(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 102123473 A
(43)申请公布日 2011.07.13

(21)申请号 201110001610.X

(22)申请日 2011.01.06

(71)申请人 山东大学
地址 250100 山东省济南市历城区山大南路27号

(72)发明人 陈曙 王晶

(74)专利代理机构 济南金迪知识产权代理有限公司 37219
代理人 于冠军

(51)Int.Cl.
H04W 40/10(2009.01)
H04W 40/12(2009.01)
H04W 40/20(2009.01)
H04W 52/28(2009.01)
H04W 56/00(2009.01)
H04W 84/18(2009.01)

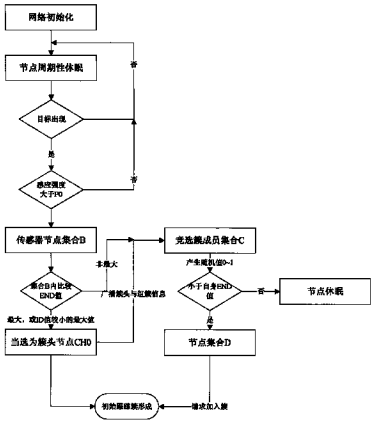
权利要求书 2 页 说明书 5 页 附图 9 页

(54)发明名称

无线传感器网络基于动态簇机制的目标跟踪方法

(57)摘要

本发明公开了一种无线传感器网络基于动态簇机制的目标跟踪方法，该方法中初始簇头的形成采用区域内广播节点END值方法，兼顾节点剩余能量和与目标的距离，在参与目标跟踪节点的选取上，不仅考虑节点的剩余能量状况，而且只选取接收目标的RSSI信号强度大于一定阈值的节点参与跟踪，动态簇的调整引入临时簇头的概念，确保在下个簇建立的过程中，目标不丢失，目标跟踪采用最小二乘法对目标轨迹进行曲线拟合。本发明基于RSSI强度的簇头选择机制，分簇算法兼顾节点剩余能量和与目标的RSSI信号强度等因素，在保证跟踪精度的同时，减少能耗，延长网络寿命。



CN 102123473 A

1. 一种无线传感器网络基于动态簇机制的目标跟踪方法，限于以下假设：定义感知区域为 F，N 个传感器节点随机等密度分布，包括普通节点和汇聚节点，目标周期性的发送射频信号，普通节点仅当收到组簇信号以后才转入工作状态，其它时间处于休眠状态，并周期性对目标信号进行检测；其特征在于，具体包括以下步骤：

(1) 目标跟踪过程

传感器节点集合为 $A = \{A_1, \cdots A_n\}$，传感器的通信半径为 R_N，目标信号的有效传输半径为 R_s，两者满足 $R_s \leqslant R_N/2$，这样就能保证所有能探测到目标的传感器节点，都在彼此的通信范围内，并且只会形成一个簇，传感器节点接收的目标信号强度 RSSI 值为 $P_i(d_i)$，其中 d_i 为节点 i 与跟踪目标的距离，节点 i 的剩余电量为 e_i；节点电量的初始值为 1；目标跟踪过程包括初始簇的创建和簇结构动态调整：

①初始簇的创建

由于传感器资源有限，为了提高传感器网络的寿命，采用动态簇结构进行分布式跟踪，在没有侦测任务的时候，所有传感器处于休眠状态，并不形成簇结构，以节省能量；当目标第一次进入传感器网络的侦测区域时，节点感应到目标的 RSSI 强度大于 P_0 时，即传感器节点与目标的距离小于阈值 R_0 时，这些节点构成集合 $B = \{B_1 \cdots B_{n_1}\}$，它们共同参与到簇头的竞选过程中；节点接收到的 RSSI 强度小于 P_0 或是没有侦测到目标信号，将继续保持休眠状态；设节点的侦测距离为 R_s，当目标距节点距离在 $R_0 \leqslant (R_s - e)$ 时总会被检测到，$e = 0.1R_s$，使用改进型无线电自由空间传播模型：

$$P(d)[dBm] = P(d_0)[dBm] - 10n\log(\frac{d}{d_0})$$

其中，P(d) 表示节点与目标距离为 d 时的接收功率强度；$P(d_0)$ 是基准距离为 d_0 时的信号强度；n 表示路径长度和路径损耗之间的比例因子，实现时取 d_0 为 1，$n = 2$；节点集合 $B = \{B_1 \cdots B_{n_1}\}$，定义：

$$END_i = e_i^{\alpha}\left[1 - \frac{d_i}{R_s}\right]^{\beta} \qquad d_i < R_0 \qquad END_i \in (0,1)$$

为节点的 END_i 值，其中 e_i 为传感器节点 B_i 当前的剩余能量，d_i 为节点 i 与跟踪目标的距离，R_s 为传感器的感知半径，α，β 为权重因子，节点 i 自动计算当前的 END_i 值，如果是目标已经在该区域出现一次以上，则使用上次更新过的 END_i 值，并广播节点的相关控制信息，同时节点 i 也会收到 B 集合中的其它节点发送的广播信息；

节点 i 比较自身的 END 值与接收到的 END 值大小，若该节点的 END 值小于任意一个节点当前的 END 值，则进入等待状态；若该节点的 END 值大于任意一个节点 END 值，则当选为簇头，并发布组簇信息，激活相应的节点参与组簇；若该节点的 END 值与其他节点相同且同时为最大值，则选择 ID 号较小的作为 CH_0，当选为 CH_0 的节点发布组簇信息，组簇信息包括节点 ID，位置信息和时间同步标签，并激活处于等待状态的节点参与组簇；

B 集合的其它等待节点，收到簇头的组簇信号，节点 k 保留 CH_0 的 ID 与位置信息，然后节点 k 产生一个 0-1 的随机数，并与自身节点的 END_k 比较，若大于 END_k，则不加入簇，节点进入重新休眠状态；若小于 END_k，则保存 CH_0 的信息，调整本地时钟与簇头同步，成为簇成员；

②簇结构动态调整，包括簇成员的调整和簇头的调整

A. 簇成员的调整

簇成员接收到的目标RSSI信号随着目标的移动而改变，节点j在发送目标位置信息的时刻同时自身进行记录，当连续三次记录的目标位置都大于R_0时，该节点j向簇头节点发出退出信号并自动退出簇结构，簇头节点收到节点j的退出信息后，将节点j从簇成员列表中删除，不再作为计算目标的数据来源，并同时广播允许新节点加入的信号，收到簇头广播信号的非簇成员节点h在连续接收目标位置都小于R_0后，计算自身的END_h值，产生0-1的随机数，若小于END_h，则选择加入簇头，请求信息包括自身ID，位置信息，END_h值等，若大于END_h，则不加入簇头，簇头在收到节点请求加入信息后，选择最大的END_h值节点加入簇；

B. 簇头的调整

随着目标的移动，目标距离簇头的距离越来越远，当目标距离簇头的距离满足条件$d_{CH} > \varepsilon R_s$时，$\varepsilon = 0.4 \sim 0.5$，启动簇头移交机制，此时当前簇头退化为临时簇头，临时簇头向簇成员发送簇头调整信号，簇成员在向簇头发送目标信息的同时，与感应到目标强度的非簇成员节点按照END值重新竞争簇头，当新的簇头确定后，原簇头CH_0发送簇结束信号，簇头退化为普通节点，原簇成员和感应到目标的新节点重新按概率加入到新的簇头，开始新一轮的跟踪；

(2) 目标的定位与预测

初始簇形成以后，加入到簇中的节点成员定义为集合$D = \{D_1 \cdots D_m\}$，目标某时刻的坐标为(x,y)，簇中的成员节点周期性将侦测目标的RSSI强度转换成相应的距离d，按时间先后顺序记录在自己的内存中，然后发送给簇头，簇头节点i根据簇成员的位置信息和距离d，利用极大似然估计法对目标位置进行定位，距离d的转换用到公式：

$$d_j = d_0 \cdot 10^{\frac{P(d_0)-P(d_j)}{10n}}$$

然后使用极大似然估计法定位当前目标的位置；

目标预测就是预测目标下一时刻的位置和速度，据此进行任务和资源的重新分配，为减少迭代等复杂计算带来的能量损耗和时间延迟，选取最小二乘法曲线估计对目标进行预测，设目标运动曲线服从下列多项式函数分布：

$$p_n(x) = \sum_{j=0}^{n} a_j x^j$$

当使下列式取最小值时，$R == \sum_{i=0}^{m}[p_n(x_i) - y_i]^2 = \sum_{i=0}^{m}[\sum_{k=0}^{n} a_k x_i^k - y_i]^2 = \min$最逼近实际函数分布，即$\frac{\partial R}{\partial a_i} = 2\sum_{i=0}^{m}(\sum_{k=0}^{n} a_k x_i^k - y_i)x_i^j = 0$，最后得到下式：

$$\begin{pmatrix} m+1 & \sum_{i=0}^{m} x_i & \cdots & \sum_{i=0}^{m} x_i^n \\ \sum_{i=0}^{m} x_i & \sum_{i=0}^{m} x_i^2 & \cdots & \sum_{i=0}^{m} x_i^{n+1} \\ \vdots & \vdots & \vdots & \vdots \\ \sum_{i=0}^{m} x_i^n & \sum_{i=0}^{m} x_i^{n+1} & \cdots & \sum_{i=0}^{m} x_i^{2n} \end{pmatrix} \cdot \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^{m} y_i \\ \sum_{i=0}^{m} x_i y_i \\ \vdots \\ \sum_{i=0}^{m} x_i^n y_i \end{pmatrix}$$

求得a_k即可得到目标的函数分布，由此预测下一时刻的位置。

无线传感器网络基于动态簇机制的目标跟踪方法

技术领域

[0001] 本发明涉及一种用于无线传感器网络的目标跟踪方法，属于目标跟踪技术领域。

背景技术

[0002] 无线传感器网络是用来对环境进行测控的，其中一个重要应用是目标跟踪。由于传感器节点体积小，价格低廉，以及传感器网络部署随机，具有自组织性，鲁棒性和隐蔽性等特点，传感器网络非常适合于移动目标的定位和跟踪。另一方面，无线传感器网络又具有网络节点的电池能量和计算能力严格受限，节点数量庞大且高密度部署，网络拓扑变化复杂等特征，使相关的课题研究极具挑战性。

[0003] 国内外许多学者对无线传感器单目标跟踪进行了深入的研究，提出了很多目标跟踪算法，主要有双元检测算法、信息驱动协作跟踪算法和传送树跟踪算法。以上算法中，基于传送树的跟踪算法是一种基于动态簇选取的分布式算法，即在目标区域建立一个动态树型结构，并随着目标的移动动态地添加或者删除节点，以达到保证目标的高效跟踪和减少节点通信开销，延长网络寿命的目的。分布式算法由于能够减少能量消耗，延长网络寿命，是现在的主流与热点研究方向。

[0004] Chen W P，Hou J C，Sha L 研究了声传感器网络进行目标跟踪问题，利用 Voronoi（泰森多边形）图实现了一种动态簇创建方法，缺陷在于簇头竞选，需要事先指定一些节点作为候选簇首集合，适用性较窄（无限传感器网络动态簇声目标跟踪研究，IEEE Transactions on Mobile Computing，3(3)，2004：258-271）。Zhang W S，Cao G H 将 DCTC 框架用于无线传感器网络自组织，该方法通过动态增加和消减节点，保证生成树同时具有较低的能量消耗和较高的信息获得量，缺陷是计算过于复杂，难以应用（基于能量有效的目标跟踪算法，Proc.of the IEEE Military Communications Conference，2003：597-602）。万江文提出时间异步条件下的动态簇算法，具有较好的跟踪精度，且平衡了网络能耗，缺陷在于初始簇头的选择为距离目标最近节点，没有考虑能量因素（时间异步无限传感器网络分布式目标跟踪．高技术通讯，2009：1026-1030）。

发明内容

[0005] 本发明针对现有无线传感器网络目标跟踪技术存在的不足，提供一种兼顾节点剩余能量和与目标的距离的无线传感器网络基于动态簇机制的目标跟踪方法。

[0006] 本发明的无线传感器网络基于动态簇机制的目标跟踪方法，限于以下假设：定义感知区域为 F，N 个传感器节点随机等密度分布，包括普通节点和汇聚 (Sink) 节点，目标周期性的发送射频信号，普通节点仅当收到组簇信号以后才转入工作状态，其它时间处于休眠状态，并周期性对目标信号进行检测；图 1 给出了简化的传感器结构系统模型：1) 节点空间部署等密度且随机，坐标位置 (x，y) 可知；2) 每个节点有唯一的 ID，具有数据融合功能；3) 节点同构对等，无线传播范围相同；4) 节点具有组簇能力。

[0007] 具体包括以下步骤：

[0008] (1) 目标跟踪过程

[0009] 传感器节点集合为 A = {A_1, …A_n}，传感器的通信半径为 R_N，目标信号的有效传输半径为 R_s（或称节点的侦测距离），两者满足 $R_s \leq R_N/2$，这样就能保证所有能探测到目标的传感器节点，都在彼此的通信范围内，并且只会形成一个簇，如图 2 所示。传感器节点接收的目标信号强度 RSSI 值为 $P_i(d_i)$，其中 d_i 为节点 i 与跟踪目标的距离，节点 i 的剩余电量为 e_i；节点电量的初始值为 1（即 100％）。动态簇跟踪算法的流程如图 3 所示，目标跟踪过程包括初始簇的创建和簇结构动态调整：

[0010] ①初始簇的创建

[0011] 初始簇的创建流程图如图 4 所示。由于传感器资源有限，为了提高传感器网络的寿命，采用动态簇结构进行分布式跟踪，在没有侦测任务的时候，所有传感器处于休眠状态，并不形成簇结构，以节省能量；当目标第一次进入传感器网络的侦测区域时，节点感应到目标的 RSSI 强度大于 P_0 时，即传感器节点与目标的距离小于阈值 R_0 时，这些节点构成集合$B=\{B_1 \cdots B_{n_1}\}$，它们共同参与到簇头的竞选过程中；节点接收到的 RSSI 强度小于 P_0 或是没有侦测到目标信号，将继续保持休眠状态；设节点的侦测距离为 R_s，当目标距节点距离在 $R_0 \leq (R_s-e)$ 时总会被检测到；通常情况下 $e = 0.1R_s$，使用改进型无线电自由空间传播模型：

[0012] $$P(d)[dBm]=P(d_0)[dBm]-10n\log(\frac{d}{d_0}) \qquad (1)$$

[0013] 其中，P(d) 表示节点与目标距离为 d 时的接收功率强度；$P(d_0)$ 是基准距离为 d_0 时的信号强度；n 表示路径长度和路径损耗之间的比例因子，依赖于建筑物的结构和使用的材料；实现时取 d_0 为 1，n = 2；节点集合$B=\{B_1 \cdots B_{n_1}\}$，定义：

[0014] $$END_i = e_i^{\alpha}\left[1-\frac{d_i}{R_s}\right]^{\beta} \qquad d_i < R_0 \qquad END_i \in (0,1) \qquad (2)$$

[0015] 为节点的 END_i 值，其中 e_i 为传感器节点 B_i 当前的剩余能量，d_i 为节点 i 与跟踪目标的距离，R_s 为传感器的感知半径，α，β 为权重因子，节点 i 自动计算当前的 END_i 值，如果是目标已经在该区域出现一次以上，则使用上次更新过的 END_i 值，并广播节点的 ID、组簇使能信号等相关控制信息，同时节点 i 也会收到 B 集合中的其它节点发送的广播信息；

[0016] 节点 i 比较自身的 END 值与接收到的 END 值大小，若该节点的 END 值小于任意一个节点当前的 END 值，则进入等待状态；若该节点的 END 值大于任意一个节点 END 值，则当选为簇头 (CH_0)，并发布组簇信息，激活相应的节点参与组簇；若该节点的 END 值与其他节点相同且同时为最大值，则选择 ID 号较小的作为 CH_0，当选为 CH_0 的节点发布组簇信息，组簇信息包括节点 ID，位置信息和时间同步标签，并按一定概率激活处于等待状态的节点参与组簇；

[0017] B 集合的其它等待节点（竞选簇成员节点集合 C，以节点 k 为例）收到簇头的组簇信号，节点 k 保留 CH_0 的 ID 与位置信息。然后节点 k 产生一个 0 ～ 1 的随机数，并与自身节点的 END_k 比较，若大于 END_k，则不加入簇，节点进入重新休眠状态；若小于 END_k，则保存 CH_0 的信息（簇动态调整时使用），调整本地时钟与簇头同步，成为簇成员。以上步骤完成了一个初始簇的创建。由公式 (2) 可知，节点 k 剩余电量 e_k 越大，距离目标 d_k 越小，相应的 END_k 值也越大，节点 k 产生的随机数小于 END_k 的概率也越大，成为簇成员的概率就越大，当节点

k 的剩余能量 e_k 越小，距离目标 d_k 越大，成为簇成员的概率越小。这样网络能量可以得到充分的利用，能量消耗也可以被平均到各个节点，延长网络寿命，同时距离目标近的节点测量误差较小，保证跟踪精度。

[0018] ②簇结构动态调整，包括簇成员的调整和簇头的调整

[0019] A 簇成员的调整

[0020] 图 5 为簇成员调整示意图。簇成员接收到的目标 RSSI 信号随着目标的移动而改变，节点 j 在发送目标位置信息的时刻同时自身进行记录。当连续三次记录的目标位置都大于 R_0 时，该节点 j 向簇头节点发出退出信号并自动退出簇结构，簇头节点收到节点 j 的退出信息后，将节点 j 从簇成员列表中删除，不再作为计算目标的数据来源，并同时广播允许新节点加入的信号，收到簇头广播信号的非簇成员节点 h 在连续接收目标位置都小于 R_0 后，计算自身的 END_h 值，产生 0-1 的随机数，若小于 END_h，则选择加入簇头，请求信息包括自身 ID，位置信息，END_h 值等，若大于 END_h，则不加入簇头，簇头在收到节点请求加入信息后，选择最大的 END_h 值节点加入簇；

[0021] B 簇头的调整

[0022] 随着目标的移动，目标距离簇头的距离越来越远，当目标距离簇头的距离满足条件 $d_{CH} > \varepsilon R_s$ 时（$\varepsilon = 0.4 \sim 0.5$），启动簇头移交机制，此时当前簇头退化为临时簇头，临时簇头向簇成员发送簇头调整信号，簇成员在向簇头发送目标信息的同时，与感应到目标强度的非簇成员节点按照 END 值重新竞争簇头，当新的簇头确定后，原簇头 CH_0 发送簇结束信号，簇头退化为普通节点，节点（原簇成员和感应到目标的新节点）重新按概率加入到新的簇头，开始新一轮的跟踪。

[0023] (2) 目标的定位与预测

[0024] 初始簇形成以后，加入到簇中的节点成员定义为集合 $D = \{D_1 \cdots D_m\}$，目标某时刻的坐标为 (x,y)，簇中的成员节点周期性将侦测目标的 RSSI 强度转换成相应的距离 d，按时间先后顺序记录在自己的内存中，然后发送给簇头，簇头节点 i 根据簇成员的位置信息和距离 d，利用极大似然估计法对目标位置进行定位。距离 d 的转换用到公式：

[0025] $$d_J = d_0 \cdot 10^{\frac{P(d_0)-P(d_J)}{10n}} \tag{3}$$

[0026] 然后使用极大似然估计法定位当前目标的位置。

[0027] 目标预测就是预测目标下一时刻的位置和速度，据此进行任务和资源的重新分配。目标预测算法有主要有序贯最小二乘预测算法，贝叶斯滤波估计，卡尔曼滤波算法，粒子滤波。为减少迭代等复杂计算带来的能量损耗和时间延迟，选取最小二乘法曲线估计对目标进行预测。设目标运动曲线服从下列多项式函数分布：

[0028] $$p_n(x) = \sum_{J=0}^{n} a_J x^J \tag{4}$$

[0029] 当使下列式取最小值时，$R == \sum_{i=0}^{m}[p_n(x_i) - y_i]^2 = \sum_{i=0}^{m}[\sum_{k=0}^{n} a_k x_i^k - y_i]^2 = \min$ 最逼近实际函数分布，即 $\frac{\partial R}{\partial a_i} = 2\sum_{i=0}^{m}(\sum_{k=0}^{n} a_k x_i^k - y_i)x_i^J = 0$。最后得到下式：

[0030] $$\begin{pmatrix} m+1 & \sum_{i=0}^{m} x_i & \cdots & \sum_{i=0}^{m} x_i^n \\ \sum_{i=0}^{m} x_i & \sum_{i=0}^{m} x_i^2 & \cdots & \sum_{i=0}^{m} x_i^{n+1} \\ \vdots & \vdots & \vdots & \vdots \\ \sum_{i=0}^{m} x_i^n & \sum_{i=0}^{m} x_i^{n+1} & \cdots & \sum_{i=0}^{m} x_i^{2n} \end{pmatrix} \cdot \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^{m} y_i \\ \sum_{i=0}^{m} x_i y_i \\ \vdots \\ \sum_{i=0}^{m} x_i^n y_i \end{pmatrix} \quad (5)$$

[0031] 求得 a_k 即可得到目标的函数分布，由此预测下一时刻的位置。

[0032] 上述目标跟踪方法，初始簇头的形成采用区域内广播节点 END(energy and distance) 值方法，兼顾节点剩余能量和与目标的距离，简单有效；在参与目标跟踪节点的选取上，不仅考虑节点的剩余能量状况，而且只选取接收目标的 RSSI 信号强度大于一定阈值的节点参与跟踪。动态簇的调整引入临时簇头的概念，确保在下个簇建立的过程中，目标不丢失。目标跟踪引入预测机制，采用最小二乘法对目标轨迹进行曲线拟合。

[0033] 本发明基于 RSSI 强度的簇头选择机制，分簇算法兼顾节点剩余能量和与目标的 RSSI 信号强度等因素，初始簇的创建简单有效，相关节点按照剩余能量和目标信号强度选择是否加入簇头，根据目标的移动，簇结构进行动态调整，实时唤醒相应节点对目标实施持续跟踪。在保证跟踪精度的同时，减少能耗，延长网络寿命。

附图说明

[0034] 图 1 是传感器结构模型示意图。

[0035] 图 2 是目标有效传输半径与普通节点通信半径示意图。

[0036] 图 3 是动态簇跟踪算法流程图。

[0037] 图 4 是初始簇的创建流程图。

[0038] 图 5 是簇成员调整示意图。

[0039] 图 6 为 t = 100s 采样时刻簇组织情况示意图，其中黑色直线为真实目标轨迹，黑色叉点为普通节点，红色叉点为竞选簇成员节点，圈为簇成员节点，加号为目标跟踪轨迹。

[0040] 图 7 为目标匀速运动时的仿真效果图；

[0041] 图 8 为此匀速运动的误差图；

[0042] 图 9 为目标做变速运动时的仿真效果图；

[0043] 图 10 为变速运动的误差图。

具体实施方式

[0044] 利用 Matlab 对本发明的方法进行仿真，以评估所提出的组簇算法的可行性。

[0045] 首先建立二维目标运动模型：

[0046] $x_k = \Phi x_{k-1} + \Gamma w_k$ (6)

[0047] (1) 当目标做匀速运动时，$x_k = (x, \dot{x}, y, \dot{y})^T$ 为目标状态变量，即 $(x,y)^T$ 为位置向量，$(\dot{x}, \dot{y})^T$ 为速度向量，Φ 为状态转移矩阵，Γ 为噪声输入矩阵，w_k 为零均值高斯白噪声，其值分别为：

[0048] $$\Phi=\begin{pmatrix}1&1&0&0\\0&1&0&0\\0&0&1&1\\0&0&0&1\end{pmatrix},\Gamma=\begin{pmatrix}0.5&0\\1&0\\0&0.5\\0&1\end{pmatrix},\quad w_k=(w_x,w_y)_k^T$$

[0049] (2) 当目标做变速运动(y 方向)时，$x_k=(y,\dot{y},\ddot{y})^T$为目标状态变量，Φ 为状态转移矩阵，Γ 为噪声输入矩阵，w_k 为零均值高斯白噪声，其值分别为：

[0050] $$\Phi=\begin{pmatrix}1&1&0.05\\0&1&1\\0&0&1\end{pmatrix},\ \Gamma=(0\ 0\ 1)^T,\ w_k=w_y$$

[0051] 假设目标在二维空间内运动，感知区域为(0m,0m)到(200m,200m)的矩形区域。参数设置如下表：

[0052]

参数	描述	取值
R_N	节点通信半径	40m
R_s	目标有效传输半径	20m
e_k	节点剩余能量	100%
α，β	权重因子	1,1
ε	簇头移交因子	0.4
N	节点个数	400
Δt	采样周期	1s

[0053]

[0054] 在仿真过程中由于节点分布的随机性，导致一个区域内的簇成员个数差异较大，会对仿真算法的复杂性(簇成员大于十个)和准确性(簇成员小于三个)造成影响，所以规定簇成员的个数为 3-6。图 6 为 t = 100s 采样时刻簇组织情况，其中黑色直线为真实目标轨迹，黑色叉点为普通节点，红色叉点为竞选簇成员节点，蓝色圈为簇成员节点，黑色加号为目标跟踪轨迹；图 7 为目标匀速运动时的仿真效果图；图 8 为此匀速运动的误差图；图 9 为目标做变速运动时的仿真效果图；图 10 为变速运动的误差图。

[0055] 通过以上仿真可以看出，无论是匀速还是变速运动，本发明的方法都能对目标进行有效地跟踪，由于采用的是动态簇机制算法，每一时刻参与跟踪的节点都是剩余能量与位置优化过的节点，且期间大部分节点处于休眠状态，因此既可以保证目标跟踪精度，又能大大的减少能耗，与传统的集中式跟踪算法相比，网络寿命可以得到显著的提高。

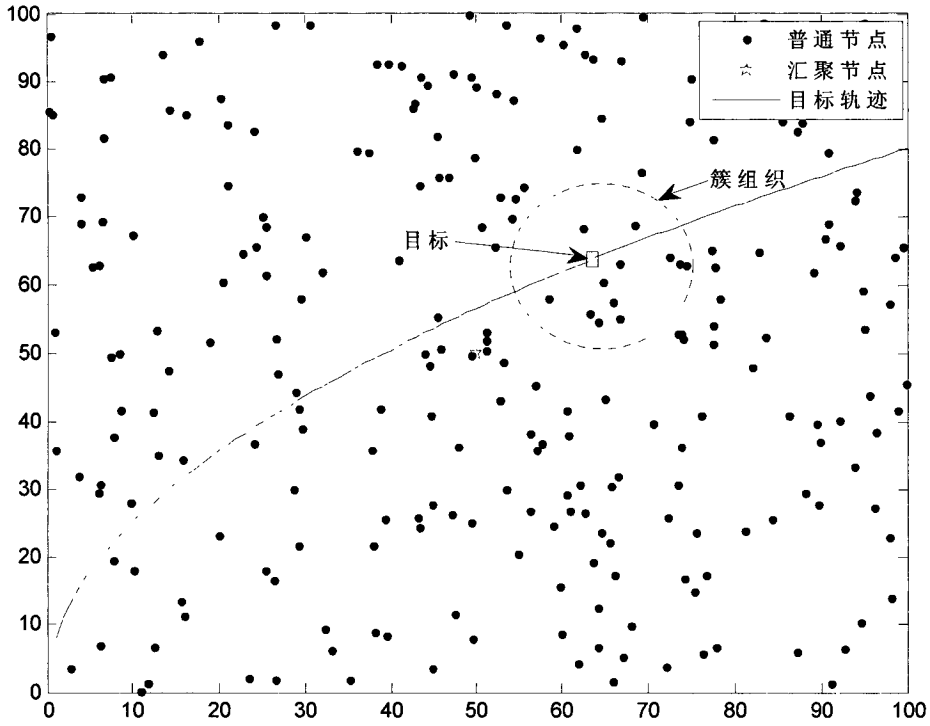

普通节点
汇聚节点
目标轨迹
簇组织
目标
100
90
80
70
60
50
40
30
20
10
0
0
10
20
30
40
50
60
70
80
90
100

图 1

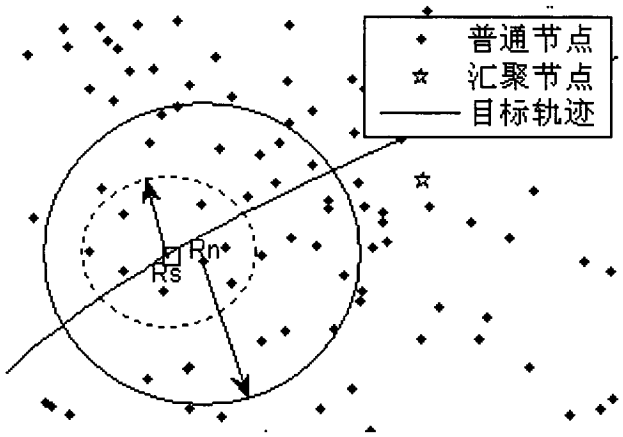

普通节点
汇聚节点
目标轨迹
Rn
Rs

图 2

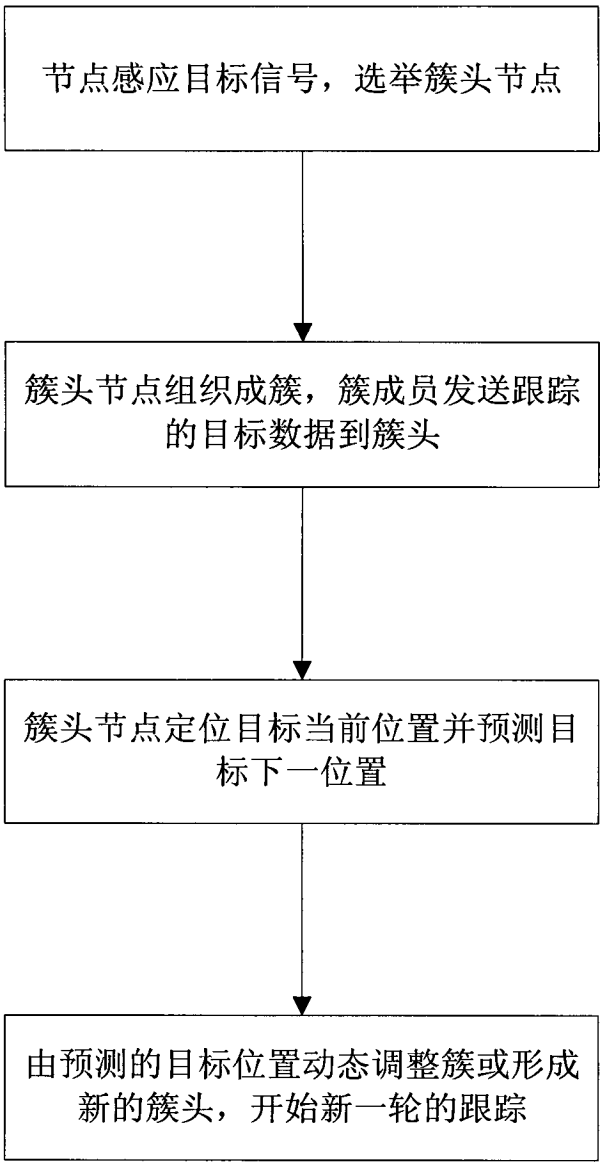

节点感应目标信号，选举簇头节点
簇头节点组织成簇，簇成员发送跟踪的目标数据到簇头
簇头节点定位目标当前位置并预测目标下一位置
由预测的目标位置动态调整簇或形成新的簇头，开始新一轮的跟踪

图 3

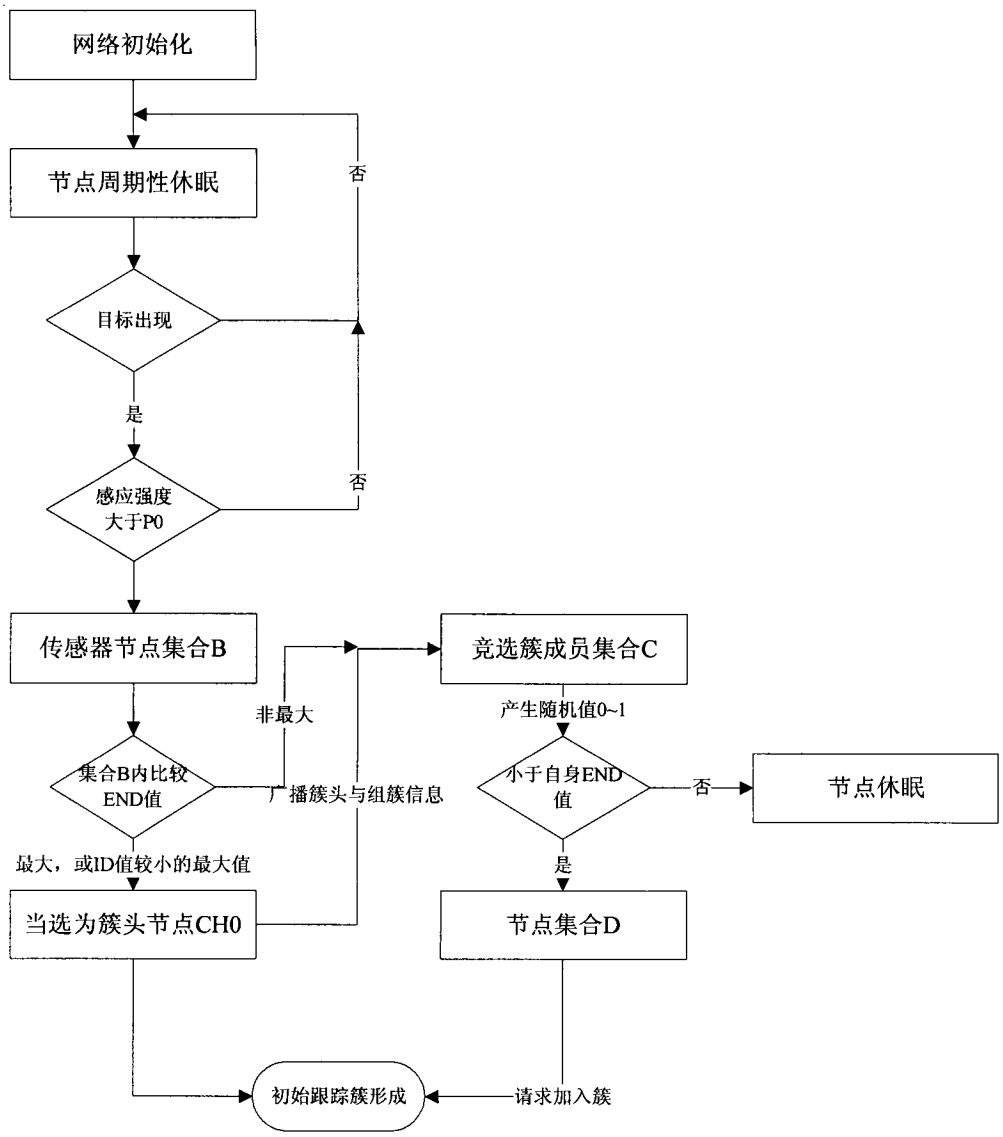

网络初始化
节点周期性休眠
否
目标出现
是
感应强度
大于P0
否
传感器节点集合B
竞选簇成员集合C
非最大
产生随机值0~1
集合B内比较
END值
广播簇头与组簇信息
小于自身END
值
否
节点休眠
最大，或ID值较小的最大值
是
当选为簇头节点CH0
节点集合D
初始跟踪簇形成
请求加入簇

图 4

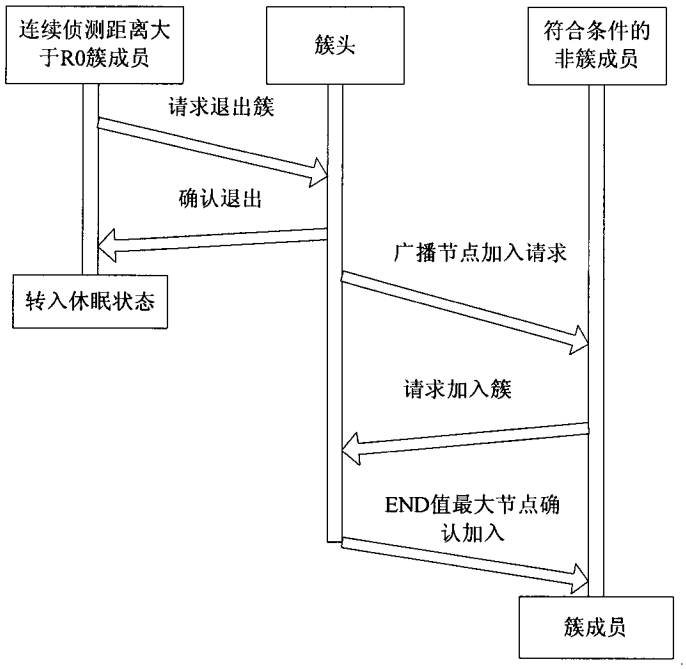

连续侦测距离大于R0簇成员
簇头
符合条件的非簇成员
请求退出簇
确认退出
广播节点加入请求
转入休眠状态
请求加入簇
END值最大节点确认加入
簇成员

图 5

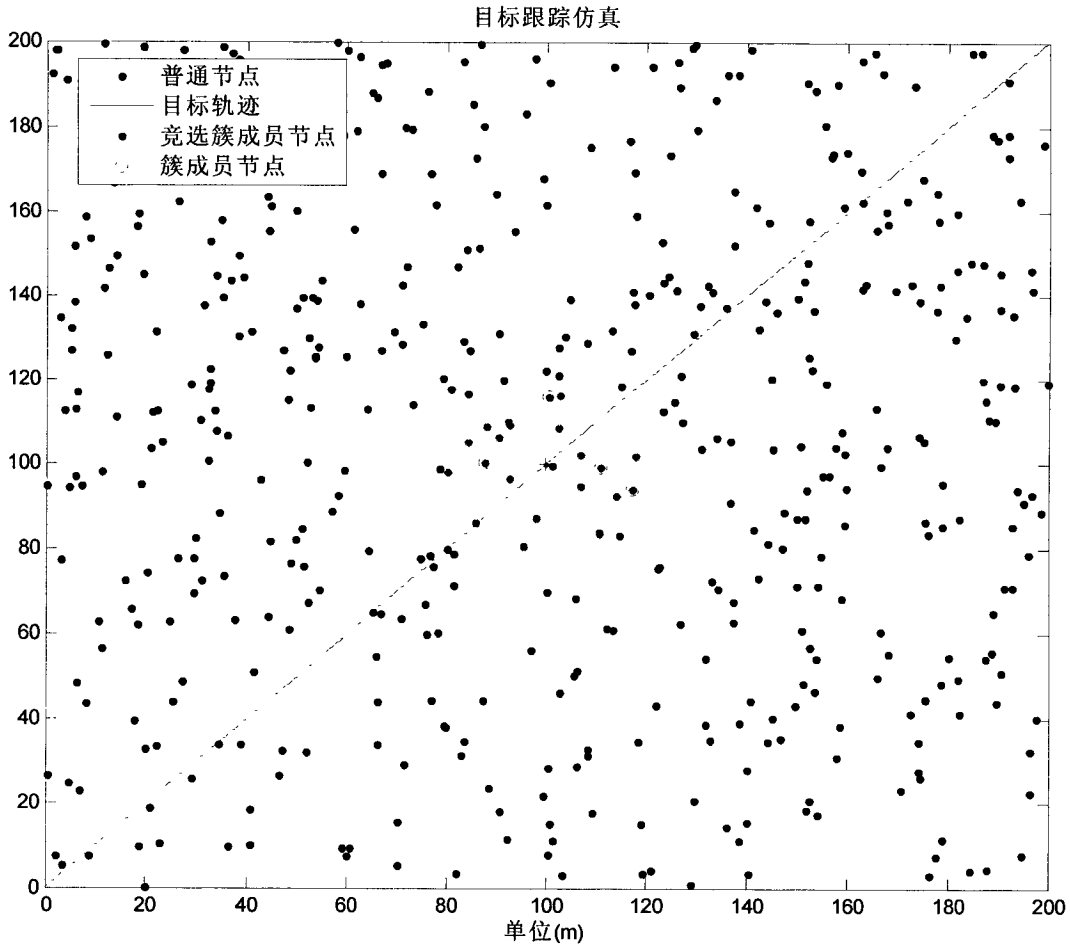

目标跟踪仿真
普通节点
目标轨迹
竞选簇成员节点
簇成员节点
200
180
160
140
120
100
80
60
40
20
0
0
20
40
60
80
100
120
140
160
180
200
单位(m)

图 6

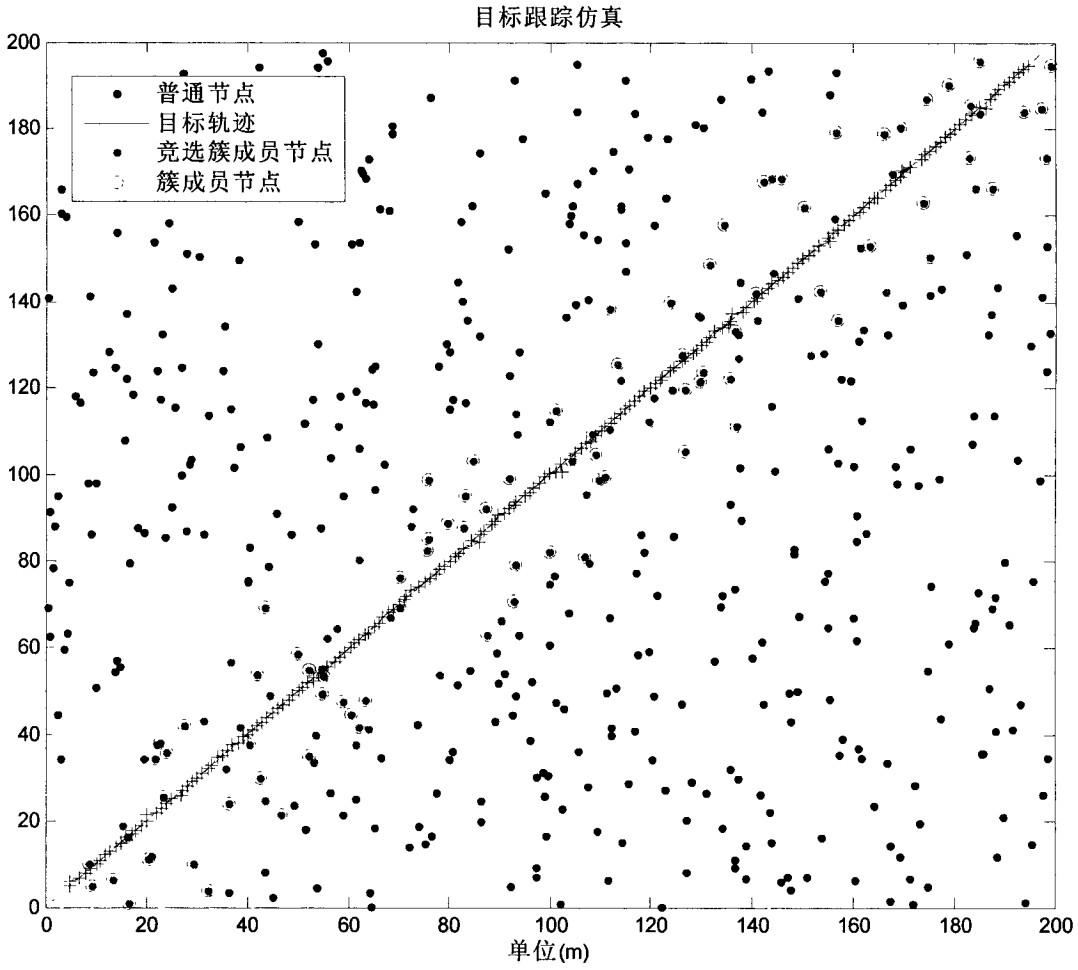

目标跟踪仿真
普通节点
目标轨迹
竞选簇成员节点
簇成员节点
200
180
160
140
120
100
80
60
40
20
0
0
20
40
60
80
100
120
140
160
180
200
单位(m)

图 7

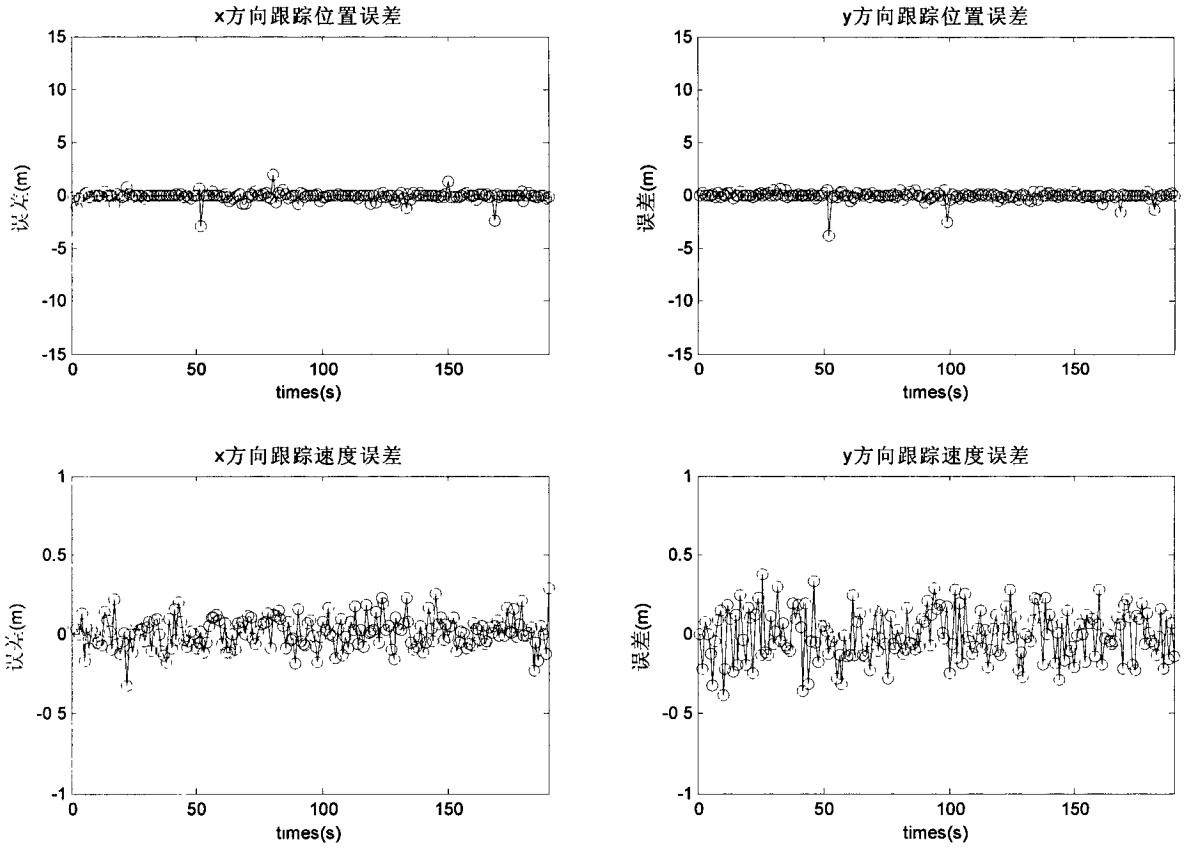

x方向跟踪位置误差
误差(m)
times(s)
y方向跟踪位置误差
误差(m)
times(s)
x方向跟踪速度误差
误差(m)
times(s)
y方向跟踪速度误差
误差(m)
times(s)

图 8

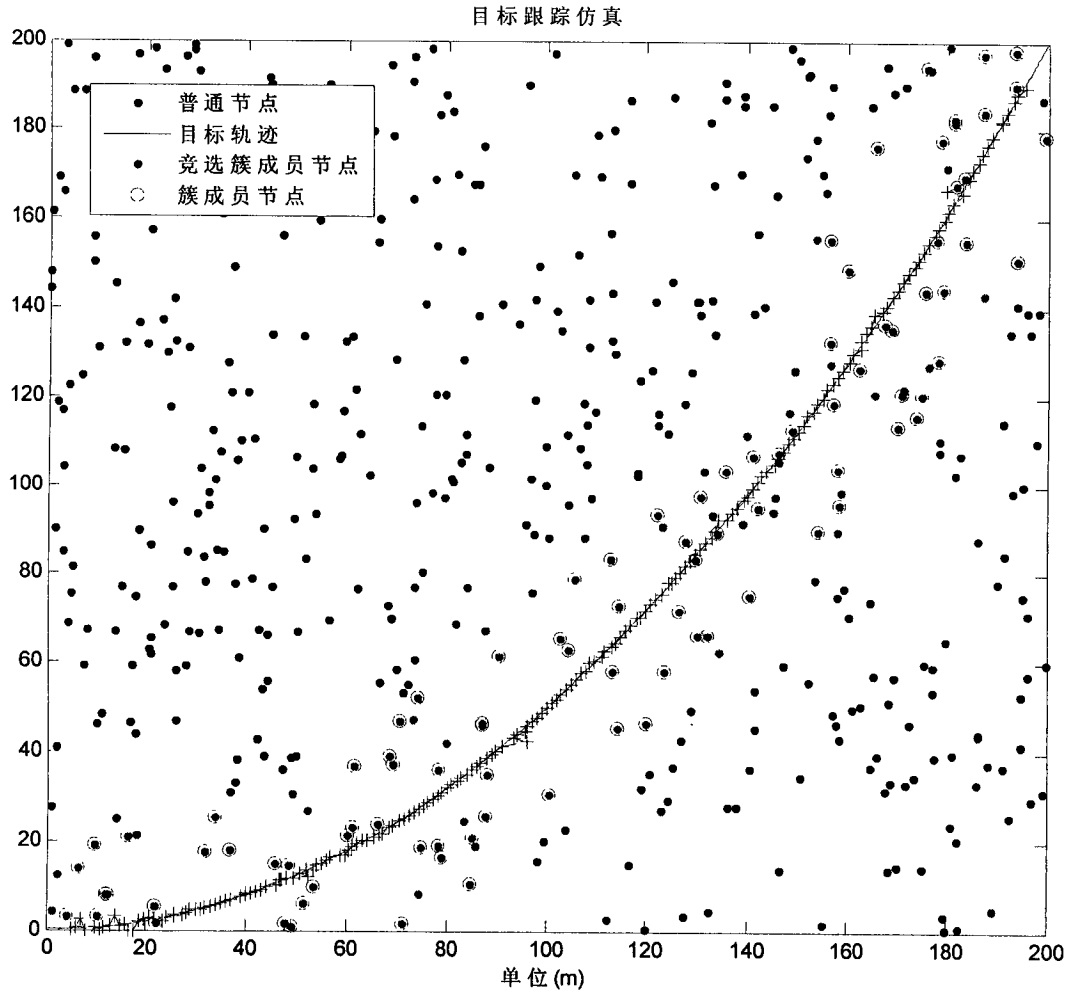

目标跟踪仿真
普通节点
目标轨迹
竞选簇成员节点
簇成员节点
200
180
160
140
120
100
80
60
40
20
0
0
20
40
60
80
100
120
140
160
180
200
单位(m)

图 9

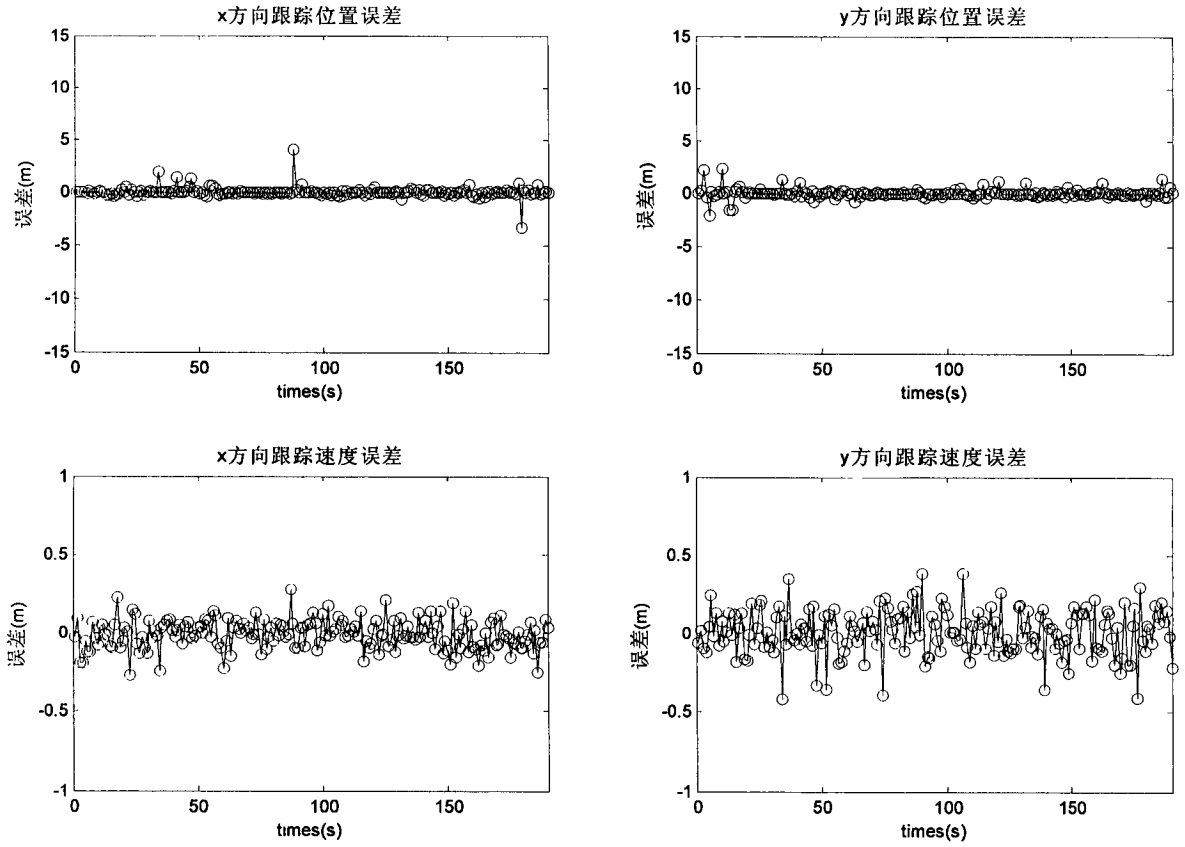

x方向跟踪位置误差
误差(m)
times(s)
y方向跟踪位置误差
误差(m)
times(s)
x方向跟踪速度误差
误差(m)
times(s)
y方向跟踪速度误差
误差(m)
times(s)

图 10