1 和 p_2 。在没有网络编码的情况下, sink 节点不得不重传所有的 4 个包。但是如果利用网络编码, 只需要再传送 2 个编码后的数据包。例如 sink 节点可以发送 $p'_1=p_1+p_2+p_3+p_4$ 和 $p'_2=p_1+2p_2+3p_3+4p_4$ 。尽管每个节点丢失的是不同的数据包, 但它们都可以顺利地解码出 4 个原始的数据包。例如, 节点 1 收到了 p_1 、 p_2 、 p'_1 和 p'_2 , 通过高斯消元法就可以解的 4 个原始包 (p_1, p_2, p_3, p_4)。在这个例子中, 利用网络编码减少了传输的次数, 从原来的 8 次减少到 6 次。直观地看, 发送数据包所消耗的能量与数据包的发送次数在一定程度上是成比例的。减少数据包的传输次数就可以节省能量消耗。若更进一步, 在 sink 节点发送源数据包的时候就考虑采用网络编码, 比如 sink 节点最初不是广播 4 个包 (p_1, p_2, p_3, p_4), 而是先利用网络编码发送 6 个编码包 (因为考虑了丢包的存在, 所以发送大于 4 个的编码包, 避免了重传), 如 p'_1 和 p'_2 ($p'_1=p_1+p_2+p_3+p_4$, $p'_2=p_1+2p_2+3p_3+4p_4$) 等, 同样因为链路的不可靠性会造成节点丢包, 但每个节点只要收到其中的任意 4 个满足系数向量线性独立的编码包, 利用高斯消元法就能顺利解码出 4 个原始包 (p_1, p_2, p_3, p_4)。使用网络编码后, 因为节点收到任意一个已编码的数据包都具有同等概率的作用, 从而增强了网络的容错性和顽健性。

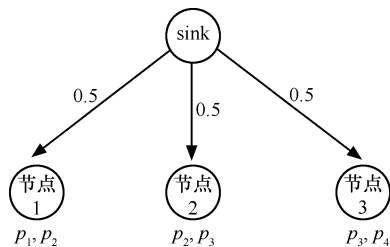


图1 利用网络编码可减少传输次数

本文首次将随机网络编码^[3]的思想应用到 sink 节点的数据广播中。在 sink 节点进行流内网络编码, 即对要发送的数据包进行编码后传送, 中间节点可进一步对其进行编码。将网络编码与广播特性相结合, 通过对接收的多个数据包进行编码信息融合, 增加单次传输的信息量, 从而减少数据包的传输次数, 降低无线传感器网络信息传输过程中的能量消耗, 从而节省节点能耗。考虑到在实际应用中, 传感器节点失效或传输链路不稳定都导致丢包情况的发生, 而使用网络编码后, 节点收到同组里的任

意一个数据包都是相同的效果, 从而增强了网络的顽健性。

2 相关研究

本文研究的 sink 节点的数据分发是一种将数据包传输给网络中所有节点的一对多的广播方式。与其相关的研究包括 2 个方面: 一个是广播通信, 另一个是网络编码。

广播通信是无线网络中的一种很普遍的通信方式。洪泛法是简单易行的广播, 但盲目洪泛会造成许多信息重复地被传送而产生冗余、竞争以及碰撞, 严重时会造成广播风暴。洪泛法会导致巨大的能量开销, 直接影响到无线节点电池的寿命, 因此如何利用现有的网络资源, 减少广播开销, 提高能量利用率成为研究的热点之一。用路由的方式实现最小化能量广播是非常困难的, Zagalj 等人^[3]证明最小化能量广播问题是 NP 完全问题。概率式洪泛法^[4]是节点按照预选确定的概率进行广播。这种广播算法的难度就在于不好控制转发概率值的大小。确定式洪泛法^[5]根据邻居节点信息确定给出信息的转发节点, 即在保证周围节点最大程度接收信息的前提下, 尽量减少转发节点的数量。

在网络编码提出后, 研究者开始利用网络编码来解决广播通信中的能量有效性问题。Fragouli 等^[6]首先结合网络编码, 针对简单圆形拓扑和网格拓扑的无线全网广播通信提出了一种最小化能量广播方法, 并用理论证明了该方法所获得的能量有效性。随后又对随机拓扑的无线网络进行了研究, 分析了节点能量传输半径、节点丢包率等因素对最小化能量广播问题的影响。然而他们研究的是网络中每个节点都对其他所有节点广播的多对多广播, 不太适合本文中研究的 sink 节点对所有节点广播的这种一对多的情况。

网络编码也被用在单播和组播中。Katti 等^[7]首次提出一种提高无线网络单播通信吞吐量的网络编码方法 COPE, 并给出了它的一个具体实现方案。Chachulski^[8]等提出了机会路由与流内网络编码相结合的 MORE, 以增大网络吞吐量为目的。CodeCast^[9]利用流内网络编码并以尽可能减少网络延时为目的。Yang 等^[10]提出了一种无线 Mesh 网络中一对多的可靠广播算法 R-Code, 主要针对网络丢包率较高情况下通过建立最小代价生成树来实现的可靠性广播。这些算法利用网络编码所达到的目

的各不相同，而本文利用网络编码主要是以减少节点能量消耗为目的。另外这些算法的复杂度和计算代价太高，不适于计算和存储能力有限的无线传感器网络节点。

与本文研究最接近的是 I-Hong 等人提出的用于传感器网络 sink 节点广播算法 AdapCode^[11]，其核心思想是节点根据邻居节点的数目确定编码机制，即编码源数据包的个数和发送编码包的个数。sink 节点直接发送若干个源数据包，中间节点都是收到了足够数量的编码包并成功解码成源数据包后，再进行编码，向下一节点转发。与 AdapCode 不同的是，本文提出的 E-Code 核心思想是在 sink 节点就进行编码，中间节点对已编码包进行再编码后转发。实现编码方式的不同将导致节点的转发机制等问题的处理也会不同，如 E-Code 是当节点能成功解码包后就不再转发包了，而 AdapCode 是节点成功解码出源数据包后再进行编码并转发。

3 基于网络编码的广播算法 E-Code

3.1 基本思想

采用流内随机网络编码 (intra-flow network coding) 的思想，sink 节点将要发送的数据包先进行编码后再发送出去。所有其他节点收到足够的已编码数据包后进行再编码并转发出去，当节点收到足够的编码包后解码出原始的数据包。具体做法如下。

Sink 节点：当 sink 节点发送数据时，将所要发送的原始数据分成若干个批次，每批次由 m 个源数据包组成。这 m 个数据包记为 x_1, x_2, \dots, x_m ，并赋予相同的批次标识。假设每个数据包为 L bit，当它与要组合的数据包长度不同时，较短的信息附加额外一串“0”。选取 m 个数，用这 m 个数作为编码系数，对原始的 m 个数据包进行线性编码，把原来的 m 个数据编码成新的 n 个数据包 y_1, y_2, \dots, y_n ，记编码第 i 个数据包时所使用的 m 个编码系数为 $g_{i1}, g_{i2}, \dots, g_{im}$ ，则编码的公式为

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} g_{11} & \dots & g_{1m} \\ \vdots & \ddots & \vdots \\ g_{n1} & \dots & g_{nm} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} \quad (1)$$

Sink 节点在对数据进行编码后，生成 y_1, y_2, \dots, y_n 个数据包，然后再把这一组数据包的批次标识与每一个数据包的编码向量作为头部信息添加到数据

首部，将数据包发出。这种把系数向量随包传送最初来自于 Chou^[12]的思想。sink 节点的整个编码过程中的数据变化如图 2 所示。

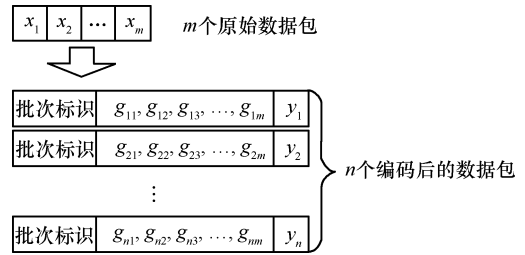


图 2 使用网络编码后的数据包

Sink 节点发送完同一批次的 n 个编码包后，停顿 T_{ms} 时间，来确保有足够的活跃节点收到这些编码包。 T_{ms} 过后，sink 节点再进行下一批次数据包的发送。

普通传感器节点：如图 2 所示，节点接收到编码包后，存储编码向量和编码后的结果，以行向量的形式，存储在解码矩阵中。最初，解码矩阵中只包含未经该节点编码的包以及与之对应的编码向量，否则为空。当接收到一个已编码包后，会从中抽取它的编码向量以及编码结果，放入到解码矩阵中。所收到的某一个编码包如果可以增加矩阵的秩，则称之为更新包。

每个接收节点收到一个编码包后，首先进行是否是更新包的判断。如果是更新包，则将新数据包存入缓存。当收到 n 个更新包后，再将这 n 个更新包进行再编码，随机选取编码系数生成一个新的编码包，以一定的概率 p 向下一节点广播出去。如果不是更新包，则丢弃该包，也不再继续向下一节点广播。

假设节点 r 进行再编码后的 k 个数据包为 y'_1, y'_2, \dots, y'_k ，记节点发送出第 i 个再编码的数据包时选取的系数为 $g'_{i1}, g'_{i2}, \dots, g'_{im}$ ，则节点 r 对收到的 n 个已编码数据包进行再编码可由式(2)表示。节点对它所收到的更新包进行再编码的意义在于通过这样的多次编码可进一步降低编码数据包的线性相关性，增大邻居节点收到更新包的机率。

$$\begin{bmatrix} y'_1 \\ y'_2 \\ \vdots \\ y'_k \end{bmatrix} = \begin{bmatrix} g'_{11} & \dots & g'_{1m} \\ \vdots & \ddots & \vdots \\ g'_{k1} & \dots & g'_{km} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad (2)$$

由式(1)和式(2)可得出节点 r 再编码后的数据

包 $(y'_1, y'_2, \dots, y'_k)$ 与源数据包 (x_1, x_2, \dots, x_m) 的关系如式(3), 新的编码系数向量 h_{ij} 如式(4), 用这组编码系数作为编码后的新系数向量放在已编码数据包的首部。

$$\begin{bmatrix} y'_1 \\ y'_2 \\ \vdots \\ y'_k \end{bmatrix} = \begin{bmatrix} h_{11} \cdots h_{1m} \\ \vdots \\ h_{k1} \cdots h_{km} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} \quad (3)$$

$$\begin{bmatrix} h_{11} \cdots h_{1m} \\ \vdots \\ h_{k1} \cdots h_{km} \end{bmatrix} = \begin{bmatrix} g'_{11} \cdots g'_{1n} \\ \vdots \\ g'_{k1} \cdots g'_{kn} \end{bmatrix} \begin{bmatrix} g_{11} \cdots g_{1m} \\ \vdots \\ g_{n1} \cdots g_{nm} \end{bmatrix} \quad (4)$$

当节点收到属于同一批次的 m 个线性独立的数据包后, 通过高斯消元法, 如式(5)就可以恢复出原始 m 个数据包。

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} = \begin{bmatrix} h_{11} \cdots h_{1m} \\ \vdots \\ h_{m1} \cdots h_{mm} \end{bmatrix}^{-1} \begin{bmatrix} y'_1 \\ y'_2 \\ \vdots \\ y'_m \end{bmatrix} \quad (5)$$

3.2 需要考虑的问题

1) m 和 n 的选取

在sink节点对同一批次的 m 个数据包进行编码后, 共发送 n 个编码的数据包。因为中间节点可对已编码包再进行编码后转发, 所以在sink节点同一批次产生编码的数据包个数 n 可小于 m 。本文 m 和 n 都取为2的整数次幂, 如 $m=4, n=2$ 和 $m=8, n=4$ 。Sink节点不进行编码的话要发送 m 个源数据包, 编码后只要发送 n 个编码包, 可减少sink节点发送数据包的个数。

2) 编码系数选取和存储代价

网络编码是发生在有限域 F_q 上的运算。若有限域 F_q 的大小 $q=2^m$ (称 m 为域的字母表长度 F_q), 则每个节点 v 需要 $\delta_i(v)m$ 个存储单元来缓存来自输入链路的信息。显然 m 越大, 则节点所需存储空间就越大。而且, 如果 q 越大, 节点选择系数并执行编解码操作的复杂性也会增加。但是如果 q 过小, 将会使信宿节点的译码成功率下降。因此, 在保证足够的译码成功率的前提下, 应尽量减少有限域的大小。文献[13]指出当 $|F|=2^{16}$ 和 $|E|=2^8$ 时解码成功率可达0.996, 而且指出 $|F|=2^8$ 就足够实际使用了。

本文考虑到传感器节点存储能力有限, 如MicaZ节点只有4KB的RAM空间, 借鉴^[11]的思想,

系数随机在0到 $q-1$ 之间取值(q 为素数), 且每个编码包的首系数都取为1。在存储编码系数时, 为了节省存储空间采用单整数来存取所有编码系数。假设要编码 m 个数据包, 首系数为1, 其余系数分别记为 g_1, g_2, \dots, g_{m-1} , 用式(6)计算出对应的整数来存储, 最多需要 $\lceil \lg q^{m-1} \rceil$ 位来存储。例如 $q=5, m=4$, 对应的系数随机取为 $g_1=2, g_2=4, g_3=1$, 则存储对应的整数 $g=2 \times 5^2 + 4 \times 5^1 + 1 \times 5^0$ 。解码的时候可以通过逆运算求得相应的编码系数。

$$g = \sum_{i=1}^{m-1} g_i q^{m-i} \quad (6)$$

3) 成功译码的可能性和编解码代价

对于 m 个源数据包的线性组合编码包, 接收节点需要收到 m 个线性无关的编码包后才可以成功解码。而在上节中提到为节省存储空间, 编码系数随机在0和 $q-1$ 中取值, 这样可能会导致一些线性组合的编码包系数线性相关。只有接收节点收到的编码包中线性无关的编码向量数目等于或大于组成编码包的源数据包的数目时, 才能保证接收节点数据包的可解性。

定理1 假设能获得编码 i 个源数据包的已编码包个数的上限为 E_i , $1 \leq i \leq m$ 。当 E_m 大于 m 时, 编码包在接收节点具有可解性。

证明 很显然 $E_m=1$, 假如已经选择了 j 个线性独立的编码向量 v_1, v_2, \dots, v_j , 如果 v_{j+1} 与前面的编码向量不线性独立, 则存在常量 c_1, c_2, \dots, c_j 满足 $\sum_{k=1}^j c_k \times v_k = v_{j+1}$ 。注意到所有向量的首系数取为1, 则有 $\sum_{k=1}^j c_k = 1$, 对于与向量组 v_1, v_2, \dots, v_j 线性相关的向

量 v_{j+1} 有 q^{j-1} 种不同的选法, 而向量 v_{j+1} 总共有 q^{m-1} 种不同的选法, 所以向量 v_{j+1} 与其他向量都线性无关的概率为 $1 - q^j/q^m$, 也就是说当已有 j 个线性无关的向量后, 还需要 $q^m/q^m - q^j$ 个向量来产生一个新的满足线性无关的向量, 即利用迭代法得式(7)。

$$E_{i+1} = E_i + \frac{q^m}{q^m - q^j} \quad (7)$$

对于所有 $i \geq 1$, 得式(8)。

$$E_m = 1 + \sum_{j=1}^{m-1} \frac{q^m}{q^m - q^j} \quad (8)$$

取 $p=5, m=4$ 或 $m=8$ 代入得 $E_4=4.37, E_8=8.30$ 都

大于所需译码的编码个数，所以编码包在接收节点具有可解性。

证毕。

文献[11]中分析了解码代价，并证明当 $q \leq 5$, $m \leq 8$ 时所需的解码代价在传感器节点上是可行的。本文只是在它基础上增加了节点对已编码包进行再编码的计算代价，而解码的代价是一样的。

4) 节点的转发机制和 Forwarding factor 参数 p 的选取

因为中间的节点对已编码包进行再编码，即产生了新的编码包后向下一节点转发，会造成数据包的大量增加，所以本文采用了与 AdapCode 完全不同的转发机制，即当节点能成功解码包后就不再进行再编码和转发了。为进一步抑制和平衡新的编码包转发到其他节点，采用了概率转发和时间控制机制相结合的方式转发到其他节点。

当节点收到 n 个的更新包后，启动一个计时器，计时器时间满足随机均匀分布 $[0, T_{max}]$ 。当计时器超时后，节点对这 n 个更新包进行再编码后以概率 p 转发。以概率 p 转发和计时器控制减少了数据包发送的次数和在 MAC 层产生冲突的机会。但这种定时器机制也会带来一定的延时。当节点能译码出同一批次的源数据包后，不再转发同一批次的编码包。本文中取 $T_{max}=30ms$ 。

Forwarding factor 参数 p 的选取是决定算法性能的重要参数之一。 p 值较大，可以保证节点对于接收到的编码包的高解码率，但是会增大节点的发送次数， p 值较小可能会造成节点没有获得足够的解码包用于解码操作。本文在仿真部分给出了 p 的不同选取值下算法的性能。

5) 接收节点不能解码的发生与解决

由于无线传感器网络中存在链路易变性、网络拓扑的动态性等原因，接收节点不能收到发送节点发来的数据包。而目的节点为恢复出原始信息，接收到的线性组合数必须与同一批次中的源数据包数一样多，如果丢失数据包将使目的节点不能正确恢复出原始的信息的情况。本文采用节点对已编码的包进行再编码后并以一定的概率向邻居节点转发来提高节点收到更新包的机会，尽可能地减少此情况的发生。接收节点不能解码情况发生的概率与编码系数 m 和转发概率 p 的选取有关。仿真部分也给出了不同转发概率 p 对无法解码情况的影响（用平均数据包传输成功率 PDR 来衡量）。AdapCode^[11]

则是当节点出现不能解码情况时，引入 NACK 机制来解决，通过向邻居节点发 NACK 请求，重发一定数量的编码包，这无疑增加了额外的数据包的发送与接收数量。

3.3 算法描述

E-Code 算法描述如图 3 所示。

```

Algorithm E-Code
1: coeffMatrix ← a m × m matrix
2: timeValue ← T
3: set countdown timer equals timerValue ms
4: while data broadcast is going on do
5:   if a packet received then
6:     sender ← the sender of the packet
7:     generationID ← the generation number of the packet
8:     run Gaussian elimination on coeffMatrix
9:     if coeffMatrix is invertible then
10:      invMatrix ← the inverse matrix of coeffMatrix
11:      return the rank of coeffMatrix
12:      rank ← Gaissoam (coeffMatrix)
13:     if rank++then
14:       received and construct coeffMatrix using coefficients
15:       wait a random timer T
16:       broadcast coded packets with Forwarding factor p
17:     else
18:       discard the coded packet
19:     if rank=m then
20:       solve all messages in the generation by
21:       stop data broadcast

```

图 3 E-Code 算法描述

4 实验结果与分析

本文进行了 2 个方面的实验，实验 1 把 E-Code 算法与最典型的概率洪泛算法在 NS2 仿真工具下进行了分析比较。实验 2 把 E-Code 算法与同样用在无线传感器网络 sink 节点广播中的 AdapCode 算法在 TOSSIM 仿真平台下进行比较。

4.1 实验 1

本实验采用 NS2^[13]仿真平台。主要仿真参数设置如下：MAC 层协议采用 IEEE 802.11b，信道容量为 2Mbit/s，节点的传输范围为 30m，节点的数据发送队列大小为 50，数据源为 CBR 流，数据包大小为 32byte。采用的网络拓扑结构为：在 200m×200m 的场景中随机和均匀分布（10×10 的网格）100 个静止节点。其他参数为 $m=4, n=2$ ，丢包率为 10%，sink 节点总共需要发 128 个源数据包， $T=300ms$ 。性能比较参数为：1) 平均数据包传输成功率 PDR (packet delivery ratio)，节点能够成功接收（和解码）

的数据包个数与sink节点需要传送的源数据包总个数的比值的平均值；2) 数据包的平均传输代价(average overhead of transmissions), 数据包总的传输次数与所有节点成功接收(和解码)的数据包个数总和的比值。

图4给出了不同转发概率时节点的平均数据包传输成功率PDR。从图4可以看出, 当链路存在10%丢包率的情况下, 相同转发概率的E-Code的平均数据包传输成功率要高于概率洪泛, 当网格分布时, 转发概率在0.8以上时(随机分布时, 转发概率在0.9以上时), E-Code能达到100%的解码率(即成功率), 而概率洪泛并不能达到100%的成功率, 其原因是当链路存在丢包情况时, 使用网络编码后因为节点收到任意一个已编码的数据包都具有同等概率的作用, 降低了网络丢包对接收节点的影响。

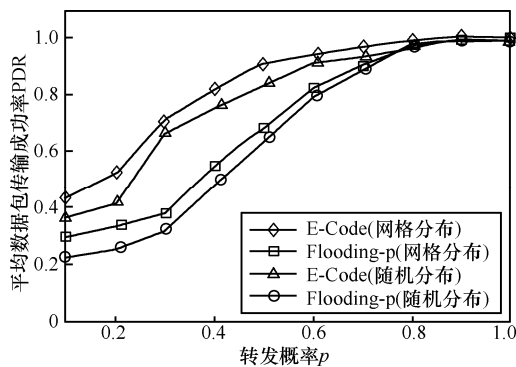


图4 不同转发概率下的平均数据包传输成功率 PDR

图5给出了不同转发概率时节点的平均数据包的传输代价。结果显示采用网络编码, 可比概率洪泛减少数据包的传输次数。不管是网格分布, 还是随机分布情况下, 当转发概率在0.8以上时, E-Code能达到100%的解码率后, 数据包平均传输代价在0.8~1区间有个极小值点, 再增大转发概率, 数据包平均传输代价反而有所增加。

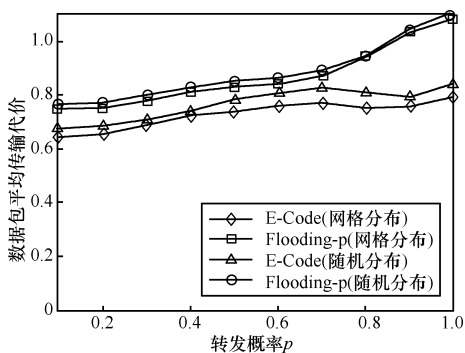


图5 不同转发概率下的数据包平均传输代价

4.2 实验 2

本实验采用 TINYOS2.0^[12]所支持的 TOSSIM 仿真平台。性能比较参数为: 1) 数据包的平均传输代价(average overhead of transmissions), 数据包总的传输次数与所有节点成功接收(和解码)的数据包个数总和的比值; 2) 平均时延(average delay), 网络中所有节点收到所有数据包的平均时间。在100个节点网格均匀分布(10×10的网格)情况下, 把E-Code算法与AdapCode算法进行性能比较。Sink节点总共发128个源数据包, 每个数据包大小为2KB, E-Code取由实验1得到的最好参数值, $m=4, n=2, p=0.8, T=300ms$ 。

图6为不同节点间距下的平均数据包的传输代价, E-Code的要优于AdapCode, 原因是AdapCode当节点出现不能解码情况时, 引入NACK机制来重发一定数量的编码包, 增加了额外的数据包的发送与接收数量。

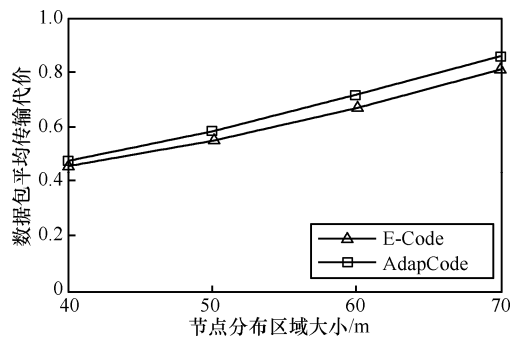


图6 不同节点间距下的平均数据包的传输代价

图7为不同节点间距下的平均时延。虽然E-Code和AdapCode都采用了网络编码, 但由于它们实现网络编码的方式不同, 即AdapCode为sink节点直接发送若干个源数据包, 中间节点都是收到了足够数量的编码包并成功解码成源数据包后, 再进行编码, 向下一节点转发; E-Code是在sink节点

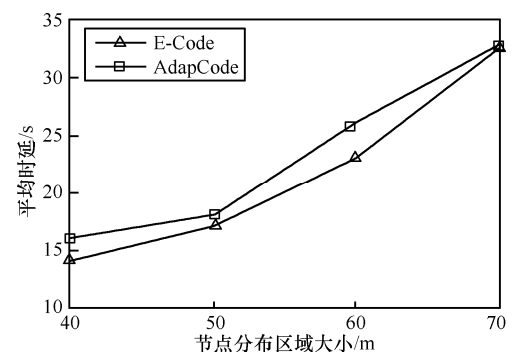


图7 不同节点间距下的平均时延

就进行编码, 中间节点对已编码包进行再编码后转发, 因此对于接收节点来说, E-Code 能更快地收到足够多的数据包来解码, 导致比 AdapCode 更小的平均时延。

5 结束语

网络编码是近几年的研究热点, 本文把随机网络编码应用在无线传感器网络中 sink 节点的数据包广播中, 并充分考虑传感器节点的储存和计算能力有限的特点, 改变了网络编码的系数向量的选取、转发机制的选取等。实验仿真显示可有效减少发送次数, 节省能量消耗。但目前的工作主要集中在网络层算法, 下一步将把 MAC 层也考虑进来, 并在真实传感器节点上来进一步验证算法的可行性, 从而更好地选取转发机制(如 forwarding factor 参数 p 的选取可动态改变)等。

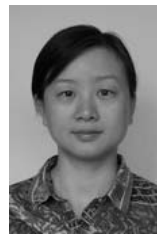
参考文献:

- [1] AHLWEDE R, CAI N, LI S, *et al.* Network information flow[J]. IEEE Transactions on Information Theory, 2000,46(4):1204-1216.
- [2] HO T, MEDARD M, KOETTER R, *et al.* A random linear network coding approach to multicast[J]. IEEE Transactions on Information Theory, 2006, 52(10): 4413- 4430.
- [3] ZAGAU M, HUBAUX J, ENZ C. Minimum-energy broadcast in all-wireless networks: NP-completeness and distribution issues[A]. ACM/IEEE Mobicom'02[C]. New York, 2002. 172-182.
- [4] SASSON Y, CAVIN D, SCHIPER A. Probabilistic broadcast for flooding in wireless mobile ad-hoc networks[A]. IEEE WCNC'03[C]. New Orleans, 2003. 1124-1130.
- [5] WU J, DAI F. Broadcasting in ad hoc networks based on self-pruning[A]. INFORCOM'03[C]. San Francisco, 2003. 2240- 2250.
- [6] FRAGOULI C, WIDMER J, BOUDEDEC J. Efficient broadcasting using network coding[J]. IEEE/ACM Transaction on Networking, 2008, 16(2): 450-463.
- [7] KATTI S, RAHUL H, HU W, *et al.* XORs in the air: practical wireless network coding[A]. SIGCOMM '06[C]. Pisa, 2006. 243-254.
- [8] CHACHULSKI S, JENNINGS M, KATTI S, *et al.* Trading structure for randomness in wireless opportunistic routing[A]. SIGCOMM '07[C].

Kyoto, 2007. 169-180.

- [9] PARK J, GERLA M, LUN D, *et al.* CodeCast : a network coding based ad hoc multicast protocol[J]. IEEE Transaction on Wireless Communication, 2006,10:76-81.
- [10] YANG Z, LI M, ZHOU W. R-code: network coding based reliable broadcast in wireless mesh networks with unreliable links[A]. GLOBECOM '09[C]. Hawaii, 2009. 1-6.
- [11] HOU I, TSAI Y, ABDELZAHER T, *et al.* AdapCode: adaptive network coding for code updates in wireless sensor networks[A]. INFOCOM'08[C]. Phoenix, 2008. 2189-2197.
- [12] NS2[EB/OL]. <http://www.isi.edu/nsnam/ns>, updated, 2007.
- [13] CHOU P, WU Y, JAIN K. Practical network coding[EB/OL]. <http://citeseerx.ist.psu.edu/viewdoc/summary?doi10.1.1.11.697,2003>.

作者简介:



肖玲(1977-), 女, 湖南衡阳人, 硕士, 湖南大学讲师, 主要研究方向为无线传感器网络、移动 ad hoc 网络。



李仁发(1957-), 男, 湖南郴州人, 博士, 湖南大学教授, 主要研究方向为嵌入式计算和无线网络。



罗娟(1974-), 女, 湖南衡阳人, 博士, 湖南大学副教授, 主要研究方向为无线传感器网络。