Anycast Routing Protocol for Wireless Sensor Networks Based on Node Disjoint Scheme *

GU Yunli*, XU Xin
(School of Software, Nanjing University of Information Science and Technology, Nanjing Jiangsu 210044, China)

Abstract: The node joint problem is often ignored in previous anycast routing protocols for wireless sensor networks (WSN). In WSN, if a node is tasked with transmitting in multiple anycast paths, it will deplete its energy quickly. For this problem, an anycast routing protocol for WSN based on node disjoint scheme is proposed. The proposed protocol adopts a node disjoint scheme to improve the network lifetime. Distributed query protocol is used in finding paths, which reduces routing query overhead. Backward path strategy is used, which makes the quality of anycast paths obtained through routing queries better. Experiments results shows that compared to those WSN anycast routing protocols based on node joint scheme, the proposed protocol has a good performance of the network lifetime, and compared to traditional WSN anycast routing protocols based on node disjoint scheme, the proposed protocol has less routing query overhead and better anycast path quality.

Key words: wireless sensor networks; node disjoint; backward path; anycast; routing protocol
EEACC: 6150P **doi**: 10.3969/j.issn.1004-1699.2025.11.020

基于节点不相交的无线传感器网络任播路由协议 *

顾云丽*, 徐 昕
(南京信息工程大学软件学院, 江苏 南京 210044)

摘 要: 以往无线传感器网络(WSN)任播路由协议往往忽略节点相交问题。在 WSN 中, 若某个节点承担多条任播路径的传输任务, 将导致该节点能量过快耗尽。针对上述问题, 提出一种基于节点不相交的 WSN 任播路由协议。该协议采用节点不相交策略, 提高网络生存期; 采用分布式查询路由, 路由查询开销较少; 采用后向路径策略, 路由查询得到的任播路径质量较优。实验结果表明, 相较基于节点相交 WSN 任播路由协议, 所提出的协议网络生存期较优; 相较传统基于节点不相交的 WSN 任播路由协议, 所提出的协议路由查询开销少, 任播路径质量较优。

关键词: 无线传感器网络; 节点不相交; 后向路径; 任播; 路由协议
中图分类号: TP393 **文献标识码**: A **文章编号**: 1004-1699(2025)11-2076-06

无线传感器网络(Wireless Sensor Networks, WSN)往往用于战场、环境等监控场景, 需要周期性将监测数据发往基站。为了节能以及能耗均衡, WSN 往往部署多个基站, 节点可以根据任播路径情况将监测数据任播(anycast)至任一基站处。

任播技术已在许多领域得到应用[1-4]。在 WSN 中, 如何查询源节点至各基站开销最优的路径成为 WSN 任播路由协议[5-8]的一个设计难题。Rachkidy 等[9]证明多路径任播路由最优化问题是一个 NP 难题, 即该优化问题只能得到近似最优解。

以往学者在解决 WSN 任播路由问题时, 往往忽略多条任播路径中有节点相交的问题。在 WSN 中, 如果某个节点承担多条任播路径的传输任务, 将导致该节点能量过快耗尽, 从而导致网络生存期较短(本文定义, 网络生存期为到第一个节点失效的时间)。

因此, 有学者提出节点不相交路由策略。如文献[10]提出一种应用于软件定义车联网的多径节点不相交路由算法; Velusamy 等[11]提出一种应用于移动 Ad Hoc 网络的基于多目标优化的节点不相交路由算法; Xu 等[12]提出一种应用于多波长光通信网络的具有可扩展性的节点不相交和边缘不相交路由算法; Ye 等[13]提出一种应用于多媒体传感器网络的基于节点不相交的多径路由算法。

项目来源: 国家自然科学基金面上项目(42075138)
收稿日期: 2023-08-29 **修改日期**: 2023-12-22

本文将节点不相交策略应用于 WSN 任播路由协议。相比较基于节点相交的路由协议,该协议能耗更均衡,网络生存期更优;该协议还采用了后向路径策略,查询到的任播路径质量更优;该协议还采用分布式查询路径策略,节点只需与相邻节点交换信息,无需掌握全局网络信息,协议开销较小。

1　问题描述

将 WSN 抽象视为连通图 $\boldsymbol{G}(\boldsymbol{V},\boldsymbol{E})$。其中,$\boldsymbol{V}$ 为传感器节点集合,$\boldsymbol{E}$ 为节点间有向通信链路集合。本文标记:$\boldsymbol{A}$ 为任播地址;$\boldsymbol{G}(\boldsymbol{A})$ 为共享 $\boldsymbol{A}$ 的任播通信组成员集合(即基站集合);A_i 为 $\boldsymbol{G}(\boldsymbol{A})$ 中第 i 个组员;N 为 $\boldsymbol{G}$ 节点数目,M 为 $\boldsymbol{G}(\boldsymbol{A})$ 组员数目。

WSN 常用于战场、环境等监控场景,需要周期性将监测数据任播发往基站。其路由的重要设计目标是寻找源节点 S 至各个基站组员的最短距离(本文路径距离是指包括节点能耗等多种指标的综合开销)的任播路径。为均衡能耗,延长网络生存期,源节点 S 每次发送任务都随机选择基站集合 $\boldsymbol{G}(\boldsymbol{A})$ 中某一基站作为此次发送的目的地。

本文标记:有向链路 $(u,v)\in\boldsymbol{E}$ 的开销记为 C_{uv},P_i 为源节点 S 至基站组员 A_i 的最短(开销最小)路径,该路径开销记为 $C(P_i)$:

$$C(P_i)=\sum_{(u,v)\in P_i}C_{uv} \tag{1}$$

S 至各个基站组员的最短路径集合记为 $\boldsymbol{P}$;用变量 $\alpha_{(u,v)}(i)$ 标记有向链路 (u,v) 是否存在于 P_i 中。若存在,$\alpha_{(u,v)}(i)=1$;反之,$\alpha_{(u,v)}(i)=0$。

我们的目标是寻找到源节点至各个基站任播路径(共 M 条),并最小化优化任播通信的总开销,即最小化各任播路径的总开销 C。

$$\min:C=\sum_{i=1}^{M}C(P_i) \tag{2}$$

约束条件:

$$\sum_{i=1}^{M}\sum_{(u,x)\in P_i,P_i\in P}\alpha_{(u,x)}(i)=1\quad\forall u\in\boldsymbol{V}\text{ 且 }u\neq S$$

该约束条件是保证任播路径之间不能使用共同的中间节点。即节点 u 为前驱节点(且 u 不是源节点 S)的全部有向链路 (u,x) 在 $\boldsymbol{P}$ 中共 M 条最短任播路径中最多只能出现一次。其中,$x\in\boldsymbol{V}$。

以往节点不相交路由协议找到并采用一条任播路径(如 P_i)后,再找其他任播路径时将会避开使用 P_i 的所有中间节点。这些算法忽略了后向路径(Backward Path)策略能够再次使用这些节点的可能,因此找到的路径质量较差。

两条路径 P_i、P_j 之间有后向路径是指:在 P_i 中有路径片段(链路数不少于 1),该路径片段的反向属于路径 P_j。

我们标记 P_i 中这部分路径片段 $a\rightarrow\cdots\rightarrow b$ 为 $P_i^{\text{back}}(a,b)$。a 是该路径片段首节点,b 是末节点。可知,$P_j^{\text{back}}(b,a)$ 与 $P_i^{\text{back}}(a,b)$ 节点完全相同,但方向相反。在 P_i 中,$P_i^{\text{back}}(a,b)$ 之前的路径记作 P_i^{prior},之后的路径记作 P_i^{subs}。若链路 (m,n) 开销值 C_{mn} 为 z,则在后向路径中,反向链路 (n,m) 的开销值 C_{nm} 为 $-z$。

图 1 中,所标数值为相应链路的开销,没有给出数值的链路开销都为 1。显然,S 至 A_1 的最短路径 $P_1=\{S\rightarrow n_5\rightarrow n_6\rightarrow n_7\rightarrow A_1\}$,开销值 $C(P_1)=4$。再继续寻找 S 至 A_2 的最短路径 P_2 时,由于 P_1 的中间节点不能再被使用,此时能找到的最短路径为 $P_2=\{S\rightarrow n_8\rightarrow n_9\rightarrow n_{10}\rightarrow A_2\}$,$C(P_2)=12$,总开销 C 为 16。

但如果采用后向路径策略,即已经被一条路径占用的节点,如果是在后向路径上,仍然可能有机会再次使用。我们找到 S 至 A_2 路径 $P_2^{\text{New}}=\{S\rightarrow n_1\rightarrow n_2\rightarrow n_7\rightarrow n_6\rightarrow n_5\rightarrow n_9\rightarrow n_{10}\rightarrow A_2\}$,后者有路径片段 $n_7\rightarrow n_6\rightarrow n_5$ 与 P_1 中路径片段 $n_5\rightarrow n_6\rightarrow n_7$ 正好相反,即后向路径。根据前文规定,后向路径中链路 (n_7,n_6) 的开销值为 -1,与 (n_6,n_7) 的开销值正好相反(后向路径的链路在随后的路径调整操作中会被删去,采用负值可以用来抵消)。同理后向链路 (n_6,n_5) 的开销值为 -1。由此,P_2^{New} 的开销值 $C(P_2)=7$,此时,P_1 和 P_2^{New} 都是图中实际最短路径,总开销 C 为 11。显然,采用后向路径策略效果较好。

但由于 P_1 和 P_2^{New} 有相同节点,不符合节点不相交要求。需要进一步的路径调整操作,调整后结果如下:

$P_1^{\text{ReA}}=\{S\rightarrow n_1\rightarrow n_2\rightarrow n_7\rightarrow A_1\}$

$P_2^{\text{ReA}}=\{S\rightarrow n_5\rightarrow n_9\rightarrow n_{10}\rightarrow A_2\}$

此时,路径 P_1^{ReA} 的开销值 $C(P_1^{\text{ReA}})$ 为 5,不再是最短路径了。但是 P_1^{ReA} 和 P_2^{ReA} 的总开销为 11,保持不变,即最终优化目标(式(2)的总开销 C)是最佳的。

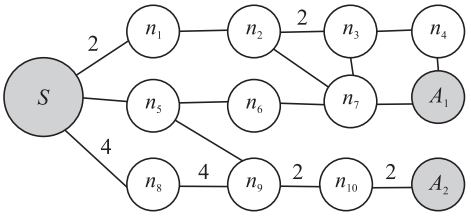


图 1　后向路径任播路由说明图

2 基于节点不相交的任播路由协议

基于上述解决策略,提出一种采用后向路径策略的基于节点不相交的 WSN 任播路由协议。

传统协议往往要求掌握网络全局信息,查询开销巨大。本文采用分布式查询路径方式,节点只需与相邻节点交换信息,无需掌握全局信息,路径查询开销较小。

本文查找任播路径包括三个阶段:查询路径 SAP(Search Anycast Path)、确认路径 CAP(Confirm Anycast Path)和路径调整 PReA(Path Readjustment)。

协议设置如下:每个节点 $u \in \boldsymbol{V}$,需要保存源节点 S 至 u 的当前最短路径 $P(u)$及其开销值 C_u。节点 u 还需要维护一张任播路径占用表。如果节点 u 被某条任播路径(如 P_i)所占用,表中记录该任播路径编号 i,以及该路径中节点 v 的前驱节点 u_i^{prep} 和后继节点 u_i^{succ};表中记录的数目记为 k_u。

2.1 查询路径 SAP

在 SAP 阶段,$\boldsymbol{G}$ 中各个节点 $u \in \boldsymbol{V}$ 需要不断向相邻节点通知 SAP 消息来寻找任播路径。节点 u 发送的 SAP 消息包括源节点 S 到 u 的最短路径 $P(u)$及其开销值 C_u。如果节点 u 已被某条任播路径占用(如 P_i),SAP 消息还需包括路径编号 i,以及该路径的后继节点 u_i^{succ}。

当有一个节点 $v \neq S$ 接收到来自相邻节点 u 的 SAP 消息$\{P(u), C_u, i, v_i^{\text{succ}}\}$,有三种可能情况需要讨论。

①$k_v=0$,即 v 没有被任何一个任播路径所占用。

A. 如果节点 u 已经被一条任播路径(如 P_i)所占用,并且其最短路径 $P(u)$上链路 $u_{P(u)}^{\text{prep}} \rightarrow u$ 不是 $\overleftarrow{P}_i$ 中的链路 $u_i^{\text{succ}} \rightarrow u$。其中,$\overleftarrow{P}_i$ 是 P_i 的反方向路径。要注意的是,由于 P_i 已确认,此处的最短路径 $P(u)$是第二次查询找到的最短路径。这种情况下,v 丢弃 u 的 SAP 消息。

B. 若上述条件不满足。计算 C_u+C_{uv}的值。如果 $C_u+C_{uv}<C_v$,这意味着新路径$(S \rightarrow \cdots \rightarrow u \rightarrow v)$短于节点 v 之前保存的最短路径 $P(v)$。由此,需更新并保存节点 v 的最短路径 $P(v)$,销值 C_v 被赋值为 C_u+C_{uv}。最短路径被更新后,v 还需向相邻节点通知 SAP 消息,消息包含以下信息$\{P(v), c_v\}$。

②$k_v=1$,即 v 已被某一个任播路径 P_i 所占用。

检查节点 v 的任播路径占用表。

A. 如果相邻节点 v 既不是该任播路径 P_i 上节点 v 的前驱节点 v_i^{prep},也不是后继节点 v_i^{succ}。即有向链路 $(u,v) \notin P_i$,其反向链路(v,u)也 $\notin P_i$。计算 C_u+C_{uv}值。如果该值比之前的旧值 C_v 小,更新节点 u 的最短路径 $P(v)$;开销值 C_v 被赋值为 C_u+C_{uv}。最短路径被更新后,$v(v \notin \boldsymbol{G}(\boldsymbol{A}))$还需向前驱节点 v_i^{prep} 通知 SAP 消息,消息包含以下信息$\{P(v), C_v, i, v_i^{\text{succ}}\}$;$v$ 无需向后继节点 v_i^{succ} 通知消息。

举例说明,在图 1 中,假设已确认 $P_1=\{S \rightarrow n_5 \rightarrow n_6 \rightarrow n_7 \rightarrow A_1\}$。由于 P_1 已确定,P_1 中各节点不能再被使用其他路径中,除非是应用于后向路径才有机会再次使用。这些节点的最短路径以及开销值需要重新计算。如 S 至节点 n_7 原先的最短路径为 $S \rightarrow n_5 \rightarrow n_6 \rightarrow n_7$,开销值为 $C_{n_7}=3$。该路径已被确认,如果需要将 n_7 再次应用到其他路径,只能用在后向路径中。当 n_7 收到邻居节点 n_5 的 SAP 消息,$(n_5,n_2) \notin P_i$,其反方向(n_2,n_5)也 $\notin P_i$。由此,至节点 n_7 的新的最短路径 $P(n_7)$ 为 $S \rightarrow n_1 \rightarrow n_2 \rightarrow n_7$,开销值 C_{n_7}的新值为 4。

B. 如果相邻节点 u 是该任播路径 P_i 上节点 v 的后继节点 v_i^{succ},即有向链路$(v,u) \in P_i$。C_v 被赋值为 C_u-C_{uv}。即如果我们的路径通过这个后向路径的这个中间节点,在后面的路径调整 PReA 中,该后向链路(v,u)也要被废弃,从而其开销要被扣除掉。我们更新并保存需更新节点 v 的最短路径 $P(v)$,开销值 C_v 被赋值为 C_u-C_{uv}。v 还需向相邻节点通知 SAP 消息,消息包含以下信息$\{P(v), C_v, i, v_i^{\text{succ}}\}$。

举例说明,在图 1 中,假设已确认 $P_1=\{S \rightarrow n_5 \rightarrow n_6 \rightarrow n_7 \rightarrow A_1\}$。当 n_6 收到邻居节点 n_7 的 SAP 消息,由于$(n_6,n_7) \in P_1$,因此,$C_{n_6}=C_{n_7}-C_{n_6n_7}=3$。同样,随后还可以求得,$C_{n_5}=2$。

③$k_v=2$,即 v 已被两条任播路径所占用。

此时,节点 v 丢弃所有 SAP 消息。即节点 v 不再参与其他任播路径的查询。

由此,SAP 过程伪代码如表 1 所示。

表 1 SAP 过程

输入:网络连通图 $\boldsymbol{G}(\boldsymbol{V},\boldsymbol{E})$;源节点 S;基站集合 $\boldsymbol{G}(\boldsymbol{A})$
WHILE $v \neq S$ 收到相邻节点 u 的 SAP 消息$\{P(u), C_u, i, v_i^{\text{succ}}\}$
CASE $k_v=0$
IF(u 是自由节点)or((若 u 被某一任播路径如 P_i 所占用 and $(u_{P(u)}^{\text{prep}}, u) \neq (u_i^{\text{succ}} \rightarrow u)$)
{IF $C_u+C_{uv}<C_v$
{更新 v 的最短路径 $P(v)$;
开销值 $C_v \leftarrow C_u+C_{uv}$;
v 向相邻节点通知 SAP 消息$\{P(v), C_v\}$;}

```
    }
  ELSE
      v 丢弃 u 的 SAP 消息;
CASE k_v=1(设 v 被 P_i 占用)
  IF u≠v_i^prep and u≠v_i^succ
  {IF C_u+C_uv<C_v
    {更新节点 v 的最短路径 P(v)
      开销值 C_v←C_u+C_uv
      v 向前驱节点 v_i^prep 通知 SAP 消息{P(v),C_v,i,
      v_i^succ}}
      }
    ELSE IF( u=v_i^succ)
    {C_v←C_u-C_uv
      更新节点 v 的最短路径 P(v)
      v 向相邻节点通知消息{P(v),C_v,i,v_i^succ}}
  CASE k_v=2
    v 丢弃 u 的 SAP 消息;
END WHILE
```

输出:源节点 S 至任一节点 $u \in \boldsymbol{V}$ 的最短路径 $P(u)$ 及其开销值 C_u

SAP 消息数分析如下:$\boldsymbol{G}$ 中每个节点需要通知其邻居 SAP 消息,如果最短路径有更新(如 $C_u+C_{uv}<C_v$),该邻居节点还需继续发送 SAP 消息。最理想情况下,SAP 消息数的复杂性为 $O(nN)$。其中 n 为 $\boldsymbol{G}$ 中节点的平均邻居节点数。最差情况下,SAP 消息会被通知到全网,消息数跟泛洪一样 $O(N^2)$。每个节点只有碰到其最短路径有更新时,才需往下通知。因此,SAP 消息数往往远低于泛洪。

2.2　确认路径 CAP

当基站集合 $\boldsymbol{G}(\boldsymbol{A})$ 中某一基站 A_i 接收到 SAP 消息,其等待足够长的时间来开始执行 CAP 流程。

A_i 根据最短路径 P_i 将 CAP 消息依次沿路通知每一个节点。

当某节点 $v \in \boldsymbol{V}$ 收到来自 A_i 的 P_i 路径传来的 CAP 消息,根据节点 v 的情况,分别操作如下。

①若 $k_v \leqslant 1$。即 v 被不超过 1 条任播路径所占用。

v 将该 CAP 消息继续沿着 P_i 路径向上游节点发送。

②若 $k_v = 2$。即节点 v 已被 2 条任播路径所占用。

此时,节点 v 已经不能再被其他任播路径所占用。由此,该任播路径作废。节点 v 将该路径作废消息沿着 P_i 路径向下游节点发送直至基站 A_i。当中间节点(如 u)收到作废消息后,$k_u \leftarrow k_u-1$,并且修改任播路径占用表;当基站 A_i 收到该作废消息后,需要等待新的任播路径并重新 CAP 流程。

当源节点 S 收到所有 M 条任播路径的 CAP 消息后,确认其中最短的一条任播路径(如 P_i),并沿 P_i 路径通知所有中间节点(如 u)的 C_u 值为+∞,并将路径 P_i 记录到 u 的任播路径占用表中,$k_u \leftarrow k_u+1$。剩下的 $M-1$ 个基站需要新一轮的 SAP 和 CAP 操作,直至所有 M 条任播路径被确认。

2.3　路径调整 PReA

源节点 S 确认所有 M 条任播路径后,需要进行路径调整(Path Readjustment,PReA)操作。

PReA 操作伪代码如表 2 所示。

表 2　PReA 过程

```
输入:任播路径集合 P;源节点 S;基站集合 G(A)
WHILE ∃(P_i∈P,P_j∈P) 存在后向路径现象
  P_i 被调整为 P_j^prior→ a→ P_i^subs
  P_j 被调整为 P_i^prior→ b→ P_i^subs
    IF ∃(v∈P_i^back(a,b) and v≠a and v≠b)
      {S 通知取消对节点 v 的占用;
        节点 v 的 k_v←k_v-2;
        修改节点 v 的任播路径占用表;}
END WHILE
输出:新的任播路径集合 P^new
```

3　实验和分析

我们采用节点不相交算法(Node Disjoint Algorithm,NDH)和节点相交算法(Node Joint Algorithm,NJH)作为对照算法。NDH 和 NJH 采用最短最短路径优先算法(Shortest-Shortest Path First),即每次确认一条至基站的全局最短任播路径;确认后,NDH 从 $\boldsymbol{G}$ 中删去该路径中间节点,而 NJH 无需删除;然后,再继续寻找至其他基站的最短路径,直至找到源节点至所有基站的 M 条任播路径;算法采用全泛洪机制来查找路径。

设置网络为正方形区域,源节点 S 设置在区域中间,随机均匀设置节点,$\boldsymbol{G}$ 中节点数目 $\boldsymbol{N}=80$,链路开销值在数值 1~3 之间随机取值,$M=3$。采用完全相同的任务,按照相同参数($\lambda=1$)的泊松分布产生监测数据分组。S 采用随机权重选择法将数据发往某个基站。S 至基站 A_i 的 P_i 路径的权重 W_i 计算如下:

$$W_i = \frac{[1/C(P_i)]^r}{\sum_{j=1}^{M}[1/C(P_j)]^r} \tag{3}$$

式中:r 为调节参数,本文设置为 1。

以上述算法为对照算法，讨论本文协议(Backward Path Algorithm，BPA)的网络生存期性能。观察各路由协议的网络生存期(到第一个节点失效的时间，单位为 s)。结果如图 2 所示。

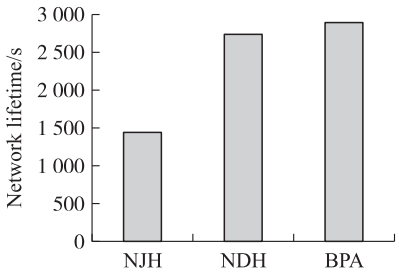


图 2　各路由协议网络生存期性能

由图 2 可知，NJH 性能最差，这是由于交叉节点承担多条路径任务，耗电过快；而相比较 NDH，BPA 查询到的路径质量较高，生存期性能较好。

调整 $\boldsymbol{G}$ 中节点数目 N(20～80)并相应调整 WSN 区域面积(保持节点密度 ρ 不变)，观察对照算法与 BPA 的查询路由开销(消息数)，结果如图 3 所示。

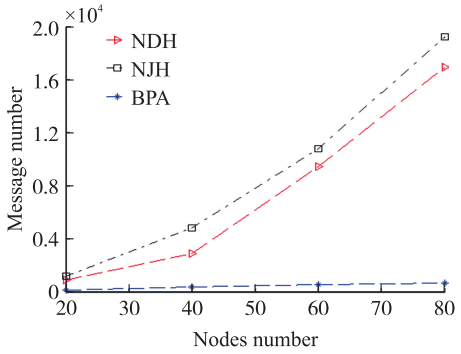


图 3　节点数目与查询消息数

由于 $M=3$，查询路径需要 3 次。由图 3 可知，BPA 的消息数(SAP 消息数)少于 NDH 和 NJH。而且，随着网络规模增大，NDH 和 NJH 消息数增长明显。这是由于全泛洪机制，消息数为 $O(N^2)$。N 的增加，导致消息数急剧增加。而 BPA 中的 SAP 消息是否需要继续往下传递，取决于该节点的开销值是否有更新，如无更新就不再继续传递。因此，BPA 查询路由开销性能较优。

在上述网络设置条件下，观察 BPA 和 NDH 的式 2 的总开销值 C。结果如图 4 所示。

由图 4 可知，BPA 的总开销优于 NDH。这是相比较 NDH，在 BPA 中，被占用的节点仍然有可能被第二次使用(后向路径策略)，从而有可能找到质量更高的路径，优化目标值(总开销值 C)更好。

设置网络节点数 $\boldsymbol{N}=80$，调节基站数目 M(3-6)，观察 BPA 和 NDH 的式(2)的总开销值 C。结果如图 5 所示。

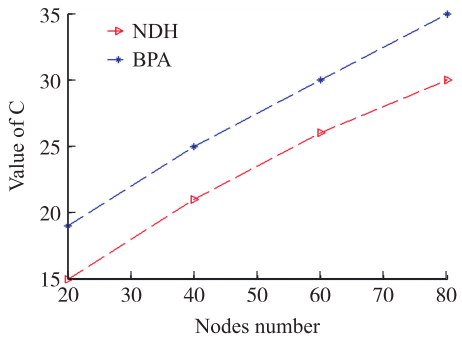


图 4　节点数目与总开销值 $\boldsymbol{C}$

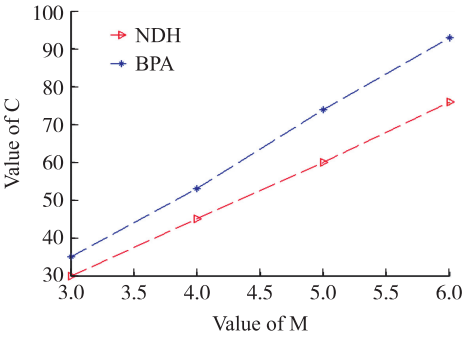


图 5　基站数目 $\boldsymbol{M}$ 与总开销值 $\boldsymbol{C}$

随着 M 值增加，意味着需要 M 次选路，而每一次选路都会导致下一次选路可用节点变少。而 BPA 对已被占用的节点能够有机会再次使用，在选路环节上有着较多的选择面。因此，在 M 值较高的情况下 BPA 仍然性能优于 NDH。

4　结束语

针对以往路由协议忽略任播路径中有节点相交问题，提出一种基于节点不相交的 WSN 任播路由协议。该协议采用节点不相交策略，提高了网络生存期；采用分布式路由查询，路由查询开销较少；采用后向路径策略，得到的任播路径质量较优。实验表明，相比较于基于节点相交 WSN 任播路由协议，该协议网络生存期较优；相比较传统基于节点不相交的 WSN 任播路由协议，该协议路由查询开销少，任播路径质量较优。

参考文献：

[1] Li C，Cheng Y N，Men H，et al. Performance Analysis of Root Anycast Nodes Based on Active Measurement[J]. Electronics，2022，11(8)：1194.

[2] Miyasaka T，Hei Y，Kitahara T. NetworkAPI：An In-Band Signalling Application-Aware Traffic Engineering Using SRv6 and IP Anycast[J]. IEICE Transactions on Information and Systems，2021，E104-D(5)：617-627.

[3] Souza G C,Giacomin J C,Heimfarth T. An Asynchronous Anycast Protocol Resilient to Changes in Communication Channel[J]. Wireless Personal Communications,2021,120(4):3243-3263.

[4] Heimfarth T, Giacomin J C, Schneider B D. An Asynchronous Anycast Cross-layer Protocol for WSN Suited to Noisy Environments[J]. Wireless Personal Communications,2021,120(2):1079-1107.

[5] Oz G. A Flooding Mechanism of Stateless Anycast Routing in Mobile Wireless Ad Hoc Networks[J]. Ad Hoc & Sensor Wireless Networks,2023,55(1-2):123-149.

[6] Jari A,Avokh A. PSO-Based Sink Placement and Load-Balanced Anycast Routing in Multi-Sink WSNs Considering Compressive Sensing Theory[J]. Engineering Applications of Artificial Intelligence, 2021, 100(99): 104164.

[7] Achour M. Minimizing End-To-End Delay with Anycast and Controlled Mobility in Sensor Networks[J]. Wireless Personal Communications,2021,118(4):2621-2630.

[8] Ye Q,Wang Y F,Tang Y L,et al. A Node-disjoint Trust-based Multipath Routing Mechanism for Multimedia Sensor Networks[J]. Journal of Internet Technology, 2021,22(1):219-228.

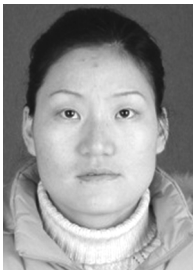
[9] Rachkidy N, Guitton A, Misson M. Routing Protocol for Anycast Communications in a Wireless Sensor Network [C]//9th International IFIP TC 6 Networking Conference, Chennai,India,Springer,2010:291-302.

[10] MalekiTabar M, Rahmani A M. A Delay-Constrained Node-Disjoint Multipath Routing in Software -Defined Vehicular Networks[J]. Peer-To-Peer Networking and Applications,2022,15(3):1452-1472.

[11] Velusamy B,Karunanithy K,Sauveron D,et al. Multi-Objective Function-Based Node-Disjoint Multipath Routing for Mobile Ad Hoc Networks[J]. Electronics, 2021, 10(15):1781.

[12] Xu Y Z,Po H F,Yeung C H,et al. Scalable Node-Disjoint and Edge-Disjoint Multiwavelength Routing[J]. Physical Review E,2022,105(4):044316.

[13] Ye Q,Wang Y F,Tang Y L,et al. A Node-disjoint Trust-based Multipath Routing Mechanism for Multimedia Sensor Networks[J]. Journal of Internet Technology, 2021,22(1):219-228.



顾云丽(1978—),女,江苏连云港人,博士,讲师,主要研究方向为无线传感器网络,guyunli@nuist.edu.cn;

徐　昕(1975—),男,江苏南京人,博士,副教授,主要研究方向为物联网技术,xuxin@nuist.edu.cn。