

基于多元约束Petri网的水利测绘无人机路径规划

姚喜, 陈衍德

引用本文

姚喜, 陈衍德. 基于多元约束Petri网的水利测绘无人机路径规划[J]. 计算机科学, 2023, 50(6A): 220700079-7.

YAO Xi, CHEN Yande. Path Planning of Hydrographic Mapping UAV Based on Multi-constraint Petri Net [J]. Computer Science, 2023, 50(6A): 220700079-7.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

一种基于网格的水利测绘无人机跟踪方法

Grid-based Tracking Method for Hydrographic Mapping UAV

计算机科学, 2023, 50(6A): 220300023-7. <https://doi.org/10.11896/jsjcx.220300023>

基于MADDPG的无人机群空中拦截作战决策研究

Study on Intelligent Decision Making of Aerial Interception Combat of UAV Group Based onMADDPG

计算机科学, 2023, 50(6A): 220700031-7. <https://doi.org/10.11896/jsjcx.220700031>

基于RRT的无人机动态航路规划算法

UAV Dynamic Route Planning Algorithm Based on RRT

计算机科学, 2023, 50(6A): 220700127-5. <https://doi.org/10.11896/jsjcx.220700127>

基于冲突搜索的多智能体路径规划研究进展

Research Progress of Multi-agent Path Finding Based on Conflict-based Search Algorithms

计算机科学, 2023, 50(6): 358-368. <https://doi.org/10.11896/jsjcx.220800151>

空-天-地一体化移动边缘计算系统的部署优化和计算卸载

Deployment Optimization and Computing Offloading of Space-Air-Ground Integrated Mobile Edge Computing System

计算机科学, 2023, 50(2): 69-79. <https://doi.org/10.11896/jsjcx.220600057>

基于多元约束 Petri 网的水利测绘无人机路径规划

姚喜 陈衍德

山东省水利勘测设计院 济南 250013

摘要 随着测绘技术的不断发展,测绘无人机在水利工程中的应用不断深入。利用无人机进行测绘作业,一方面革新了测绘作业工作模式,提高了工作效率;另一方面也因无人机无人驾驶、持续飞行时间有限、航测图片拼接约束等客观原因,需要进行科学的无人机测绘作业路径规划,满足无人机飞行安全、航测数据有效及作业效率优化等要求。鉴于此,提出了一种基于多元约束 Petri 网的水利测绘无人机路径规划方法。对问题场景进行了描绘;定义了多元约束 Petri 网,并给出了可达性分析方法;构建了面向测绘无人机路径规划的多元约束 Petri 网模型;基于可达标识图析出了路径规划最优方案。实验证明了所提方法在面向水利测绘无人机的路径规划方案寻优上具有优越性。

关键词: 多元约束 Petri 网;水利测绘;无人机;路径规划;可达性分析

中图分类号 TP311

Path Planning of Hydrographic Mapping UAV Based on Multi-constraint Petri Net

YAO Xi and CHEN Yande

Shandong Survey and Design Institute of Water Conservancy, Jinan 250013, China

Abstract With the development of surveying and mapping technology, the application of unmanned aerial vehicle(UAV) in hydraulic engineering has been deepened. The use of UAV has revolutionized the working mode of surveying and improved working efficiency. Due to the reasons such as unmanned aerial vehicle, limited duration of flight and restriction of aerial survey picture splice, it is necessary to carry out scientific route planning. It can meet the requirements of flight safety, validity of survey data and operation efficiency. In view of this, a path planning method based on multi-constraint Petri net is proposed. The problem scene is described. The multi-constraint Petri net is defined and the method of reachability analysis is given. The multi-constraint Petri net model for path planning is constructed. The optimal route planning scheme is obtained based on the reachability marking diagram. Experimental results show that this method has superiority in UAV path planning scheme optimization.

Keywords Multi-constraint Petri net, Hydrographic mapping, UAV, Path planning, Reachability analysis

1 引言

水利工程建设自古有之,兴修水利对于促进农业生产,维护沿水地区社会稳定,保障群众生命财产安全,具有重要意义^[1]。测绘作为水利工程建设的前驱环节,能够对拟施工地区进行测量和绘图,获取及标记反映地面现状的图形和位置,为水利工程科学规划提供决策支持^[2]。随着信息技术及智能控制技术的不断发展,测绘作业模式不断革新,测绘智能化装备不断出现。测绘无人机可搭载高频差分模块、GNSS 系统、高光谱相机等装备,以其机动灵活、专业高效的特点,在水利工程中不断深入应用^[3]。如通过测绘无人机可对堰塞湖、洪水发生地等人员较难进入的地区进行空中勘测,实现了水利测绘业务的拓展^[4];利用测绘无人机对大片区域进行测绘,与传统人工徒步实地测绘相比,能够有效提高测绘作业效率^[5];利用测绘无人机对危险及环境恶劣地区进行测绘,可发挥无人驾驶测绘作业优势,提高了测绘人员的安全作业系数^[6]。利用无人机进行测绘作业,一方面革新了测绘作业工作模式,提高了工作效率;另一方面也因无人机无人驾驶、持续飞行时间有限^[7]、航测图片拼接约束^[8]等客观原因,需要进行科学的无人机测绘作业路径规划,满足无人机飞行安全、航测数据

有效及作业效率优化等要求。

当前,面向无人机的路径规划的研究较为丰富。Xue 等提出了一种基于复杂地块凸划分优化的多无人机覆盖路径规划方法^[9],采用改进的 Douglas-Peucker 算法,将复杂地块边界压缩为复杂多边形边界,进而使用随机路标法寻找最短辅线,解决了对复杂地块及复杂障碍物无法进行凸划分的问题,使得无人机在全覆盖过程中路径更短。Liao 提出了一种多无人机协同对地任务规划方法^[10],在传统合同网算法基础上增加“状态更新”和“心跳保持”两个步骤,实现了离线和实时两种模式的任务分配。引入多指标约束的效能函数及负载均衡机制,在任务分配时增加任务买卖和任务交换调配操作,提高了全局负载能力,优化了任务目标收益价值。Deng 敏等提出了一种基于 k 度平滑的多无人机协调路径规划方法^[11],首先通过改进的蚁群优化算法搜索最短路径,进而应用 k 度平滑方法平滑初始路径,协调多架无人机在 k 度时间间隔内到达指定地点,实现了多机高效路径规划。He 等提出了一种基于改进蜂群算法的无人机路径规划方法^[12],针对人工蜂群算法存在的初始蜜源随机、后期收敛速度慢、易早熟等问题,运用最大最小距离积分法处理初始蜜源,把 k 均值聚类算法与人工蜂群算法相结合对人工蜂群算法进行改进,实现了无人机

路径规划效率及质量的优化。Guo 等提出了一种基于改进鲸鱼算法的无人机三维路径规划方法^[13],该算法在迭代过程中加入了 levy 飞行,对最优解进行随机扰动,并引入了信息交流机制,通过当前全局最优解和个体记忆最优解以及邻域最优解来更新个体的位置,能够更好地均衡局部收敛和全局开发,实现了航迹优化。Pandey 等采用萤火虫群优化算法对无人机进行三维路径规划^[14],实现了三维空间内从初始点到目标点的无障碍路径寻优,并加快了算法收敛速度。Nielsen 等将无人机覆盖范围分割成若干多边形子地块,通过对子地块进行组合优化,达到无人机整体覆盖路径最短的效果^[15]。Dewangan 等使用灰狼优化器对无人机进行三维路径规划^[16],将无人机路径规划视为 Np-Hard 问题,进而构建问题模型,并利用灰狼算法进行问题求解,得到优化的三维空间无人机飞行路径。Hassan 等提出了一种基于捕猎者-猎物方法的路径寻优算法^[17],在覆盖范围已知条件下实时响应边界变化,实现新型自适应覆盖路径规划,使路径成本最小。Zhang 提出了一种基于强化学习的多无人机路径规划方法^[18],以深度强化学习算法 DQN 为基础,借鉴人工势场的思想来设计强化信号,构建了 APF-DON 模型,克服了强化学习过程中由稀疏奖励信号导致的模型收敛慢等问题,实现了在未知环境下多无人机路径的有效规划。Yao 等提出了一种无人机编队地貌监测方法^[19],由地面控制站分配任务,分布式控制无人机编队,结合定位模块,利用目标点具有吸引力、障碍物具有排斥力的势场法进行路径规划,达到了避障效果。

水利测绘无人机的作业特点有:1)测绘无人机具有续航时间约束,如果作业中的无人机电量低于安全阈值,则应立即返航,以保证设备安全;2)测绘无人机是在三维空间内进行作业,需要在测绘区域上方,即一般不存在障碍物。故当前已有无人机路径规划算法无法有效适用于水利测绘无人机的路径规划需求。

鉴于此,本文提出了一种基于多元约束 Petri 网的水利测绘无人机路径规划方法。对问题场景进行了描绘;定义了多元约束 Petri 网,并给出了可达性分析方法;构建了面向测绘无人机路径规划的多元约束 Petri 网模型,并给出了无人机最优路径析出算法。实验证明了本文方法在面向水利测绘无人机的路径规划方案寻优上具有优越性。

2 问题场景及技术路线

本文所述的场景是:利用一架测绘无人机对区域 Q 进行航空测绘作业。按照一般性水利工程地貌特征进行区域 Q 的地貌特点假设,即河流在山川中流过,河流上建设有水利大坝。测绘区域 Q 的地貌起伏较大,有低洼的河流、凹下的河床、凸起的水利大坝、耸立的山峦。测绘区域示意图如图 1 所示。

本文所解决的问题是对一架测绘无人机的作业路径进行规划,实现对 Q 区域的全覆盖性航拍。在保证航测质量和无人机安全的前提下,寻求测绘工作效率的最优化。Q 区域水利测绘无人机约束及场景约束假设为:1)无人机持续飞行时间为 T ,行进速度为 V ,返回到降落点时的剩余飞行时间应不小于 T_j ;2)假设测绘区域连续,测绘区域外边界内的区域均需勘测。测绘区域由边界离散点依次连而成的线包围而成;3)测绘无人机的起降点在测绘区域内,且为同一地点。

系统起降、悬停、补给时间忽略。

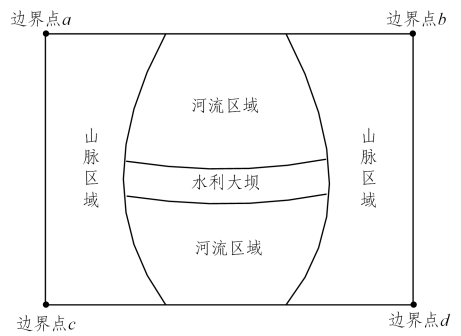


图 1 水利无人机测绘区域示意图

Fig. 1 Survey area of water conservancy UAV

本文所述的基于多元约束 Petri 网的水利测绘无人机路径规划方法的技术路线为:首先,对勘测区域进行平面网格划分;然后,建立面向测绘区域的多元约束 Petri 网模型;最后,基于可达标识图生成无人机最优路径规划策略。

3 具体方法

3.1 测绘网格划分

本文将勘测区域进行网格化,并对测绘网格进行信息标定。网格划分可减小无人机路径规划算法的数据处理基数,提高计算效率,降低计算复杂度。下面给出测绘区域、测绘网格的形式化定义,并对测绘网格划分算法进行描述。

定义 1(测绘区域) 二元组 $Q=(Id, D[n][2])$ 为一个测绘区域,其中:1) Id 唯一标识一个测绘区域;2) $D[n][2]$ 为测绘区域边界线上的离散点坐标二维数组。测绘区域的赋值方法为:用户按照逆时针顺序依次输入测绘区域边界线上的离散点坐标值。若当前为用户输入的第 i 个坐标,且经纬度坐标值为 (j, w) ,则令 $D[i][0]=j, D[i][1]=w; Q.Id$ 由系统生成。

定义 2(测绘网格) 十三元组 $Wg=(Id, d_1, d_2, d_3, d_4, s, zs, z, zx, ys, y, yx, x)$ 为一个测绘网格,其中:1) Id 唯一标识一个测绘网格;2) d_1, d_2, d_3, d_4 分别标识了测绘网格左上、左下、右上、右下 4 个顶点坐标;3) $s, zs, z, zx, ys, y, yx, x$ 分别标识了此测绘网格上边、左上、左边、左下、右上、右边、右下、下边相邻的测绘网格的 Id 。

算法 1 测绘网格划分算法

输入:测绘区域 Q, 无人机测绘相机单幅照片覆盖区域宽度 Kd 、长度 Cd

输出:测绘网格集合 W

第 1 步 创建 4 个变量 $J_{max}=J_{min}=W_{max}=W_{min}=0$; 遍历 $Q.D[n][2]$, 假设当前元素为 $D[i][0]$, 若 $D[i][0]<J_{min}$, 则令 $J_{min}=D[i][0]$; 若 $D[i][0]>J_{max}$, 则令 $J_{max}=D[i][0]$; 假设当前元素为 $D[i][1]$, 若 $D[i][1]<W_{min}$, 则令 $W_{min}=D[i][0]$; 若 $D[i][0]>W_{max}$, 则令 $W_{max}=D[i][0]$ 。

第 2 步 创建变量 $f_1=J_{max}, f_2=J_{min}, f_3=W_{max}, f_4=W_{min}$, 令 f_5 为测绘网格长度, 且 $f_5=Cd, f_6$ 为测绘网格宽度, 且 $f_6=Kd$, 测绘网格集合 $W=Null$. For ($f_2=J_{min}, f_2<(f_1+f_3), f_2=f_2+f_5$) do, For ($f_4=W_{min}, f_4<(f_3+f_6), f_4=f_4+f_6$) do, 创建一个新的测绘网格 $Wg_k=(Id, d_1, d_2, d_3, d_4, s, zs, z, zx, ys, y, yx, x); Wg_k.d_1=(f_2, f_4), Wg_k.d_2=(f_2+f_5, f_4), Wg_k.d_3=(f_2, f_4+f_6), Wg_k.d_4=(f_2+f_5, f_4+f_6)$; 将 Wg_k 放入集合 W 中。

第 3 步 遍历集合 W, 若当前元素为 Wg_i , 令变量 $f_7=0$, 遍历 Wg_i 的

4个顶点,若当前顶点为 $d_k=(x_k, y_k)$,如果 $(x_k < J_{\min})$ 或 $(x_k > J_{\max})$ 或 $(y_k < W_{\min})$ 或 $(y_k > W_{\max})$,则 $f_7 = f_7 + 1$ 。创建变量 f_8, f_9 ,布尔变量 $f_{10} = \text{false}$, $\text{For}(f_8 = 0, f_9 = (\lfloor Q \cdot D[n][2] \rfloor / 2 - 1), f_8 < \lfloor Q \cdot D[n][2] \rfloor / 2, f_9 = f_8 + 1)$,如果 $Q \cdot D[f_8][1] > y_k$ 且 $Q \cdot D[f_9][1] < y_k$,或者 $Q \cdot D[f_8][1] < y_k$ 且 $Q \cdot D[f_9][1] > y_k$,则如果 $x_k < ((Q \cdot D[f_9][0] - Q \cdot D[f_8][0]) * (y_k - Q \cdot D[f_8][1]) / (Q \cdot D[f_9][1] - Q \cdot D[f_8][1]) + Q \cdot D[f_8][0])$,则令布尔变量 f_{10} 求反。上述 For 语句执行完毕后,若 f_{10} 为 false,则令 $f_7 = f_7 + 1$ 。当 W_{g_i} 的 4 个顶点遍历完成后,若 f_7 为 4,则将测绘网格元素 W_{g_i} 从集合 W 中删除。

第 4 步 令 $f_1 = 1$,遍历集合 W ,若当前元素为 W_{g_i} ,则 $W_{g_i} \cdot \text{Id} = f_1$, $f_1 = f_1 + 1$ 。

第 5 步 遍历集合 W ,若当前元素为 W_{g_i} 。遍历集合 W 中除 W_{g_i} 之外的其他元素,若当前元素为 W_{g_k} ,若 $W_{g_k} \cdot d_2 = W_{g_i} \cdot d_1$ 且 $W_{g_k} \cdot d_4 = W_{g_i} \cdot d_3$,则令 $W_{g_i} \cdot z = W_{g_k} \cdot \text{Id}$;若 $W_{g_k} \cdot d_2 = W_{g_i} \cdot d_1$ 且 $W_{g_k} \cdot d_3 = W_{g_i} \cdot d_4$,则令 $W_{g_i} \cdot s = W_{g_k} \cdot \text{Id}$;若 $W_{g_k} \cdot d_4 = W_{g_i} \cdot d_2$ 且 $W_{g_k} \cdot d_3 = W_{g_i} \cdot d_1$,则令 $W_{g_i} \cdot x = W_{g_k} \cdot \text{Id}$;若 $W_{g_k} \cdot d_1 = W_{g_i} \cdot d_2$ 且 $W_{g_k} \cdot d_3 = W_{g_i} \cdot d_4$,则令 $W_{g_i} \cdot y = W_{g_k} \cdot \text{Id}$;若 $W_{g_k} \cdot d_4 = W_{g_i} \cdot d_1$ 则令 $W_{g_i} \cdot zx = W_{g_k} \cdot \text{Id}$;若 $W_{g_k} \cdot d_1 = W_{g_i} \cdot d_4$ 则令 $W_{g_i} \cdot ys = W_{g_k} \cdot \text{Id}$;若 $W_{g_k} \cdot d_3 = W_{g_i} \cdot d_2$ 则令 $W_{g_i} \cdot yx = W_{g_k} \cdot \text{Id}$;若 $W_{g_k} \cdot d_2 = W_{g_i} \cdot d_3$ 则令 $W_{g_i} \cdot zs = W_{g_k} \cdot \text{Id}$ 。

第 6 步 输出测绘网格集合 W 。

由算法 1 可知,测绘网格划分方法分为 5 个环节。首先,找出测绘区域边缘离散点坐标中经纬度的最大值及最小值。然后,以(经度最大值,纬度最大值)(经度最小值,纬度最小值)(经度最大值,纬度最小值)(经度最小值,纬度最大值)为顶点确定网格预划分区域。其次,从坐标(经度最小值,纬度最小值)开始,按照拟划定测绘网格的长宽值,对网格预划分区域进行连续的网格创建,并明确每个网格的顶点坐标及高度值。再次,判断所创建的测绘网格是否在测绘区域内,对不在测绘区域的网格进行删除。最后,对在测绘区域内的测绘网格 Id 号,及其上边、左上、左边、左下、右上、右边、右下、下边相邻测绘网格 Id 属性值进行赋值。将所划分网格放入集合并输出,完成测绘网格划分。

图 2 给出了一个平面网格划分的例子。由图 2 可知,测绘区域由 6 个边缘离散点依次连接包围而成,经过算法 1 所述的平面网格划分后,该测绘区域被划分为 32 个测绘网格。如标号 13 的测绘网格 $W_{g_{13}}$ 的属性值为 $W_{g_{13}} \cdot \text{Id} = 13, W_{g_{13}} \cdot s = 14, W_{g_{13}} \cdot zs = 8, W_{g_{13}} \cdot z = 7, W_{g_{13}} \cdot zx = 6, W_{g_{13}} \cdot ys = 20, W_{g_{13}} \cdot y = 19, W_{g_{13}} \cdot yx = 18, W_{g_{13}} \cdot x = 12$ 。

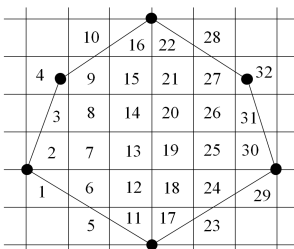


图 2 测绘网格划分

Fig. 2 Mapping grid division

3.2 测绘区域形式化模型构建

3.2.1 多元约束 Petri 网定义

定义 3(网) $N=(P, T; F)$ 是一个网^[20],其中:

(1) P 是一个有限库所集;

(2) T 是一个有限变迁集,且 $P \cup T \neq \emptyset$ 及 $P \cap T = \emptyset$;

(3) $F \subseteq (P \times T) \cup (T \times P)$ 是一个有限弧集;

(4) $\text{dom}(F) \cup \text{cod}(F) = P \cup T$,其中

$$\text{dom}(F) = \{x \in P \cup T \mid \exists y \in P \cup T: (x, y) \in F\}$$

$$\text{cod}(F) = \{x \in P \cup T \mid \exists y \in P \cup T: (y, x) \in F\}$$

定义 4(前置库所及后置库所) $x \in P \cup T$ 是一个 $N=(P, T; F)$ 中的节点。 $\cdot x = \{y \mid (y, x) \in F\}$ 表示 x 的前置库所; $x \cdot = \{y \mid (x, y) \in F\}$ 表示 x 的后置库所。如果 $X \subseteq P \cup T$,则 X 的前置库所和后置库所分别表示为 $\cdot X = \bigcup_{x \in X} \cdot x$ 及 $X \cdot = \bigcup_{x \in X} x \cdot$ 。

定义 5(Petri 网) $PN=(P, T; F, M_0)$ 是一个 Petri 网^[21],其中:

(1) $N=(P, T; F)$ 是一个网;

(2) $M: P \rightarrow Z$ 是一个标识函数,其中 M_0 是初始标识, $Z = \{0, 1, 2, \dots\}$ 是一个自然数集;

(3) 变迁引发规则:

1) 如果 $\forall p \in \cdot t: M(p) \geq 1$,则 t 在标识 M 下是使能的,表示为 $M[t]$;

2) 如果 t 在标识 M 下是使能的,则 t 能引发,且一个新的标识 M' 从标识 M 中计算得来,表示为 $M[t]M'$,其中

$$M'(p) = \begin{cases} M(p) + 1, & \text{if } p \in t \cdot - \cdot t \\ M(p) - 1, & \text{if } p \in \cdot t - t \cdot \\ M(p), & \text{else} \end{cases}$$

为了便于应用 Petri 网构建测绘区域形式化模型,本文对 Petri 网定义进行改进,提出了多元约束 Petri 网(Multi-Constraint Petri Net, MCPN)。下面给出多元约束 Petri 网的形式化定义,并对其动态性质进行分析。

定义 6(多元约束 Petri 网) 五元组 $MCPN=(P, T; F, D, M)$ 为一个多元约束 Petri 网,当且仅当:

(1) $N=(P, T; F)$ 是一个网;

(2) $M: P \rightarrow \{0, 1\}$ 是一个标识函数,且 $\forall p \in P, M(p)$ 表示 p 中的托肯(token)数;

(3) $F \subseteq (P \times T) \cup (T \times P)$ 是一个有向弧集;

(4) D 是一个面向库所的多元属性集函数。根据需要,本文定义多元属性集内容为:对于 $\forall p \in P, D(p) = \{Id, d_1, d_2, d_3, d_4, s, zs, z, zx, ys, y, yx, x\}$,其中 Id 标识了库所的编号,且唯一标识一个库所; d_1, d_2, d_3, d_4 为变量,且与特定测绘网格 4 个顶点坐标及测绘网格高度相对应; $s, zs, z, zx, ys, y, yx, x$ 为变量,且与特定测绘网格上边、左上、左边、左下、右上、右边、右下、下边相邻的测绘网格的 Id 相对应。

(5) 变迁触发规则:如果 $\exists p_i \in \cdot t: M(p_i) = 1$,且 $\exists p_j \in t \cdot: M(p_j) = 0$;或者, $\exists p_i \in t \cdot: M(p_i) = 1$,且 $\exists p_j \in \cdot t: M(p_j) = 0$;则 t 在标识 M 下是使能的,表示为 $M[t]$;此变迁 t 触发后,一个新的标识 M' 从标识 M 中计算得来,表示为 $M[t]M'$,其中 $M'(p_i) = 0, M'(p_j) = 1$,其他库所标识不变。

3.2.2 多元约束 Petri 网可达性分析

算法 2 MCPN 一步可达标识计算算法

输入:多元约束 Petri 网 $MCPN=(P, T; F, D, M)$,变迁 $t \in T$,当前标识 M_0 ;

输出:若变迁 t 使能,且 $M[t]M'$,输出为 M' ,否则为 M 。

第 1 步 变迁 t 使能判定。遍历 MCPN. P ,假设当前库所为 p_i ,若 $p_i \in \cdot t$,且 $M_0(p_i)$ 为 1,则遍历 MCPN. P ,假设当前库所为

p_j , 若 $p_j \in t'$, 且 $M_0(p_j)$ 为 0, 则变迁 t 使能, 即 $M_0[t]$ 。遍历 MCPN, P , 假设当前库所为 p_i , 若 $p_i \in t'$, 且 $M_0(p_i)$ 为 0, 则遍历 MCPN, P , 假设当前库所为 p_j , 若 $p_j \in t'$, 且 $M_0(p_j)$ 为 1, 则变迁 t 使能, 即 $M_0[t]$ 。

第 2 步 令 $M' = M_0$, 若 $M_0[t]$, 则令 $M'(p_i) = 0, M'(p_j) = 1$ 。

第 3 步 输出 M' 。

定义 7 (可达标识图) 令 $MCPN = (P, T; F, D, M)$ 为一多元约束 Petri 网, 多元约束 Petri 网 MCPN 的可达标识图表示为 $RG(MCPN) = (R(M_0), E, W)$, 其中 $E = \{(M_i, M_j) | M_i, M_j \in R(M_0), \exists t_k \in T: M_i[t_k]M_j\}$, 且 $M_i(p_i) = 0, M_j(p_i) = 1, M_i(p_j) = 1, M_j(p_j) = 0$ 。 $W: E \rightarrow (T, P), W(M_i, M_j) = (t_k, p_j, p_i)$, 且 $M_i[t_k]M_j$ 。 $R(M_0)$ 表示可达标识图的顶点集合, E 表示可达标识图的弧集。对于 $W(M_i, M_j) = (t_k, p_j, p_i), (t_k, p_j, p_i)$ 表示弧 (M_i, M_j) 上的标注。

下面给出多元约束 Petri 网可达标识图的构建方法。

算法 3 $RG(MCPN)$ 的构建算法

输入: 多元约束 Petri 网 $MCPN = (P, T; F, D, M)$;

输出: 可达标识图 $RG(MCPN) = (R(M_0), E, W)$ 。

第 1 步 定义链表 Q 的数据结构。链表 Q 中结点的数据结构定义为两部分。一是数据区, 用 $data$ 表示, 数据区用来存储多元约束 Petri 网的标识; 另一部分是指针类型的二维数组区, 用 $next[n][4]$ 表示。

第 2 步 建立并初始化 Q 的头结点 $Head$ 。令 $Head.data = M_0, Head.next = Null$, 且令 $Head$ 为当前结点 X 。

第 3 步 遍历 MCPN 的所有变迁 T , 假设当前变迁为 $t \in T$ 。如果 $X.data[t]$, 则由算法 2 得到新标识 M' , 即 $X.data[t]M'$, 且 $X.data(p_i) = 0, M'(p_i) = 1, X.data(p_j) = 1, M'(p_j) = 0$ 。遍历链表 Q 的所有结点, 假设当前结点为 $Q_k \in Q$, 如果 $Q_k.data = M'$, 则遍历 $X.next[n][4]$ 数组中的元素。假设数组 $X.next[n][4]$ 的当前元素为 $X.next[y]$, 如果 $X.next[y][0] = Null$, 则令 $X.next[y][0] = Q_k, X.next[y][1] = t, X.next[y][2] = p_j, X.next[y][3] = p_i$ 。否则, 新建一个结点 K , 令 $K.data = M', K.next = Null$ 。遍历 $X.next[n][4]$ 数组中的元素。假设数组 $X.next[n][4]$ 的当前元素为 $X.next[y]$, 如果 $X.next[y][0] = Null$, 则 $X.next[y][0] = K, X.next[y][1] = t, X.next[y][2] = p_j, X.next[y][3] = p_i$ 。令 K 为当前结点 X , 返回到第 3 步进行迭代。

第 4 步 遍历链表 Q 的所有结点。假设当前结点为 $Q_k \in Q$ 。由定义 7 对可达标识图的形式化描述, 把标识 $Q_k.data$ 放入可达标识图 $RG(MCPN)$ 的顶点集合 $R(M_0)$ 中。遍历 $Q_k.next[n][4]$ 数组中的元素, 假设数组 $Q_k.next[n][4]$ 的当前元素为 $Q_k.next[y]$ 。如果 $Q_k.next[y][0] \neq Null$, 则由定义 7 对可达标识图的形式化描述, 把 $(Q_k.data, Q_k.next[y][0].data)$ 放入 $RG(MCPN)$ 的弧集 E 中, $W: (Q_k.data, Q_k.next[y][0].data) \rightarrow (Q_k.next[y][1], Q_k.next[y][2], Q_k.next[y][3])$ 。

第 5 步 输出可达标识图 $RG(MCPN) = (R(M_0), E, W)$ 。

算法 3 给出了多元约束 Petri 网可达标识图的构建算法, 通过算法 3 可以得到多元约束 Petri 网的所有可达标识, 以及标识变化时所触发的变迁、库所托肯变化等情况, 构建的多元约束 Petri 网可达标识图能够展现出 MCPN 所包含的全部运动状态。下面给出一个构建 MCPN 可达标识图的例子。存在多元约束 Petri 网 $MCPN_k = (P, T; F, D, M)$, 如图 3 所示, 其中, $P = \{a, b, c\}, T = \{t_1, t_2\}, F = \{(a, t_1), (t_1, b), (b, t_3), (t_3, c), (c, t_2), (t_2, a)\}$, 若初始标识 $M_0 = (1, 0, 0)$, 则根据

算法 3, 可以得到可达标识为 $\{(1, 0, 0), (0, 1, 0), (0, 0, 1)\}$, 其可达标识图 $RG(MCPN)$ 如图 4 所示。由图 4 可知, 标识 $(1, 0, 0)$ 变化到标识 $(0, 1, 0)$ 时, 变迁 t_1 触发, 且库所 a 的托肯数由 1 变为 0, 库所 b 的托肯数由 0 变为 1。若初始标识 $M_0 = (1, 0, 1)$, 则根据算法 3, 可达标识为 $\{(1, 0, 1), (0, 1, 1), (1, 1, 0)\}$ 。若初始标识 $M_0 = (1, 1, 1)$, 则根据算法 2, 可达标识仅为 M_0 。

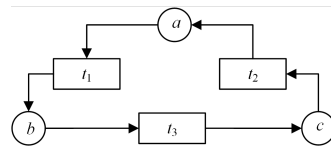


图 3 多元约束 Petri 网 $MCPN_k$

Fig. 3 Multi-constraint Petri net $MCPN_k$

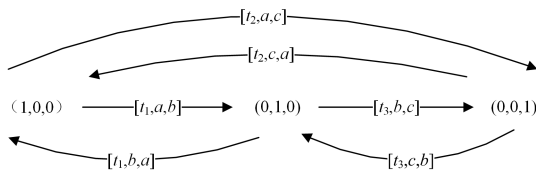


图 4 可达标识图 $RG(MCPN)$

Fig. 4 Reachable marking graph $RG(MCPN)$

3.2.3 测绘区域的 MCPN 模型构建

算法 4 给出了测绘区域的 MCPN 模型构建方法, 首先对测绘区域进行网格划分, 然后根据测绘网格属性值对多元约束 Petri 网的库所集、变迁集、弧集、多元属性集进行赋值, 实现了对测绘区域的 MCPN 模型构建。

算法 4 测绘区域的 MCPN 模型构建算法

输入: 测绘区域 $Q = (Id, D[n][2])$;

输出: 测绘区域 Q 的 MCPN 模型。

第 1 步 由算法 1 对测绘区域 Q 进行测绘网格划分, 并得到测绘网格集合 W 。

第 2 步 新建一个多元约束 Petri 网 $MCPN_k = (P, T; F, D, M)$, 且 $P = T = F = D = \emptyset$ 。遍历测绘网格集合 W , 假设当前元素为 $W_{gk} = (Id, d_1, d_2, d_3, d_4, s, zs, z, zx, ys, y, yx, x)$, 新建库所 p_i , 将 p_i 放入集合 $MCPN_k, P$ 中, 令 $D(p_i). d_1 = W_{gk}. d_1, D(p_i). d_2 = W_{gk}. d_2, D(p_i). d_3 = W_{gk}. d_3, D(p_i). d_4 = W_{gk}. d_4, D(p_i). Id = W_{gk}. Id$ 。

第 3 步 遍历 $MCPN_k, P$, 假设当前元素为 p_j , 遍历测绘网格集合 W , 假设当前元素为 W_{gi} , 若 $D(p_j). Id$ 与 $W_{gi}. Id$ 相同, 则令 $D(p_j). s = W_{gi}. s, D(p_j). zs = W_{gi}. zs, D(p_j). z = W_{gi}. z, D(p_j). zx = W_{gi}. zx, D(p_j). ys = W_{gi}. ys, D(p_j). y = W_{gi}. y, D(p_j). yx = W_{gi}. yx, D(p_j). x = W_{gi}. x$; 遍历 $MCPN_k, P$, 假设当前元素为 p_i , 若 $D(p_i). Id$ 等于 $s, zs, z, zx, ys, y, yx, x$ 中的一个, 则新建一个变迁 t_i , 将 t_i 放入 $MCPN_k, T$ 中; 新建有向弧 $f_i = (p_j, t_i), f_j = (p_i, t_i)$, 将 f_i 和 f_j 放入 $MCPN_k, F$ 中, 并删除相同元素。

第 4 步 输出 $MCPN_k$ 。

3.3 最优路径规划策略析出

本节给出测绘无人机最优作业路径析出方法, 其基本思想为: 首先, 构建测绘区域的 MCPN 模型, 并对测绘区域进行网格划分; 然后, 由无人机起降点确定 MCPN 的初始标识, 并生成可达标识图; 其次, 基于可达标识图获取全部无人机运行轨迹, 并对不符合无人机续航约束的轨迹进行删除; 最后, 析出航迹最短的路径规划方案。具体流程如算法 5 所示。

算法 5 最优路径析出算法

输入:测绘区域 Q_i , 无人机起降点经纬度坐标 (j, w) , 无人机持续飞行时间 T_{ci} , 无人机行进速度 V_i , 无人机返回到降落点时的最小剩余飞行时间 T_{ri} , 测绘网格宽度 K_{di} , 测绘网格长度 C_{di} ;

输出:路径规划策略。

第 1 步 由算法 4 构建测绘区域的 MCPN 模型, 假设为 $MCPN_k = (P, T; F, D, M)$ 。由算法 1 对测绘区域 Q 进行网格划分, 假设得到的测绘网格集合为 W 。新建一个网格变量 W_{gk} , 遍历集合 W , 若当前元素为 W_{gt} , 且 $W_{gt}.d_2 = (x_1, y_2)$, $W_{gt}.d_3 = (x_2, y_1)$; 若 $x_1 \leq j \leq x_2$, 且 $y_2 \leq w \leq y_1$, 则令 $W_{gk} = W_{gt}$ 。

第 2 步 令 $MCPN_k.M$ 为 0, 遍历 $MCPN_k.P$, 假设当前元素为 p_i , 若 $D(p_i).Id = W_{gk}.Id$, 则令 $M(p_i) = 1$ 。

第 3 步 令 $MCPN_k$ 的初始标识为 M , 由算法 3 对 $MCPN_k$ 进行可达性分析, 假设获取的可达标识图为 $RG(MCPN_k) = (R(M_0), E, W)$ 。新建一个标识变量 $M_k = M$, 新建队列变量 L_i 和队列集合 L 。

第 4 步 遍历可达图中标记为 M_k 的顶点的输出弧。假设当前弧为 $e_z = (M_k, M_j)$, 且 $W(M_k, M_j) = (t_a, p_m, p_n)$, 则新建队列变量 L_u , 令 $L_u = L_i$, 令 $M_k = M_k \parallel M_j$, 将 p_m, p_n 依次插入到队列 L_u 的队尾。若队列中相邻元素相同, 则删除相同元素, 仅保留一个。若队列 L_u 中存在两个相同的环路则退出。若 $\forall M_k(p_i) = 1, p_i \in MCPN_k.P$, 将队列 L_u 放入集合 L 中, 并删除相同项; 否则, 令 $M_k = M_j, L_i = L_u$, 返回第 4 步进行迭代。

第 5 步 遍历队列集合 L , 假设当前队列元素为 L_u 。从队首遍历队列 L_u , 假设当前元素为 p_m , 将当前元素替换为 $MCPN_k.D(p_m).Id$ 。

第 6 步 新建变量 v_1, v_2 , 遍历队列集合 L , 假设当前队列元素为 L_u 。假设队首元素为 Id_1 , 队尾元素为 Id_2 , 若 $Id_1 \neq Id_2$, 则遍历集合 W , 假设当前元素为 W_{gt} , 若 Id_2 等于 $W_{gt}.Id$, 则假设 $W_{gt}.d_1 = (x_2, y_2)$; 若 Id_1 等于 $W_{gk}.Id$, 假设 $W_{gk}.d_4 = (x_4, y_4)$; 若 $(x_4 = x_2) \& \& (y_2 > y_4)$, 则令 $v_1 = y_2, v_2 = y_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.x$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.x$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = y_6$ 。若 $(x_4 = x_2) \& \& (y_2 < y_4)$, 则令 $v_1 = y_2, v_2 = y_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.s$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.s$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = y_6$ 。若 $(y_4 = y_2) \& \& (x_2 < x_4)$, 则令 $v_1 = x_2, v_2 = x_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.y$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.y$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = x_6$ 。若 $(y_4 = y_2) \& \& (x_2 > x_4)$, 则令 $v_1 = x_2, v_2 = x_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.z$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.z$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = x_6$ 。若 $(y_4 < y_2) \& \& (x_2 < x_4)$, 且 $(y_2 - y_4)/K_{di}$ 小于 $(x_4 - x_2)/C_{di}$, 则令 $v_1 = y_2, v_2 = y_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.yx$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.yx$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = y_6$ 。令 $v_1 = x_2, v_2 = x_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.y$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.y$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = x_6$ 。若 $(y_4 < y_2) \& \& (x_2 < x_4)$, 且 $(y_2 - y_4)/K_{di}$ 大于 $(x_4 - x_2)/C_{di}$, 则令 $v_1 = x_2, v_2 = x_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.yx$ 插入到队列 L_u 的队尾, 遍历

集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.yx$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = x_6$ 。令 $v_1 = y_2, v_2 = y_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.x$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.x$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = y_6$ 。若 $(y_4 < y_2) \& \& (x_2 > x_4)$, 且 $(y_2 - y_4)/K_{di}$ 小于 $(x_2 - x_4)/C_{di}$, 则令 $v_1 = y_2, v_2 = y_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.zx$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.zx$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = y_6$ 。令 $v_1 = x_2, v_2 = x_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.z$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.z$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = x_6$ 。若 $(y_4 < y_2) \& \& (x_2 > x_4)$, 且 $(y_2 - y_4)/K_{di}$ 大于 $(x_4 - x_2)/C_{di}$, 则令 $v_1 = x_2, v_2 = x_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.zx$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.zx$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = x_6$ 。令 $v_1 = y_2, v_2 = y_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.x$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.x$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = y_6$ 。若 $(y_4 > y_2) \& \& (x_2 > x_4)$, 且 $(y_4 - y_2)/K_{di}$ 小于 $(x_2 - x_4)/C_{di}$, 则令 $v_1 = y_2, v_2 = y_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.zs$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.zs$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = y_6$ 。令 $v_1 = x_2, v_2 = x_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.z$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.z$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = x_6$ 。若 $(y_4 > y_2) \& \& (x_2 > x_4)$, 且 $(y_4 - y_2)/K_{di}$ 大于 $(x_2 - x_4)/C_{di}$, 则令 $v_1 = x_2, v_2 = x_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.zs$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.zs$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = x_6$ 。令 $v_1 = y_2, v_2 = y_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.s$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.s$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = y_6$ 。若 $(y_4 > y_2) \& \& (x_2 < x_4)$, 且 $(y_4 - y_2)/K_{di}$ 小于 $(x_4 - x_2)/C_{di}$, 则令 $v_1 = y_2, v_2 = y_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.ys$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.ys$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = y_6$ 。令 $v_1 = x_2, v_2 = x_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.y$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.y$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = x_6$ 。若 $(y_4 > y_2) \& \& (x_2 < x_4)$, 且 $(y_4 - y_2)/K_{di}$ 大于 $(x_4 - x_2)/C_{di}$, 则令 $v_1 = x_2, v_2 = x_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.ys$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.ys$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = x_6$ 。令 $v_1 = y_2, v_2 = y_4$, while $(v_1 \neq v_2)$, 将 $W_{gt}.s$ 插入到队列 L_u 的队尾, 遍历集合 W , 假设当前元素为 W_{gy} , 若 $W_{gt}.s$ 等于 $W_{gy}.Id$, 则假设 $W_{gy}.d_4 = (x_6, y_6)$; 则令 $W_{gt} = W_{gy}, v_1 = y_6$ 。

第 7 步 新建数字变量 $v_1 = 0$ 和网格变量 w_1 。遍历队列集合 L , 假设当前队列元素为 L_u , 队首元素为 Id_1 , 且测绘网格 $W_{gy}.Id = Id_1$, 令 $w_1 = W_{gy}$; 从队首遍历 L_u , 假设当前变量为 Id_i , 若 $Id_i = Id_1$, 则令 $v_1 = 0$; 遍历集合 W , 假设当前元素为 W_{gt} , 若 Id_i 等于 $W_{gt}.Id$, 且 $(W_{gt} = w_1.s) \parallel (W_{gt} = w_1.x)$, 则 $v_1 = v_1 + K_{di}$; 若 Id_i 等于 $W_{gt}.Id$, 且 $(W_{gt} = w_1.z) \parallel (W_{gt} = w_1.y)$, 则 $v_1 = v_1 + C_{di}$; 若 Id_i 等于 $W_{gt}.Id$, 且 $(W_{gt} = w_1.zs) \parallel (W_{gt} =$

$w_1, ys) \parallel (W_{g_t} = w_1, zx) \parallel (W_{g_t} = w_1, yx)$, 则假设 $K_{d_i}^2 + C_{d_i}^2$ 的平方根为 $kc, v_1 = v_1 + kc$; 令 $w_1 = W_{g_t}$, 若 $(T_{c_i} - T_{f_i}) * V_i < v_1$, 则从集合 L 中删除 L_u 。

第 8 步 新建队列变量 L_{out} , 变量 $v_1 = v_2 = +\infty$; 遍历队列集合 L , 假设当前队列元素为 L_u ; 从队首遍历 L_u , 假设当前变量为 Id_i ; 遍历集合 W , 假设当前元素为 W_{g_t} , 若 Id_i 等于 $W_{g_t}. Id$, 且 $(W_{g_t} = w_1, s) \parallel (W_{g_t} = w_1, x)$, 则 $v_1 = v_1 + K_{d_i}$; 若 Id_i 等于 $W_{g_t}. Id$, 且 $(W_{g_t} = w_1, z) \parallel (W_{g_t} = w_1, y)$, 则 $v_1 = v_1 + C_{d_i}$; 若 Id_i 等于 $W_{g_t}. Id$, 且 $(W_{g_t} = w_1, zs) \parallel (W_{g_t} = w_1, ys) \parallel (W_{g_t} = w_1, zx) \parallel (W_{g_t} = w_1, yx)$, 则假设 $K_{d_i}^2 + C_{d_i}^2$ 的平方根为 $kc, v_1 = v_1 + kc$; 令 $w_1 = W_{g_t}$ 。若 $v_2 > v_1$, 则令 $v_2 > v_1, L_{out} = L_u$ 。

第 9 步 输出队列 L_{out} 。

算法 5 给出了测绘无人机最优路径析出算法, 算法输出为队列 L_{out} , 队列中的元素依次为无人机从起飞点开始飞行经过的测绘网格 Id 号, 无人机在经过起降点时可以进行补给。

4 实验结果与分析

为了验证本文所述的水利测绘无人机路径规划方法的有效性及其优越性, 本节在 PC 机上搭建仿真平台进行实验。实验硬件环境: 处理器为 Intel(R) Core(TM) i5-6500, CPU 主频 3.2 GHz, 内存为 8GB; 软件环境: Windows 7 Service Pack 1 操作系统, Microsoft Visual Studio 2015 编程软件; 实验平台为 C# 语言构建的 WinForm 应用程序。

实验流程设计为: 1) 构建测绘区域。在仿真平台用户界面中输入离散点个数, 然后依次输入离散点坐标, 点击生成测绘区域按钮后, 可实现测绘区域的构建。此功能实现思想是基于定义 1, 如用户输入的 4 个离散点分别为 $(0, 0), (0, 100), (100, 0), (100, 100)$, 则利用 C# 语言定义一个二维数组 $D[4][2]$, 将离散点信息赋值给数组生成测绘区域。2) 划分测绘网格。用户在平台界面中输入测绘网格的长度和宽度, 点击划分测绘网络按钮后, 可实现对测绘区域的网格划分。此功能实现思想是基于算法 1, 并依据定义 2 对测绘网格数据结构进行了 C# 程序定义, 包括测绘网络的 Id 号、4 个顶点坐标, 以及上、左上、左、左下、右上、右、右下、下面相邻的测绘网格的 Id 号。3) 用户在平台界面中点击构建 MCPN 模型按钮, 系统根据算法 4 生成测绘区域的形式化模型。4) 用户在平台界面中输入无人机起降点坐标、飞行速度、持续飞行时间、最小剩余飞行时间后, 点击输出最优路径按钮, 系统基于算法 5 析出最优路径规划方案, 并在文本框中显示。仿真平台界面如图 5 所示。



图 5 仿真平台界面
Fig. 5 Simulation platform interface

图 6 给出了一个简单的实例, 从中可知实验设置离散点数为 6, 离散点坐标为 $(6, 0), (1, 9), (0, 4), (6, 12), (12, 4), (11, 9)$, 网格长度为 2, 网格宽度为 2, 起降点坐标为 $(5, 1)$, 飞行速度为 2, 持续飞行时间为 25, 最小剩余时间为 3。依次点击生成测绘区域、测绘网格, 构建 MCPN 模型, 生成最优路径按钮后, 得到无人机路径规划方案。无人机依次经过的坐标点为 $(5, 1), (5, 3), (5, 5), (5, 7), (5, 9), (5, 11), (3, 11), (3, 9), (3, 7), (3, 5), (3, 3), (3, 1), (1, 3), (1, 5), (1, 7), (1, 9), (3, 7), (5, 5), (5, 3), (5, 1), (7, 1), (7, 3), (7, 5), (7, 7), (7, 9), (7, 11), (9, 11), (9, 9), (9, 7), (9, 5), (9, 3), (9, 1), (11, 3), (11, 5), (11, 7), (11, 9), (9, 7), (7, 5), (5, 3), (5, 1)$ 。从程序后台可以获得网格划分情况, 如图 2 所示, 无人机轨迹依次经过的网格为 11, 12, 13, 14, 15, 16, 10, 9, 8, 7, 6, 5, 1, 2, 3, 4, 8, 13, 12, 11, 17, 18, 19, 20, 21, 22, 28, 27, 26, 25, 24, 23, 29, 30, 31, 32, 26, 19, 12, 11。



图 6 仿真实例
Fig. 6 Simulation example

仿真实验目标是: 通过实验验证本文所述方法是否能够在仿真场景中找到最优路径规划方案, 并对寻优效率及效果进行比较分析。本文主要进行两项测试: 1) 在同一个测绘区域内变更测绘网格大小, 并保持持续飞行时间不变; 2) 在同一个测绘区域内变更持续飞行时间, 并保持测绘网格大小不变。实验设置离散点数为 4 个, 离散点坐标分别为 $(0, 0), (0, 600), (600, 0), (600, 600)$ 。实验结果如图 7 所示。由图 7 可知, 本文在上述两项测试中均能完成路径规划寻优, 且在同一个测绘区域内保持测绘网格不变, 无人机持续飞行时间越长, 算法规划时间越短; 保持无人机持续飞行时间不变, 测绘网格面积越大, 算法规划时间越短。

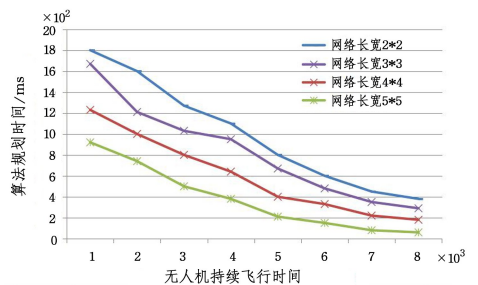


图 7 实验结果
Fig. 7 Experimental results

文献[9]、文献[15]、文献[17]给出了无人机全覆盖路径规划方法, 本文方法与上述文献技术相比具有如下优势: 1) 考虑了无人机续航时间, 即无人机动力有限, 在动力低时需要进行动力补充以免发生安全事故, 即本文方法接近应用实际,

方法适用性更强;2)根据无人机测绘相机单幅照片覆盖区域范围定义了测绘网格,在保证测绘精度的前提下,缩小了数据处理基数,提高了算法效率;3)基于可达标识图,面向全部航迹情况进行路径寻优,能够析出路径最短方案。在无人机续航时间无穷大时,方法比较结果如图 8 所示。由图 8 可知,本文方法在算法效率和航迹长度上具有优势,且保障了测绘无人机飞行安全。

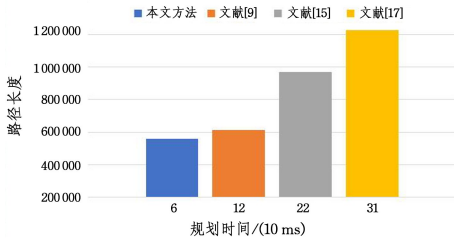


图 8 效果比较

Fig. 8 Effect comparison

结束语 鉴于水利无人机测绘作业具有智能路径规划需求,本文提出了一种基于多元约束 Petri 网的水利测绘无人机路径规划方法。对测绘区域进行了网格划分,减小了无人机路径规划算法的数据处理基数,提高了计算效率;提出了多元约束 Petri 网,给出了测绘区域形式化建模方法;基于可达性分析,获取了模型可达标识图,并给出了无人机测绘作业最优路径析出算法。实验证明,与相关文献方法相比,本文方法考虑了无人机续航时间,能够在保证飞行安全的前提下,基于起降点找到一条覆盖测绘区域的飞行路径;进行了网格划分,优化了算法效率。下一步将对基于网格的测绘无人机图像处理与拼接技术进行研究和探索。

参 考 文 献

[1] YU D X. Exploring the application of tilt photography in hydraulic engineering mapping [J]. *Pearl River Water Transport*, 2021(23):103-104.

[2] MAO J C, ZHAO S J. Research on the application of low-altitude remote sensing in hydraulic engineering mapping [J]. *Architectural Technology Development*, 2020, 47(19): 85-86.

[3] TIAN Z, MEGHDAD H S, AYMAN H. Tightly-coupled camera/LiDAR integration for point cloud generation from GNSS/INS-assisted UAV mapping systems [J]. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2021, 180(4): 336-356.

[4] LIU C T, GUO, Y, LI N, et al. Multiuser Oriented Multi-UAV Mission Assignment With Cooperative Information Sharing [J]. *IEEE Wireless Communications Letters*, 2021, 10(4): 907-911.

[5] WANG J F, JIA G W, LIN J C, et al. Cooperative task allocation for heterogeneous multi-UAV using multi-objective optimization algorithm [J]. *Journal of Central South University*, 2020, 27(2): 432-448.

[6] MA Y H, ZHAO Y F, BAI S Y, et al. Collaborative task allocation of heterogeneous multi-UAV based on improved CBGA algorithm [C] // 16th IEEE International Conference on Control, Automation, Robotics and Vision (ICARCV 2020). 2020: 795-800.

[7] XU L F, YANG Z Z, HUANG Z S, et al. Route design method

of plant protection UAV combined with hybrid particle swarm optimization algorithm [J]. *Small microcomputer system*, 2020, 41(9): 1826-1832.

- [8] GEBREHIWOT A A, BENI L H. Three-Dimensional Inundation Mapping Using UAV Image Segmentation and Digital Surface Model [J]. *ISPRS International Journal of Geo-Information*, 2021, 10(3): 139-144.
- [9] XUE Z T, CHEN J, ZHANG Z C, et al. Multi-UAV coverage path planning based on optimization of convex division of complex plots [J]. *Acta Aeronautica et Astronautica Sinica*, 2022, 43(X): 325990.
- [10] LIAO C C. Research on multi-uav cooperative mission planning [D]. Chengdu: Sichuan University, 2021.
- [11] DENG M, CHEN Z. Coordinated path planning for multiple uavs based on k-degree smoothing [J]. *Computer Engineering and Design*, 2021, 42(8): 2387-2394.
- [12] HE J R, HE G J, YU X S. The UAV path planning based on improved artificial bee colony algorithm [J]. *Fire Control & Command Control*, 2021, 46(10): 103-106.
- [13] GUO Q C, DU X Y, ZHANG Y Y, et al. Three-dimensional path planning for UAV based on improved whale algorithm [J]. *Computer Science*, 2021, 48(12): 304-311.
- [14] PANDEY P, SHUKLA A, TIWARI R. Three-dimensional path planning for unmanned aerial vehicles using glowworm swarm optimization algorithm [J]. *International Journal of System Assurance Engineering and Management*, 2018, 9(4): 836-852.
- [15] NIELSEN L D, SUNG I, NIELSEN P. Convex decomposition for a coverage path planning for autonomous vehicles; interior extension of edges [J]. *Sensors*, 2019, 19(19): 4165-4716.
- [16] DEWANGAN R K, SHUKLA A, GODFREY W W. Three dimensional path planning using Grey wolf optimizer for UAVs [J]. *Applied Intelligence*, 2019, 49(6): 2201-2217.
- [17] HASSAN M, LIU D K. PPCPP: a predator-prey-based approach to adaptive coverage path planning [J]. *IEEE Transactions on Robotics*, 2020, 36(1): 284-301.
- [18] ZHANG S C. Path planning for multiple uavs based on reinforcement learning [D]. Chengdu: Sichuan University, 2021.
- [19] YAO C P, XU J, LI X Y. Terrain monitoring of unmanned aerial vehicle formation [J]. *Science and Technology Information in China*, 2020(22): 68-69.
- [20] DU Y Y, NING Y H, LIANG Q. Reachability analysis of Logic Petri Nets using incidence matrix [J]. *Enterprise Information Systems*, 2014, 8(6): 630-647.
- [21] DU Y Y, YNING H. Property analysis of Logic Petri Nets using reachable marking graphs [J]. *Frontiers of Computer Science in China*, 2014, 8(4): 684-692.



YAO Xi, born in 1984, postgraduate, senior engineer. Her main research interests include data processing, equipment design of mapping UAV, photogrammetry and remote sensing.