

基于电子海图的水面无人艇全局路径规划研究

庄佳园 万 磊 廖煜雷 孙寒冰

(哈尔滨工程大学水下智能机器人技术国防科技重点实验室 哈尔滨 150001)

摘 要 为解决水面无人艇全局路径规划问题,提出一种基于电子海图的距离寻优 Dijkstra 算法。该算法使用动态网格模型,克服了传统 Dijkstra 算法占用内存大的问题,可以减少规划时间,提高规划精度。仿真结果表明,采用的环境模型表示方法以及路径规划算法可以生成安全、合理的航线。

关键词 水面无人艇,电子海图,全局路径规划,Dijkstra 算法

中图法分类号 TP24 文献标识码 A

Global Path Planning of Unmanned Surface Vehicle Based on Electronic Chart

ZHUANG Jia-yuan WAN Lei LIAO Yu-lei SUN Han-bing

(National Key Laboratory of Science and Technology on Autonomous Underwater Vehicle, Harbin Engineering University, Harbin 150001, China)

Abstract To solve the problem of global path planning of USV(Unmanned Surface Vehicle), a search of shortcut Dijkstra algorithm based on electronic chart was presented. The algorithm uses a dynamic grid model, overcomes the occupy memory problem of traditional Dijkstra algorithm, reduces planning time and improves planning precision. Simulation results show that the environment model and path planning algorithm can generate safety and reasonable routes.

Keywords Unmanned surface vessel, Electronic chart, Global path planning, Dijkstra algorithm

1 引言

水面无人艇,简称无人艇或 USV(Unmanned Surface Vehicle),它的研发晚于陆地机器人、无人机和水下机器人,目前已有多种无人艇应用于军事、科研等领域。美国、以色列等国的研究已经取得了一定成果,其中以美国的“SPARTAN”、以色列的“Protector”为代表^[1]。文献[2,3]总结了无人艇的研究历史、现状以及发展趋势。水面无人艇是一个复杂的系统,它集船舶设计、智能控制、人工智能、信息处理等专业技术为一体,其研究内容涉及自动驾驶、自主避障、规划与导航、模式识别等多方面^[4]。

水面无人艇的路径规划方法可分为两类:一种是基于环境完全信息的全局路径规划,另一种是基于传感器信息的局部路径规划。全局路径规划按环境模型的表示方式可分为两种比较典型的方法,即构型空间法^[5]和自由空间法^[6]。构型空间法的基本思想是将无人艇简化为一点,同时根据无人艇的形状和尺寸将障碍物进行相应的膨胀处理,其中研究比较成熟的是可视图法。自由空间法的基本思想是采用预先定义的基本形状构造自由空间,并将自由空间表示为连通图,然后通过对图的搜索来规划路径。Bijlsma 等则将随机动态规划方法运用在随机动态网络中,找出最短航线,完成了全局路径规划^[7];饶森应用激活值和遗传算法,利用分层思想完成了全

局路径规划^[8]。

水面无人艇的全局路径规划能力体现出水面无人艇的智能水平。海洋的探索 and 开发以及军事上的应用,对水面无人艇提出了更高的要求。水面无人艇是一种自主式海洋运载器,其自主能力的真正含义是具有与外部环境进行交互的能力。这种交互的一个重要方面就是具有全局路径规划以及突发事件下的全局重规划和躲避障碍的能力。能够在复杂的海洋环境下完成指定的使命,也是水面无人艇智能行为的重要体现。

电子海图可以提供完备的海图信息,如果能够充分运用这些信息自动获取水面无人艇到目的地的所有可航区域和不可航区域,则将会为水面无人艇设计计划航线提供重要参考。基于这一思路,本文在完成对电子海图数据读取、建立环境模型的基础上,提出了一种基于距离寻优 Dijkstra 算法的水面无人艇全局路径规划方法。

2 环境模型的建立

水面无人艇在航行过程中,可以通过雷达、摄像机等传感器获得与障碍物的相对位置关系信息。这些信息只能描述局部环境,无法提供全局环境信息。为了解决这些问题,需要根据电子海图来获取全局环境信息。而水面无人艇无法利用电子海图进行路径规划,需要将电子海图转换成可以直接利用

到稿日期:2010-10-23 返修日期:2011-01-26 本文受国家自然科学基金(50579007),水下智能机器人技术国防科技重点实验室研究基金(HEUFT08017)资助。

庄佳园(1985—),男,博士生,主要研究方向为水面无人艇的运动规划,E-mail:zhuangjiayuan@hrbeu.edu.cn;万磊(1964—),男,研究员,主要研究方向为水面无人艇系统设计与运动控制;廖煜雷(1985—),男,博士生,主要研究方向为水面无人艇的智能控制与运动仿真;孙寒冰(1985—),女,博士生,主要研究方向为水面无人艇的操纵性。

的环境模型。创建环境模型必须解决环境模型的表示、存储等问题。

2.1 不可航区域的获取

电子海图文件是 Shapefile 文件格式,每个 Shapefile 文件由 3 种文件组成:主文件(*.shp)包含几何图形,索引文件(*.shx)包含数据的索引,数据库文件(*.dbf)包含图形的属性^[9]。水面无人艇关心的不可航区域主要包括大陆、岛、半岛、暗礁、沉船以及其他障碍物等。由相应图层索引文件(*.shx)可读取对应面状目标在主文件(*.shp)中的存储位置(位移量、记录长度),根据这个索引值可以在主文件(*.shp)中读取该面状目标的具体数据,其中,每个面状目标由一个或多个闭合子环组成。

大连地区不可航区域如图 1 所示。

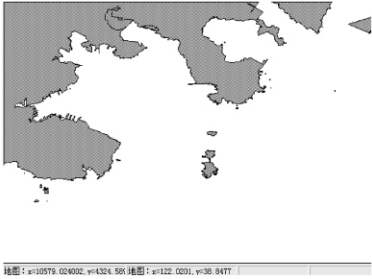


图 1 不可航区域显示

2.2 环境模型的表示和存储

环境建模方法一般分为栅格法、几何法和拓扑法 3 类。根据空间数据的几何特点又可分为图像栅格和图形数据。图像栅格是用像素来描述空间对象,不同的像素存储结构及空间单元对应不同的栅格结构,像素值表示空间对象的特征。图形数据用点、线和面等地理元素描述空间对象,这些元素可以表示简单和复杂的对象以及空间对象的拓扑关系,因此适合基于图搜索的路径规划。因为基于 Dijkstra 算法的全局路径规划方法是一种基于图搜索的路径规划算法,所以本文采用图形数据来表示环境模型。

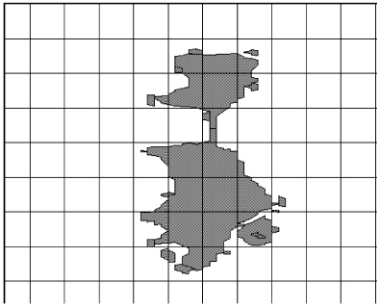


图 2 环境模型

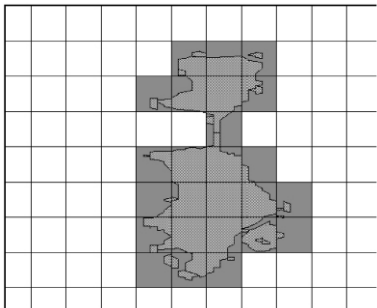


图 3 可航行判断后的环境模型

环境模型的建立是通过网格化方法把电子海图划分为若干大小相等的网格,即把数字海图转化为网格环境地图,利用环境地图中各点的标记值存储路径规划所需要的环境信息,例如大陆、岛、碍航区等障碍物的地理坐标,以及对应位置的海洋环境信息等。

环境模型的存储采用一个结构数组 Danger everypoint [],其中 Danger 的数据结构如下:

```
struct Danger
{
    double point_x;    //该网格的经度坐标
    double point_y;    //该网格的纬度坐标
    BOOL FeasibleorNot; //可航性判断标志
    BOOL AvailableorNot; //网格有效性判断标志
};
```

环境模型的表示如图 2 所示。

设置网格大小为 $D \times D$ 海里,分别判断所有节点的可航性,即判断该网格是否在不可航区域内。可航区域 FeasibleorNot 值为 TRUE,不可航区域 FeasibleorNot 值为 FALSE。网格可航性判断后的环境模型如图 3 所示。

2.3 动态网格模型的建立

本文采取网格长度递减策略动态地降低网格模型中的网格数。对前述的网格模型进行如下改进:先对整个航行区域进行“粗”分割,在“粗网格”状态下,找寻到最短可航路径,得到“有效网格”(最短可航路径上的网格及其相邻网格);剔除“有效网格”外的所有网格,再细分剩余的“有效网格”,并在细分后的网格状态下继续寻找最短路径。如此循环,进行若干次细分,直到网格边长达到航行要求的精度。最后得到最短可航路径经过的网格。

此方法是一边找寻最短路径,一边淘汰明显无效网格区域,是动态的方法。如此处理可以去掉大量的无用网格,降低细分后的网格数量。其整个过程是先找到比较宽粗的航线,然后逐渐细化,直至寻找到满足航行要求的航线。动态网格模型如图 4、图 5 所示。

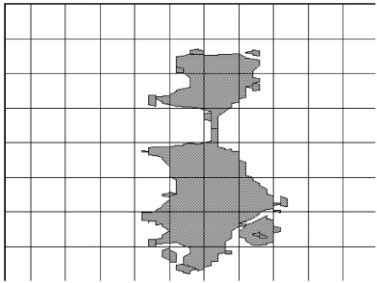


图 4 航行精度为 500m 的网格模型

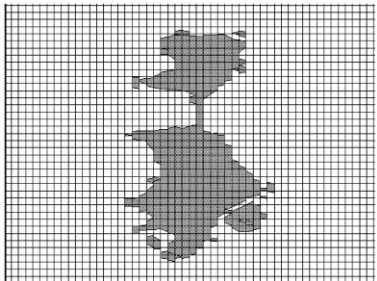


图 5 航行精度为 100m 的网格模型

3 算法的描述与实现

3.1 Dijkstra 搜索算法基本思想

Dijkstra 算法的基本思想是:设置两个顶点的集合 S 和 $T=V-S$,集合 S 中存放已找到最短路径的顶点,集合 T 存放当前还未找到最短路径的顶点^[10]。初始状态时,集合 S 中只包含源点 u_0 ,然后不断从集合 T 中选取到顶点 u_0 路径长度最短的顶点 v 加入到集合 S 中。集合 S 每加入一个新的顶点 v ,都要修改顶点 u_0 到集合 T 中剩余顶点的最短路径长度值。集合 T 中各顶点新的最短路径长度值为原来的最短路径长度值与顶点 v 的最短路径长度值加上 v 到该顶点的路径长度值中的较小值。此过程不断重复,直到集合 T 的顶点全部加入到 S 中为止。

Dijkstra 算法在计算两点间最短路径问题的过程中,通常执行了许多与最短路径无关的顶点的运算,增加了额外的运算量,从而降低了算法的效率。尤其当集合 T 中顶点较多时,传统 Dijkstra 算法的效率是非常低的。为提高运算效率,我们使用了距离寻优的 Dijkstra 算法^[11]。

3.2 距离寻优 Dijkstra 搜索算法的实现

在两个指定顶点 u_0 和 v_0 之间寻找一条最短距离的路径问题,称之为距离寻优。

根据几何学知识,任意两点 u, v 的直线距离记为:

$$J(u, v) = \sqrt{(x_u - x_v)^2 + (y_u - y_v)^2}$$
 (1)

两个指定顶点 u_0 和 v_0 之间的最短距离下限为这两点间的直线距离:

$$J(u_0, v_0) = \sqrt{(x_{u_0} - x_{v_0})^2 + (y_{u_0} - y_{v_0})^2} \leq d(u_0, v_0)$$
 (2)

假设已经求得 u_0 到 u_i 的距离,标识为 $L(u_i)$,那么 u_0 经过 u_i 到达 v_0 的距离下限应该为:

$$L(u_i) + J(u_i, v_0) = L(u_i) + \sqrt{(x_{u_i} - x_{v_0})^2 + (y_{u_i} - y_{v_0})^2}$$
 (3)

在距离寻优算法中,优化的目标就是提高 S 集合的效率,最大程度地减少 S 中与最短距离路径无关的顶点的计算。

优化采取的方法是:

(1)计算顶点 v 的 $L(v)$ 值($v \in T$)时,就计算 v 和目标点 v_0 之间的直线距离 $J(v, v_0)$,认为目前从 u_0 经过 v 到 v_0 的距离下限为 $L(v) + J(v, v_0)$ 。

(2) T 中顶点加入 S 集合的规则为:在集合 T 中寻找一点 u_{i+1} ,从 u_0 经过 u_{i+1} 到 v_0 的距离的下限为最小值,即满足

$$L(u_{i+1}) + J(u_{i+1}, v_0) = \min \{ L(v) + J(v, v_0) \} \quad (v \in T)$$
 (4)

把顶点 $u_{i+1} (u_{i+1} \in T)$ 加入集合 S 。

具体算法如下:

①初始化:

$$L_{u_0} = 0;$$

$$J(u_0, v_0) = \sqrt{(x_{u_0} - x_{v_0})^2 + (y_{u_0} - y_{v_0})^2};$$

$$L_v = \infty \quad v \neq u_0;$$

$$S = u_0;$$

$$i = 0;$$

②判断 u_i 是否为 v_0 ,若是,转④,否则转③:

③添加顶点 u_{i+1} 到集合 S

$$L(v) = \min \{ L(v), L(u_i) + W(u_i, v) \};$$

$$J(v, v_0) = \sqrt{(x_v - x_{v_0})^2 + (y_v - y_{v_0})^2};$$

$$\forall v \in T \text{ 并且 } \exists e, \varphi(e) = u_i v;$$

$$\text{计算 } \min \{ L(v) + J(v, v_0) \} \quad v \in T;$$

存在 u_{i+1} 使得:

$$L(u_{i+1}) + J(u_{i+1}, v_0) = \min \{ L(v) + J(v, v_0) \} \quad v \in T;$$

$$S \cup \{ u_{i+1} \} \rightarrow S, i = i + 1;$$

④结束: $d(u_0, v_0) = L(v_0)$ 。

由算法过程可以看出,在传统 Dijkstra 算法中不断把距离 u_0 最近的顶点加入集合 S ,而该过程没有把 u_0 和 v_0 联系起来考虑。在距离寻优 Dijkstra 算法中,向着从 u_0 到 v_0 最短路径不断逼近的目标而选择顶点加入集合 S 。因此,集合 S 中的顶点基本在 u_0 到 v_0 的最短路径的局部范围内,而那些与最短路径相距较远的顶点几乎不会被计算到。这种情况下距离寻优 Dijkstra 算法的 S 集合要比传统 Dijkstra 算法的 S 集合小得多,进而提高了计算效率。

3.3 路径优化

如果把规划出来的路径点连接起来作为水面无人艇的航行路径,折线会过多。路径中的多余节点指的是那些去掉以后不会影响路径的有效性和安全性的节点。出现的多余节点使得规划出来的路径有时会出现阶梯和锯齿状的线段。将规划的结果作为航行路径要经常改变航向,这样对水面无人艇的控制提出了很高的机动性要求。因为多余节点的存在,使得规划出来的路径并不理想,这样的情况是我们不希望的。为此需要对规划的路径进行优化,以减少路径中不必要的路径点,使折线优化为直线,增加路径的光滑度。

本文中优化采取的方法是:对规划出来的路径,依次取出连续的节点 p_i, p_{i+1}, p_{i+2} 。若 p_i 与 p_{i+2} 之间的连线不穿过任何障碍物,则删去 p_{i+1} ,然后连续判断 p_i 与 p_{i+2} 之后节点的连线,若不穿过任何障碍物,则删去 p_{i+2} 。依次类推,直到 p_i 与后面的某节点连接穿过障碍物,则从该节点开始向后取出连续的 3 个节点作为 p_i, p_{i+1}, p_{i+2} ,继续刚才的操作,直至取完路径中的所有节点。航线优化效果如图 6、图 7 所示。

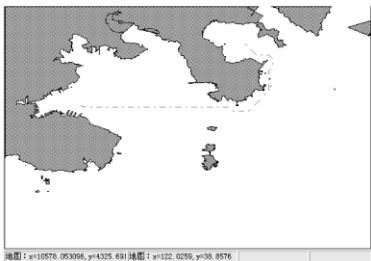


图 6 优化前的规划路径

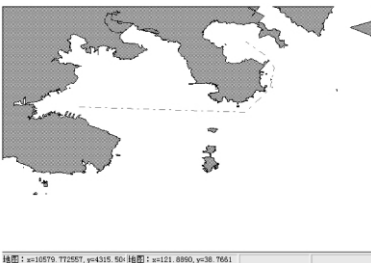


图 7 优化后的规划路径

图 6 中优化前的航线包含节点 62 个,图 7 中优化后的航线仅包含节点 6 个。由此可以看出,对规划后的航线进行路径规划可以减少航行节点数,缩短航行时间,减轻运动控制系统的负荷,对水面无人艇的安全快速航行有着重要的意义。

4 仿真结果

为了说明环境模型和算法的有效性,下面给出仿真结果。仿真环境用 VC 开发,运行于 PC 机。基于 MapObjects 开发的电子海图显示平台界面如图 8 所示。

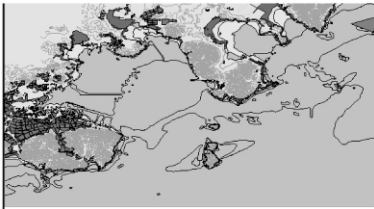


图 8 电子海图显示平台界面

针对大连海区(区域坐标范围为东经 121.571 度至122.029 度,北纬 38.8167 度至 39.0667 度)给出如图 9、图 10 所示的仿真结果,图中虚线为最终规划航线。

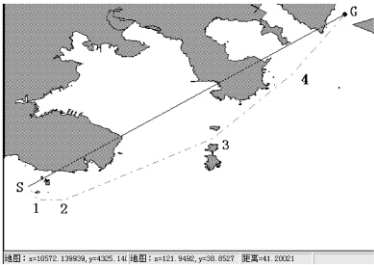


图 9 仿真结果 1

表 1 列出仿真结果 1 中路径规划各点坐标。

表 1 仿真结果 1 中路径规划各点坐标

路由节点	经度坐标(度)	纬度坐标(度)
初始点 S	121.600	38.8428
节点 1	121.611	38.8267
节点 2	121.626	38.8267
节点 3	121.826	38.9017
节点 4	121.926	38.9817
目标点 G	121.990	39.0539

初始点 S 到目标点 G 的直线距离为 41.2 海里,规划后总航程为 45.39 海里。

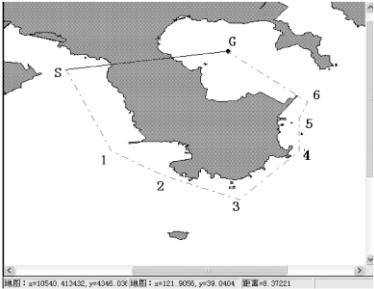


图 10 仿真结果 2

表 2 列出仿真结果 2 中路径规划各点坐标。

表 2 仿真结果 2 中路径规划各点坐标

路由节点	经度坐标(度)	纬度坐标(度)
初始点 S	121.764	39.0168
节点 1	121.791	38.9667
节点 2	121.821	38.9517
节点 3	121.866	38.9367
节点 4	121.901	38.9667
节点 5	121.901	38.9867
节点 6	121.906	38.9967
目标点 G	121.859	39.0283

初始点 S 到目标点 G 直线距离为 8.37 海里,规划后总航程为 26.81 海里。

针对动态网格模型给出如图 11、图 12 所示的仿真结果。

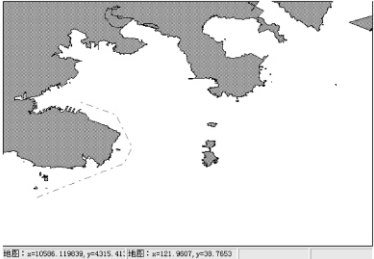


图 11 无动态网格模型规划结果

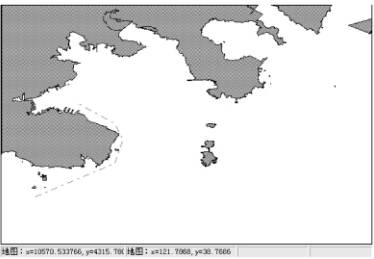


图 12 使用动态网格模型规划结果

图 11、图 12 规划起点和终点均为(121.613,38.8238), (121.667,38.9406)。

表 3 列出动态网格模型规划结果对比。

表 3 动态网格模型规划结果对比

	无动态网格模型	有动态网格模型
网格精度(m)	500	局部 100
航程(海里)	22.51	21.02
占用内存(kB)	200264	29504
运行时间(ms)	1469	641

由此可见,应用动态网格模型可以使航行节点布置更加合理,并且可有效地缩短航程,减少程序占用的内存和运行时间。

由仿真结果可以看出,本文给出的环境模型和路径规划算法可以在预定区域快速生成一条安全航线。节点位置合理,在避开大陆、半岛、暗礁等障碍物的同时保证了航行距离相对最小。

结束语 本文给出的基于电子海图的水面无人艇全局路径规划方法可以在给定的海图区域任意两个可航点之间,快速、准确地生成一条可以避开陆地、岛、半岛、暗礁、沉船以及其他障碍物的最短安全航线。针对研究对象水面无人艇的环境特点,选择了能够解决大范围环境下路径规划的动态网格建模方法。由于环境建模以及仿真环境都是在真实电子海图环境下完成,生成的航线可以直接为运动控制系统调用,因此

(下转第 219 页)

tionistic fuzzy sets; straightforward approaches may not work [J]. Studies in Computational Intelligence, 2008, 109: 415-430

[5] Szmidt E, Kacprzyk J. Distances between intuitionistic fuzzy sets and their applications in reasoning [J]. Studies in Computational Intelligence, 2005, 2: 101-116

[6] Szmidt E, Kacprzyk J. An intuitionistic fuzzy set based approach to intelligent data analysis: an application to medical diagnosis [C]// Abraham A, Jain L, Kacprzyk J, eds. Recent advances in intelligent paradigms and applications. Studies in Fuzziness and Soft Computing, Germany: Physica-Verlag GmbH Heidelberg, 2003: 57-70

[7] Hung W L, Yang M S. Similarity measure of intuitionistic fuzzy sets based on L_p metric [J]. International Journal of Approximate Reasoning, 2007, 46(1): 120-136

[8] Yuan Xue-hai, Li Hong-xing. Cut sets on interval-valued intuitionistic fuzzy sets [C]// IEEE Sixth International Conference on Fuzzy Systems and Knowledge Discovery, FSDK. Tianjin, China, vol. 6, 2009: 167-171

[9] Yuan Xue-hai, Li Hong-xing, Sun Kai-biao. Theory based on interval-valued level cut sets of Zadeh fuzzy sets [J]. Fuzzy Info and Eng, 2009, 2: 501-510

[10] Yager R R. Some aspects of intuitionistic fuzzy sets [J]. Fuzzy Optim Decis Making, 2009, 8: 67-90

[11] Xu Ze-shui. Some similarity measures of intuitionistic fuzzy sets and their applications to multiple attribute decision making [J]. Fuzzy Optim Decis Making, 2007, 6: 109-121

[12] Xu Ze-shui, Yager R R. Intuitionistic and interval-valued intuitionistic fuzzy preference relations and their measures of similarity for the evaluation of agreement within a group [J]. Fuzzy Optim Decis Making, 2009, 8: 123-139

[13] Li Deng-feng. Topsis-based Nonlinear-programming methodology for multiattribute decision making with Interval-valued intuitionistic fuzzy sets [J]. IEEE Transactions on Fuzzy Systems, 2010, 18(2): 299-311

[14] Li Deng-feng. Mathematical-Programming approach to matrix games with payoffs represented by Atanassov's Interval-valued intuitionistic fuzzy sets [J]. IEEE Transactions on Fuzzy Systems, 2010, 18(6): 1112-1128

[15] Garcia J G, Rodabaugh S E. Order-theoretic, topological, categorical redundancies of interval-valued sets, grey sets, vague sets, interval-valued intuitionistic sets, intuitionistic fuzzy sets and topologies [J]. Fuzzy Sets and Systems, 2005, 156: 445-484

[16] 徐泽水. 区间直觉模糊集相似性测度及其在模式识别中的应用 [J]. 东南大学学报: 英文版, 2007, 23(1): 139-143

[17] Zhang Ying-jun, Ma Pei-jun, Su Xiao-hong. Pattern recognition using interval-valued intuitionistic fuzzy set and its similarity degree [C]// IEEE International Conference on Intelligent Computing and Intelligent Systems, ICIS. Shanghai, China, 2009: 361-365

[18] Zhang Qian-sheng, Yao Hai-xiang, Zhang Zhen-hua. Some similarity measures of interval-valued intuitionistic fuzzy sets and application to pattern recognition [J]. International Journal of Applied Mechanics and Materials, 2011(44-47): 3888-3892

[19] Zhang Qian-sheng, Jiang Sheng-yi, Jia Bao-guo, et al. Some information measures for interval-valued intuitionistic fuzzy sets [J]. Information Sciences, 2010, 180(12)

[20] Zhang Qian-sheng, Yao Hai-xiang, Zhang Zhen-hua. An interval-valued fuzzy reasoning approach based on weighted similarity measure [J]. Advanced Materials Research, 2011(143/144): 161-165

[21] 雷阳, 雷英杰, 华继学, 等. 基于自适应直觉模糊推理的目标识别方法 [J]. 系统工程与电子技术, 2010, 32(7): 1471-1475

[22] 申晓勇, 雷英杰, 周创明, 等. 基于直觉模糊集的不确定时序逻辑模型 [J]. 计算机科学, 2010, 37(5): 187-189

[23] 申晓勇, 雷英杰, 华继学, 等. 基于 IFTPN 的不确定时间知识描述和推理方法 [J]. 控制与决策, 2010, 10(25): 1457-1462

[24] 张继国, 林文浩. Vague 集向 Fuzzy 集转化的新模型 [J]. 模糊系统与数学, 2010, 24(4): 146-150

[25] 徐凤生. Vague 集向 Fuzzy 集转化的两种新方法 [J]. 计算机工程与应用, 2010, 46(21): 160-161

[26] 范平, 戴文华, 桂学勤. Vague 集向 Fuzzy 集转化的方法分析 [J]. 计算机工程与应用, 2010, 46(24): 48-50

[27] 王鸿绪. 从 Fuzzy 值数据向 Vague 值数据的转化公式 [J]. 计算机工程与应用, 2010, 46(25): 47-48

[28] 王鸿绪. 单值数据转化为 Vague 值数据的定义和转化公式 [J]. 计算机工程与应用, 2010, 46(24): 42-44

(上接第 214 页)

保证了运动控制系统的实时性, 满足了无人艇航速快、机动性强等特点。

在海洋环境下有风、浪、流的作用, 特别是针对水面无人艇这一特殊的对象, 这些影响更加突出。环境模型的建立仅仅基于电子海图, 没有考虑风、浪、流等水文气象因素对航线设计的影响, 这有待于进一步的研究和改进。

参 考 文 献

[1] 徐玉如, 苏玉民, 庞永杰. 海洋空间智能无人运载器技术发展展望 [J]. 中国舰船研究, 2006, 1(3): 2-4

[2] Manely J E. Unmanned Surface Vehicles, 15 Years of Development [C]// Proc. Oceans 2008 MTS/IEEE Quebec Conference and Exhibition. Quebec City: Ocean'08, 2008: 1-4

[3] Veers J, Bertran V. Development of the USV Multi-mission Surface Vehicle III [C]// 5th Int. Conf. Computer and IT Application in the Maritime Industries, COMPIT. 2006: 345-355

[4] 廖煜雷, 庞永杰, 庄佳园. 无人水面艇嵌入式基础运动控制系统研究 [J]. 计算机科学, 2010, 9(37): 214-217

[5] Kim M-S, Moon S-R, Lee K-H. Motion Planning with Planar Geometric Models [C]// Proceedings of the IEEE International Conference on Robotics and Automation. Sacramento, California, April 1991

[6] Conn R A, Robot M K. Motion Planning on N -dimensional Star Worlds Among Moving Obstacles [J]. IEEE Transactions on Robotics and Automation, 1998, 14(2): 320-325

[7] Bijlsma S J. On the applications of the principle of optimalevolution in ship routing [J]. Journal of the Institute of Navigation, 2004, 51(2): 93-100

[8] 饶森. 水面无人艇的全局路径规划技术研究 [D]. 哈尔滨: 哈尔滨工程大学, 2007

[9] 韩鹏. 地理信息系统开发——MapObjects 方法 [M]. 武汉: 武汉大学出版社, 2004: 18-40

[10] Dijkstra E. A note on two problems in connexion with graphs [M]. Numerische Mathematik, 1959: 269-271

[11] 朱静. Dijkstra 算法在 GIS 中的优化实现 [J]. 计算机与现代化, 2005, 9: 19-24