

能量捕获无线传感网的信道质量和能量感知高传递率多播方案

池凯凯^{1,2} 戴志泉¹ 李燕君¹ 程 珍¹

(浙江工业大学计算机科学与技术学院 杭州 310023)¹

(浙江省可视媒体智能处理技术研究重点实验室 杭州 310023)²

摘要 能量捕获无线传感器网络(EH-WSNs)由于可以无限期持续工作而具有非常广泛的应用前景。目前已有的EH-WSNs单跳多播方案存在以下问题:没考虑到链路质量会动态变化、节点储能易发生溢出而浪费能量等。提出了一种高传递率、高能效的EH-WSNs单跳多播方案。该方案采用纠错码,综合考虑节点当前储能、能量捕获速率以及当前信道质量等3方面因素,分析出当前待接收数据块的正确接收数据包数目期望值,只有当该期望值大于等于一定值时或者如果不接收该数据块则将发生储能溢出时才接收该数据块,有效地降低了信道质量差造成数据块接收失败和节点储能溢出的发生频率,高效利用极其有限的捕获能量。仿真结果表明,相比已有方案,本方案能显著提高EH-WSNs单跳多播的数据包成功传递率。

关键词 无线传感器网络,能量捕获,单跳多播,数据包传递率

中图分类号 TN 911.2 **文献标识码** A

Link-quality and Energy Aware High-rate Multicast Scheme for Energy Harvesting Wireless Sensor Networks

CHI Kai-kai^{1,2} DAI Zhi-quan¹ LI Yan-jun¹ CHENG Zhen¹

(School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China)¹

(Key Laboratory of Visual Media Intelligent Process Technology of Zhejiang Province,

Zhejiang University of Technology, Hangzhou 310023, China)²

Abstract As the energy harvesting wireless sensor networks(EH-WSNs) can work forever, they have a lot of promising applications. The available one-hop multicast schemes of EH-WSNs have the shortcomings that they have not taken the feature of dynamic link quality into account and waste the energy due to the overflow of energy storage, ect. This paper presented one energy-efficient one-hop multicast scheme with high packet delivery rate. This scheme integrates the erasure code and considers the node energy, energy harvesting rate and current link quality to analyze the expected number of correctly received packets of the follow data block. This scheme only let the sensor receive the block whose expected number of correctly received packets is greater than or equal to one threshold and the block followed by the occurrence of energy overflow once the sensor does not receive it, effectively reducing the occurrence probabilities of energy overflow and the failure decoding due to non-perfect link quality. Simulation results demonstrate that, compared to the available scheme, the proposed scheme is able to greatly improve the one-hop multicast packet delivery rate of EH-WSNs.

Keywords Wireless sensor networks, Energy harvesting, One-hop multicast, Packet delivery rate

1 引言

无线传感器网络(Wireless Sensor Networks, WSNs)有着非常广泛的应用前景。但是,在传统传感器网络中,传感器节点由电池来供电,而电池的容量是有限制的,电池储能的耗尽意味着传感器节点生命的结束。即使采用经过优化的通信和网络协议,节点的生命也非常有限,这严重地限制了WSNs的实际应用。

由能从环境中捕获能量的传感器节点组成的无线传感器网络(Energy-Harvesting Wireless Sensor Networks, EH-

WSNs)^[1-3]提供了一种崭新的有效突破能量供应瓶颈的传感器网络方案,具有非常广泛的实际应用前景。目前为止,不少相关文献已经对EH-WSNs展开了研究,比如文献[4,5]介绍了基于太阳能捕获的EH-WSNs,而文献[6]介绍了基于振动能量捕获的EH-WSNs。

在EH-WSNs中,在对节点的工作参数进行重新配置、更新节点的系统等场合需要用到多播或广播通信。由于传感器节点捕获能量非常有限,使用以基站为汇聚节点的单跳星形拓扑是一种相对容易实现的网络结构。目前已经少量的相关工作来研究EH-WSNs的单跳多播方案设计。Yang等

本文受国家自然科学基金(61472367,61202204),浙江省自然科学基金(LY15F020029,LY15F020027)资助。

池凯凯(1980—),男,博士,副教授,主要研究方向为无线通信,E-mail:kkchi@zjut.edu.cn;戴志泉(1991—),男,硕士生,主要研究方向为无线传感器网络;李燕君(1982—),女,博士,副教授,主要研究方向为无线传感器网络;程 珍(1981—),女,博士,副教授,主要研究方向为网络安全。

人^[7,8]考虑了将多播源节点作为能量捕获传感器节点的单跳百分比可靠多播方案设计,通过优化发送功率和发送速率来最小化多播传输时间。2014年,Kuan等人^[9]提出了一种应用于源节点具有充足能量、通过捕获得到接收节点能量这种场景的单跳多播传输方案。该方案采用纠错码来提高传输可靠性,将 k 个源数据包通过编码得到 n 个编码包,这 n 个编码包被称为一个数据块(block),接收端只要正确接收到一个数据块中的任意 k 个数据包就可以通过解码来恢复 k 个源数据包。重要的是,该方案一方面具有剩余能量意识,在一个新数据块的第一个数据包就判定是否接收该数据块,只有当前剩余能量足以接收 n 个数据包时才启动一个数据块的接收,从而避免因能量不足而导致数据块接收中断的发生;另一方面,该方案当正确接收到一个数据块中的 k 个数据包时,或者即使能正确接收当前数据块所有后续数据包也不能够解码时,则提前终止当前数据块的接收,从而有效地节省能量消耗。

本文研究EH-WSNs如下单跳多播场景:源节点具有充足能量而接收节点的能量是通过捕获而得来。上述Kuan等人针对该场景提出的方案虽然具有剩余能量意识和采用了数据块提前终止接收机制,但还存在两个方面的不足:(1)没有考虑到传感器节点在数据块接收过程中还会捕获一定的能量,这将会导致以下情况发生:节点由于当前能量不足以接收整个当前数据块而不启动该数据块接收,但是节点电池由于该数据块持续期间的能量捕获而发生储能溢出;(2)未考虑无线链路丢包率动态变化特性,不论当前数据块接收期间的丢包率有多高,只要当前能量足够接收本数据块中所有数据包,就接收该数据块。这两个方面会浪费所捕获的极其有限的能量,造成能量利用率不高。基于已有方案的以上不足,本文提出一种高吞吐量、高能效的单跳多播方案。该方案综合考虑节点当前储能、能量捕获速率以及当前信道质量这3方面因素,分析出当前待接收数据块的正确接收数据包的数目期望值,只有当该期望值大于等于一定值或者如果不接收该数据块则将发生储能溢出时才接收该数据块,有效地降低了信道质量差造成接收失败和节点储能溢出这两种事件的发生频率,从而能高效利用极其有限的捕获能量。

2 系统模型

2.1 网络模型

本文考虑一跳多播网络。网络由一个基站(即源节点)和多个具有能量捕获特性的传感器节点构成。传感器节点在源节点的直接通信的范围内。源节点具有充足能量,而接收节点能量通过捕获环境能量而得来。

源节点有一定数目的等长数据包需要发送给周围的多个传感器节点。将时间划分为时隙,时隙的时间长度用 T 表示。源节点在每个时隙的时长内能够发送一个数据包。源节点采用纠错码提高传输的可靠性,每 k 个源数据包通过编码得到 n 个编码包,如前所述,这 n 个编码包被称为一个数据块。由于一个数据块的发送持续时间较短,因此假设传感器节点在一个数据块的时间内的丢包率恒定不变,而不同的数据块接收期间的丢包率会动态变化。每个传感器节点基于过往一段时间的自己的数据包正确接收情况来统计出当前新数据块发送持续时间内的自身丢包率。节点 i 在当前新数据块发送持续时间的丢包率表示为 $p_i^{(t)}$ 。

2.2 能量捕获模型

本文考虑传感器节点的能量捕获速率较稳定的场景,即单位时间内捕获的能量随着时间的推移而变化非常缓慢。比如太阳能捕获的场景,太阳不会时隐时现,相对于毫秒级(或至多秒级)的数据块发送时间而言,光照强度的变化非常缓慢。用 $s_k^{(t)}$ 表示节点 i 在当前新数据块传递时间段内的能量捕获速率。 $s_k^{(t)}$ 的值在节点 i 上往往是可以特定方式知道的,比如说可以用当前多播数据流发起时刻的能量捕获速率来近似代表接下来的每个数据块传递时间段内的能量捕获速率,也可以利用指数加权移动平均等方法来估计 $s_k^{(t)}$ 的值。

需要说明的是,本文虽然以数据块内能量捕获速率恒定不变的能量捕获模型来进行多播方案的介绍,但该多播传输方案同样适用其它的能量捕获可预知或估计的场景。用 C 表示节点上的能量存储容量,如果节点存了 C 能量,后续捕获的能量将被溢出而浪费掉。

2.3 能耗模型

与文献[9]中的能耗模型相同,在传感器节点上,数据接收消耗相当的能量,用 E_R 表示接收一个数据包所消耗的能量。另外,传感器节点的带电运转会消耗一定的能量,这个能耗与是否接收数据包无关。用 E_M 表示一个时隙内该项能量消耗值。用 E_L 表示每个时隙上由于泄露而损失的能量。

因此,传感器节点在每个接收数据包的时隙上消耗的能量为

$$E_{pkt,rec} = E_R + E_M + E_L \quad (1)$$

而在每个空闲的时隙上消耗的能量为

$$E_{idle} = E_M + E_L \quad (2)$$

3 高吞吐量多播方案

该方案采用纠错码来提高传输可靠性,综合考虑节点当前储能、能量捕获速率以及当前信道质量这3方面因素,分析出当前待接收数据块中能被正确接收的数据包数目期望值,只有当该期望值大于等于一定值时或者如果不接收该数据块则将发生储能溢出时才接收该数据块,有效地降低了信道质量差造成接收失败和节点储能溢出这两种事件的发生频率,高效地利用了极其有限的捕获能量。

下面分别详细介绍该多播方案在源节点端和传感器节点端(即接收端)的操作。

3.1 源节点端的数据包发送

源节点采用纠错码提高传输的可靠性,将每 k 个源数据包通过系统纠错码(Systematic Erasure Code)编码得到 n 个包。这 n 个包被称为一个数据块。由于采用的是系统纠错码,一个数据块中具有 k 个包是 k 个源数据包,又称这些数据包为明文数据包,而其它 $n-k$ 个包中的任意一个包是 k 个源数据包的线性组合,称这些数据包为编码包。

传感器节点的能量捕获速率通常比较低。给定节点的能量捕获速率,节点在单位时间内可进行的数据包接收次数也就可以被确定出来。源节点单位时间内发送的数据包数目越多,则节点能执行接收的数据包的比率越低。为了避免过低的数据包接收率,源节点每发送完一个包后空闲一定数目的时隙才发送下一个包。用 M 表示发送周期,即 M 个时隙内

只发送一个包。因此,一个数据包的发送需经历 nM 个时隙。 M 的取值取决于传感器节点的平均能量捕获速率。如果传感器节点的平均能量捕获速率较低,为了保证相对高的数据包传递率, M 值需要取相对大的值,不过也会导致相对大的传输延迟;如果传感器节点的平均能量捕获速率较高, M 值可取相对小的值,以在保证一定的数据包传递率的同时可以获得相对高的吞吐量。

3.2 传感器节点端的数据包接收

传感器节点端的数据包接收以数据包为单位,包含两方面的操作:(a)判定是否接收当前新数据包;(b)如果要接收当前新数据包,则进行该数据包的接收,并在正确接收到 k 个编码包时进行解码。下面介绍这两方面的操作。

3.2.1 新数据包接收与否的判定

传感器节点上编码包接收失败主要有两方面的原因:(1)由于当前的包接收时间段内信道质量不够好,导致包接收出错;(2)由于包的接收过程中节点能量低于某个门限值,导致节点内部电路模块无法工作,从而导致当前包接收中断。

采用纠错码就是为了应对原因(1)所导致的数据包接收失败的情况,只要能正确接收一个数据包中 n 个包的任意 k 个包,就能通过解码来恢复出 k 个源数据包。但是纠错码的采用仍然无法解决以下问题:如果一个数据包内正确接收的编码包个数小于 k ,则不能通过译码得出 k 个源数据包,即数据包解码失败,会造成较大的能量浪费。如图 1 中的例子所示,采用($n=6, k=4$)纠错码,数据包包含 P_1, P_2, \dots, P_6 个包,其中 P_1, P_2, P_3, P_4 是明文数据包。某传感器节点在该数据包中仅正确接收到了 3 个包,因此没法正确解码,在正确接收的 3 个包 P_1, P_2 和 P_3 中只有明文包 P_1 和 P_2 有用,而其它 4 个包的接收能量白白浪费掉,包括 P_5 包的接收能量。因此,高效的多播方案应该在信道质量差时放弃当前数据包的接收,以把非常有限的捕获能量用于信道质量好时的数据包接收,从而提高数据包的成功接收率和吞吐量。

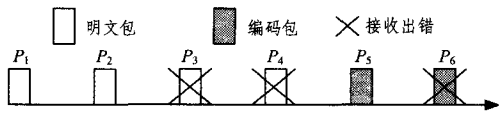


图 1 采用($n=6, k=4$)纠错码的数据包解码失败例子

另外,由于原因(2)所导致的数据包接收中断,导致当前数据包的接收失败,则当前数据包的部分数据接收已经消耗的能量都浪费掉。在能量捕获速率较低的场景下情况则更加糟糕,因为能量捕获速率较低时节点没有足够能量来进行当前数据包的后续数据包接收,导致当前数据包译码失败,从而在该数据包先前接收的数据包中,除了正确接收的明文数据包,别的数据包接收所消耗的能量都浪费掉。如图 2 中例子所示,在接收数据包 P_5 时节点因能量不足而造成接收中断,因能量捕获速率较低而导致 P_6 包也不能接收,最终导致接收 P_3, P_4 和 P_5 所耗能量都浪费掉。

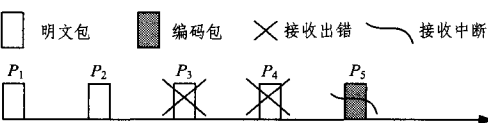


图 2 采用($n=6, k=4$)纠错码的数据包接收中断例子

综上所述,低信道质量和剩余能量不足而接收中断这两方面会联合会造成可观的能量浪费,大大降低数据包传递率。本文的多播方案综合考虑这两个问题,使用能达到较低数据包译码失败比率的数据块接收与否的判定规则,使得节点将非常有限的捕获能量用来接收那些很可能被正确译码的数据块。

数据包接收与否的判定方法:基于新的数据包到来时刻的信道质量、剩余能量及能量捕获速率,计算出能正确接收的数据包数期望值 N ,如果该期望值大于等于一个预设的门限值 N_{th} ,则判定为要接收这个新的数据包;另外,如果不接收当前数据包将发生储能溢出,则也判定为要接收这个新的数据包;否则,判定为不接收这个新的数据包。

N_{th} 的一个比较直观、易理解的取值是 $N_{th} = k$,因为如果真的接收到 k 个包,则可以成功译码。在性能评估部分会对 N_{th} 的不同取值进行测试,以得到其较优取值。另外,对于一些信道质量较低即满足 $N < N_{th}$ 的数据块,会发生以下情况:在数据块的初始时刻节点储能比较多,不接收该数据包则将发生储能溢出,造成能量浪费。在所提出的判定方法中,对于这种数据包,虽然 $N < N_{th}$ (即能正确接收 k 个或更多数据包的可能性相对较小),传感器节点还是接收该数据包,以避免储能溢出而浪费能量。

该判定方法的伪代码如下所示:

新数据包接收与否的判定方法

```
N=RecPktNum(); //计算正确接收的数据包数期望值
```

```
if( $N \geq N_{th}$ ) //期望的正确接收数据包数目大于等于  $N_{th}$ 
```

```
Start to receive the current data block;
```

```
else if( $E_i(1) - nME_{idle} + nMT_s \cdot s_i^{(j)} > C$ ) //如果不接收该数据包则发生储能溢出
```

```
Start to receive the current data block;
```

```
else //不接收该数据包
```

```
Give up receiving the current data block;
```

```
end if
```

其中,分析计算数据包中能被正确接收的数据包数期望值 N 的算法 $RecPktNum()$ 描述如下。

用 $E_i(j)$ 表示节点 i 在当前数据包的第 j 个包接收时刻的储能, $j=1, \dots, n$, 其中 $E_i(1)$ 也就是节点 i 在数据包初始时刻的储能。本多播方案中,一旦节点判定为接收当前数据包,传感器节点的数据块接收方式为对该数据包中的每一个即将从源节点发出的数据包,如果在该数据包发出时刻传感器节点的储能大于等于接收一个数据包所需要的能量 E_{pkt_rec} ,则接收该数据包;否则不接收该数据包。数据包中的数据包的接收一直进行到正确接收的包为 k 个(意味着可以正确译码)或错误接收的包为 $n-k+1$ 个(意味着即使正确接收当前数据包的所有后续包也不能正确译码)。基于以上所采用的数据包接收方式,节点 i 在数据包初始时刻根据 $E_i(1)$ 和能量捕获速率 $S_i^{(j)}$ 的值,计算出一旦接收该数据包则能正确接收的数据包个数期望值 N 。 N 值计算方法的伪代码如下:

数据包中能被正确接收的数据包个数估计算法 $RecPktNum()$

```
 $E_i'(1) = E_i(1), j=1, N=0$  //初始化
```

```
while( $j \leq n$ )
```

```
if( $E_i'(j) \geq E_{pkt\_rec}$ ) //接收该数据包
```

$N=N+1$ //记录接收的数据包数目

$$E_i'(j+1)=\min\{E_i'(j)-E_{\text{pkt_rec}}-(M-1)E_{\text{idle}}+MT_s \cdot s_h^{(i)}, C\}$$

else //不接收该数据包

$$E_i'(j+1)=\min\{E_i'(j)-ME_{\text{idle}}+MT_s \cdot s_h^{(i)}, C\}$$

end if

$j=j+1$;

end while

$N=N \cdot (1-p_e^{(i)})$ //基于丢包率得出正确接收的数据包个数期望值

Return N

3.2.2 数据块接收和解码

一旦节点根据上述的方法判定为要接收当前新到来的数据块,则启动数据块接收模式;对于每个到来的数据包,节点根据能量状况来判断是否接收该数据包。如果在该数据包到达时刻,当前能量小于接收一个数据包所需要的能量,则不接收该数据包;否则接收该数据包。如果正确接收的包为 k 个或错误接收的包为 $n-k+1$ 个,则停止接收当前数据块的后续包。

如果传感器节点正确接收的包为 k 个,则进行纠错码解码以恢复出 k 个源数据包。

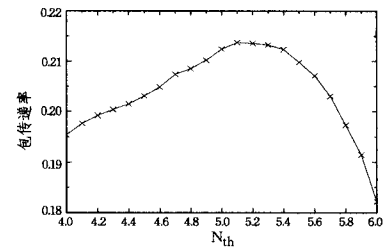
4 性能评估

本节通过仿真来将本文所提出的 EH-WSNs 单跳多播方案与文献[9]中的单跳多播方案进行包传递率比较。包传递率为正确接收的包数目与源节点发送的源数据包数目的比值。不难知道,对于能量捕获速率越高和信道质量越好的场景,其包传递率越高,且传感器节点的吞吐量正比于包传递率。

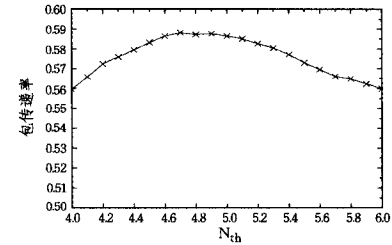
节点在一个数据块内的丢包率保持恒定,其值在 $[0, 0.4]$ 区间内等概率分布,数据块之间的丢包率相互独立。 $E_{\text{pkt_rec}}=10$, $E_{\text{idle}}=0.5$,源节点所采用的纠错码参数为 $n=6$, $k=4$ 。节点的储能容量 $C=80$,该值保证满储能的情况下足以接收一个数据块。

图3给出了 $M=3$ 时不同阈值 N_{th} 下的包传递率。令 $E_h = T_s \cdot s^{(i)} h$, 即 E_h 是每个时隙捕获的能量。图3(a)和图3(b)分别是 $E_h=1.0$ 和 $E_h=2.0$ 时的包传递率。从该图可以看出阈值 N_{th} 存在较优的取值,比如 $E_h=1.0$ 的场景中 N_{th} 的较优取值是 5.1 左右,而 $E_h=2.0$ 的场景中 N_{th} 的较优取值是 4.7 左右,其原因如下:因为只要一个数据块所估计的正确接收包数目大于等于 N_{th} , 就执行该数据块接收操作,所以相比于 N_{th} 取较高的值,当 N_{th} 取较低值(比如 $N_{th}=4.0$)时,执行接收的数据块平均译码失败率较高,造成了所浪费的能量与所捕获的能量的比值相对较高,从而导致较低的包传递率。但是, N_{th} 取过高的值也会导致较低的包传递率。因为 N_{th} 取较高值时,如果一个数据块是由于其估计的正确接收包数目大于等于 N_{th} 而执行接收,那么能正确接收到 k 个数据包的概率是较高的,可惜的是,绝大多数数据块所估计的正确接收包数目是小于 N_{th} 的,因此除了极少数由于其估计的正确接收包数目大于等于 N_{th} 而执行接收的数据块外,只有那些不接收则将储能溢出的数据块才执行接收操作,而这些数据块的通信质量是没法保障的,可能是较差的(即能正确接收到 k 个数据包的概率是较低的)。综上所述, N_{th} 存在一个较优的取值区间,如图3所示。另外,不同的场景下最优的 N_{th} 取值

也略有差别,不过不同场景下的仿真结果表明,在不同场景下 N_{th} 取 5.0 左右都是较优的。



(a) 每时隙捕获的能量 $E_h=1.0$



(b) 每时隙捕获的能量 $E_h=2.0$

图3 所提出方案在不同阈值 N_{th} 下的包传递率

接下来比较所提出的多播方案与文献[9]中的多播方案。首先比较不同 M 取值下的性能。如图4所示,随着发送周期 M 的增大,两种方案的包传递率都是增大的。这是因为发送周期越大意味着执行接收的数据块比例越高,从而包传递率也越大,不过这是以增大包传输时延为代价的。另外,从图4也可以看出,所提出的多播方案在发送周期 M 取不同值下都能显著提高包传递率。比如在 $M=4$ 时,包传递率提高了 25.3%。

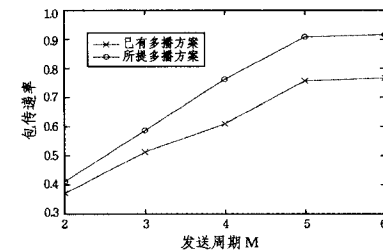


图4 不同 M 值下的包传递率比较 ($E_h=2.0$, $N_{th}=5.0$)

图5给出了不同能量捕获速率下的包传递率。随着能量捕获速率的增大,两种方案的包传递率都是增大的。这是因为能量捕获速率越大意味着执行接收的数据块比例越高,从而包传递率也越大。另外,从图5可以看出,除了具有较高能量捕获速率的场景(这种场景比较少),所提出的多播方案都能显著提高包传递率。比如在每时隙捕获的能量为 2.5 的场景中,包传递率提高了 24.7%。

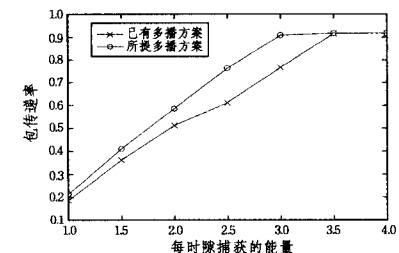


图5 不同能量捕获速率下的包传递率比较 ($M=3$, $N_{th}=5.0$)

结束语 本文提出了一种能量捕获无线传感器的高吞吐量单跳多播方案。该方案采用纠错码,综合考虑节点当前储能、能量捕获速率以及当前信道质量这3方面因素,计算出当前待接收数据块的正确接收数据包数期望值,只有当该期望值大于等于一定值时或者如果不接收该数据块则将发生储能溢出时才接收该数据块,有效地降低了因信道质量差而造成的接收失败和节点储能溢出这两种事件的发生频率。仿真结果表明,本方案能显著提高单跳多播的吞吐量,具体提高程度取决于能量捕获速率等参数。

参考文献

[1] Wan Z G, Tan Y K, Yuen C. Review on energy harvesting and energy management for sustainable wireless sensor networks [C]// Proceedings of IEEE 13th ICCT. Jinan, China; IEEE, 2011;362-367

[2] Seah W, Eu Z A, Tan H P. Keynote: Wireless sensor networks powered by ambient energy harvesting [C]// Proceedings of IEEE Wireless VITAE 2009. Aalborg, Denmark; IEEE, 2009

[3] Sudevalayam S, Kulkarni P. Energy harvesting sensor nodes;

Survey and implications[J]. IEEE Commun. Surv. Tuts., 2011, 13(3):443-461

[4] Hsu J, Kansal A, Friedman J, et al. Energy harvesting support for sensor networks[C]// Proceedings of IPSN'05. Los Angeles, USA; IEEE, 2005

[5] Jiang X, Polastre J, Culler D E. Perpetual environmentally powered sensor networks[C]// Proceedings of IPSN'05. Los Angeles, USA; IEEE, 2005

[6] Ammar Y, Buhrig A, Marzencki M, et al. Wireless sensor network node with asynchronous architecture and vibration harvesting micro power generator[C]// Proceedings of sOc-EUSAI'05. New York, USA; IEEE, 2005

[7] Ozel O, Yang J, Ulukus S. Broadcasting with a battery limited energy harvesting rechargeable transmitter[C]// Proceedings of IEEE WiOpt, 2011

[8] Yang J, Ozel O, Ulukus S. Broadcasting with an energy harvesting rechargeable transmitter [J]. IEEE Transactions on Wireless Communications, 2012, 11(2):571-583

[9] Kuan C, Lin G, Wei H. Reliable Multicast and Broadcast Mechanisms for Energy-Harvesting Devices [J]. IEEE Transactions on Vehicular Technology, 2014, 63(4):1813-1826

(上接第244页)

理位置随机生成,节点的通信距离为100m。 $T=1$ 和 $E_{h,max}=1.0$ 。图3给出了不同节点数目即不同节点密度下的数据流平均吞吐量的提高比值, N 值越大则节点密度越大。从该图中可以看出,数据流吞吐量提高比值随着节点密度的增加而升高。例如,当 $N=70$ 时使用所提出的路由方案带来的平均吞吐量提高为21.75%,而 $N=100$ 时使用所提出的路由方案带来的平均吞吐量提高为25.64%。

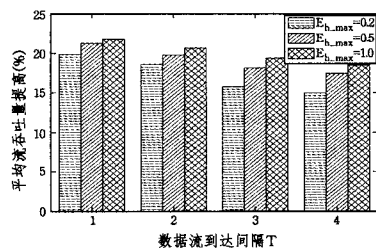


图2 不同数据流负载下平均数据流吞吐量的提高比值

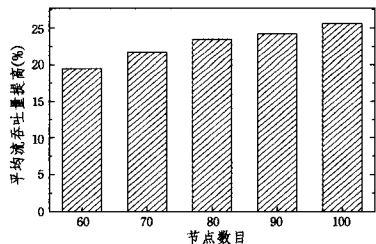


图3 不同节点密度下平均数据流吞吐量的提高比值

结束语 本文首先建模出EH-WSNs端到端吞吐量最大化路由问题,为深入了解该问题提供理论基础,然后提出了一种EH-WSNs端到端吞吐量最优路由方案。相比已有的路由方案,本路由方案能显著提高数据流的端到端吞吐量。当网络负载越大、节点的能量捕获速率越不均匀和节点密度越高时,吞吐量提高比值也越大。

参考文献

[1] Wan Z, Tan Y, Yuen C. Review on energy harvesting and energy management for sustainable wireless sensor networks [C]// IEEE 13th ICCT. Jinan, China, 2011

[2] Seah W, Eu Z, Tan H. Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP)-survey and challenges [C]// IEEE Wireless VITAE 2009. Aalborg, Denmark, 2009

[3] Sudevalayam S, Kulkarni P. Energy harvesting sensor nodes; Survey and implications [J]. IEEE Communication Surveys & Tutorials, 2011, 13(3):443-461

[4] Voight T, Ritter H, Schiller J. Utilizing solar power in wireless sensor networks [C]// LCN'03. 2003

[5] Intanagonwiwat C, Govindan R, Estrin D. Directed diffusion; a scalable and robust communication paradigm for sensor networks [C]// ACM MOBICOM. 2000

[6] Kansal A, Hsu J, Srivastava M, et al. Harvesting aware power management for sensor networks [C]// 43rd IEEE/ACM Design Automation Conference. 2006

[7] Kwon H, Noh D, Kim J, et al. Low-latency routing for energy-harvesting sensor networks [C]// 4th International Conference on Ubiquitous Intelligence and Computing. 2007

[8] Zeng K, Ren K, Lou W, et al. Energy-aware geographic routing in lossy wireless sensor networks with environmental energy supply [C]// QShine'06. 2006

[9] Eu Z, Tan H. Adaptive opportunistic routing protocol for energy harvesting wireless sensor networks [C]// IEEE ICC. 2012

[10] Martinez G, Li S, Zhou C. Wastage-Aware Energy-Harvesting Routing in Wireless Sensor Networks [J]. IEEE Sensors Journal, 2014, 14(9):2967-2974

[11] Johnson D, Maltz D. Dynamic source routing in ad hoc wireless networks [J]. Mobile Computing, 1996, 353:153-181