

Fast Searching on Convex Deltahedra

Xiaodong Han

College of Math, Physics and Information Technology, Zhejiang Normal University, Jinhua Zhejiang
Email: 1328447839@qq.com

Received: Jul. 27th, 2018; accepted: Aug. 9th, 2018; published: Aug. 16th, 2018

Abstract

In this paper, we study fast searching graph searching model, which differs from the classical edge searching in one way: every edge is traversed exactly once and searchers are not allowed to jump. The graph of this paper is convex deltahedra. We will examine the fast search number (*i.e.*, the minimum number of searchers required for capturing the fugitive) of all the convex deltahedra. We provide an explicit formula for the fast search number of convex deltahedra.

Keywords

Fast Searching, Graph Searching, Pursuit-Evasion Strategy, Cops and Robber Game

凸三角多面体的快速搜索

韩小东

浙江师范大学数理与信息工程学院, 浙江 金华
Email: 1328447839@qq.com

收稿日期: 2018年7月27日; 录用日期: 2018年8月9日; 发布日期: 2018年8月16日

摘要

本文研究的图搜索模型是快速搜索, 该模型与边搜索的区别仅是: 搜索者不能“跳跃”且每条边只能被访问一次。本文研究的图是凸三角多面体, 求解所有凸三角多面体的快速搜索数, 提出关于凸三角多面体快速搜索数的相关推论。

关键词

快速搜索, 图搜索, 追逃对策, 警察和小偷博弈



1. 引言

上世纪六七十年代 Breisch [1]与 Parsons [2]研究组建一支搜救队在洞穴里寻找丢失人员所需要的勘探人员的最少人数,称之为洞穴问题。Kirousis 等在 1986 年提出了节点搜索[3]。Megiddo 等在 1988 年提出了离散型的洞穴问题,称之为边搜索[4],其核心问题是:在图上成功抓获入侵者最少需要多少个搜索者。节点搜索和边搜索是搜索博弈的两个基本模型。

本文研究的快速搜索是 Dyer 和 Yang 在 2008 年提出的一个改版的边搜索[5]。每条边只允许被搜索者访问一次且搜索者不允许有“跳跃”动作的单调的搜索博弈模型。数个搜索者在图上搜索一个自由且不可见的入侵者,即搜索者不知道入侵者的位置,入侵者却掌握搜索者的位置移动等信息。入侵者藏在要么顶点要么边上,并且可以在任一时间从当前顶点 u 沿着一条不含有搜索者的路径以非常快的速度移动到顶点 v 。当搜索者和入侵者占据同一个顶点时,入侵者被搜索者抓捕。搜索者的目的是成功抓捕入侵者,入侵者的目的是避免被抓捕并且总是选择最优策略进行逃脱。

在快速搜索中,搜索者可以有以下两种动作:

- 1) 放置: 某一搜索者被放置到图上的一个顶点;
- 2) 移动: 搜索者从当前顶点沿着一条未被清理过的邻边移动到邻点。

满足以下两个条件之一,搜索者就可以从顶点 u 沿邻边 uv 移动到邻点 v :

- 1) 如果搜索者从顶点 u 移动到顶点 v 后,顶点 u 上至少还有一个搜索者;
- 2) 除边 uv 以外,顶点 u 的所有邻边都已被清理。

如果一条边 uv 可能藏有入侵者,则称该边是被污染的。如果可以确定一条边不藏有入侵者,则称该边是干净的。在某一时刻,如果图 G 的某一个顶点上至少有一个搜索者,则称该点被搜索者守卫。每条边通过搜索者的移动动作被清理。在博弈开始时,图上的每一条边都是被污染的,我们的目的是放置最少的搜索者来一步一步的清理该图的每一条边。

在快速搜索中,清理图 G 所需的最少搜索者的数量称为图 G 的快速搜索数,记为 $f_s(G)$ 。一个快速搜索策略是由满足最后一个动作结束整个图 G 被清理的一系列移动动作所组成的集合。以 $f_s(G)$ 个搜索者清理图 G 的快速搜索策略被称为最优快速搜索策略。

单调性是搜索博弈的一个核心问题。一条边被搜索者清理后保持干净状态直至整个搜索策略结束,称这样的搜索策略具有单调性。以往的研究已证明边搜索、节点搜索和混合搜索具有单调性[6] [7],连通边搜索不具有单调性[8]。

Dyer 和 Yang 设计了一个线性时间复杂度的图搜索算法用来计算树的快速搜索数,并对二部图的快速搜索数做了相关研究[5]。2011 年 Stanley 和 Yang 提出了一个快速搜索数的下界,并设计了计算立方体图和哈林图等的快速搜索数的图搜索算法[9]。2013 年 Yang 等撰写了关于图搜索及其相关问题的综合性研究论文[10]。2016 年 Yang 等得出了完全 k 部图的快速搜索数的一个上界与下界,并对完全二部图和完全分割图这两个特殊的完全 k 部图做了详细的研究[11]。2017 年 Xue 和 Yang 提出了快速搜索的一个新下界,并提出了欧拉图和路径的笛卡尔积图的快速搜索数[12]。在两个基本模型的基础上,研究人员还提出了混合搜索、快速边搜索等搜索博弈的模型[13] [14] [15] [16] [17]。

图搜索在计算机科学、数学、物理领域引起了研究人员的浓厚兴趣。随着网络信息安全越来越得到

重视,图搜索在网络安全、计算机游戏设计等领域得到了广泛的应用[18][19]。经过多年的演进,在人工智能的很多分支领域都有所涉及图搜索[20][21]。

本文研究在凸三角多面体上的快速搜索,目标求出凸三角多面体的快速搜索数。本文结构如下:第二部分给出相关的概念,第三部分求解凸三角多面体的快速搜索数并得出相关推论,第四部分进行全文总结。

2. 概念

本文分别用本文分别用 $V(G)$ 、 $E(G)$ 表示图 $G=(V,E)$ 的顶点集和边集,文中仅考虑简单连通图,即无环和无重边的图。

我们用 uv 表示连接顶点 u 和 v 的边,如果 $uv \in E(G)$,则称 u 和 v 是相邻的。对于图 G 中的任一顶点 v 的度是指 v 的邻边个数,记为 $\deg_G(v)$ 。如果一个顶点的度为奇数,称该顶点为奇点。类似的,当一个顶点的度数为偶数,称该顶点为偶点。当一个图的顶点都是奇点,称该图为奇图。对于任一点 $v \in V(G)$,集合 $\{u:uv \in E(G)\}$ 表示点 v 的所有邻点,记为 $N_G(v)$ 。 $\delta(G) = \min(|N_G(v)|:v \in V(G))$,表示图 G 中最小的顶点度数。

三角多面体是指每一个面都是等边三角形的多面体。在凸三角多面体中,正四面体、正八面体和正二十面体是正则多面体。其余图属于约翰逊固体,一些顶点度数为 3,一些顶点度数为 4,一些顶点度数为 5。值得注意的是,不存在 18 个面的凸三角多面体。

3. 凸三角多面体的快速搜索

首先求解所有凸三角多面体的快速搜索数,接着提出相关推论。

2011 年 Stanley 和 Yang [6] 提出如下推论和定理:

推论 1: 对于一个没有叶子节点的图 $G=(V,E)$, 有 $fs(G) \geq \frac{1}{2}|V_{odd}| + 2$ 。

定理 1: 对于一个连通立体图 $G=(V,E)$, 设 τ 是图 G 的切点块图的叶子节块的数量, 如果图 G 不包含切点, 则 $\tau=2$, 那么 $fs(G) = \frac{|V|}{2} + \tau$ 。

2011 年 Yang [13] 提出如下定理:

定理 2: 对于一个满足 $\delta(G) \geq 3$ 的连通图 G , 有 $fs(G) \geq \max \left\{ \delta(G)+1, \left\lceil \frac{\delta(G)+|V_{odd}(G)|-1}{2} \right\rceil \right\}$

引理 1: 对于正四面体 C_4 , 有 $fs(C_4) = 4$ 。

证: 正四面体是一个连通立体图,显然不含有切点,又已知 $|V(C_4)| = 4$, 根据定理 1 得: $fs(C_4) = 4$ (图 1)。

引理 2: 对于双三角锥 C_6 , 有 $fs(C_6) = 4$ 。

证: 首先提出一个由 4 个搜索者组成的快速搜索策略:

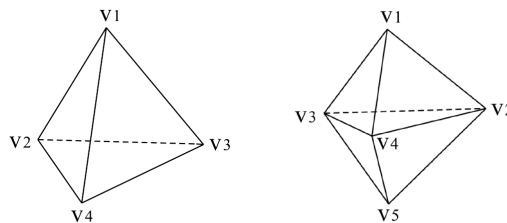


Figure 1. Tetrahedron, triangular bipyramid

图 1. 完全正四面体 C_4 、双三角锥 C_6

- 1) 我们把搜索者 λ_1 、 λ_2 、 λ_3 均放置到顶点 v_1 ，把 λ_4 放置到顶点 v_2 ；
- 2) λ_1 、 λ_2 、 λ_3 分别沿边 v_1v_2 、 v_1v_3 、 v_1v_4 移动，使得 v_2 、 v_3 、 v_4 分别被 1 个搜索者守卫；
- 3) λ_4 从 v_2 沿环 $v_2v_3v_4v_2$ 清理回到 v_2 ；
- 4) λ_1 、 λ_2 、 λ_3 分别沿 v_2v_5 、 v_3v_5 、 v_4v_5 移动到 v_5 ，完成对图 C_6 的清理。即 $fs(C_6) \leq 4$ 。

其次已知 $\delta(C_6) = 3$ ， $|V_{odd}(C_6)| = 2$ ，根据定理 2 得： $fs(C_6) = 4$ (图 1)。

引理 3: 对于正八面体 C_8 ，有 $fs(C_8) = 5$ 。

证：首先提出一个由 5 个搜索者组成的快速搜索策略：

- 1) 我们把搜索者 λ_1 、 λ_2 、 λ_3 、 λ_4 均放置到顶点 v_1 ，把 λ_5 放置到顶点 v_2 ；
- 2) λ_1 、 λ_2 、 λ_3 、 λ_4 分别沿边 v_1v_2 、 v_1v_3 、 v_1v_4 、 v_1v_5 移动，使得 v_2 、 v_3 、 v_4 、 v_5 分别被 1 个搜索者守卫；
- 3) λ_4 从 v_2 沿环 $v_2v_3v_4v_5v_2$ 清理回到 v_2 ；
- 4) λ_1 、 λ_2 、 λ_3 、 λ_4 分别沿 v_2v_6 、 v_3v_6 、 v_4v_6 、 v_5v_6 移动到 v_6 ，完成对图 C_8 的清理。即 $fs(C_8) \leq 5$ 。

其次已知 $\delta(C_8) = 4$ ， $|V_{odd}(C_8)| = 0$ ，根据定理 2 得： $fs(C_8) = 5$ (图 2)。

引理 4: 对于双五角锥 C_{10} ，有 $fs(C_{10}) = 5$ 。

证：首先提出一个由 5 个搜索者组成的快速搜索策略清理图 C_{10} ：

- 1) 我们把搜索者 λ_1 、 λ_2 、 λ_3 、 λ_4 均放置到顶点 v_2 ，把 λ_5 放置到顶点 v_6 ；
- 2) λ_1 、 λ_2 、 λ_3 、 λ_4 分别沿边 v_2v_1 、 v_2v_3 、 v_2v_5 、 v_2v_6 移动，使得 v_1 、 v_3 、 v_5 、 v_6 分别被至少 1 个搜索者守卫， λ_5 从 v_6 沿环 $v_6v_4v_3v_7v_6$ 清理回到 v_6 ；
- 3) λ_1 、 λ_3 分别沿 v_1v_4 、 v_3v_7 移动，使得 v_4 、 v_7 被一个搜索者守卫；
- 4) λ_5 从 v_6 沿环 $v_6v_4v_3v_7v_6$ 清理回到 v_6 ， v_6 上的 λ_5 清理 v_7v_4 ，完成对图 C_8 的清理。即 $fs(C_{10}) \leq 5$ 。

其次已知 $\delta(C_{10}) = 4$ ， $|V_{odd}(C_{10})| = 2$ ，根据定理 2 得： $fs(C_{10}) = 5$ (图 2)。

引理 5: 对于变棱双五角锥 C_{12} ，有 $fs(C_{12}) = 5$ 。

证：首先提出一个由 5 个搜索者组成的快速搜索策略：

- 1) 把搜索者 λ_1 、 λ_2 、 λ_3 、 λ_4 均放置到顶点 v_2 ，把 λ_5 放置到顶点 v_4 ；
- 2) λ_1 、 λ_2 、 λ_3 、 λ_4 分别沿边 v_2v_1 、 v_2v_3 、 v_2v_4 、 v_2v_5 移动，使得 v_1 、 v_3 、 v_4 、 v_5 分别至少被 1 个搜索者守卫， λ_5 从 v_4 沿环 $v_4v_5v_3v_1v_4$ 清理回到 v_4 ；
- 3) λ_1 、 λ_5 分别沿 v_1v_6 、 v_4v_6 移动(此时 v_6 被 2 个搜索者守卫)；
- 4) λ_4 沿 v_4v_7 移动到顶点 v_7 ， v_6 上的 λ_5 沿路径 $v_6v_7v_5v_8$ 移动到 v_8 ；
- 5) λ_4 沿 v_7v_8 移动到 v_8 ，然后 v_8 上的 λ_5 沿 v_8v_3 移动到 v_3 ；
- 6) v_3 上的 λ_5 清理 v_3v_6 ，完成对图 C_{12} 的清理。即 $fs(C_{12}) \leq 5$ 。

其次已知 $\delta(C_{12}) = 4$ ， $|V_{odd}(C_{12})| = 4$ ，根据定理 2 得： $fs(C_{12}) = 5$ (图 3)。

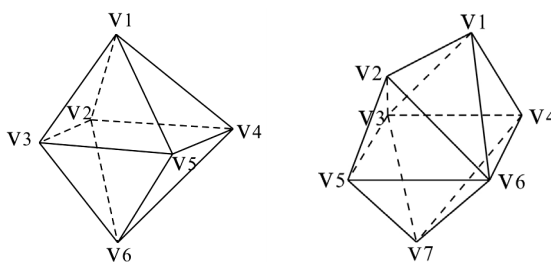


Figure 2. Octahedron, pentagonal bipyramid

图 2. 完全正八面体 C_8 、双五角锥 C_{10}

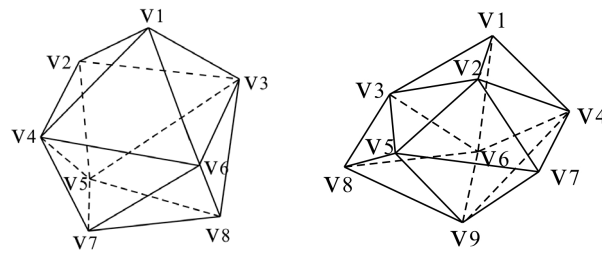


Figure 3. Snub disphenoid, triaugmented triangular prism
 图 3. 变棱双五角锥 C_{12} 、三侧锥三角柱 C_{14}

引理 6: 对于三侧锥三角柱 C_{14} , 有 $fs(C_{14})=6$ 。

证: 三侧锥三角柱的顶点布局为: 3×3^4 、 6×3^5 。根据搜索者放置的初始顶点的度数, 我们分为两种情况:

Case 1: 初始顶点的度数为 4, 不失一般性, 我们选取 v_1 作为初始顶点。

- 1) 搜索者 λ_1 、 λ_2 、 λ_3 、 λ_4 均放置到顶点 v_1 ;
- 2) λ_1 、 λ_2 、 λ_3 、 λ_4 从 v_1 沿邻边进行清理, 移动到我 4 个邻点 v_2 、 v_3 、 v_4 、 v_6 (此时这 4 个搜索者不满足继续移动清理图的条件而只能守卫);
- 3) 不失一般性, 放置 λ_5 到 v_3 ;
- 4) λ_5 沿环 $v_3v_2v_6v_4v_3$ 清理回到 v_3 (此时 v_3 有两条邻边未被清理且有 2 个搜索者 λ_2 、 λ_5);
- 5) λ_2 、 λ_5 从 v_3 分别沿邻边进行清理, 移动到 v_8 、 v_5 ;

此时 5 个搜索者所在的顶点上只有 1 个搜索者且其未被清理的邻边不止一条, 所以这 5 个搜索者不满足继续移动清理图的条件。

Case 2: 初始顶点的度数为 5, 不失一般性, 我们选取 v_2 作为初始顶点。

- 1) 放置 λ_1 、 λ_2 、 λ_3 、 λ_4 、 λ_5 到 v_2 , 然后 λ_1 、 λ_2 、 λ_3 、 λ_4 、 λ_5 分别沿邻边进行清理, 移动到其 5 个邻点 λ_1 、 λ_3 、 λ_4 、 λ_5 、 λ_7 , 此时 5 个搜索者所在的顶点上只有 1 个搜索者且其未被清理的邻边不止一条, 因此这 5 个搜索者不满足继续移动清理图的条件。

由上面分析可以得出: 5 个搜索者不足以成功地清理图 C_{14} , 即 $fs(C_{14}) > 5$ 。

其次需要证明 $fs(C_{14}) \leq 6$ 。接着 Case 1 的第 5 步:

- 2) 我们再放置第 6 个搜索者 λ_6 到 v_6 , 然后 λ_6 从 v_6 沿路径 $v_6v_8v_5v_2v_7$ 清理移动到 v_7 ;
- 3) λ_4 从 v_6 沿邻边 v_6v_9 移动到 v_9 , 然后 λ_2 从 v_8 沿路径 $v_8v_9v_4v_7v_5v_9v_7$ 清理, 完成对图 C_{14} 的清理, 即 $fs(C_{14}) \leq 6$ 。综上得证: $fs(C_{14})=6$ (图 3)。

引理 7: 对于双四角锥反角柱 C_{16} , 有 $fs(C_{16})=7$ 。

证: 首先由于双四角锥反角柱的对称性, 只要证明必要的几何元素具有某种性质, 就足以概括其余, 这是显然的。双四角锥反角柱的顶点布局为: 2×3^4 、 8×3^5 。根据搜索者放置的初始顶点的度数, 我们分为两种情况:

Case 1: 初始顶点的度数为 5, 不失一般性, 我们选取顶点 v_3 作为初始顶点。

- 1) 放置 λ_1 、 λ_2 、 λ_3 、 λ_4 、 λ_5 到 v_3 ;
- 2) λ_1 、 λ_2 、 λ_3 、 λ_4 、 λ_5 从 v_3 分别沿邻边进行清理, 到我 5 个邻点 v_1 、 v_2 、 v_6 、 v_9 、 v_4 (此时这 5 个搜索者不满足继续移动清理图的条件而只能守卫);
- 3) 放置 λ_6 到顶点度数为 4 的顶点 v_1 的邻点 v_2 处(此时 v_2 上有 2 个搜索者 λ_2 、 λ_6);
- 4) λ_6 沿环 $v_2v_6v_9v_4v_1v_2$ 清理回到 v_2 (此时 v_1 有一条邻边未被清理且有 1 个搜索者 λ_1);

5) λ_1 沿着邻边 v_1v_5 到 v_5 , λ_2 沿邻边 v_2v_5 到 v_5 (此时 v_2 有一条邻边未被清理且有 1 个搜索者 λ_6 , v_5 上有 2 个搜索者 λ_1 、 λ_2);

6) λ_6 沿邻边 v_2v_7 到 v_7 ;

由于 v_5 上有 2 个搜索者, 移动其中 1 个。此时根据 v_5 的邻边未被清理的边的个数, 可以分为以下两种情形:

(A) 选择有两条邻边未被清理的顶点

a) λ_2 沿 v_3v_4 移动到 v_4 (此时 v_4 上有 2 个搜索者 λ_2 、 λ_5 , 且只有一条邻边 v_4v_8 未被清理);

b) λ_2 沿 v_4v_8 移动到 v_8 ;

此时清理情况: λ_1 、 λ_2 、 λ_3 、 λ_4 及其所有邻边已经被清理; 顶点 v_5 、 v_6 、 v_9 有两条邻边未被清理(各点上 1 个搜索者); v_7 、 v_8 、 v_{10} 有两条邻边未被清理(各点上 1 个搜索者), 此时不满足继续移动清理图的条件;

(B) 选择有三条邻边未被清理的顶点

a) λ_2 沿 v_3v_7 移动到 v_7 , 此时 v_7 上有 2 个搜索者 λ_2 、 λ_6 , 可以移动其中 1 个; 但是有三条邻边 v_7v_6 、 v_7v_8 、 v_7v_{10} 未被清理, 此时不满足继续移动清理图的条件;

b) 如果 λ_6 沿边 v_7v_8 移动到 v_8 , 由于此时 v_8 上 1 个搜索者且四条邻边, 此时不满足继续移动清理图的条件;

c) 如果 λ_6 沿边 v_7v_{10} 移动到 v_{10} , 由于此时 v_{10} 上 1 个搜索者且四条邻边, 此时不满足继续移动清理图的条件;

i) 如果 λ_6 沿 v_7v_6 移动到 v_6 (此时 v_6 有 2 个搜索者 λ_3 、 λ_6 , 可以移动其中 1 个; 且只有一条邻边 v_6v_{10} 未被清理);

ii) λ_3 沿 v_6v_{10} 移动到 v_{10} ;

此时清理情况: v_1 、 v_2 、 v_3 、 v_6 及其所有邻边已经被清理(v_6 上 1 个搜索者); v_4 、 v_5 、 v_7 、 v_9 有两条邻边未被清理(各顶点上有 1 个搜索者); v_8 有五条邻边未被清理(0 个搜索者); v_{10} 有三条邻边未被清理(1 个搜索者), 此时不满足继续移动清理图的条件;

Case 2: 初始顶点的度数为 4, 不失一般性, 我们选取顶点 v_1 作为初始顶点。

1) 放置 λ_1 、 λ_2 、 λ_3 、 λ_4 到 v_1 ; 然后 λ_1 、 λ_2 、 λ_3 、 λ_4 从 v_1 分别沿邻边进行清理, 到其四个邻点 v_2 、 v_3 、 v_4 、 v_5 , 此时这 4 个搜索者只能守卫而不能移动;

2) 不失一般性, 为了使清理继续下去, 放置 λ_5 到顶点 v_2 ;

3) λ_5 沿环 $v_2v_3v_4v_5v_2$ 清理回到 v_2 (此时 v_2 上上有 2 个搜索者 λ_1 、 λ_5 且有两条未被清理的边);

4) v_2 上的 λ_1 、 λ_5 沿邻边 v_2v_7 、 v_2v_6 到移动 v_7 、 v_6 ;

此时清理情况: v_1 、 v_2 及其所有邻边已经被清理; v_3 、 v_4 、 v_5 有两条邻边未被清理(各点上 1 个搜索者); v_6 、 v_7 有三条邻边未被清理(各点上 1 个搜索者); v_8 、 v_9 、 v_{10} 及其所有邻边已经被清理;

显然此时这 5 个搜索者只能守卫而不能移动;

5) 为了使清理继续下去, 我们需要放置第 6 个搜索者 λ_6 到图中, 根据放置顶点处, 未被清理边的个数, 可以分为以下 3 种情形:

(A) 不失一般性, 放置 λ_6 到 v_5 (此时 v_5 上有 2 个搜索者 λ_4 、 λ_6);

a) λ_4 、 λ_6 移动到其邻点 v_7 、 v_8 (此时 v_7 上有 2 个搜索者 λ_1 、 λ_4);

b) 移动 λ_4 到 v_{10} , 此时 v_{10} 上只有 1 个搜索者, 清理不能继续下去;

c) 移动 λ_4 到 v_8 , 此时 v_8 上有 2 个搜索者 λ_4 、 λ_6 , 移动 λ_6 到其一个邻点 v_4 , 再移动 λ_6 到 v_4 一个邻

点 v_9 ，此时 v_3 、 v_6 、 v_7 、 v_8 、 v_9 上有且只有 1 个搜索者且不止一条未被清理的邻边；显然此时这 6 个搜索者只能守卫而不能移动，清理不能继续下去；

(B) 放置 λ_6 到有四条邻边未被清理的顶点上；类上同证！

(C) 放置 λ_6 到有五条邻边未被清理的顶点上；类上同证！

由上面分析可以得出：6 个搜索者不足以成功清理图 C_{16} ，即 $fs(C_{16}) > 6$ 。

其次需要证明 $fs(C_{16}) \leq 7$ 。接着 Case 1(a) 的第 2 步结束后，只需要在 v_7 处放置 1 个搜索者 λ_7 ；然后 λ_7 移动到 v_8 ； v_7 上的 λ_6 沿着路径 $v_7v_6v_{10}v_9v_8v_4v_3v_5v_8v_{10}$ 移动到 v_{10} ，即完成对图 C_{16} 的清理。

因此 7 个搜索者足以成功地清理图 C_{16} ，即 $fs(C_{16}) = 7$ (图 4)。

引理 8：对于正二十面体 C_{20} ，有 $fs(C_{20}) = 9$ 。

证明：首先需要证明 $fs(C_{20}) > 8$ 。由于正二十面体的高度对称性，只需要证明一种情形，就足以概括其余，这是显然的。

1) 不失一般性，放置搜索者 λ_1 、 λ_2 、 λ_3 、 λ_4 、 λ_5 到顶点 v_1 ，然后 λ_1 、 λ_2 、 λ_3 、 λ_4 、 λ_5 分别沿着其邻边移动到 v_2 、 v_3 、 v_4 、 v_5 、 v_6 (此时显然不满足继续移动清理的条件)；

2) 分别放置 λ_6 、 λ_7 、 λ_8 到 v_2 、 v_3 、 v_4 上；

3) 移动 λ_1 、 λ_6 分别到 v_7 、 v_8 ，移动 λ_2 、 λ_7 分别到 v_8 、 v_9 ，移动 λ_3 、 λ_8 分别到 v_9 、 v_{10} (此时 v_8 上有 2 个搜索者 λ_6 、 λ_2 ，三条未被清理的邻边； v_9 上有 2 个搜索者 λ_7 、 λ_8 ，三条未被清理的邻边)；

4) λ_6 、 λ_7 分别沿 v_8v_{12} 、 v_9v_{12} 移动到 v_{12} ，然后 λ_6 分别沿 $v_{12}v_{11}$ 移动到 v_{11} ；

此时未被清理的顶点上都有 1 个搜索者，并且这些顶点不止一条邻边未被清理，所以 8 个搜索者不足以成功地清理图 C_{20} ，即 $fs(C_{20}) > 8$ 。

其次未被清理的边构成欧拉子图且其每个顶点上都有 1 个搜索者，那么我们只需要放置 1 个搜索者到该欧拉子图的任一顶点上，该搜索者沿着欧拉回路就可以完成对图 C_{20} 的清理。因此 9 个搜索者足以成功地清理图 C_{20} ，即 $fs(C_{20}) \leq 9$ 。综上得证： $fs(C_{20}) = 9$ (图 4)。

根据引理 1~8 我们得出如下推论：

推论 1：对于凸三角多面体 C_n ，有 $fs(C_n) = \max \left\{ \delta(C_n) + 1, \left\lfloor \frac{\delta(C_n) + |V_{\text{odd}}(C_n)|}{2} \right\rfloor + 1 \right\}$ 。

4. 总结

本文研究的是凸三角多面体的快速搜索问题。首先求解了凸三角多面体的快速搜索数，最后提出 $fs(C_n) = \max \left\{ \delta(C_n) + 1, \left\lfloor \frac{\delta(C_n) + |V_{\text{odd}}(C_n)|}{2} \right\rfloor + 1 \right\}$ 。在后续的研究中，我们也将对其他具有研究意义的

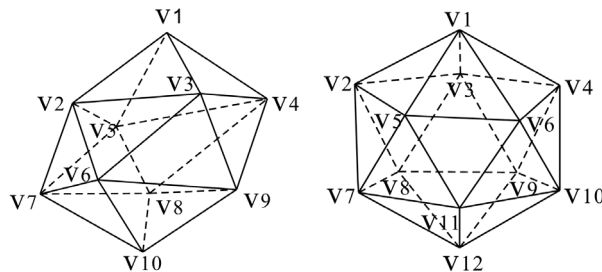


Figure 4. Gyroelongated square bipyramid, icosahedron

图 4. 双四角锥反角柱 C_{16} 、正二十面体 C_{20}

快速搜索问题展开研究。例如，树 T_n 和路径 P_n 的笛卡尔积图的快速搜索等都有待研究。

参考文献

- [1] Breisch, R. (1967) An Intuitive Approach to Speleotopology. *Southwestern Cavers*, **5**, 72-78.
- [2] Parsons, T.D. (1978) Pursuit-Evasion in a Graph. *Theory and Applications of Graphs*, **642**, 426-441. <https://doi.org/10.1007/BFb0070400>
- [3] Kirousis, L.M. and Papadimitriou, C.H. (1986) Searching and Pebbling. *Theoretical Computer Science*, **47**, 205-218. [https://doi.org/10.1016/0304-3975\(86\)90146-5](https://doi.org/10.1016/0304-3975(86)90146-5)
- [4] Megiddo, N., Hakimi, S.L., Garey, M.R., Johnson, D.S. and Papadimitriou, C.H. (1988) The Complexity of Searching a Graph. *Association for Computing Machinery*, **35**, 18-44. <https://doi.org/10.1145/42267.42268>
- [5] Dyer, D., Yang, B. and Yaşar, Ö. (2008) On the Fast Searching Problem. *AAIM 2008, LNCS*, **5034**, 143-154. https://doi.org/10.1007/978-3-540-68880-8_15
- [6] Stanley, D. and Yang, B. (2011) Fast Searching Games on Graphs. *Journal of Combinatorial Optimization*, **22**, 763-777. <https://doi.org/10.1007/s10878-010-9328-4>
- [7] Yang, B. (2013) Fast-Mixed Searching and Related Problems on Graph. *Theoretical Computer Science*, **507**, 100-113. <https://doi.org/10.1016/j.tcs.2013.04.015>
- [8] Xue, Y., Yang, B., Zhong, F. and Zilles, S. (2016) Fast Searching on Complete k-Partite Graphs. *Conference on Combinatorial Optimization and Applications*, **10043**, 159-174. https://doi.org/10.1007/978-3-319-48749-6_12
- [9] Xue, Y. and Yang, B. (2017) The Fast Search Number of a Cartesian Product of Graphs. *Discrete Applied Mathematics*. <https://doi.org/10.1016/j.dam.2017.03.003>
- [10] Dendris, N.D., Kirousis, L.M. and Thilikos, D.M. (1995) Fugitive-Search Games on Graphs and Related Parameters. *Graph-Theoretic Concepts in Computer Science*. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-59071-4_59
- [11] Yang, B. and Cao, Y. (2008) Monotonicity in Digraph Search Problems. *Theoretical Computer Science*, **407**, 532-544. <https://doi.org/10.1016/j.tcs.2008.08.025>
- [12] Richerby, D. and Thilikos, D.M. (2007) Graph Searching in a Crime Wave. *International Conference on Graph-Theoretic Concepts in Computer Science*, **4769**, 21-32. https://doi.org/10.1007/978-3-540-74839-7_3b
- [13] Yang, B. (2011) Fast Edge Searching and Fast Searching on Graphs. *Theoretical Computer Science*, **412**, 1208-1219. <https://doi.org/10.1016/j.tcs.2010.12.065>
- [14] Bonato, A. and Yang, B. (2013) Graph Searching and Related Problems. *Handbook of Combinatorial Optimization*, 1511-1558. https://doi.org/10.1007/978-1-4419-7997-1_76
- [15] Bienstock, D. and Seymour, P. (1991) Monotonicity in Graph Searching. Academic Press, Inc., 239-245.
- [16] Lapaugh, A.S. (1993) Recontamination Does Not Help to Search a Graph. *Journal of the ACM*, **40**, 224-245. <https://doi.org/10.1145/151261.151263>
- [17] Yang, B., Dyer, D. and Alspach, B. (2009) Sweeping Graphs with Large Clique Number. Elsevier Science Publishers B. V.
- [18] Flocchini, P., Fraigniaud, P. and Santoro, N. (2002) Capture of an Intruder by Mobile Agents. *ACM*, 200-209.
- [19] Kinnersley, N.G. (1992) The Vertex Separation Number of a Graph Equals Its Path-Width. *Information Processing Letters*, **42**, 345-350. [https://doi.org/10.1016/0020-0190\(92\)90234-M](https://doi.org/10.1016/0020-0190(92)90234-M)
- [20] Isaza, A., Lu, J., Bulitko, V., et al. (2008) A Cover-Based Approach to Multi-Agent Moving Target Pursuit. *Artificial Intelligence and Interactive Digital Entertainment Conference*, Stanford, California, 22-24 October 2008, 54-59.
- [21] Moldenhauer, C. and Sturtevant, N.R. (2009) Evaluating Strategies for Running from the Cops. *International Joint Conference on Artificial Intelligence*, Morgan Kaufmann Publishers Inc., 584-589.

知网检索的两种方式：

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>
下拉列表框选择：[ISSN]，输入期刊 ISSN：2163-1476，即可查询
2. 打开知网首页 <http://cnki.net/>
左侧“国际文献总库”进入，输入文章标题，即可查询

投稿请点击：<http://www.hanspub.org/Submission.aspx>

期刊邮箱：orf@hanspub.org