

doi: 10.7690/bgzdh.2022.09.010

## 基于 UWB 测距辅助的车载自组网链路优化

王天和<sup>1</sup>, 刘钊良<sup>1</sup>, 任春华<sup>1</sup>, 卢飞平<sup>2</sup>, 李鑫锴<sup>1</sup>, 何柯锦<sup>1</sup>, 任禹衡<sup>1</sup>

(1. 重庆大学光电工程学院光电传感与测控研究室, 重庆 400030; 2. 上海电控研究所, 上海 200082)

**摘要:** 针对车载自组网(vehicular ad hoc network, VANET)拓扑结构速度过慢、处理数据过多, 导致信道容易阻塞、实时性差的问题, 提出一种基于地理信息和超宽带的小型自组网链路优化算法。利用 UWB 测距信息和地理信息进行系统设计和理论分析, 并搭建基于 Matlab 的仿真平台。仿真结果表明: 基于该优化算法的链路模型在所有可能的链路模型中位于次优或者最优这一档, 相对同样物理拓扑结构下其他可能模型, 最大时延优化可达 25%。

**关键词:** 车载自组网; 链路优化; 地理信息; 超宽带技术

**中图分类号:** TP393 **文献标志码:** A

### Link Optimization of Vehicular Ad Hoc Network Based on UWB Ranging

Wang Tianhe<sup>1</sup>, Liu Zhaoliang<sup>1</sup>, Ren Chunhua<sup>1</sup>, Lu Feiping<sup>2</sup>, Li Xinkai<sup>1</sup>, He Kejin<sup>1</sup>, Ren Yuheng<sup>1</sup>

(1. Lab of Photoelectric Sensing, Measurement and Control, College of Optoelectronic

Engineering, Chongqing University, Chongqing 400030, China;

2. Shanghai Electric Control Research Institute, Shanghai 200082, China)

**Abstract:** Aiming at the problems of vehicular ad hoc network (VANET) topology, such as slow speed, too much data processing, channel blocking and poor real-time performance, a small ad hoc network link optimization algorithm based on geographic information and ultra-wideband (UWB) is proposed. The system design and theoretical analysis are carried out by using UWB ranging information and geographic information, and the simulation platform based on Matlab is built. The simulation results show that the link model based on the optimization algorithm is in the sub-optimal or optimal level among all possible link models, and the maximum delay optimization can reach 25% compared with other possible models under the same physical topology.

**Keywords:** vehicular ad hoc network; link optimization; geographic information; ultra-wideband technology

## 0 引言

车载自组网(VANET)是指在户外交通环境中, 车辆与车辆之间、车辆与固定节点之间以及车辆与行人之间建立的移动网络。

车载自组网内的成员可以获取当前路段其他车辆的方位、速度等信息, 给驾驶人带来超视距的感知能力, 能够有效地解决拥堵问题、降低交通事故发生率。

现今车载自组网存在的问题主要有<sup>[1]</sup>: 1) 车载自组网拓扑结构变化迅速, 车辆识别拓扑结构速度过慢; 2) 节点处理数据过多导致信道容易阻塞、实时性差。

针对信道容易阻塞、实时性差的问题, 文献[2]提出一种基于鱼群优化的车载自组织网络路由算法, 显著改善 VANET 中消息到达的平均时延和消息到达率。文献[3]基于车载自组网通信终端和运动信息的容忍时延网络分簇路由算法, 与传统分簇路

由(cluster-based routing protocol, CBRP)算法和动态源路由(dynamic source routing, DSR)协议相比, 增加了消息投递率, 减少了网络开销、簇结构重组次数。文献[4]针对移动自组网提出了一种新的按需带状路由协议, 解决了多个移动自组网路由协议的不足之处, 降低了时延。文献[5]利用网络编码减小数据包的传输次数, 从而减小了数据流端到端的时延, 依据路径的时延分配数据量降低整体的时延。

针对拓扑网络变化问题, 文献[6]仿真多种不同的切换策略理论, 提出垂直切换算法来管理 VANET 异构网络。文献[7]提出一种面向动态拓扑的深度强化学习智能路由技术, 实现了路由智能体对不同网络拓扑的泛化, 具有比传统的最短路由算法更高的网络吞吐量。文献[8]提出一种基于网络连通性和节点删除法相结合的关键节点识别方法, 具有比节点收缩法和半局部中心度法更好的识别效果, 比中介性方法时间复杂度低, 能够适应网络拓扑动态性。

针对上述 2 个问题, 笔者提出一种基于地理信

收稿日期: 2022-05-22; 修回日期: 2022-06-20

作者简介: 王天和(1999—), 男, 福建人, 硕士, 从事陀螺惯性导航系统研究。E-mail: 610372370@qq.com。

息和超宽带测距的小型自组网链路优化算法。该算法能够充分利用 UWB 测距信息和地理信息，完成对车群拓扑结构的实时性更新，并规划相应拓扑结构下的最优通信链路。

## 1 组网节点信息的获取

为识别车载自组网的拓扑结构，需获取各自节点的位置信息。笔者采用基于 TDOA 的 Fang 协同定位算法来获取各自节点的位置信息。

如图 1 所示，TDOA<sup>[9-10]</sup>算法依靠到达的时间差来进行定位。未知节点测得 UWB 信号到达多个锚节点的时间差与光速相乘来计算达到不同锚节点的距离差，而后以每 2 个锚节点作为焦点，以它们到未知节点的距离差作为长轴作一条双曲线，未知节点即在双曲线上。从多个锚节点中取出多组双曲线，交点即为未知节点的位置。TDOA 的优点在于：获取的测量值是时间差，未知节点与锚节点的时钟不同步被相互抵消，只需要保证锚节点间时钟同步即可。

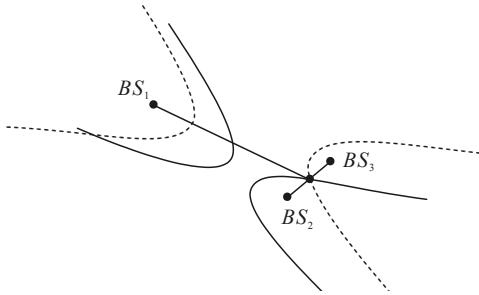


图 1 TDOA 算法

TDOA 算法计算过程：3 个已知的锚节点分别为  $S_1(x_1, y_1)$ 、 $S_2(x_2, y_2)$ 、 $S_3(x_3, y_3)$ ，未知节点为  $T(x, y)$ ，未知节点到锚节点  $i$  和锚节点  $j$  的距离差为  $\Delta R_{i,j}$ ，根据两点间距离公式可以写出方程组：

$$\begin{aligned} \Delta R_{2,1} &= \sqrt{(x_2 - x)^2 + (y_2 - y)^2} - \sqrt{(x_1 - x)^2 + (y_1 - y)^2}; \\ \Delta R_{3,1} &= \sqrt{(x_3 - x)^2 + (y_3 - y)^2} - \sqrt{(x_1 - x)^2 + (y_1 - y)^2}. \end{aligned} \quad (1)$$

基于 TDOA 的 Fang 协同定位算法实现过程如下，TDOA 算法中给出了已知节点  $S_1(x_1, y_1)$ 、 $S_2(x_2, y_2)$ 、 $S_3(x_3, y_3)$  与未知节点  $T(x, y)$  之间的关系：

$$R_i = \sqrt{(x_i - x)^2 + (y_i - y)^2}, \quad i=1, 2, 3. \quad (2)$$

将式(2)左右同时平方可得：

$$\begin{aligned} R_i^2 &= M_i - (2x_i x + 2y_i y) + x^2 + y^2; \\ M_i &= x_i^2 + y_i^2. \end{aligned} \quad (3)$$

假设第  $i$  个锚节点与未知节点的距离和第 1 个锚节点与未知节点距离的差值为  $\Delta R_{i,1}$ ，化简可得：

$$\begin{aligned} 2R_1 \Delta R_{i,1} + \Delta R_{i,1}^2 &= M_i - M_1 - 2x(x_i - x_1) - \\ & 2y(y_i - y_1), \quad i=2, 3. \end{aligned} \quad (4)$$

以第 1 个锚节点为坐标原点，第 1、2 个锚节点所在直线为  $x$  轴，此时锚节点坐标为  $S_1(0, 0)$ 、 $S_2(x_2, 0)$ 、 $S_3(x_3, y_3)$ 。化简式(4)可得：

$$\left. \begin{aligned} -2R_1 \Delta R_{2,1} &= \Delta R_{2,1}^2 - x_2^2 + 2xx_2 \\ -2R_1 \Delta R_{3,1} &= \Delta R_{3,1}^2 - (x_3^2 + y_3^2) + 2(xx_3 + yy_3) \end{aligned} \right\}. \quad (5)$$

在式(5)中消去  $R_1$  后，令

$$\left. \begin{aligned} V &= (x_3^2 + y_3^2 - \Delta R_{3,1}^2 + \Delta R_{3,1} \Delta R_{2,1} - \\ & (\Delta R_{3,1} / \Delta R_{2,1}) * x_2^2) / 2y_3 \\ U &= (\Delta R_{3,1} x_2 - \Delta R_{2,1} x_3) / (\Delta R_{2,1} y_3) \end{aligned} \right\}. \quad (6)$$

可得：

$$y = Ux + V. \quad (7)$$

根据式(5)–(7)可得：

$$Ax^2 + Bx + C = 0;$$

$$A = 4\Delta R_{2,1}^2 (1 + U^2) - 4x_2^2;$$

$$B = 8UV\Delta R_{2,1}^2 - 4\Delta R_{2,1}^2 x_2 + 4x_2^3;$$

$$C = 4V^2\Delta R_{2,1}^2 - (x_2^2 - \Delta R_{2,1}^2)^2.$$

求解方程组即可得到未知节点的坐标。相较于 Chan 算法和 Taylor 算法，Fang 算法计算量小，可以减小节点计算压力。

## 2 拓扑识别算法

在获取各自节点的位置信息后，对于整个识别算法，已知只有各节点的位置及它们的间距，而关键物理拓扑结构的识别，只有利用已知不同类型拓扑结构的数学模型去逐个验证计算。笔者通过最小二乘法拟合抛物线，得到简单线性回归模型参数的估计量，再通过验证绝对残差之和来确认拓扑类型。

假设抛物线的方程为  $y = Ax^2 + Bx + C$ ，偏差综合为：

$$E = \sum_{k=1}^N (Ax_k^2 + Bx_k + C - y_k)^2. \quad (8)$$

令偏导数  $\partial E / \partial A$ 、 $\partial E / \partial B$ 、 $\partial E / \partial C$  都为 0，化简可得：

$$\left. \begin{aligned} \sum_{k=1}^N y_k x_k^2 &= \left( \sum_{k=1}^N x_k^4 \right) A + \left( \sum_{k=1}^N x_k^3 \right) B + \left( \sum_{k=1}^N x_k^2 \right) C \\ \sum_{k=1}^N y_k x_k &= \left( \sum_{k=1}^N x_k^3 \right) A + \left( \sum_{k=1}^N x_k^2 \right) B + \left( \sum_{k=1}^N x_k \right) C \\ \sum_{k=1}^N y_k &= \left( \sum_{k=1}^N x_k^2 \right) A + \left( \sum_{k=1}^N x_k \right) B + NC \end{aligned} \right\}. \quad (9)$$

此方案仅适用于标准形式的抛物线，对于非标

准形式的抛物线(即在坐标系中有一定的旋转,不能以标准方程代表的抛物线)不适用。

对此问题,笔者解决的方案为:不断将节点坐标进行旋转,每次固定旋转 1° 并进行拟合且记录拟合的绝对残差和,通过 360 个绝对残差和数据来判断物理拓扑结构是否为抛物线形状。

假设有 2 个直角坐标系  $XOY$  和  $X'O'Y'$ , 它们之间的夹角为  $\theta$ , 规定顺时针夹角为正值;反之,逆时针夹角为负值,点  $O'$  到  $O$  的距离为  $(x_0, y_0)$ 。假设有一点  $P$ , 在 2 个坐标系中的位置坐标分别为  $(x, y)$  和  $(x', y')$ , 可以得到从  $XOY$  坐标系转到  $X'O'Y'$  坐标系的公式:

$$\begin{cases} x = (x' - x_0)\cos(\theta) + (y' - y_0)\sin(\theta) \\ y = -(x' - x_0)\sin(\theta) + (y' - y_0)\cos(\theta) \end{cases} \quad (10)$$

可以把 2 个齐次式子简化为旋转矩阵:

$$X = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} X'' \quad (11)$$

其中  $X'' = [x' - x_0, y' - y_0]^T$ 。

依据式(11)的变换, 首先把节点坐标变换到坐标原点附近, 再每次偏转 1°, 不断循环 360 次, 即可满足对非标准抛物线的旋转。

### 3 链路的布置和优化

笔者采用节点分簇的算法优化传输链路规划。经典的分簇算法有很多, 但并没有一个非常简单高效并普适的方法, 大多数算法都是在某一方面优越, 但是依据应用的场景, 可以将几个算法结合使用。比较经典的算法有最小节点 ID 分簇算法、最高节点度分簇算法、最低节点移动性分簇算法和低能量自适应分簇算法。本文中的需求:

- 1) 簇的结构稳定, 不会频繁更改;
- 2) 2 个簇中的节点靠近簇头且不会在簇间游离。

因此, 笔者采用了最小平均距离和算法与最低节点移动性分簇算法相结合, 将最低节点移动性分簇算法和最小平均距离和算法按照平均距离和: 节点移动性=6:4 的加权系数组合在一起, 满足需求。

最低节点移动性计算公式如下<sup>[11]</sup>:

$$M(yx) = 10\log\left(R_x P_{rx \rightarrow y}^{new} / R_x P_{rx \rightarrow y}^{old}\right) \quad (12)$$

式中:  $M(yx)$  表示节点  $y$  相对于节点  $x$  的移动性;  $R_x P_{rx \rightarrow y}^{new}$  和  $R_x P_{rx \rightarrow y}^{old}$  表示当前和上一次测量的节点  $y$  收到来自节点  $x$  的接收功率。

自由空间传播模型的无线信号功率计算公

式为:

$$P_r(d) = (P_t G_t G_r \gamma^2) / ((4\pi)^2 d^2 L) \quad (13)$$

式中:  $P_r(d)$  为接收信号的功率;  $P_t$  为发射功率;  $G_t$ 、 $G_r$  分别为发射和接收增益;  $\gamma$  为波长;  $d$  为发射端与接收端的距离;  $L$  为与传播无关的损耗。

假设 UWB 发射端功率、发射接收增益都一致, 那么可得:

$$R_x P_{rx \rightarrow y}^{new} / R_x P_{rx \rightarrow y}^{old} = d_{old}^2 / d_{new}^2 \quad (14)$$

将式(14)代入式(12)可得:

$$M(yx) = 10\log(d_{old}^2 / d_{new}^2) = 20\log(d_{old} / d_{new}) \quad (15)$$

计算所有可能性中, 簇头  $y$  与其簇内节点的平均相对移动性为:

$$M(y) = \sum_{i=1}^k M(yx_i) / k \quad (16)$$

式中:  $k$  为簇头  $y$  的度数(簇内邻居节点个数);  $x_i$  为第  $i$  个邻居节点。

同理可得最小平均距离和公式:

$$D(y) = \sum_{i=1}^k D(yx_i) / k \quad (17)$$

后续仿真实验中对 6 辆节点组成的自组网进行仿真分析, 以 6 个节点为例, 把 6 个点分为 2 簇, 排列组合可以知道共有 15 种可能性, 将这 15 种可能性都利用算法将其 2 种特征值算出对比, 即可知道簇头的最佳选择, 最后其他节点分到离得最近的簇头所在簇。简单的计算 15 种情况下 2 个特征值的大小, 如按 6:4 加权作和进行对比就能选出比较满意的 2 个簇头, 其他节点与离得最近的簇头组成 1 簇。

## 4 Matlab 仿真

### 4.1 拓扑结构识别及通信链路规划算法仿真

仿真步骤设计如下:

- 1) 随机确定小车位置;
- 2) 车群通过拓扑结构识别算法确定当前最优拓扑结构;
- 3) 车群通过通信链路规划确定当前最佳通信链路;
- 4) 设定随机故障点, 车群通过短路补偿算法确定当前最佳通信链路。

仿真结果如图 2、3 所示, 图 2、3 中节点为小车位置, 节点间的实线连线为运行算法后所得的通信链路。图 3 中点线表示故障线路, 虚线表示故障后重新规划的路线。

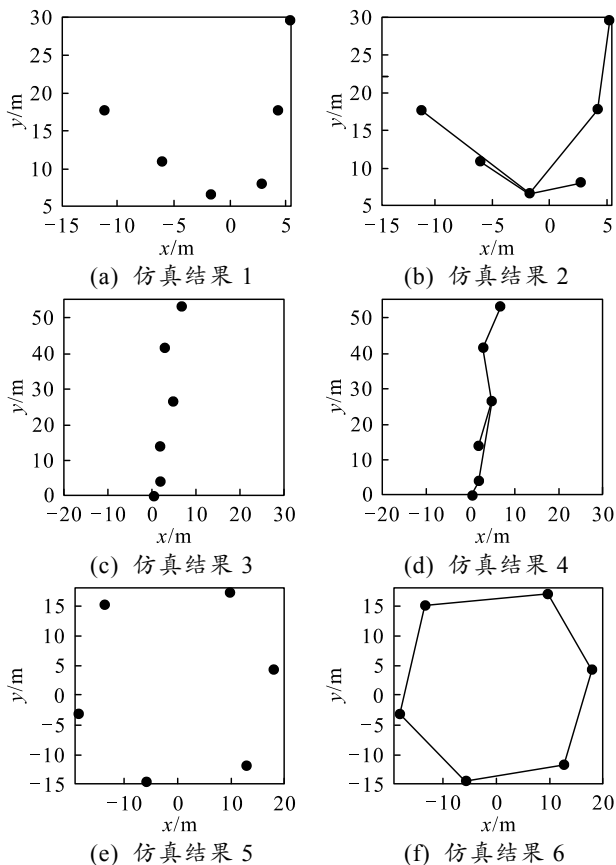


图 2 规划最佳通信链路

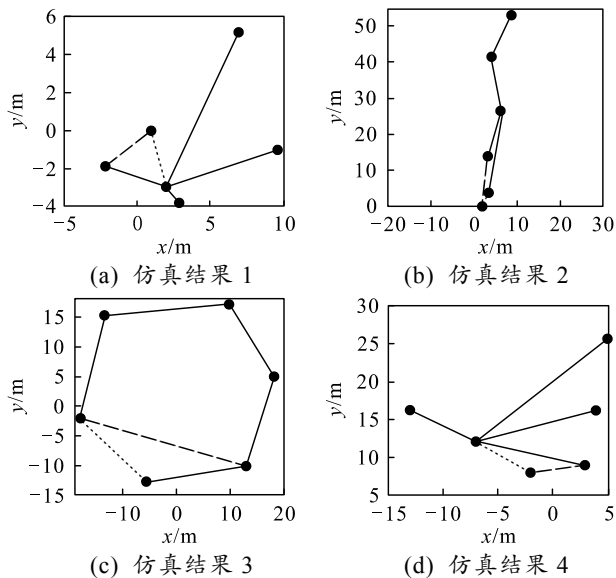


图 3 断路补偿

通过仿真可得：算法在理论上能够根据 UWB 测距信息和地理信息，寻找车群当前时刻的拓扑结构、规划通信链路。在节点位置发生变化时，实时更新距离信息，重新寻找拓扑结构，规划通信链路。

### 4.2 链路规划算法优化

笔者选用的分簇算法为最小平均距离和算法与

最低节点移动性分簇算法以一定加权结合，判断算法性能的指标根据上述 2 种方法各自的指标按相同加权比例计算而得。计算所得的指标值越低，代表链路通信的时延越低，链路通信的实时性能更好。

以图 2 中的(b)链路结构为例，在其基础上改变链路结构，如图 4 所示的链路安排为其他可能的通信链路。计算由笔者所提算法所得的通信链路及其他可能通信链路的性能指标，判断算法性能。

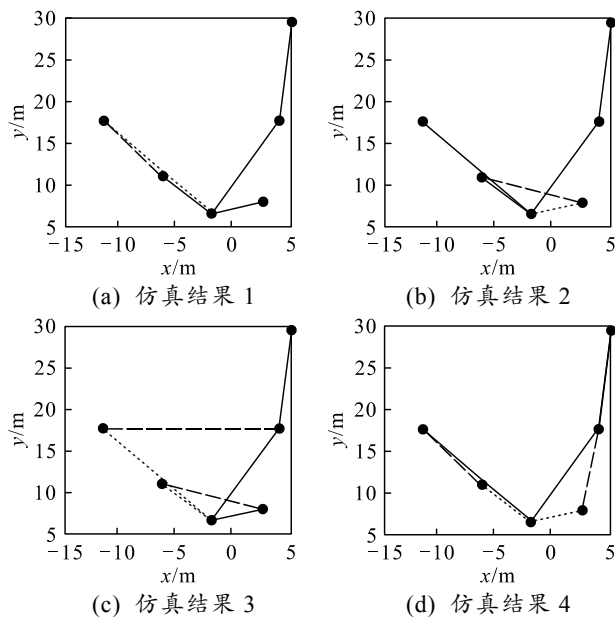


图 4 链路规划对比

上图中：点线表示原有路线，虚线表示修改后新增的路线。

取其中的(a)、(c)、(d)为例。经计算，所得结果如表 1 所示，其中：标准型为本算法运行所得链路结构；(a)、(c)、(d)为图 4 所示的其他链路结构； $d$  为平均距离； $n$  为平均中间节点； $P(2:8)$  为当节点间传输距离和中间节点数按 2:8 加权组合时的算法性能指标； $P(3:7)$  为当节点间传输距离和中间节点数按 3:7 加权组合时的算法性能指标。由表 1 可得在加权比为 2:8 的情况下，笔者所提算法所得链路为最优链路。在加权比为 3:7 的情况下，笔者所提算法所得链路为次优链路，相较于其他类型的通信链路优化最高可达 25.52%。

表 1 链路规划对比计算结果

参数	标准型	(a)	(b)	(d)
$d$	21.024 9	20.148 9	23.865 6	28.084 9
$n$	0.866 7	1.200 0	1.133 3	1.200 0
$P(2:8)$	4.898 3	4.989 8	5.679 8	6.577 0
$P(3:7)$	6.914 1	6.884 7	7.953 0	9.265 5