

复杂无向网络连通性的一种高效判定算法

王卓^{1,2} 秦博东³ 徐雍⁴ 鲁仁全⁴ 魏庆来⁵

摘要 通信网络的拓扑结构连通性是多智能体系统一致性控制或编队控制等的理论前提. 以往, 各种多智能体系统一致性控制或编队控制方面的文献仅侧重于控制协议、智能体动力学模型和控制律设计, 而缺乏对多智能体通信网络拓扑结构的连通性研究. 网络连通性高效判定算法不仅是大规模多智能体系统一致性控制或编队控制的保证, 而且在图论、现代移动通信、计算机与交通等各种网络中有着重要和广泛的应用. 针对复杂无向网络的连通性问题, 本文给出了一种新的高效判定算法, 以及该算法的时间复杂度和空间复杂度的上界. 该算法具有非常低的时间复杂度和空间复杂度, 且便于计算机实现, 因而具有重要的理论意义和广泛的实用价值.

关键词 复杂无向网络, 图论, 连通性, 多智能体系统, 高效算法

引用格式 王卓, 秦博东, 徐雍, 鲁仁全, 魏庆来. 复杂无向网络连通性的一种高效判定算法. 自动化学报, 2020, 46(10): 2129–2136

DOI 10.16383/j.aas.c190246

An Efficient Algorithm for Determining the Connectivity of Complex Undirected Networks

WANG Zhuo^{1,2} QIN Bo-Dong³ XU Yong⁴ LU Ren-Quan⁴ WEI Qing-Lai⁵

Abstract The topological connectivity of communication network is the theoretical premise of consistency control or formation control of multi-agent systems. The past literature on consistency control or formation control of multi-agent systems focused only on control protocol, agent dynamics model and control law design, lacking the research on the topological connectivity of multi-agent communication network. The efficient determination algorithm of network connectivity is not only the guarantee of consistency control or formation control for the large-scale multi-agent systems, but also has important and extensive applications in graph theory and various networks such as modern mobile communication networks, computer networks and transportation networks. This paper presents a new efficient determination algorithm with the upper bounds of the time complexity and space complexity of the algorithm, for the connectivity problem of complex undirected networks. The algorithm has very low time complexity and space complexity, and can be realized easily with computer programs, which makes it have important theoretical significance and wide practicability.

Key words Complex undirected networks, graph theory, connectivity, multi-agent systems, efficient algorithm

Citation Wang Zhuo, Qin Bo-Dong, Xu Yong, Lu Ren-Quan, Wei Qing-Lai. An efficient algorithm for determining the connectivity of complex undirected networks. *Acta Automatica Sinica*, 2020, 46(10): 2129–2136

收稿日期 2019-03-25 录用日期 2019-06-06
Manuscript received March 25, 2019; accepted June 6, 2019
国家自然科学基金 (61673041, 61722312, 61876041, 61425009, U1611262), 北京量子信息科学研究院 (Y18G34) 资助
Supported by National Natural Science Foundation of China (61673041, 61722312, 61876041, 61425009, U1611262), Beijing Academy of Quantum Information Sciences Research Program (Y18G34)

本文责任编辑 刘艳军

Recommended by Associate Editor LIU Yan-Jun

1. 北京航空航天大学前沿科学技术创新研究院北航-首医大数据精准医疗高精尖创新中心 北京 100191 2. 北京量子信息科学研究院 北京 100193 3. 北京航空航天大学仪器科学与光电工程学院 北京 100191 4. 广东工业大学自动化学院智能决策与协同控制广东省重点实验室 广州 510006 5. 中国科学院自动化研究所复杂系统管理与控制国家重点实验室 北京 100190

1. The Research Institute of Frontier Science, and Beijing Advanced Innovation Center for Big Data-Based Precision Medicine, Beihang University, Beijing 100191 2. Beijing Academy of Quantum Information Sciences, Beijing 100193 3. The The School of Instrumentation and Optoelectronic Engi-

自多智能体系统理论提出以来, 其协调控制方面^[1]的研究受到了国际学术界的广泛关注, 相关内容仅涉及多智能体系统一致性控制^[2–3]或编队控制协议^[4–5]、智能体动力学模型^[6–7]和控制律设计^[8–10], 而多智能体通信网络拓扑结构的连通性^[11–12]研究未见有文献报道. 通信网络拓扑结构的连通性是多智能体系统一致性控制与编队控制等的理论前提. 网络连通性高效判定算法不仅是大规模多智能体系统一致性控制或编队控制的保证, 而且在图论^[13–15]、现代移动通信^[16–17]、计算机与交

neering, Beihang University, Beijing 100191 4. The School of Automation, Guangdong University of Technology, and Guangdong Provincial Key Laboratory of Intelligent Decision and Co-operative Control, Guangzhou 510006 5. The State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190

通^[18] 等各种网络中有着重要和广泛的应用.

在多智能体系统一致性控制或编队控制中, 如果两个智能体在规定的时间内至少有一次数据交换, 则认为这两个智能体在通信方面是连通的. 如此, 进行数据交换的多个智能体构成了一个移动通信网络. 若所有智能体构成的这个移动通信网络在规定的时间内是连通的, 则称该网络的拓扑结构是连通的. 这种多智能体通信网络常被建模为复杂无向网络 (含自环和多重边), 而其拓扑结构的连通性检测和判定问题就可转化为复杂无向网络的连通性检测和判定问题.

迄今为止, 判定网络连通性的算法主要有两类: 基于深度优先搜索技术的算法, 如 Tarjan 算法^[19] 和 Gabow 算法^[20] 等; 基于可达矩阵的算法和基于关系传递闭包的 Warshall 算法^[21] 等. Tarjan 算法和 Gabow 算法是一类面向网络的直接搜索算法, 其优点是算法复杂度低 (均为 $O(n+m)$, 其中 n 为节点数, m 为边数), 缺点是仅适合简单网络 (无自环或多重边) 且不易编程 (需使用堆栈和标号技术). Warshall 算法的优点是结构异常简洁且可采用效率更高的位运算方法, 缺点是算法复杂度高 ($O(2n^3)$) 且同样仅适用于简单网络. 基于可达矩阵的连通性判定算法虽可用于复杂无向网络, 但由于其算法复杂度太高 ($O(n^4)$) 而很难适用于大规模移动通信网络或计算机互联网络的连通性判定.

针对具有 n 个节点的复杂无向网络, 本文给出了一种新的连通性判定算法, 该算法的时间复杂度和空间复杂度分别为 $O(3n^2)$ 和 $O(n^2+n)$. 与基于可达矩阵的连通性判定算法和基于关系传递闭包的 Warshall 算法相比, 本文所给算法在时间复杂度上具有显著优势, 因而更适合大规模移动通信网络或计算机互联网络的连通性检测与判定.

1 问题描述

给定一个无向网络 G , 我们用 v_i ($i = 1, 2, \dots$) 与 e_j ($j = 1, 2, \dots$) 分别表示 G 中的节点和边, 用 $V(G) = \{v_1, v_2, \dots\}$ 与 $E(G) = \{e_1, e_2, \dots\}$ 分别表示 G 中所有节点和所有边的集合, 则网络 G 可用一个偶对 (V, E) 表示, 记作 $G = (V, E)$.

若用 $n = |V(G)|$ 表示 $V(G)$ 中节点的个数, $m = |E(G)|$ 表示 $E(G)$ 中边的个数, 则网络 G 可另记为 $G = (n, m)$. n 称为网络 G 的阶数, 一个具有 n 个节点的网络称为 n 阶网络. n 和 m 均为有限值的无向网络 G 称为有限无向网络.

除特别声明外, 本文所讨论的网络均指有限复杂无向网络, 原因在于大规模移动通信网络和计算机互联网络常被建模为复杂无向网络, 且复杂无向网络较简单无向网络具有更大的普适性.

定义 1. 给定 n 阶无向网络 G 和网络的节点集 $V(G) = \{v_1, v_2, \dots, v_n\}$, 称 $\bar{A} = \bar{A}(\bar{a}_{ij}) \in \mathbf{R}^{n \times n}$ 为 G 的邻接矩阵, 其中

$$\bar{a}_{ij} = \begin{cases} m_{ij}, & v_i \text{ 与 } v_j \text{ 之间有 } m_{ij} \text{ 条边邻接} \\ 0, & v_i \text{ 与 } v_j \text{ 不邻接} \end{cases} \quad (1)$$

定义 1 中, $m_{ij} \geq 1$ 为整数. 当 G 为简单无向网络时, $m_{ij} = 1$; 当 G 为复杂无向网络时, 则至少存在一个 $m_{ij} > 1$. 因本文仅考虑 G 的连通性, 所以将 m_{ij} 视为 1 并不影响 G 的连通性.

定义 2. 给定 n 阶无向网络 G 和网络的节点集 $V(G) = \{v_1, v_2, \dots, v_n\}$, 设

$$a_{ij} = \begin{cases} 1, & \text{若 } v_i \text{ 与 } v_j \text{ 邻接, 或 } i = j \\ 0, & \text{若 } v_i \text{ 与 } v_j \text{ 不邻接} \end{cases} \quad (2)$$

则称: 1) 由元素 a_{ij} ($i, j = 1, 2, \dots, n$) 构成的 n 阶矩阵为网络 G 的邻居矩阵, 记作 $A(G)$, 或简记为 A ; 2) A 的第 i 行为网络 G 的第 i 个邻交集, 记作 $A_i(G)$, 或简记为 A_i .

我们以图 1 和图 2 所示无向网络为例来具体说明定义 2 中所涉及的概念.

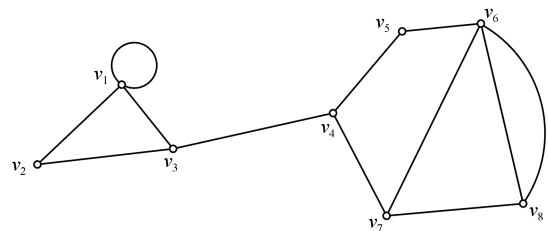


图 1 G_1
Fig. 1 G_1

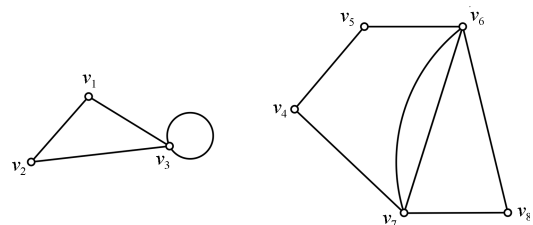


图 2 G_2
Fig. 2 G_2

图 1 和图 2 所示的无向网络各有 8 个节点, 它们的邻居矩阵分别是

$$A(G1) = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

$$A(G2) = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

邻居集 $A_i(G_j)$ ($i = 1, 2, \dots, n; j = 1, 2$) 表示节点 v_i 与其邻接节点的关系, 其中取值为 1 的元素所对应的节点就是包含 v_i 在内的一个邻居组, 如 $A_1(G1)$ 中包含 v_1 在内的邻居组是 $\{v_1, v_2, v_3\}$. 我们用 $|A_i|$ 表示 A_i 中取值为 1 的元素个数, 即邻居集 A_i 中邻居的个数, 如 $|A_1(G1)| = 3$. 显然, 每个邻居组中的节点及与它们相邻接的边一起构成网络 G 的一个连通子网络. 此外, 我们称元素全为 0 的邻居集为零邻居集 (对应空网络), 元素全为 1 的邻居集为全邻居集 (对应连通网络).

给定一个无向网络, 由定义 2 和图论中邻接矩阵的定义可知, 复杂无向网络邻居矩阵与简单无向网络邻接矩阵的差别仅是对角线元素的取值有所不同, 邻居矩阵的对角线元素全为 1, 而邻接矩阵的对角线元素全为 0.

2 算法理论基础与算法设计

2.1 算法理论基础

由定义 2 可知, n ($n \geq 2$) 阶无向网络 G 的邻居集 $A_i(G)$ 具有如下特点: 1) 邻居集中只有 2 种元素 1 和 0, 且至少有一个 1; 2) 各邻居集中元素的个数彼此相等; 3) 每个邻居集对应一个邻居组, 每个邻居组彼此邻接且对应网络 G 的一个连通子网络.

定义 2 是一个新定义, 也是本文所给算法的基础. 在具体给出基于邻居矩阵的网络连通性判定算法 (下面简称本文所给算法) 之前, 我们需先给出邻居集的相关运算规则.

定义 3. 给定 n ($n \geq 2$) 阶无向网络 G 及相应的邻居集 A_i ($i = 1, 2, \dots, n$), 邻居集之交与并的运算规则如下:

1) $A_i \cap A_j = \{z_k = x_k \cap y_k | x_k \in A_i, y_k \in A_j, k = 1, 2, \dots, n\}$;

2) $A_i \cup A_j = \{z_k = x_k \cup y_k | x_k \in A_i, y_k \in A_j, k = 1, 2, \dots, n\}$.

上式中各元素之间的运算 (\cap 和 \cup) 均为布尔运算, 可采用效率更高的位运算方法实现. 上述运算规则用于邻居集时具有明确意义:

1) 若两邻居集 A_i, A_j ($i \neq j$) 的交集 $A_i \cap A_j$ 为零邻居集, 则由 A_i, A_j 决定的两个邻居组没有公共邻居, 反之则有公共邻居;

2) 没有公共邻居的两个邻居集的并集不构成新的邻居组, 即不构成 G 的新的连通子网络;

3) 有公共邻居的两个邻居集的并集 $A_i \cup A_j$ 是一个新的邻居集, 该邻居集对应 G 的一个新的连通子网络; 新连通子网络是 A_i, A_j 所对应的连通子网络的并, 且显然有 $|A_i \cup A_j| \geq \max(|A_i, A_j|)$.

本文所给算法主要涉及 2 种运算: 1) 求两个邻居集 A_i, A_j ($i \neq j$) 的交集, 以验证它们是否有公共邻居, 若无则不求 $A_i \cup A_j$; 2) 若有公共邻居, 则求 $A_i \cup A_j$.

两个邻居集 A_i, A_j ($i \neq j$) 的交并运算可按如下简便方法一次完成:

1) 求 $A_i \cap A_j$, 对于 $k = 1, 2, \dots, n$, 依次检查 $x_k \cap y_k$ 的取值, 当首次出现 $x_k \cap y_k = 1$ 时, 停止本次求交运算;

2) 求 $A_i \cup A_j = \{z_k = x_k \cup y_k | x_k \in A_i, y_k \in A_j, k = 1, 2, \dots, n\}$.

本文所给算法的主要思路是: 1) 给定 n ($n \geq 2$) 阶无向网络, 生成邻居矩阵 $A(a_{ij}) \in \mathbf{R}^{n \times n}$ 和相应的邻居集 A_i ($i = 1, 2, \dots, n$); 2) 求 $A_1 \cap A_i$ ($i = 2, 3, \dots, n$) 以确定两个邻居集之间是否具有公共邻居; 若无, 令 $i = i + 1$, 继续进行求交运算; 若有, 求 $A_1 \cup A_i$ 并将运算结果赋予 A_1 ; 上述求并与赋值运算结束后, 令 $i = i + 1$, 继续进行这种求交和求并运算; 3) 若上述运算进行到最后一步后, $|A_1| < n$ 或 A_1 与所有剩余的邻居集之间没有相同的邻居, 则可判定所给网络不连通, 否则, 可判定所给网络连通.

2.2 算法设计

给定 n ($n \geq 2$) 阶无向网络 G , 设: G 的邻居矩阵 $A(a_{ij}) \in \mathbf{R}^{n \times n}$ 和相应的邻居集 A_i ($i = 1, 2, \dots, n$) 已经生成; i 为 $A(a_{ij})$ 的行循环变量, j 与 k 为 $A(a_{ij})$ 的列循环变量 ($i, j, k = 1, 2, \dots, n$); $H = \{h_1, h_2, \dots, h_n\}$ 为标号集, $h_i = 1$ 表示 A_1 与 A_i 进行过求并运算, $h_i = 0$ 表示 A_1 与 A_i 未进

行过求并运算. 下面我们具体给出复杂无向网络 G 的连通性判定算法.

基于邻居矩阵的复杂无向网络连通性判定算法

- 1) 赋初值: $i = j = k = 0$, $H = \{h_i = 0 | i = 1, 2, \dots, n\}$;
- 2) $j = j + 1$;
- 3) 如果 $a_{1j} = 0$ 且 $j < n$, 转到 2); 如果 $a_{1j} = 0$ 且 $j = n$, 转到 13); 如果 $a_{1j} = 1$, $i = 1$, 转到 4);
- 4) $i = i + 1$;
- 5) 如果 $h_i = 1$ 且 $i < n$, 转到 4); 如果 $h_i = 1$ 且 $i = n$, 转到 11); 如果 $h_i = 0$, 转到 6);
- 6) 如果 $a_{ij} = 0$ 且 $i < n$, 转到 4); 如果 $a_{ij} = 0$ 且 $i = n$, 转到 11); 如果 $a_{ij} = 1$, $k = 1$, 转到 7);
- 7) $k = k + 1$;
- 8) 如果 $a_{ik} = 0$ 且 $k < n$, 转到 7); 如果 $a_{ik} = 0$ 且 $k = n$, 转到 10); 如果 $a_{ik} = 1$, $a_{1k} =: a_{ik}$, 转到 9);
- 9) 如果 $k < n$, 转到 7); 如果 $k = n$, 转到 10);
- 10) $h_i =: 1$; 如果 $i < n$, 转到 4); 如果 $i = n$, 转到 11);
- 11) 如果 $j < n$, 转到 2); 如果 $j = n$, $|A_1| = \sum_{j=1}^n a_{1j}$, 转到 12);
- 12) 如果 $|A_1| = n$, 输出结果: 图 G 连通, 算法结束; 如果 $|A_1| < n$, 转到 13);
- 13) 输出结果: 图 G 不连通; 算法结束.

步骤 8) 和 10) 中的符号 “ $=:$ ” 表示赋值运算.

定理 1. 任给 n ($n \geq 2$) 阶复杂无向网络 G , 使用基于邻居矩阵的判定算法时, 网络 G 连通的充分必要条件是算法结束后 $|A_1| = n$.

证明. 由本文所给算法可知, 当有公共邻居时, 每进行一次求并运算, A_1 中所包含的邻居数 (取值为 1 的元素数) 或者保持不变或者增加. 当保持不变且 A_1 包含网络 G 的全部节点, 即 $|A_1| = n$ 时, 网络 G 连通; 当邻居数增加并最终导致 $|A_1| = n$ 时, 网络 G 也连通.

反之, 设 $|A_1| < n$ 时, 网络 G 连通. $|A_1| < n$ 有两种情况: 1) $|A_1| = 1$, 这表明节点 v_1 是悬挂点, 没有邻接边, 所以网络 G 不连通; 2) 至少存在一个 A_i ($i \in \{2, 3, \dots, n\}$), 它在算法结束后与 A_1 没有相同的邻居, 即 A_i 对应的子网络与 A_1 对应的子网络没有公共节点, 这便导致 $|A_1| < n$, 网络 G 不连通. 上述两种情况都与假设矛盾, 则若要网络 G 连通, 必须要 $|A_1| = n$. \square

图 3 是本文所给算法的框图.

2.3 算法应用举例

我们以图 G_1 和图 G_2 所示无向网络为例来说明本文所给算法的具体应用.

例 1. 图 G_1 所示无向网络有 8 个节点, 其邻居矩阵 $A(G_1) \in \mathbf{R}^{8 \times 8}$. 算法运行前, $|A_1(G_1)| = 3$, 算法运行后: $j = 1$, $|A_1(G_1)| = 4$; $j = 2$, $|A_1(G_1)| = 4$; $j = 3$, $|A_1(G_1)| = 6$; $j = 4$, $|A_1(G_1)| = 8$; $j = 5, 6, 7, 8$ 时, $|A_1(G_1)| = 8$. 因此, 可判定图 G_1 所示无向网络连通.

例 2. 图 G_2 所示无向网络有 8 个节点, 其邻居矩阵 $A(G_2) \in \mathbf{R}^{8 \times 8}$. 算法运行前, $|A_1(G_2)| = 3$, 算法运行后: $j = 1$, $|A_1(G_2)| = 3$; $j = 2$, $|A_1(G_2)| = 3$; $j = 3$, $|A_1(G_2)| = 3$; $j = 4, 5, 6, 7, 8$ 时, $|A_1(G_2)| = 3 < 8$. 因此, 可判定图 G_2 所示无向网络不连通.

3 算法复杂性分析

3.1 算法的时间复杂度

给定 n ($n \geq 2$) 阶无向网络 G , 设其相应的邻居矩阵、邻居集分别为 $A(a_{ij}) \in \mathbf{R}^{n \times n}$ 和 A_i ($i = 1, 2, \dots, n$). 本文所给算法涉及 a_{ij} 是否为 1 的比较运算, 将 a_{ik} ($i \geq 2$) 的值赋予 a_{1k} 的赋值运算, 以及两个元素 a_{1j} 、 a_{1k} ($j \neq k$) 的布尔或运算. 按算法复杂性分析的常规做法, 我们将不考虑赋值运算并将一次比较运算视为一次加法运算. 下面, 我们分 3 部分具体分析本文所给算法的时间复杂度.

1) 对于每一个 $j \in \{1, 2, \dots, n\}$, 本文所给算法需要比较 a_{ij} ($1 \leq i \leq n$) 是否等于 1, 用于确定是否存在某个 i ($2 \leq i \leq n$) 使得 $a_{1j} = a_{ij} = 1$, 即确定 $A_1 \cap A_i$ 为非零邻居集. 对于每一给定的 j , 这种比较运算共需 n 次, 相当于 n 次加法运算. 考虑最坏情况, 即当所有的 $a_{1j} = 1$ ($1 \leq j \leq n$) 时, 这种比较运算总共需要 n^2 次加法运算.

2) 当 $A_1 \cap A_i$ ($2 \leq i \leq n$) 为非零邻居集时, 需进行 $A_1 \cup A_i$ 运算. 考虑最坏情况, 即当所有的 $a_{ik} = 1$ ($2 \leq k \leq n$) 时, $A_1 \cup A_i$ 需要 $2(n-1)$ 次布尔或运算, 当 i 遍历所有的取值时, 总共需要 $2(n-1)^2$ 次布尔或运算.

3) 本文所给算法最后需要求 $|A_1|$ 并判定 $|A_1|$ 是否小于 n , 这部分总共需要 n 次加法运算.

设 J 为总的计算次数, 由上述 3 部分计算结果可知

$$J = n^2 + 2(n-1)^2 + n = 3n^2 - 3n + 2$$

因此, 本文所给算法的时间复杂度的上界为 $O(3n^2)$. 相比之下可知, 就复杂无向网络的连通性判定而言, 本文所给算法的时间复杂度远低于 Warshall 算法 ($O(2n^3)$) 和基于可达矩阵算法 ($O(n^4)$) 的时间复杂度.

3.2 算法的空间复杂度

本文所给算法仅需生成和使用一个邻居矩阵

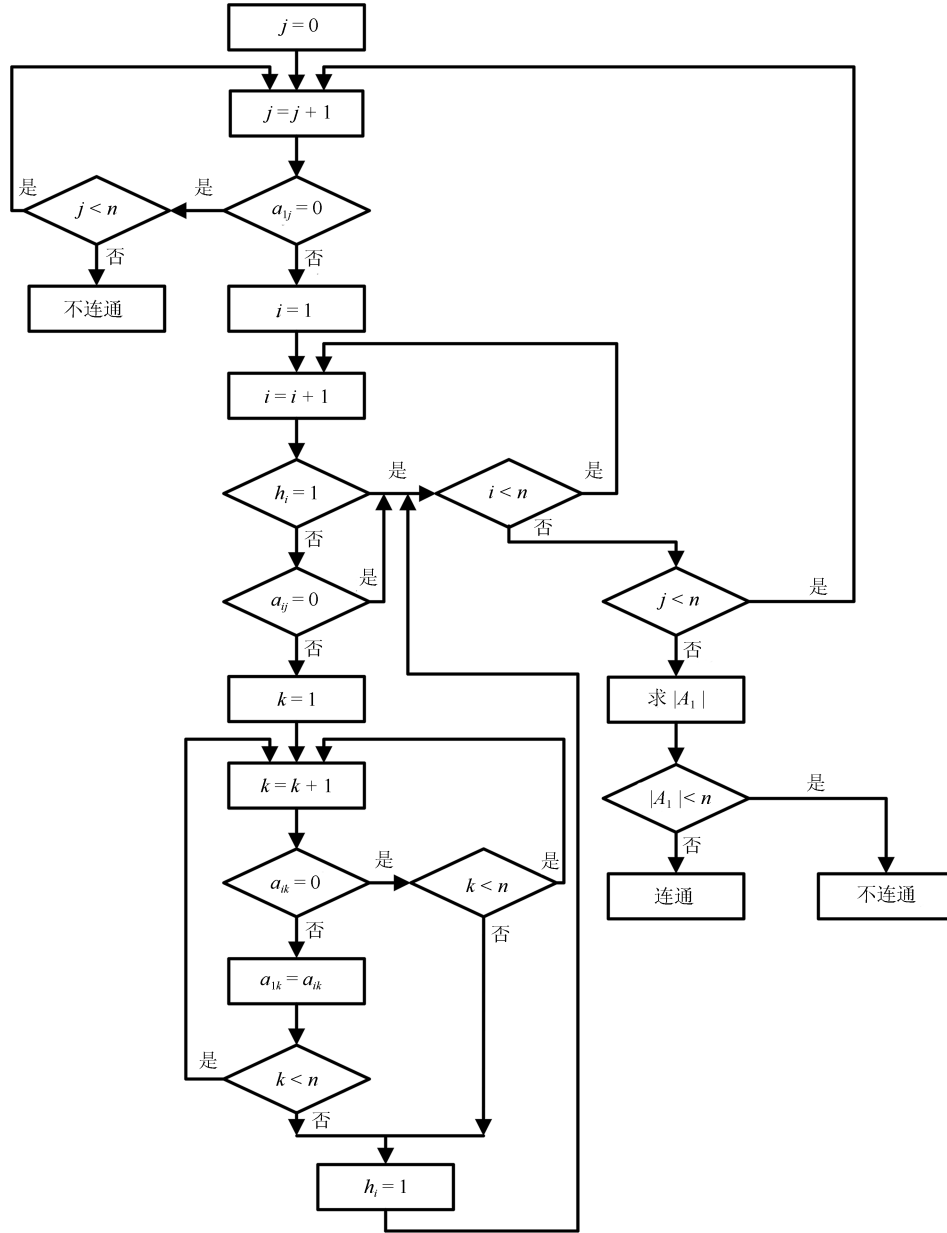


图3 基于邻居矩阵的复杂无向网络连通性判定算法框图

Fig. 3 Block diagram of the connectivity determination algorithm for complex undirected networks based on the neighbor matrix

$A(a_{ij}) \in \mathbf{R}^{n \times n}$ 和一个标号集 $H = \{h_i = 0 \text{ 或 } h_i = 1 | i = 1, 2, \dots, n\}$. 因邻居矩阵元素的取值或为 0 或为 1, 所以生成邻居矩阵共需占用 n^2 个存储单元. 同理, 标号集 H 共需占用 n 个存储单元. 因此, 本文所给算法的空间复杂度的上界为 $O(n^2 + n)$. 不难理解, 本文所给算法比 Warshall 算法和基于可达矩阵的算法具有更低的空间复杂度.

3.3 算法的最优分布式实现

分布式计算是一种分组并行计算, 可极大地提高计算效率. 由图论和算法结构理论可知, 本文所给

算法适合分布式计算. 下面, 我们将简要讨论这个问题.

给定 $n (n \geq 2)$ 阶无向网络 G , 我们可以按如下规则将 n 个节点分成 $k (2 \leq k \leq n)$ 组, 使得:

- 1) 每个组中的节点数不少于 1 个;
- 2) 任意两个组不包含相同的节点.

如此:

- 1) 当我们将每个组中的全部节点集结成 (视作) 一个节点时, G 就成为具有 k 个节点的网络;
- 2) 每个组中的全部节点连同它们的邻接边一起

构成 G 的一个子网络, 且全部 k 个子网络的并就是 G ;

3) 每个子网络都连通且全部 k 个子网络也连通, 则 G 连通, 否则, G 不连通.

如果能将本文所给算法分别用于上述每个子网络及全部 k 个子网络时, 除仍可判定 G 的连通性外, 计算量将会大幅度减小. 现在的问题是, 如何分组可使总的计算量达到最小?

定理 2. 任给 n ($n \geq 2$) 阶无向网络 G , 将 n 个节点分成 k 组. 若分组数 $k = \left(\frac{n^2}{2}\right)^{\frac{1}{3}}$ 且每个组中的节点数均为 $m = (2n)^{\frac{1}{3}}$, 则将本文所给算法用于 G 的连通性判定时, 计算量将会达到最小.

证明. 本文所给算法的时间复杂度为 $O(3n^2)$, 即最大可能的计算量为 $3n^2$. 设将 n 个节点分成 k ($2 \leq k \leq n$) 组, 每个组中的节点数为 m_i ($1 \leq i \leq k$), $\sum_{i=1}^k m_i = n$, J 为判定 G 的连通性所需最大计算量. 由前面的分析可知, 判定每个子网络 (m_i 个节点) 连通性所需最大计算量为 $3m_i^2$, 判定 k 个子网络连通性所需最大计算量为 $3k^2$. 如此

$$J = 3k^2 + \sum_{i=1}^k [3m_i^2] = 3 \left\{ k^2 + \sum_{i=1}^{k-1} m_i^2 + m_k^2 \right\} = 3 \left\{ k^2 + \sum_{i=1}^{k-1} m_i^2 + (n - \sum_{i=1}^{k-1} m_i)^2 \right\} \quad (3)$$

对于 $1 \leq i \leq (k-1)$, 我们有

$$\frac{\partial J}{\partial m_i} = 6 \left[m_i - \left(n - \sum_{i=1}^{k-1} m_i \right) \right] = 6(m_i - m_k) \quad (4)$$

令 $\frac{\partial J}{\partial m_i} = 0$, 可得 $m_1 = m_2 = \dots = m_{k-1} = m_k$. 这表明, 若要 J 达到最小, 每个组中的节点数必须相等. 设每个组的节点数为 $m_i = m$ ($1 \leq i \leq k$), 代入前式可得: $J = 3(k^2 + km^2) = 3 \left(k^2 + \frac{n^2}{k} \right)$.

令 $\frac{\partial J}{\partial k} = 3 \left(2k - \frac{n^2}{k^2} \right) = 0$, 可得 $k = \left(\frac{n^2}{2}\right)^{\frac{1}{3}}$. 另一方面, $J = 3(k^2 + km^2) = 3 \left(\frac{n^2}{m^2} + nm \right)$. 令 $\frac{\partial J}{\partial m} = 3 \left(n - \frac{2n^2}{m^3} \right) = 0$, 可得 $m = (2n)^{\frac{1}{3}}$. 这表

明, 当 $k = \left(\frac{n^2}{2}\right)^{\frac{1}{3}}$, $m = (2n)^{\frac{1}{3}}$ 时, 使用本文所给算法可使计算量达到最小. \square

定理 2 所确定的分组方式就是本文所给算法的最优分布式实现方式. 分组与不分组情况下的计算量之比为 $3(2n)^{-\frac{2}{3}}$. 显然, n 越大, 计算量下降的幅度也越大. 当 $n = 500$ 时, 分组情况下的计算量仅是不分组情况下的计算量的 3%.

需要说明的是, 定理 2 中的 k 和 m 需取整数. 下面, 我们举例说明如何确定 k 和 m 的取值问题.

例 3. 设某一无向网络 G 具有 $n = 497$ 个节点, 试确定节点的最优分组结果.

由定理 2 可得: $k = \left(\frac{n^2}{2}\right)^{\frac{1}{3}} = \left(\frac{1}{2} \times 247\,009\right)^{\frac{1}{3}} < \left(\frac{1}{2} \times 250\,000\right)^{\frac{1}{3}} = 50$, 即 $k < 50$; 同理可得, $k = \left(\frac{n^2}{2}\right)^{\frac{1}{3}} = \left(\frac{1}{2} \times 247\,009\right)^{\frac{1}{3}} > \left(\frac{1}{2} \times 235\,298\right)^{\frac{1}{3}} = 49$, 即 $k > 49$. 我们按两种方式分组: 1) $k = 50$, $m_i = 10$ ($1 \leq i \leq 49$), $m_{50} = 7$, 最大可能的计算量为 $J = 3(50^2 + 49 \times 10^2 + 7^2) = 22\,347$; 2) $k = 49$, $m_i = 10$ ($1 \leq i \leq 42$), $m_i = 11$ ($43 \leq i \leq 49$), 最大可能的计算量为 $J = 3(49^2 + 42 \times 10^2 + 7 \times 11^2) = 20\,166$. 显然, 第 2 种分组方式可使最大可能的计算量达到最小. 因此, k 和 m 的取值应按第 2 种分组方式确定.

在大规模多智能体编队控制问题中, 各个智能体的相对位置比较固定, 定理 2 所给的最优分组方法在其通信网络的连通性判定方面更能发挥作用.

4 结论

在引言部分, 我们论述了网络连通性判定算法、特别是高效连通性判定算法的理论与现实意义.

本文第 1 节给出了复杂无向网络邻居矩阵和邻居集的定义及相关概念, 它们是基于邻居矩阵的复杂无向网络连通性判定算法, 即本文所给算法的前提.

本文第 2 节给出了邻居集交与并的运算规则、相关概念及其集合解释. 在此基础上, 给出了基于邻居矩阵的复杂无向网络连通性判定算法. 该算法是本文的主要贡献, 显然它也适用于简单无向网络的连通性判定.

第 3 节分析给出了本文所给算法的时间复杂度 $O((3n^2))$ 和空间复杂度 $O((n^2+n))$, 它较 Warshall 算法和基于可达矩阵的算法具有低得多的时间复杂度和空间复杂度. 此外, 本节还给出了可使最大可能

的计算量达到最小的最优分布式计算规则, 相关结果则是本文的另一有意义的贡献。

与现有算法相比, 本文所给算法的时间复杂度和空间复杂度均非常低、易于分布式计算, 更适合大规模动态复杂无向网络的连通性判定。需要说明的是, 本文所给算法不能直接用于复杂有向网络的连通性判定。我们今后的工作是将上述结果推广到复杂有向网络, 以期得到更一般和更高效的连通性判定算法。

References

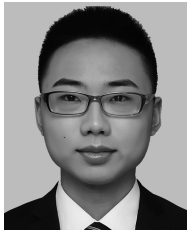
- Chen Shi-Ming, Guan Jun-Jie, Gao Yan-Li, Pei Hui-Qin, Qiu Yun. Event-triggered fast consensus algorithm for multi-agent systems under jointly-connected topology. *Acta Automatica Sinica*, 2018, **44**(12): 2269–2277
(陈世明, 管俊杰, 高彦丽, 裴惠琴, 邱昀. 组合连通拓扑下基于事件触发的多智能体快速一致性算法. *自动化学报*, 2018, **44**(12): 2269–2277)
- Yang Dong-Yue, Mei Jie. Disturbance observer based consensus of linear multi-agent systems under a directed graph. *Acta Automatica Sinica*, 2018, **44**(6): 1037–1044
(杨东岳, 梅杰. 有向图中基于扰动观测器的线性多智能体系统一致性. *自动化学报*, 2018, **44**(6): 1037–1044)
- Yan Wei-Sheng, Li Jun-Bing, Wang Yin-Tao. Consensus for damaged multi-agent system. *Acta Automatica Sinica*, 2012, **38**(11): 1880–1884
(严卫生, 李俊兵, 王银涛. 受损多智能体系统的信息一致性. *自动化学报*, 2012, **38**(11): 1880–1884)
- Luo Xiao-Yuan, Yang Fan, Li Shao-Bao, Guan Xin-Ping. Generation of optimally persistent formation for multi-agent systems. *Acta Automatica Sinica*, 2014, **40**(7): 1311–1319
(罗小元, 杨帆, 李绍宝, 关新平. 多智能体系统的最优持久编队生成策略. *自动化学报*, 2014, **40**(7): 1311–1319)
- Min Hai-Bo, Liu Yuan, Wang Shi-Cheng, Sun Fu-Chun. An overview on coordination control problem of multi-agent system. *Acta Automatica Sinica*, 2012, **38**(10): 1557–1570
(闵海波, 刘源, 王仕成, 孙富春. 多个体协调控制问题综述. *自动化学报*, 2012, **38**(10): 1557–1570)
- Cao Ran, Mei Jie. Group consensus for networked Euler-Lagrangian systems under a directed graph without relative velocity information. *Acta Automatica Sinica*, 2018, **44**(1): 44–51
(曹然, 梅杰. 有向图中网络 Euler-Lagrange 系统无需相对速度信息的群一致性. *自动化学报*, 2018, **44**(1): 44–51)
- Huang C, Zhai G S, Xu G S. Necessary and sufficient conditions for consensus in third order multi-agent systems. *IEEE/CAA Journal of Automatica Sinica*, 2018, **5**(6): 1044–1053
- Luo Xiao-Yuan, Shao Shi-Kai, Guan Xin-Ping, Zhao Yuan-Jie. Dynamic generation and control of optimally persistent formation for multi-agent systems. *Acta Automatica Sinica*, 2013, **39**(9): 1431–1438
(罗小元, 邵士凯, 关新平, 赵渊洁. 多智能体最优持久编队动态生成与控制. *自动化学报*, 2013, **39**(9): 1431–1438)
- Wang Q, Wang Y Z, Zhang H X. The formation control of multi-agent systems on a circle. *IEEE/CAA Journal of Automatica Sinica*, 2018, **5**(1): 148–154
- Maithripala D H A, Jayasuriya S. Rigid formation keeping and formation reconfiguration of multi-agent systems. *IFAC Proceedings Volumes*, 2008, **41**(2): 5155–5160
- You Ke-You, Xie Li-Hua. Survey of recent progress in networked control systems. *Acta Automatica Sinica*, 2013, **39**(2): 101–118
(游科友, 谢立华. 网络控制系统的最新研究综述. *自动化学报*, 2013, **39**(2): 101–118)
- Huo Zhi-Hong, Fang Hua-Jing. Fault-tolerant control research for networked control system under communication constraints. *Acta Automatica Sinica*, 2006, **32**(5): 659–666
- Chung F. Graph theory in the information age. *Notices of the American Mathematical Society*, 2009, **57**(6): 726–732
- Zhang S H, Yan Y H, Bao W, Guo S W, Jiang J D, Ma M R. Network topology identification algorithm based on adjacency matrix. In: *Proceedings of the 2017 IEEE Innovative Smart Grid Technologies*. Auckland: IEEE, 2017. 1–5
- Ghosh R K. Parallel algorithms for connectivity problems in graph theory. *International Journal of Computer Mathematics*, 1986, **18**(3-4): 193–218
- Wagenpfeil J, Trachte A, Hatanaka T, Fujita M, Sawodny O. A distributed minimum restrictive connectivity maintenance algorithm. *IFAC Proceedings Volumes*, 2009, **42**(16): 365–370
- Pisano A, Franceschelli M, Pilloni A, Shtessel Y, Usai E. Retaining connectivity in mobile communication mesh networks. *IFAC-PapersOnLine*, 2017, **50**(1): 800–807
- Nagamochi H. Graph algorithms for network connectivity problems. *Journal of the Operations Research Society of Japan*, 2004, **47**(4): 199–223
- McNunn G S, Bryden K M. A proposed implementation of Tarjan's algorithm for scheduling the solution sequence of systems of federated models. *Procedia Computer Science*, 2013, **20**: 223–228
- Hwang F. Control algorithms for rearrangeable clos networks. *IEEE Transactions on Communications*, 1983, **31**(8): 952–954
- Kenneth H R. *Discrete Mathematics and Its applications*. 7th Edition. 1221 Avenue of the Americas, New York, NY: McGraw-Hill, 2012. 603–606



王卓 北京航空航天大学前沿科学技术创新研究院教授。2013 年获得美国伊利诺伊大学芝加哥分校电子与计算机工程系博士学位。主要研究方向为基于数据的系统辨识、建模、分析、优化与控制, 自适应动态规划方法, 非线性自适应控制, 基于原子自旋效应的惯性测量技术, 自旋原子系综控制(操控)方法。

E-mail: zhuowang@buaa.edu.cn

(**WANG Zhuo** Professor at the Research Institute of Frontier Science, Beihang University, China. He received his Ph.D. degree in Electrical and Computer Engineering Department from University of Illinois at Chicago, USA in 2013. His research interest covers data-based system identification, modeling, analysis, optimization and control, adaptive dynamic programming methods, nonlinear adaptive control, atomic-spin-effect-based inertial measurement technology, and atomic ensemble control (manipulation) methods.)



秦博东 北京航空航天大学仪器科学与光电工程学院硕士研究生. 主要研究方向为精密量子测量仪器传感数据处理.

E-mail: qinbodong.qbd@163.com

(**QIN Bo-Dong** Master student at the School of Instrumentation and Optoelectronic Engineering, Beihang University. His main research interest is

sensing data processing for high precision quantum measurement in struments.)



徐 雍 广东工业大学自动化学院副教授. 2014 年获得浙江大学控制科学与工程学院博士学位. 主要研究方向为网络化控制系统, 状态估计, 正定系统.

E-mail: xuyong809@163.com

(**XU Yong** Associate professor at the School of Automation, Guangdong University of Technology. He received

his Ph.D. degree from the College of Control Science and Engineering, Zhejiang University in 2014. His research interest covers networked control systems, state estimation, and positive systems.)



鲁仁全 广东工业大学自动化学院教授. 2004 年获得浙江大学控制科学与工程学院博士学位. 主要研究方向为复杂系统, 网络控制系统, 非线性系统. 本文通信作者. E-mail: rqlu@gdut.edu.cn

(**LU Ren-Quan** Professor at the School of Automation, Guangdong University of Technology. He received his

Ph. D. degree from the College of Control Science and Engineering, Zhejiang University in 2004. His research interest covers complex systems, networked control systems, and nonlinear systems. Corresponding author of this paper.)



魏庆来 中国科学院自动化研究所复杂系统管理与控制国家重点实验室研究员. 2009 年获得东北大学控制理论与控制工程专业博士学位. 主要研究方向为智能控制, 人工智能, 自学习系统, 自适应动态规划, 自适应最优控制, 数据驱动控制, 神经网络控制, 工业控制系统优化, 智能电网.

E-mail: qinglai.wei@ia.ac.cn

(**WEI Qing-Lai** Professor at the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences. He received his Ph.D. degree in control theory and control engineering from Northeastern University in 2009. His research interest covers intelligent control, artificial intelligence, learning systems, adaptive dynamic programming, adaptive optimal control, data-based control, neural network-based control, optimization in industrial systems, and smart grid.)