一种可验证和高效的多秘密共享门限方案

步山岳[1] 王汝传[2]
(淮阴工学院计算机工程学院 淮安 223002)[1] (南京邮电大学计算机学院 南京 210003)[2]

摘 要 已公开的门限多秘密共享方案大都是利用RSA,ECC等公钥体制来提高安全性,其占用的资源较多,速度慢。提出了一种新的多秘密共享(t,n)门限方案,该方案是在shamir秘密共享方案的基础上,利用拉格朗日插值多项式方法进行秘密分割和重构,利用NTUR算法和单向散列函数进行数据合法性验证。方案设计简单、计算量小、存储量少,能有效检测出各种欺骗、伪造行为,以确保恢复的秘密是安全和可信的。

关键词 多秘密共享,NTUR算法,门限方案,可验证
中图法分类号 TP309 **文献标识码** A

Verifiable and Efficient Multi-secret Sharing Threshold Scheme

BU Shan-yue[1] WANG Ru-chuan[2]
(School of Computer Engineering, Huaiyin Institute of Technology, Huaian 223002, China)[1]
(College of Computer, Nanjing University of Post & Telecommunications, Nanjing 210003, China)[2]

Abstract In most of the multi-secret sharing schemes already published, RSA, ECC or other public key cryptosystems are used to improve security. But such schemes would take up lots of resources and result in low speed. We proposed a new multi-secret sharing(t,n) threshold scheme based on shamir secret-sharing scheme, using the Lagrange interpolating polynomial to split and reconstruct the secrets and the NTRU and one-way hashing function to verify the validity of data. The scheme is simple in design and requires limited calculation and limited storage space. It can detect effectively a variety of cheating or forgery behaviors and guarantee that the reconstruction of the secret is the secure and trustworthy.

Keywords Multi-secret sharing, NTRU algorithm, Threshold scheme, Verifiable

1 引言

秘密共享技术就是通过秘密分发、保存和重构算法来保护重要秘密或数据。1979年,密码学家shamir[1]提出了一种基于插值多项式的(t,n)门限秘密共享方案。shamir门限方案的基本思想是将一个秘密信息或密钥分解成n个不同的子秘密,分发给n个成员分别保管。在n个成员当中,至少有t个成员相互合作才可能恢复出秘密信息。

文献[2]中,Harn提出了一种基于(t,n)门限的可验证多秘密共享方案,通过验证,其可以识别管理员欺骗和成员欺骗问题。文献[3]通过使用双线性映射计算方程,提出了一种新的公开可验证的秘密共享方案,它更容易检测出管理员和成员的不诚实行为。文献[4]设计了一个可验证多门限多秘密共享方案,每个门限值可以共享多个秘密,减少了运行时间和存储量。文献[5]基于双密钥思想提出了一种改进的多秘密共享认证方案,它可以有效抵御成员的欺骗,重构时采用并行算法来降低运算复杂度。文献[6]给出的方案试图防止管理者和成员的欺诈,秘密分发时不需要维护安全信道等。该方案的动态性差,当秘密个数发生变化时,就要重新构造$k+n-1$次多项式。文献[7]结合基于身份(ID)的公钥密码技术,给出了利用成员私钥作为其主份额的秘密份额方法,它能够在分发秘密的同时分发秘密份额。文献[9-11]给出了一些改进的多秘密共享方案,对成员的子秘密重用和扩展性方面进行了研究。不过,上述方案的安全性都是基于传统的离散对数或大数分解的密码算法。文献[8]提出了基于NTRU算法的秘密共享方案,但该方案只能恢复单个秘密,存储量也比较大。

本文在shamir的lagrange插值多项式门限方案的基础上,提出了一种基于NTRU算法的(t,n)门限多秘密共享方案。本方案的主要特点是:方案的安全性是建立在快速、高效的NTRU算法基础上,而不再是基于传统的离散对数或大数分解的密码算法,减少了系统运行成本;能有效地抵御成员之间欺骗行为,防止各种类型的攻击;在确保安全的前提下,方案设计合理、简单,计算量小,存储要求低,与目前同类型的方案相比具有较大的优越性。

2 NTRU算法回顾

NTRU算法运行在多项式环$R=Z[X]/(X^N-1)$上,所有多项式为次数等于$N-1$的一元多项式,所有多项式的系数为整数或0。NTRU算法需要初始化三个整数参数$(N,p,$

到稿日期:2010-02-23 返修日期:2010-05-07 本文受国家自然科学基金项目(60973139)资助。
步山岳(1959—),男,副教授,主要研究方向为网络安全、密码学,E-mail:bushanyue@121.com;王汝传(1943—),男,教授,博士生导师,主要研究方向为计算机信息安全、计算机软件、虚拟现实技术等。

q)和四个具有 $N-1$ 阶多项式 f,g,r,m。其中 N 为素数，p 可以是多项式或整数，q 为整数，但要保证 $\gcd(p,q)=1$。NTRU 算法过程如下：

(1)密钥产生

随机选择 2 个多项式 F,g，计算：

$$f=1+p*F, f*f_q=1(\bmod q) \tag{1}$$

式中，f_q 表示 f 的模 q 逆，$*$ 表示卷积运算。计算：

$$h=p*g*f_q(\bmod q) \tag{2}$$

得到 NTRU 算法的密钥对为(h,f)。

(2)加密

设有明文多项式 m，随机选择多项式 r，计算密文 e：

$$e=r*h+m(\bmod q) \tag{3}$$

(3)解密

还原明文时计算：

$$m=f*e(\bmod q)(\bmod q) \tag{4}$$

有关 NTRU 算法的优势和更详细说明请看文献[12,13]。

3 多秘密共享方案

3.1 参数说明

P：方案中所有成员的集合，当方案中有 n 个成员时，$P=\{P_1,P_2,\cdots,P_n\}$。

D：方案中秘密管理员，$D\notin P$。D 的主要职责就是将与秘密相关的子秘密等信息分发给每个成员 $P_i(i=1,2,\cdots,n)$，为方案中的每个 P_i 提供恢复秘密的公开参数，并对方案中的成员信息进行管理等。

NB：电子公告牌。D 可以在 NB 上发布公开信息，可以对 NB 中内容进行修改、添加和删除。方案中的其他成员 P_i 只能查看、读取 NB 中的信息，不能，也无法对 NB 中的内容进行修改等操作。方案对没有公布在 NB 上的所有信息都进行严格的保密。

id：方案中所有成员的身份标志符集，当方案中有 n 个成员时，$id=\{id_1,id_2,\cdots,id_n\}$。$D$ 为方案中每个成员 P_i 分配唯一身份标志符 $id_i(i=1,2,\cdots,n)$，每个 id_i 值互不相同。D 在 NB 上公开每个 id_i 值。

(h,f)：由 NTRU 算法产生的密钥对，h 由 D 秘密保管，f 放到 NB 上。

H：表示方案中使用的单向散列函数，可以是 MD5 或 SHA 算法。

S：表示方案要共享的 m 个不同的秘密集，$S=(S_1,S_2,\cdots,S_m)$。

3.2 构造成员子密钥

按照 shamir 的(t,n)门限秘密共享方案，管理员 D 随机生成 t 个不同整数，$t-1$ 阶插值多项式 $a(x)$：

$$a(x)=a_0+\sum_{j=1}^{t-1}a_jx^j \tag{5}$$

式中，$t\leqslant n<N, a_i\in Z[X]/(X^N-1)$，针对 NTRU 算法时，可以用向量表示。

D 根据式(6)为方案中每个成员 P_i 分配一个子密钥 $x_i\in Z[X]/(X^N-1)$。

$$x_i=a(id_i)=\sum_{j=0}^{t-1}a_j*(id_i)^j \tag{6}$$

D 针对多项式 $a(x)$ 的系数 a_j(j=0,1,…,t-1)，按照式(7)计算验证向量 $v=(v_0,v_1,\cdots,v_{t-1})$。

$$v_j=r*h+a_j(\bmod q) \tag{7}$$

D 通过安全信道将 x_i 密送给 P_i，将 v 放到 NB 上。当方案中的其他成员 P_i 收到 x_i 后，可以按照式(8)验证 x_i 的真实性。如果式(8)不成立，则认为存在 D 对 P_i 有欺诈行为，这时成员可以向管理员发出抱怨。否则表示 D 给成员 P_i 的子密钥 x_i 是诚实的。

$$x_i=f*\sum_{j=0}^{t-1}v_j*(id_i)^j(\bmod p) \tag{8}$$

3.3 构造多秘密的秘密凭证

设二元集合(e,c) 是 S 的秘密凭证。管理员 D 随机选择不同的参数 r 和 $e,e_j\in e(j=1,2,\cdots,m)$，$m$ 表示方案中秘密的个数，$r,e_j\in Z[X]/(X^N-1)$，按照式(9)为每个秘密 $S_j\in S$ 生成二元秘密凭证(e_j,c_j)。

$$c_j=r*h+e_j(\bmod q) \tag{9}$$

3.4 构造多秘密的影子子密钥

设 k 是 S 的影子子密钥集，D 按照式(10)为每个秘密 $S_j\in S$ 生成一个影子子密钥 $k_j\in k(j=1,2,\cdots,m)$。

$$k_j=S_j\oplus H(a_0*c_j) \tag{10}$$

D 将(e_j,c_j,k_j)公布到 NB 上。

3.5 多秘密重构

设 $\Gamma(|\Gamma|=t\leqslant n)$是 P 中成员的一个访问结构，并满足单调属性。$B\in\Gamma$ 为 Γ 的最小授权子集。即 B 是 P 中要重构秘密 $S_j\in S$ 的 t 个成员的集合，B 中的每个成员都可以作为秘密重构者进行秘密重构。

(1)计算秘密交换凭证

成员 $P_i\in B$ 从电子公告牌 NB 上获得 c_j，用自己的子密钥 x_i 按照式(10)计算秘密交换凭证 A_{ij}，然后 P_j 将 A_{ij} 发送给 B 的其他合作者。

$$A_{ij}=x_i*c_j \tag{11}$$

(2)验证秘密交换凭证

当方案中的其他成员收到 A_{ij} 后，从 NB 上获得 e_j，按照式(12)验证 A_{ij} 的真实性。如果式(12)不成立，则认为成员中有欺诈行为，宣布合作成员中提供的 A_{ij} 有错误，停止秘密恢复，并要求 P_i 重新发送正确的 A_{ij}。

$$f*A_{ij}(\bmod p)=f*\sum_{k=0}^{t-1}v_k(id_i)^k*e_j(\bmod p) \tag{12}$$

(3)多秘密重构

当 B 中所有成员的 A_{ij} 值都通过验证后，就可以从 NB 上获得 k_j，按照式(13)重构出秘密 S_j。

$$S_j=k_j\oplus H\Big(\sum_{P_i\in\Gamma}A_{ij}*\prod_{P_k\in\Gamma,P_k\neq P_i}\frac{-id_k}{id_i-id_k}\Big) \tag{13}$$

4 方案的正确性证明

定理 1 方案 P 中任何一个成员 P_i 可以按照式(8)验证管理员 D 给 P_i 的子密钥 x_i 的真实性。

证明：

$$\begin{aligned}&f*\sum_{j=0}^{t-1}v_j*(id_i)^j(\bmod p)\\&=f*\sum_{j=0}^{t-1}(r*h+a_j)(\bmod q)(id_i)^j(\bmod p)\\&=\sum_{j=0}^{t-1}a_j(id_i)^j=x_i\end{aligned}$$

所以，如果 x_i 满足式(8)，则 x_i 是真实的。

定理 2 方案 B 中任何一个成员 P_i 均可以按照式(12)

验证其他合作成员提供的秘密交换凭证 A_{ij} 的真实性。

证明：

$$f*A_{ij}(\bmod p)=f*x_i*c_j(\bmod p)$$

$$=f*\sum_{k=0}^{t-1}a_k(id_i)^k*(r*h+e_j)(\bmod q)(\bmod p)$$

$$=\sum_{k=0}^{t-1}a_k(id_i)^k*e_j$$

$$f*\sum_{k=0}^{t-1}v_k(id_i)^k*e_j(\bmod p)$$

$$=f*\sum_{k=0}^{t-1}r\cdot h+a_k(\bmod q)(id_i)^k*e_j(\bmod p)$$

$$=\sum_{k=0}^{t-1}a_k(id_i)^k\cdot e_j$$

即：$f*A_{ij}=f*\sum_{k=0}^{t-1}v_k(id_i)^k*e_j$。

所以，如果 A_{ij} 满足式(12)，则 A_{ij} 是真实的。

定理 3 在保证方案所提供的信息是真实的情况下，方案 B 中任何一个成员 P_i 均可以按照式(13)重构出多秘密 $S_j\in S, j=1,2,\cdots,m$。

证明：

$$a_0*c_j=a(0)*c_j=c_j*\sum_{P_i\in\Gamma}a(id_i)*\prod_{P_k\in\Gamma,P_k\neq P_i}\frac{-id_k}{id_i-id_k}$$

$$\sum_{P_i\in\Gamma}A_{ij}*\prod_{P_k\in\Gamma,P_k\neq P_i}\frac{-id_k}{id_i-id_k}$$

$$=\sum_{P_i\in\Gamma}x_i*c_j*\prod_{P_k\in\Gamma,P_k\neq P_i}\frac{-id_k}{id_i-id_k}$$

$$=c_j*\sum_{P_i\in\Gamma}a(id_i)*\prod_{P_k\in\Gamma,P_k\neq P_i}\frac{-id_k}{id_i-id_k}$$

即：$a_0*c_j=\sum_{P_i\in\Gamma}A_{ij}*\prod_{P_k\in\Gamma,P_k\neq P_i}\frac{-id_k}{id_i-id_k}$

所以有：$S_j=k_j\oplus H(\sum_{P_i\in\Gamma}A_{ij}*\prod_{P_k\in\Gamma,P_k\neq P_i}\frac{-id_k}{id_i-id_k})$

5 方案的安全性分析

本方案的安全性建立在如下 3 个条件下：

(1)基于 NTRU 公开密钥加密体制。即在格中求解最短向量的数学问题，在格中求解最短向量的数学问题是非常困难的，根据公开信息直接推导出秘密信息在计算量上是不可行的。

(2)成员的子密钥 x_i 是由管理员 D 根据 $t-1$ 阶多项式 $a(x)$ 计算出来的。只要 $a(x)$ 没有被泄密，x_i 就是安全的。

(3)根据式(8)、式(12)可以避免管理员和成员、成员与成员之间交换数据时发生的欺骗行为。

下面给出本方案可能遇到的几种攻击，进一步分析方案的安全性。

攻击 1 管理员 D 的疏忽问题。由于疏忽，当 D 为其他成员提供了错误的子密钥 x_i，或者是方案中 $a(x)$ 等数据需要更改时 D 没有及时更新 x_i，或者是在传输数据过程中 x_i 值被恶意修改，这时成员可以通过式(8)验证 x_i 的正确性。任何不真实的 x_i 都无法通过式(8)的条件，除非攻击者成功破解了 NTRU 算法，获取了保密密钥 h 等信息。显然，这也是不现实的。

攻击 2 已知部分秘密企图重构其他秘密。由于式(10)、式(13)中使用了单向散列函数 H，攻击者即使通过合法的手段获得已恢复的部分秘密 S_j 和 k_j，也无法利用异或运算⊕恢复 a_0 等重要信息。所以，在没有其他合作者的帮助下，攻击者也无法恢复出其他秘密，除非攻击者成功破解单向散列函数 H，获取了 a_0 等重要信息，显然，这是不现实的。

攻击 3 成员之间的欺骗问题。在秘密重构时，方案设计了 S 的秘密凭证二元集合(e,c)，并对秘密凭证进行了 NTRU 算法的加密处理，确保了公开的(e,c)和影子子密钥集 k 无法修改。如果有不诚实的成员 P_i 故意提供错误的秘密交换凭证 A_{ij}，使其他成员无法恢复正确的秘密，而自己却可以得到正确的秘密，这种企图同样是不能实现的。因为根据定理 3 的证明以及 c_j 等安全性，任何不正确的 A_{ij} 都无法按照式(13)得到正确的 S_j。同时当 P_i 提供虚假的 A_{ij} 时，P_i 也很容易被揭发出来。

攻击 4 成员之间合谋攻击问题。假如方案中有两个成员 P_i 和 P_j 合谋破坏恢复共享秘密，他们可能交换彼此的 x_i,x_j，即 P_i 持有 x_j，P_j 持有 x_i。如果 P_i 按照式(11)计算出 A_{ij}，并提供给其他成员，则由于 P_i 的身份标志符 id_i 已经公布到 NB 上，对于成员来说几乎是不可能修改的，因此 P_i 提供的 A_{ij} 无法通过式(12)的验证，其他成员也很容易识别出 P_i,P_j 的合谋行为。

6 方案的动态性分析

当方案中对需要增加新的秘密 S_{new} 时，管理员 D 只需要按照式(9)和式(10)计算出新的 e_{new},c_{new},k_{new}，并放到 NB 上就可以了。当方案中需要删除某个秘密 S_{del} 时，D 只要从 NB 上删除 e_{del},c_{del},k_{del} 就可以了，方案中其他信息不需要改变。

当方案中需要增加一个新的成员 P_{new} 时，D 只需要为 P_{new} 分配一个新的 id_{new}，并按照式(6)和式(7)，计算出新的 x_{new},v_{new}，将 x_{new} 分配给 P_{new}，将 v_{new} 发布到电子公告牌 NB 上，方案中其他信息不需要改变。

当方案中需要删除某个成员 P_{del} 时，D 要从 NB 上删除原来的 v,id_{del},c,k 等信息。重新计算一个新的 NTRU 算法密钥对(h_{new},f_{new})，按照式(7)、式(9)、式(10)重新计算 $v'_{new}=(v_1,v_2,\cdots,v_n)$，$c'_{new}=(c_1,c_2,\cdots,c_m)$，$k'_{new}=(k_1,k_2,\cdots,k_m)$，将 $f_{new},v'_{new},c'_{new},k'_{new}$ 发布到 NB 上。方案中其他成员的子密钥等信息不需要改变。被删除的 P_{del} 利用自己的 x_{del} 企图参与秘密恢复时，将无法通过式(8)和式(12)的验证。

7 方案的效率分析

一些文献在分析方案的效率时，主要是依据方案中执行幂运算的次数，幂运算次数越少效率越高。本方案设计独特，没有使用幂运算，只使用加、减乘、模运算。而使用的乘运算量也比其他文献[5,14,15]中使用的乘运算量要少。如，本方案中使用的乘运算量只有文献[14]的三分之一。所以本方案的计算量和复杂度与其他方案相比具有无可比拟的优势。同时，本方案需要的通信量和存储量也有所减少。

结束语 本文提出了一种新的基于(t,n)门限的多秘密共享方案。管理员首先为每个成员分配一个子密钥，并对成员的子密钥提供了可验证向量；管理员还为每个秘密设计了秘密凭证和影子子密钥；重构秘密时，每个成员可以根据公告牌上提供的信息，先计算秘密交换凭证，其他成员在验证秘密交换凭证的真实性后才能重构秘密。在保证安全、可信的基础上，本方案与现有方案相比具有设计简单、执行效率高等优点。当方案需要增加成员或秘密个数发生变化时，成员的子密钥可以重用，需要处理的数据量小，方案具有良好的扩展

性。本方案需要删除成员时，需要处理的数据量仍然较大。今后我们将在此方案的基础上进一步改进多秘密共享方案。

参 考 文 献

[1] Sham ir A. How to share a secret [J]. Communications of the ACM, 1979, 22(11): 612-613

[2] Harn L. Efficient sharing(broadcasting) of multiple secrets[J]. IEE Computers and Digital Techniques, 1995, 142(3): 237-240

[3] Heidarvand S, Villar J L. Public verifiability from pairings in secret sharing schemes[C]// SAC 2008, LNCS 5381: 294-308

[4] Tartary C, Pieprzyk J, Wang Huaxiong. Verifiable multi-secret sharing schemes for multiple threshold access structures[C]// Inscrypt 2007, LNCS 4990: 167-181

[5] 甘元驹，谢仕义，等. 对安全有效的(t,n)多秘密共享认证方案的改进[J]. 电子与信息学报，2007，29(7)：1642-1644

[6] 贺军，李丽娟，李喜梅. 一种新的可验证多秘密共享方案[J]. 计算机工程，2009，35(9)：119-120

[7] 庞辽军，裴庆祺，等. 基于ID的门限多重秘密共乡享方案[J]. 软件学报，2008，19(10)：2739-2745

[8] 步山岳，于坤，王汝传. 一种基于NTRU算法的秘密共享方案[J]. 小型微型计算机系统，2009，30(10)：1986-1987

[9] 刘锋，何业锋，程学翰. 动态的(t,n)门限多秘密分享方案[J]. 计算机应用研究，2008，25(1)：241-245

[10] 王天成，张建中. 一个动态门限多重秘密共享方案[J]. 计算机工程与应用，2009，45(33)：75-76

[11] 殷凤梅，侯整风. 可选子密钥的门限多秘密共享方案[J]. 计算机应用，2007，27(9)：2187-2188

[12] IEEE. Standard specifications for public-key cryptographic techniques based on hard problems over lattices(version D12), October 2008[OL]. http://grouper.ieee.org/groups/1363/lattPK/draft.html

[13] 步山岳，张有东，王汝传. NTRU公开密钥体制快速实现算法[J]. 微电子学与计算机，2008，25(5)：216-220

[14] 祝建华，崔国华，等. 一种可信多重密钥共享认证方案[J]. 小型微型计算机系统，2008，29(4)：635-638

[15] 李锋，李大兴. 一种改进的多重密钥共享的门限方案[J]. 计算机工程，2008，34(5)：11-13

(上接第93页)

下，给整个路由发现过程中带来的额外延迟是极小的，对最终的网络端到端延迟几乎不会带来影响。在节点最大移动速度超过15m/s时，HN-AODV较AODV端到端延迟有所增加，这是因为节点移动导致的链路失效情况增多，重复的路由发现使节点能量迅速下降，剩余能量比值Re的减小导致$RREQDelay$计算公式中分量$\exp(1/(E_r+K_1))$带来的延迟增大，从而略微增加了网络端到端延迟。

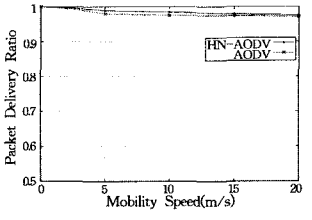

图4 不同速率下的网络包投递率比较

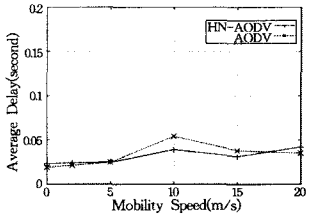

图5 不同速率下的网络平均端到端延迟比较

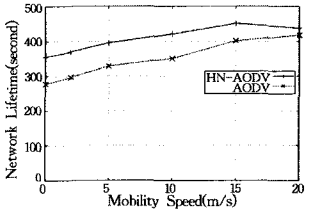

图6 不同速率下的网络生命期比较

图6对比了在不同节点移动速度下采用HN-AODV和AODV的网络的生命周期。可以看到，使用HN-AODV的网络的生命期最高可提高26.3%，平均提高约17%。这是由于HN-AODV使能量相对较低的节点在路由建立过程中被尽量避开；同时照顾到了数据传输过程中能量消耗较大的节点，由其邻居节点分担其转发任务，从而延长了这些节点的存活时间，进而均衡了网络的能耗，延长了网络生命周期。

从以上分析可见，HN-AODV在基本不损失AODV转发性能的前提下显著延长了网络生命周期。

结束语 本文提出了无线自组织网络中的一种高能量节点驱动的AODV协议(HN-AODV)。HN-AODV改进了AODV的HELLO报文和邻居表结构。提出了计算路由发现过程中RREQ报文的转发延时的新方法，在数据传输过程中加入了节点能量感知的路由重发现机制，使得能量充裕的邻居节点可帮助低能量节点承担数据转发任务。使用这些方法，HN-AODV相对于AODV改进了节点能耗的均衡性，从而延长了网络生命周期。这种高能量节点驱动的方案同样可以运用在类似的反应式路由发现协议中。仿真结果显示，HN-AODV协议相对于AODV，在保证了良好端到端转发延时和端到端投递率的基础上，均衡了网络能耗，延长网络生命周期达26.3%。

参 考 文 献

[1] Perkins C E, Royer E M. Ad Hoc On-demand Distance Vector Routing[C]// Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications(WMCSA). 1999: 90-100

[2] Perkins C E, Royer E M, Das S R. Ad Hoc On-demand Distance Vector(AODV) Routing[S]. RFC3561

[3] Shaikh A, Shin K. Destination-driven routing for low-cost multicast[J]. IEEE Journal on Selected Areas in Communications, 1997, 15(3): 373-381

[4] Tian K, Zhang B, Mouftah H, et al. Destination-driven On-demand Multicast Routing Protocol for Wireless Ad Hoc Networks[C]// IEEE International Conference on Communications (ICC). 2009

[5] 刘雯雯，马锐，许海滨. 均衡无线传感器网络能耗的AODV改进方案[J]. 计算机工程，2008，34(22)：143-147

[6] 郝聚涛，赵晶晶，李明禄. 基于能量和链路状态的AODV路由请求转发机制研究[J]. 计算机科学，2009，36(7)：68-70

[7] UCLA Parallel Computing Laboratory and Wireless Adaptive Mobility Laboratory. GloMoSim: A Scalable Simulation Environment for Wireless and Wired Network Systems[D]

[8] Bai F, et al. The IMPORTANT Framework for Analyzing the Impact of Mobility on Performance of Routing for Ad Hoc Networks[J]. Ad Hoc Networks, 2003, 1(4): 383-403