非线性随机网络编码研究

张东秋
(牡丹江师范学院 计算机与信息技术学院,牡丹江 157011)

摘 要 针对网络编码里的"全有或全无"以及因线性网络编码纠错能力过低而导致重传代价过大的问题,提出了非线性随机网络编码的方法. 该法用有限域上非线性函数的系数代替线性网络编码里的线性函数系数,在中间节点用一般的非线性函数对上游消息进行复合函数操作,在信宿节点用查表法进行译码. 实验结果表明: 非线性随机网络编码比线性网络编码具有更低的能量消耗、更低的时延,码的长度相同时能纠正更多的错误.

关键词 网络编码; 非线性; 纠错
中图分类号 TP39 **文献标识码** A **文章编号** 1672-4321(2017)04-0116-05

Study on Random Non-linear Network Coding

Zhang Dongqiu
(School of Computer and Information Technique, Mudanjiang Normal University, Mudanjiang 157011, China)

Abstract The error-correcting capacity of the linear network coding is limited, and the concept of non-linear network coding is presented. A new concept of non-linearly independent is presented to replace the existing concept of linearly independent. When the received symbols are non-linearly dependent, there is a fair possibility to decode the original messages without receiving more symbols. This scheme is just to utilize the pre-existing dependent symbols reasonably to decode original messages, instead of re-transmitting new symbols. Moreover, network coding is performed over binary field to save computational overhead. The simulation results show that this scheme reduces much energy and has low time delay.

Keywords network coding; non-linear; error-correcting

网络编码技术可以增强多播网络的吞吐量,具有巨大的应用前景[1]. 目前网络编码主要采用线性编码方案[2],线性网络编码主要采用有限域上的线性函数进行运算. 在线性随机网络编码中,每个数据包的头部都携带一个编码向量,该向量为待解 n 维消息向量的系数向量[3]. 假若中间节点从 K 条入边收到 K 个消息,对于其每一条出边,会在本地随机产生一个 K 维的有限域上的向量,利用该向量与收到的消息包相乘,然后将得到新的数据包在该条出边上发送出去,中间节点对每一条出边都从事上述操作. 信宿节点如果收到含 n 个消息的头部编码向量组成的矩阵满秩,利用线性方程理论就可以解码出原始消息.

线性网络编码方案具有实现方案简单、计算复杂度低等特点. 李硕彦证明了针对多播网络,线性网络编码可以达到多播容量上限[1]. 但线性网络编码依然具有几个明显缺点. 首先,在非多播即一般网络里,勒曼已经证明线性网络编码并不能达到所有类型非多播网络的容量上限[4]. 其次,线性网络编码存在"全有或全无"问题[5],当随机线性网络编码里的编码向量彼此不独立时,无法得出唯一解,造成解码失败. 此时,即使想恢复信源消息的一小部分也不可能[5,6]. 第三,即使是在多播网络里,虽然线性网络编码可以达到多播容量上限,但必须有一个前提条件是网络不存在错误,如果有错误发生,因为中继节点对上游消息的多次组合操作,所以即使很小的错误也有可能扩散至全网而造成信宿节点不可译码[7].

针对线性网络编码的上述问题,提出非线性网络编码的概念[8]. 非线性网络编码和线性网络编码

收稿日期 2017-09-20
作者简介 张东秋(1983-),女,博士,研究方向: 计算机教育技术和传感器网络,E-mail: 307642064@qq.com
基金项目 国家自然科学基金资助项目(61571150)

的本质区别是其采用非线性函数来代替线性函数. 针对 n 个原始消息,编码包头部的编码向量不再是 n 维的,而是 q^n 维的,它代表有限域 F_q 上的每个变量最高为 $q-1$ 次的多项式的系数. 非线性网络编码可以部分地克服线性网络编码的上述缺点.

1　随机线性网络编码

1.1　线性网络编码模型

设 $G(V,E)$ 是无延迟的通信网络. 信源节点集: $\{s_1,s_2,\cdots\}\subset V$,信宿节点集: $R=\{r_1,r_2,\cdots\}\subset V$,边 e 的头节点用 $h(e)=v$ 表示,边 e 的尾节点用 $t(e)=v'$ 表示. $X(v,i)$ 表示源节点 v 的 n 长消息的第 i 个字符. 称这样的编码为线性网络编码,如果对于网络中的每一条边 $e=(v,v')$ 的传输符号均满足:

$$I(e)=\begin{cases}\sum\limits_{e'\cdot h(e')=i(e)}\beta_{e',e}I(e') & \text{若 } v \text{ 不是信源节点}\\ \sum a_{i,e}X(v,i)+\sum\limits_{e'h(e')=t(e)}\beta_{e',e}I(e') & \text{否则}\end{cases},\tag{1}$$

其中 $\alpha_{i,e},\beta_{e',e}\in F_q$.

从函数映射角度,对边集 E 中的每条边 $e=(v,v')$,存在一种映射:

$$f_e:\prod_{e'h(e')=v}F_q\to F_q\quad,\tag{2}$$

这里,f_g 是有限域 F_q 上的线性函数. 本文主要考虑一个信源节点的多播网络,此多播网络最大流最小割为 n,所以信源消息是一个 n 长的一维向量.

随机线性网络编码模型如图 1 所示. 源节点 v 发出的的原始待解消息数据包 $X(v,i)$ 用 X_i 表示. 节点从入边收到的包个数为 K. 如果是信宿节点,由于最大流最小割是 n,其最大有效数据包个数为 n,即 $K=n$. 信宿收到 n 个数据包为 $Y_1,\cdots,Y_i,\cdots,Y_n$. Y_i 头部包含一个 n 长的编码向量 $g_{i,1},g_{i,2},\cdots,g_{i,n}$. 收到的 n 个数据包 $Y=(Y_1,Y_2,\cdots,Y_n)$ 为一组,针对的待解信源消息为 $X=(X_1,X_2,\cdots,X_n)$.

网络中存在很多这样的组,为了区别哪些数据包为一组,用数据包的头部数据区 Groupid 进行唯一标识. 为了节省头部编码向量标标识带来的通信开销,这里进行成组传输,每组容纳的 $Y_1,\cdots,Y_i,\cdots,Y_n$ 个数为 b,b 越大,数据包的头部数据区 Groupid 带来的开销将被稀释得越厉害[3].

1.2　线性网络编码的不足

第一是线性网络编码不能达到所有类型非多播网络的容量上限. 勒曼指出,针对某些非多播网络,

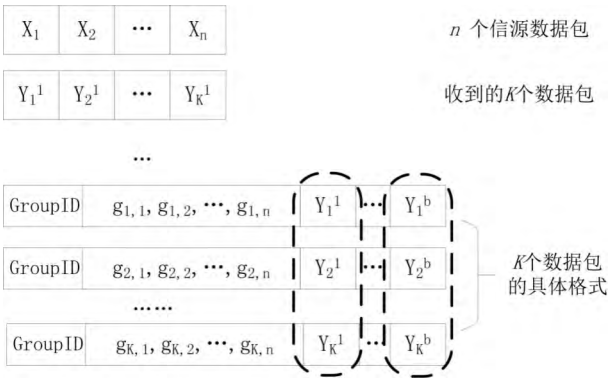

图 1　随机线性网络编码成组传输数据包格式

Fig. 1　The data-packet format of group transmission in random network coding

非线性网络编码能带来更多的传输增益[4].

第二是解码时存在"全有或全无"问题. 由图 1 可知,信宿节点解码时会将 n 个编码向量组合在一起形成一个关于未知数 $X_1,\cdots,X_i,\cdots,X_n$ 的 n 元一次线性方程组. 方程组的系数矩阵记为:

$$G=\begin{bmatrix}g_{1,1},g_{1,2},\cdots,g_{1,n}\\ \cdots\\ g_{i,1},g_{i,2},\cdots,g_{i,n}\\ \cdots\\ g_{n,1},g_{n,2},\cdots,g_{n,n}\end{bmatrix},\tag{3}$$

方程记为: $GX=Y$. 如果 G 满秩,则能解出信源消息 X,但如果 G 不满秩,则不能得到 X 的任何一部分. 上述问题是线性网络编码里的"全有或全无"问题. 一般的解决方法是重传更多的数据包直到 G 满秩. 但这样会加重传输负担,降低传输效率.

第三是线性网络编码的纠错能力不强. 网络编码中的错误和点对点通信中的错误不同,由于中间节点的混合操作,原始发生的错误会不断被扩散至下游节点而造成信宿节点不可译码.

2　非线性随机网络编码

2.1　非线性随机网络编码模型

如果将(2)式的函数 f_g 调整为有限域 F_q 上的非线性函数,那么线性网络编码将成为非线性网络编码.

2.1.1　非线性编码向量

其数据包格式和图 1 基本相同. 不同的主要是编码向量由 $g_{i,1},g_{i,2},\cdots,g_{i,n}$ 改为 $g_{i,q^1},g_{i,q^2},\cdots,g_{i,q^n}$,也就是编码向量长度由 n 变为 q^n. 为了减小消息包头部编码向量的开销,有限域设定为{0,1},即 $q=2$. 虽然非线性编码向量的长度比线性编码的长度大

很多,但是如果不断加大图1中成组传输里的b的数值,由此带来的通信开销会不断减小.

2.1.2 中间节点的非线性编码复合函数

若中间节点从K条入边收到K个消息,对于其每一条出边,会在本地随机产生一个K元多项式函数f_e,可以看作是有限域F_q上的的每个元的最高次幂为$q-1$的多项式. 因为$F_q=\{0,1\}$,所以每个元最高次幂为1,则该函数有2^K个系数. 其通式如下:

$$f(x_1,x_2,\cdots,x_K)=\sum_{a_1,a_2,\cdots,a_k\in(0,1,\cdots,|F|-1)} c(a_1,a_2,a_K)\,\alpha_1^{a_1}x_2^{a_2}\cdots x_K^{a_K}, \tag{4}$$

当某一中间节点收到3个消息时,即$K=3$,则:

$$f_e=x_1+x_1x_2x_3+x_2+x_2x_3+x_1x_3, \tag{5}$$

系数$c(a_1,a_2,\cdots,a_K)$从有限域F_q上随机选取. K个数据包头部的编码向量实际上对应以原始n个数据包$X_1,\cdots,X_i,\cdots X_n$,为变量的有限域$F_q$上的多项式,用$g_i(X_1,\cdots,X_i,\cdots,X_n)(1\leqslant i\leqslant K)$表示该多项式函数. 针对出边$e$,节点的局部编码函数不再是线性编码里的公式(1),而是复合函数操作,即

$$f(g_1(X_1,\cdots,X_i,\cdots,X_n),\cdots,g_k(X_1,\cdots,X_i,\cdots,X_n))=fg(X_1,\cdots,X_i,\cdots,X_n)=\sum_{a_1,a_2,\cdots,a_k\in F_2,\lambda_1\in F_2)}\lambda_1\cdot X_1^{a_1}X_2^{a_2}\cdots X_n^{a_n}. \tag{6}$$

复合函数操作之后函数的多项式系数组成新的数据包,然后发送到出边e上. 该数据包头部的编码向量即为$\lambda_1,\lambda_2,\cdots,\lambda_n$,它与图1中$\lambda$的$g_{i,1},g_{i,2},\cdots,g_{i,n}$是同一个具体值,即$(\lambda_1,\lambda_2,\cdots,\lambda_n)=(g_{i,1},g_{i,2},\cdots,g_{i,n})$. 非线性复合函数的运算过程可以用图2表示.

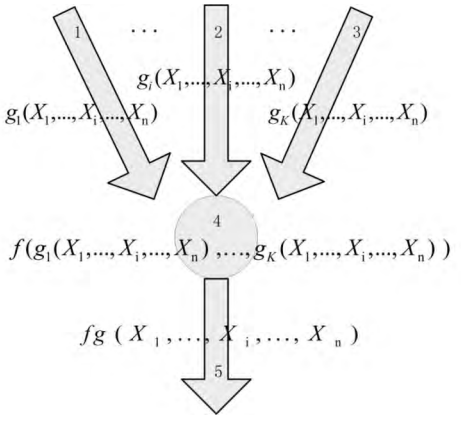

图2 中间节点的非线性编码复合函数

Fig. 2 The composite function of an intermediate node for non-linear coding

图2中,有三条入边,一条出边以及一个中间节点. 三条入边本身被赋予相应的全局编码函数,中间节点具有一个局部函数. 这三个全局编码函数在经过中间节点局部编码函数的复合之后形成新的全局编码函数,然后将这个新的全局编码函数赋予出边. 这样就完成了中间节点的非线性网络编码操作.

2.1.3 无错误时的信宿节点的译码

不像线性网络编码的译码,非线性编码里信宿节点通过查表法进行译码. 信宿节点会在本地产生n个随机多项式函数,每个随机多项式对入边的消息进行复合函数运算,得出一个编码包,然后传到信宿节点中的一条想象出边上. 因为有n个原始消息,所以对应有n条想象出边. 假设计算出来的n条想象出边上的数据包头部编码向量对应的多项式函数为$f_1(X_1,X_2,\cdots,X_n),\cdots,f_n(X_1,X_2,\cdots,X_n)$记为$f_1,\cdots,f_n$. 根据这些编码向量绘制解码函数数值对应表,如表1所示.

表1 解码函数

Tab. 1 The decoding function

$(X_1,\cdots,X_n)$	$(0,\cdots,0)$	$(0,\cdots,1)$	…	$(1,\cdots,1)$
$(f_1,\cdots,f_n)$	$(a_1,\cdots,a_n)$	$(b_1,\cdots,b_n)$		$(c_1,\cdots,c_n)$

当$(a_1,\cdots,a_n)$,$(b_1,\cdots,b_n)$,$(c_1,\cdots,c_n)$为$(X_1,\cdots,X_n)$的值确定时,$(f_1,\cdots,f_n)$的具体取值也确定. 信宿节点的n条想象出边上收到的n个数据包的数值部分,即图1中的$Y=(Y_1,\cdots,Y_i,\cdots,Y_n)$也对应$(f_1,\cdots,f_n)$的一组具体值,其为$(f_1,\cdots,f_n)=(Y_1,\cdots,Y_i,\cdots,Y_n)$. 在表1中找到$(Y_1,\cdots,Y_i,\cdots,Y_n)$,其对应的$(X_1,\cdots,X_n)$即为原始信源消息.

上述译码成功的前提是表1确定的映射关系是一一映射. 假设针对$(Y_1,\cdots,Y_i,\cdots,Y_n)$有两组或以上的$(X_1,\cdots,X_n)$与之对应组合,则译码失败.

2.2 非线性随机网络编码的优势

2.2.1 解决全有或全无问题

如果是线性编码,当G不满秩时,$GX-Y$总是得不到解. 当$n=3$时,表2和表3是线性编码和非线性编码时译码函数不是完全一一映射函数时的各自的一个具体例子. 可以发现,线性编码时,针对每一个$(f_1,\cdots,f_n)$组合,总有两个具体的$(X_1,\cdots,X_n)$组合与其对应. 非线性编码时,虽然对某些$(f_1,\cdots,f_n)$组合,总有多个的具体$(X_1,\cdots,X_n)$组合与其对应,但还有很多$(f_n,\cdots,f_n)$和$(X_1,\cdots,X_n)$的具体数值一一对应. 如果收到的$(f_1,\cdots,f_n)$落入不是一一对应的列,则译码失败. 而非线性编码下的解码函数不具有一一映射性,如果落到一一映射的列,依然可以译码. 所以,当信宿节点收到n个数据包时,如果数据包具有相关性,对应线性编码质. 这就是“全有或

全无”问题. 当为线性函数时,总是解码失败,此时需要重传数据包. 但是当为非线性编码时,还有一定的机会解码成功,此时不需要重传. 这样,非线性网络编码节省了重传次数.

表 2　线性函数编码时 G 不满秩时的解码函数

Tab. 2　The decoding function when G for linear function is not full rank

(X_1, X_2, X_3)	(0,0,0)	(0,0,1)	(0,1,0)	(0,1,1)	(1,0,0)	(1,0,1)	(1,1,0)	(1,1,1)
(f_1, f_2, f_3)	(0,1,0)		(1,0,0)		(1,1,1)		(1,0,1)	

表 3　非线性函数编码时不是一一映射时的解码函数

Tab. 3　The decoding function when non-linear function is not one-to-one mapping

(X_1, X_2, X_3)	(0,0,0)	(0,0,1)	(0,1,0)	(0,1,1)	(1,0,0)	(1,0,1)	(1,1,0)	(1,1,1)
(f_1, f_2, f_3)	(0,0,1)	(0,1,0)	(0,1,1)	(1,0,0)	(1,1,1)		(1,0,1)	

2.2.2　非线性编码的纠错

线性编码的纠错方案一般通过加入一定的冗余来识别错误. 线性纠错码里,为了增加检错或纠错能力,在信息里增加多余的码元,以扩大码字之间的差别,新增加的多余码元即为冗余. 在此时需要传输的消息记为 $u = (u_1, u_2, \cdots, u_k)$,通过一个$(n, k)$ 的最大距离可分码的生成矩阵对 $u = (u_1, u_2, \cdots, u_k)$ 进行编码,编码后的消息向量为 $X = (X_1, \cdots, X_i, \cdots, X_n)$. 此时冗余的大小为 $n - k$. 此编码的最小距离为 $n - k + 1$. 如果是点对点通信,只要错误个数 t 满足 $t \leqslant (n - k)/2$,则以可正确译码. 但在线性网络编码中因为中间节点的混合操作使得 t 个错误发生扩散,扩散后的错误个数可能远远大于 t,所以不能译码[10,11].

如果是非线性网络编码,设计一个和线性编码的生成矩阵对应的编码函数矩阵. 编码过程如下.

$$\begin{pmatrix} X_1 \\ X_2 \\ \cdots \\ X_n \end{pmatrix} = \begin{pmatrix} \beta_1(u_1, u_2, \cdots, u_k) \\ \beta_2(u_1, u_2, \cdots, u_k) \\ \cdots \\ \beta_n(u_1, u_2, \cdots, u_k) \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ \cdots \\ u_k \end{pmatrix}. \tag{7}$$

其中·为复合函数操作. 只要函数 $\beta_1, \beta_2, \cdots, \beta_n$ 之间的独立性达到最大,则其可以对抗尽可能多的错误. 当 $\beta_1, \beta_2, \cdots, \beta_n$ 为线性函数时,则此编码退化为线性编码时的生成矩阵. 非线性的函数个数远远大于线性函数个数,$\beta_1, \beta_2, \cdots, \beta_n$ 之间的汉明距离更大,所以对抗的错误更多. 这一性质可以局部改善线性网络编码对抗错误能力弱的问题.

3　仿真

对比纯路由传输,线性网络编码和非线性网络编码三种传输方案的正确译码率以及能量消耗. q 大小设置为 2,3,5,7,以 2 为主. q 如设置太大,编码向量太长,将加重通信负担.

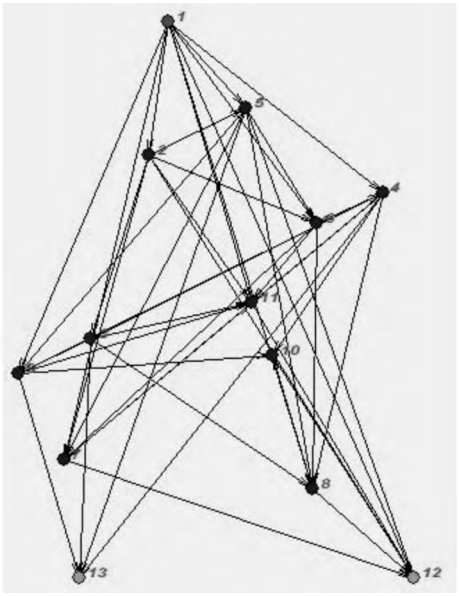

图 3　多播网络的拓扑

Fig. 3　The topology of a multicast network

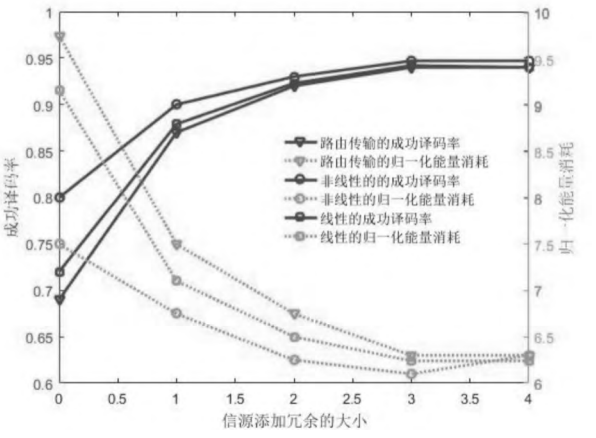

图 4　三种传输方案对比

Fig. 4　The comparison among three transmission schemes

图 3 是实验采用的一个多播网络的拓扑. 在图 3 里,深色的节点 1 表示信源节点,浅色的节点 12,13 表示两个信宿节点. 其余的节点为中间节点,总的节点数 13. 拓扑由拓扑生成器随机生成的. 拓扑生成

器是Visual C++编写的一个拓扑生成软件. 图4表明,译码的成功率和能量消耗随着冗余的增加而变化的情况. 其中能量评价函数采用文献[12]中的模型,其特点是用消耗的时间来表征消耗的能量. 能量评价函数用公式

$$T = \frac{\sum_{i=1}^{H} H_i}{R} + h(\lceil \log_2 q \rceil + n)^3 \qquad (8)$$

来表示. 其中H_i和R的定义和文献[12]中的定义相同. $\frac{\sum_{i=1}^{H} H_i}{R}$表示的是传送需要的能量. $h(\lceil \log_2 q \rceil + n)^3$是信宿节点解码耗费的能量. 在网络编码里,编码域的大小q的值应该不小于信宿节点的值. 但在实际编码中为减少构造难度,往往采用随机网络编码,为保证解码成功率,q的值往往要设置的大一些. 较大的q值在增加解码成功率的同时又会加大运算量,所以,q的取值应在二者之间取得一个较好的平衡.

结果显示,随机非线性网络编码具有更高的成功译码率,更低的能量消耗. 主要原因是即使受到消息之间相关性的影响,非线性网络编码仍有一定机会恢复原始消息. 当有错误存在时,其能容纳更多的错误. 以上两点减少了重传的机率,因而能量消耗也相应减少.

4 结语

相比线性网络编码,非线性网络编码较好地解决了"全有或全无"问题,也能纠正更多的错误. 但其编码向量长,增加了一定的通信负载. 解码算法基于查表法,效率有待提高. 相比线性网络编码,非线性网络编码还有更多的潜力有待开发.

参 考 文 献

[1] Li S Y, Yeung R W, Cai N. Linear network coding [J]. IEEE Transactions on Information Theory, 2003, 49(2): 371-381.

[2] 姚世雄,陈 晶,向 琨,等. 网络编码理论及应用综述[J]. 中南民族大学学报(自然科学版),2017,36(2):115-128.

[3] Wang L, Yang Z, Xu L, et al. NCVCS: Network-coding-based video conference system for mobile devices in multicast networks [J]. Ad Hoc Networks, 2016, 45: 13-21.

[4] Lehman A R, Lehman E. Complexity classification of network information flow problems [C]// ACM. 15th Acm-Siam Symposium on Discrete Algorithms. Society for Industrial and Applied Mathematics. New Orleans: ACM, 2004: 142-150.

[5] Kwon M, Park H, Frossard P. Compressed network coding: Overcome all-or-nothing problem in finite fields [C]//IEEE. Wireless Communications and Networking Conference. Istanbul: IEEE, 2014: 2851-2856.

[6] Yan Z, Xie H, Suter B W. Rank deficient decoding of linear network coding [C]// IEEE. International Conference on Acoustics, Speech and Signal Processing. Florence: IEEE, 2013: 5080-5084.

[7] Sanna M, Izquierdo E. A survey of linear network coding and network error correction code constructions and algorithms [J]. International Journal of Digital Multimedia Broadcasting, 2011: 1687-7578.

[8] Shang T, Zhang C, Li K, et al. Nonlinear quantum network coding with classical communication resource [C]// IEEE. GlobeCOM Workshops. San Diego: IEEE, 2015: 1-6.

[9] Katti S, Rahul H, Hu W, et al. XORs in the Air: Practical wireless network coding [J]. IEEE/ACM Transactions on Networking, 2008, 16(3): 497-510.

[10] Langberg M, Effros M. Network coding: Is zero error always possible? [C]//IEEE. Communication, Control and Computing. Monticello: IEEE, 2012: 1478-1485.

[11] Gadouleau M, Yan Z. Packing and covering properties of subspace sodes for error control in random linear network coding [J]. IEEE Transactions on Information Theory, 2010, 56(5): 2097-2108.

[12] Z. Guo, B. Wang, P. Xie, et al. Efficient error recovery with network coding in underwater sensor networks [J]. Ad Hoc Networks, 2009, 7: 791-802.