

无线网络中寻找非干扰不相交路径的拟人算法

董高秀 凌珊 陈卫东

(华南师范大学计算机学院 广州 510631)

摘要 针对无线网络中寻找从源点 s 到汇点 t 的两条非干扰不相交路径这一 NP 难问题,提出了一种拟人算法。该算法首先基于网络流方法得到两条点不相交的 $s-t$ 路径,然后通过一种拟人化的策略逐步调整这两条路径,力图使得它们变为两条非干扰不相交的 $s-t$ 路径。模拟实验表明,与现有的算法相比,拟人算法可以快速地以更高的概率找到两条长度较短的非干扰不相交路径。

关键词 无线网络,不相交路径,非干扰不相交路径,NP 难度,拟人算法

中图分类号 TP301 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.08.015

Quasi-human Algorithm for Finding Non-interfering Disjoint Paths in Wireless Networks

DONG Gao-xiu LING Shan CHEN Wei-dong

(School of Computer, South China Normal University, Guangzhou 510631, China)

Abstract For the NP-hard problem of finding two non-interfering disjoint paths between two nodes s and t in a wireless network, a quasi-human algorithm was proposed. Starting with two node-disjoint paths from node s to node t obtained by using the flow technique, this algorithm tries to change these two paths into two non-interfering disjoint paths from s to t by adjusting them gradually through a quasi-human strategy. Experimental results show that compared with the existing algorithms, the quasi-human algorithm can quickly find two shorter non-interfering disjoint paths from s to t with a higher probability.

Keywords Wireless networks, Disjoint paths, Non-interfering disjoint paths, NP-hard, Quasi-human algorithm

1 引言

多道路由可用来增加网络系统传输率和增强传输的可靠性,是实现负载均衡和各种容错路由的重要方法,在传输网络、VLSI 布局以及高速网络中虚拟线路路由等领域有重要应用^[1]。网络常常建模为图,图的点表示网络中的节点,图的边表示网络中节点之间的直接的物理链接。于是,网络中多道路由问题可归结为在图中找不相交路径来解决^[2]。不相交路径问题有两种基本类型:

1) $s-t$ 不相交路径:任给源点 s 和汇点 t ,在图中找 k 条从 s 到 t 的不相交路径。

2) k -pairs 不相交路径:任给源点 $s_1, s_2, \dots, s_k$ 和汇点 $t_1, t_2, \dots, t_k$,在图中找 k 条从 s_i 到 t_i 的不相交路径($i=1, 2, \dots, k$)。

k -pairs 不相交路径问题即使在 $k=2$ 时仍是 NP 难的^[3]。 $s-t$ 不相交路径问题可用最大流技术解决,但如果考虑某些优化标准,问题通常也是 NP-难的。例如,为了提高网络可靠性、实现负载均衡和避免拥塞,可采用如下简单的路由策略:为源-汇点之间准备两条不相交的路径,当主路径失效后迅速启用备用路径进行路由。Min-Max 问题是寻找两条不相交路径使得较长者的长度达到最小^[4], Min-Min 问题是寻找两

条不相交路径使得较短者的长度达到最小^[5], QoS 约束 Min-Sum 问题是指两条路径时延满足 QoS 约束且路径长度之和达到最小的不相交路径问题^[6,7],它们都是 NP 难的。

不相交路径有边不相交路径和点不相交路径之分。点不相交路径问题和边不相交路径问题通常可在多项式时间内相互转化^[3]。边不相交路径要求不同路径之间不存在公共边,可用于负载均衡,但当不同路径之间有公共节点时会出现拥塞现象。点不相交路径要求不同路径之间不存在公共节点,能避免数据传输过程可能产生的拥塞现象,但无法避免节点之间数据的潜在干扰。例如,在无线传感器网络中,当两节点距离小到一定程度时,如果两节点传输的是不同数据包,则可能会产生相互干扰从而导致数据包出错甚至丢失^[8]。针对这种情况,人们提出了非干扰不相交路径的概念^[9]。非干扰不相交路径是独立性更强的不相交路径。具体来说, k -pairs 非干扰不相交路径问题要求在图中找 k 条点不相交路径来连接图中任意给定的 k 个点且满足任何两条路径之间没有相邻节点;非干扰不相交 $s-t$ 路径问题则要求在图中寻找多条从 s 到 t 的路径且满足任何两条路径之间除了点 s 和 t 外没有其它的公共节点和相邻节点。然而,一般图中即使 2-pairs 非干扰不相交路径问题、两条非干扰不相交 $s-t$ 路径问题也都是 NP 难的^[10]。针对 k -pairs 非干扰不相交路径问题,文献^[11]

到稿日期:2013-05-25 返修日期:2013-07-21 本文受国家自然科学基金(61370003),教育部留学回国人员科研启动基金资助。

董高秀(1988—),男,硕士生,主要研究方向为算法设计;凌珊(1988—),女,硕士生,主要研究方向为算法设计;陈卫东(1968—),男,博士,教授,主要研究方向为图论和算法理论,E-mail:chenwd2007@hotmail.com(通信作者)。

提出了一个近似比为 $O(\sqrt{m})$ 的贪心算法 (m 为图的边数), 且证明了这是基于 m 为参数的最好近似比算法。针对两条非干扰不相交 $s-t$ 路径问题, 目前在无线网络中求解该问题的主要方法是基于网络中节点的物理位置表示的“隔离带”方法^[8,12]。然而, 很多情况下, 我们仅仅知道无线网络的拓扑结构, 并不知道网络中节点的物理位置。无线网络的拓扑结构可以表示为单位饼图 UDG(Unit Disk Graph), 这是一类无向图^[13]。因此, 针对一般无向图来研究寻找两条非干扰不相交 $s-t$ 路径的有效算法是有意义的。

针对 NP 难问题, 精确算法复杂度太高不现实, 通常寻求各种启发式算法求解。拟物拟人方法是处理 NP 难度问题的一种有效的途径, 已经在一些典型的组合最优化问题上得到成功应用^[14,15]。这类方法是将待求解的问题与大自然或人类社会的某些现象做类比, 借鉴自然界的规律和人们处理问题的经验和策略, 形式化为解决问题的算法^[16]。本文将针对一般无向图中如何有效构造两条非干扰不相交 $s-t$ 路径问题展开研究, 提出了一种拟人算法, 其基本思路是先用网络流方法构造两条点不相交 $s-t$ 路径, 如果这两条路径“太近”导致有干扰, 我们就采用一种拟人的策略来“拉开”它们, 力图使它们逐渐远离并最终形成两条非干扰不相交 $s-t$ 路径。

本文第 2 节介绍基本概念和相关算法; 第 3 节基于拟人策略提出了一个新算法; 第 4 节通过模拟实验评估拟人算法的性能; 最后是结论。

2 基本概念及相关工作

无线网络可以用无向图来建模(确切地讲, 是用单位饼图来建模)。本节介绍图、非干扰不相交路径的概念及相关记号。

设 G 是一个简单无向图(简称为图), 其节点集记作 $V(G)$, 边(链)集记作 $E(G)$ 。图 G 中与节点 v 相邻点的个数称作是点 v 的度。对于图 G 中任意两个不同的节点 s 和 t , 一条从 s 到 t 的路径称作是一条 $s-t$ 路径。设 P 是一条 $s-t$ 路径, 用 $P^* = P \setminus \{s, t\}$ 表示从 P 移去其端点 s 和 t 后剩下的路径。

路径的潜在干扰点集和边集: 设 P 是图 G 中任意一条 $s-t$ 路径, 记 $V(P)$ 为路径 P 上的点集, P 上所有点的邻点集记作 $N(P) = \{v \in V(G) \setminus V(P) \mid \exists u \in V(P) \text{ s. t. } (u, v) \in E(G)\}$ 。称 $IV(P) = V(P) \cup N(P)$ 为路径 P 的潜在干扰点集, 称 $IE(P) = \{(u, v) \in E(G) \mid u, v \in IV(P)\}$ 为路径 P 的潜在干扰边集。

非干扰不相交路径: 给定图 G 中 k 条路径 $P_1, P_2, \dots, P_k$, 其中 P_i 是一条 s_i-t_i 路径 ($1 \leq i \leq k$)。如果 $s_1, s_2, \dots, s_k, t_1, t_2, \dots, t_k$ 是图中 $2k$ 个互不相同的点, 且对于任意 $i \neq j$ ($1 \leq i, j \leq k$), 均有 $IV(P_i) \cap V(P_j) = \emptyset$ 且 $V(P_i) \cap IV(P_j) = \emptyset$, 则称路径 $P_1, P_2, \dots, P_k$ 为一组 k -pairs 非干扰不相交路径。特别地, 如果对于任意 i ($1 \leq i \leq k$), 均有 $s_i = s, t_i = t$, 且对任意 $i \neq j$ ($1 \leq i, j \leq k$), 有 $IV(P_i^*) \cap V(P_j^*) = \emptyset$ 且 $V(P_i^*) \cap IV(P_j^*) = \emptyset$, 则称路径 $P_1, P_2, \dots, P_k$ 为一组非干扰不相交的 $s-t$ 路径。

求两条非干扰不相交 $s-t$ 路径的现有算法有如下两类。

1) 基于“隔离带”的构造方法

文献[9] 提出了采用“隔离带”思想来构造两条非干扰不

相交 $s-t$ 路径的算法, 其基本思路是在节点 s 和 t 之间画一条直线, 根据两节点相互干扰的最大距离(用 L 表示)来设定一个“隔离带”, 即在该直线两侧与该直线相距 $L/2$ 处画两条平行线, 将两条平行线所包围的区域定为“隔离带”。显然, 如果两条路分别分布在“隔离带”两侧, 则它们必定是非干扰不相交路径(见图 1)。该算法需要知道网络中节点的物理位置, 且其成功率显然不高, 因为对靠边位置的 s 或 t 节点而言, 该算法不可能找到两条非干扰不相交 $s-t$ 路径, 即使它们是明显存在的。本文研究针对无线网络拓扑图(无向图)来构造非干扰不相交路径, 因此不考虑该类算法。

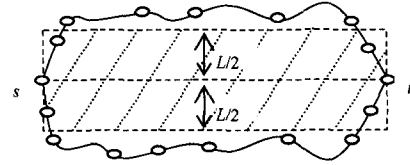


图 1 “隔离带”方法寻找两条非干扰不相交 $s-t$ 路径(阴影区为隔离带)

2) 基于 k -pairs 非干扰不相交路径的贪心算法的构造方法

文献[11]针对在一般图中构造 k -pairs 非干扰不相交路径给出了一个近似比为 $O(\sqrt{|E(G)|})$ 的贪心算法, 其主要思路是优先构造潜在干扰边集最小的路径, 然后从图中删除该路径的所有潜在干扰点集, 在当前图中继续寻找潜在干扰边集最小的下一条路径, 直到找到所要求的 k 条路径, 或者再也找不到下一条路径为止。显然, 先后找出的这些路径必定是非干扰不相交路径。

上述算法可启发产生出求解两条非干扰不相交 $s-t$ 路径的如下两个构造算法。

算法 1 首先寻找一条潜在干扰边集最小的 $s-t$ 路径, 删除该路径上所有中间节点及其邻点(s, t 除外), 然后在当前图中寻找下一条潜在干扰边集最小的 $s-t$ 路径。

算法 2 设 s 的所有邻点为 $s_1, s_2, \dots, s_p$, t 的所有邻点为 $t_1, t_2, \dots, t_q$, 则可形成 $pq(p-1)(q-1)/2$ 个不同的 2-pairs 非干扰不相交路径问题。对每一个这样的问题, 均可用上述 k -pairs 非干扰不相交路径的贪心算法求解, 只要其中一个这样的问题求解成功, 就得到两条非干扰不相交 $s-t$ 路径。

为了找到两条非干扰不相交的 $s-t$ 路径, 算法 2 可能会穷尽所有的 $pq(p-1)(q-1)/2$ 个 2-pairs 非干扰不相交路径问题来求解。因此算法 2 成功找到两条非干扰不相交的 $s-t$ 路径的机会通常应比算法 1 要大, 但其运行时间要长很多。容易看出, 算法 1 和算法 2 的时间复杂度分别为 $O(n^2)$ 和 $O(n^6)$, 其中 n 是图 G 的点数。

3 拟人算法

不难发现上述算法 1 和算法 2 有两点缺陷: 1) 一条路径一旦构造出来就固定了, 不能调整和修改; 2) 根据其贪心准则优先构造的是潜在干扰边集最小的路径, 而很多时候图中存在有两条所需路径时它们不一定满足这个准则, 其潜在的干扰边集可能较大。一个具体例子见图 2。在图 2 实例中, 潜在干扰边集最小的路径是中间那条路径 P_0 , 算法一旦优先构造出了路径 P_0 , 在删除该路径上所有中间节点及其邻点(s, t

除外)后,图中不再存在有下一条 $s-t$ 路径。但 P_1 和 P_2 显然是符合要求的两条非干扰不相交 $s-t$ 路径。

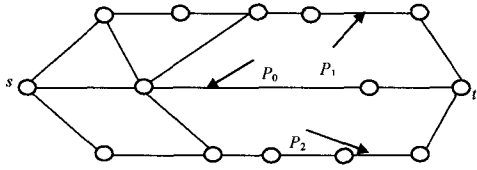


图2 两条非干扰不相交 $s-t$ 路径问题的一个实例

我们提出的拟人算法能在一定程度上避免上述这些缺陷。为了描述我们的算法,先引入干扰度的概念。

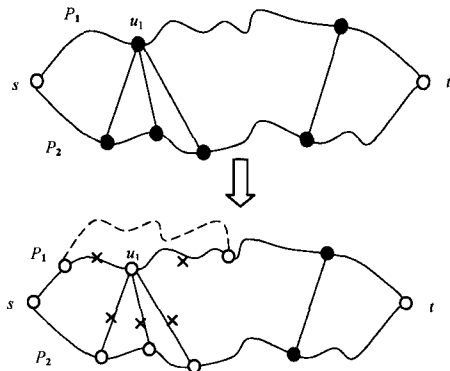
• 干扰度

给定图 G 中的两条点不相交 $s-t$ 路径 P_1 和 P_2 。对于路径 P_1 上的每个中间结点 $u(u \neq s, t)$,定义 u 的干扰度 $I(u)$ 为点 u 在路径 P_2 上的邻点个数(s 和 t 除外),计算如下: $I(u) = |\{v \in P_2 \mid (u, v) \in E(G)\} \setminus \{s, t\}|$ 。同样可定义路径 P_2 上的各中间结点的干扰度。定义路径 P_1 和 P_2 的干扰度 $I(P_1, P_2)$ 为这两条路径上所有中间结点的干扰度之和。

显然,如果路径 P_1 和 P_2 的干扰度为 0,则我们得到了两条非干扰不相交 $s-t$ 路径 P_1 和 P_2 ;否则这两条路径上一定都存在干扰度大于 0 的中间节点。

• 拟人算法

拟人算法的基本策略如下:首先采用网络流方法构造两条点不相交 $s-t$ 路径 P_1 和 P_2 。如果这两条路径“太近”纠缠在一起导致有干扰,则我们要想方设法去“拉开”它们,逐渐使得它们远离并最终形成两条非干扰不相交 $s-t$ 路径。具体来说,当两条路有干扰时,我们将调整其中一条路使之绕过某个干扰点(比如说最大干扰点),得到干扰点更少的两条路(见图3)。重复这个过程,直到得到两条非干扰不相交 $s-t$ 路径。如果调整不成功,则将其中一条路上的最大干扰点从图中移去。在新图上重启上述过程,直到成功找到所需要的两条路,或者当前图中不再存在两条点不相交 $s-t$ 路径为止。算法的非形式化描述见算法3。



路径上灰色节点是干扰点,调整后 P_1 绕过了干扰点 u_1

图3 拟人算法调整过程示意图

算法3 拟人算法

输入:无向图 $G=\langle V, E \rangle$,源点 $s \in V$,汇点 $t \in V$

输出:图 G 中两条非干扰不相交 $s-t$ 路径 P_1 和 P_2 ;或者没有找到

1. $\text{Flag} \leftarrow \text{TRUE}$
2. **WHILE** (TRUE)
3. 图 G 中用网络流方法找两条点不相交 $s-t$ 路径 P_1 和 P_2 。若不能

找到,则令 $\text{Flag} \leftarrow \text{FALSE}$ 并退出 **WHILE** 循环;否则进入 Step 4。

4. 计算干扰度 $I(P_1, P_2)$,并求 P_1 和 P_2 上的最大干扰点 u_1 和 u_2 。
如果 $I(P_1, P_2)=0$,则退出 **WHILE** 循环;否则进入 Step 5。
5. 调整路径 P_1 上的干扰点 u_1 。如果调整成功则得到新路径 P_1 ,返回 Step 4;否则进入 Step 6。
6. 调整路径 P_2 上的干扰点 u_2 。如果调整成功则得到新路径 P_2 ,返回 Step 4;否则进入 Step 7。
7. 在图 G 中移去 P_1 上的最大干扰点 u_1 ,所得图仍记作 G 。
8. **END-WHILE**
9. **IF** Flag **THEN**
10. 成功返回两条非干扰不相交 $s-t$ 路径 P_1 和 P_2
11. **ELSE**
12. 返回没有找到的信息
13. **END-IF**

算法中的关键步骤是 Step 3—Step 6,下面分别予以说明。

Step 3 将图 G 的每一条边看成方向相反的两条边,得到一个有向图 G' 。将图 G' 中每个节点 u 裂变为两点 u_1 和 u_2 ,并且添加从 u_1 到 u_2 的边。对于图 G' 原来每一条有向边 $\langle u, v \rangle$ 则得到有向边 $\langle u_2, v_1 \rangle$,由此得到有向图 G'' 。使用网络流方法求 G'' 中从 s_2 到 t_1 的大小为 2 的流,即得到两条边相交的 s_2-t_1 路径,它们对应于原图 G 中两条点不相交 $s-t$ 路径。这一步的时间为 $O(n^2)$, n 是 G 中点数。

Step 4 对于两条点不相交 $s-t$ 路径 P_1 和 P_2 ,计算其干扰度 $I(P_1, P_2)$ 。如果 $I(P_1, P_2)=0$,则我们得到了两条非干扰的 $s-t$ 点不相交路径 P_1 和 P_2 ;否则可找出每条路上干扰度最大的中间节点 u_1 和 u_2 。这个过程需要时间 $O(n^2)$ 。

Step 5 针对路径 P_1 上干扰点 u_1 ,调整策略如下:将路径 P_1 上除去 u_1 外的所有点以 u_1 为界分为两部分即 Part1 和 Part2,其中 Part1 包含 P_1 上的 u_1 之前的所有点,Part2 包含 P_1 上的 u_1 之后的所有点。从点集 $V(G) \setminus IV(P_2^*)$ 中取点依次扩展 Part1 和 Part2。每次将 Part1(Part2)中点的相邻点都加入到 Part1(Part2)中,直到 Part1 和 Part2 有公共点为止(调整成功),或者它们不能再扩展为止(调整失败)。调整成功时,干扰点 u 被绕过去了,所得两条点不相交的 $s-t$ 路径仍记作 P_1 和 P_2 ,其干扰点总数至少减少 1。这个过程称作是一个干扰点的调整,其时间为 $O(n^2)$ 。

Step 6 与 Step 5 相类似。

关于拟人算法,我们有下面定理。

定理 给定无向图 G ,设 $n=|V(G)|$ 。对于任意两个不同的节点 $s, t \in V(G)$,拟人算法的时间复杂度为 $O(n^4)$,且当输出两条路径 P_1, P_2 时,它们必定是两条非干扰不相交的 $s-t$ 路径。

证明:根据上述分析可知,一个干扰点的调整时间为 $O(n^2)$,而最多有 $O(n)$ 个干扰点需要调整,因此进入 Step 7 之前的时间为 $O(n^3)$ 。注意到,每当两条路径调整都不成功时,就进入 Step 7,从图中移去干扰节点 u_1 ,然后重新从构造两条点不相交 $s-t$ 路径开始。由于重启的次数是 $O(n)$,因此算法总的时间为 $O(n^4)$ 。

由上述 Step 3—Step 6 的讨论可知,一旦拟人算法输出路径 P_1, P_2 ,则它们必定是两条非干扰不相交的 $s-t$ 路径。证毕。

例 图 4 是拟人算法在图 2 实例上的调整过程。

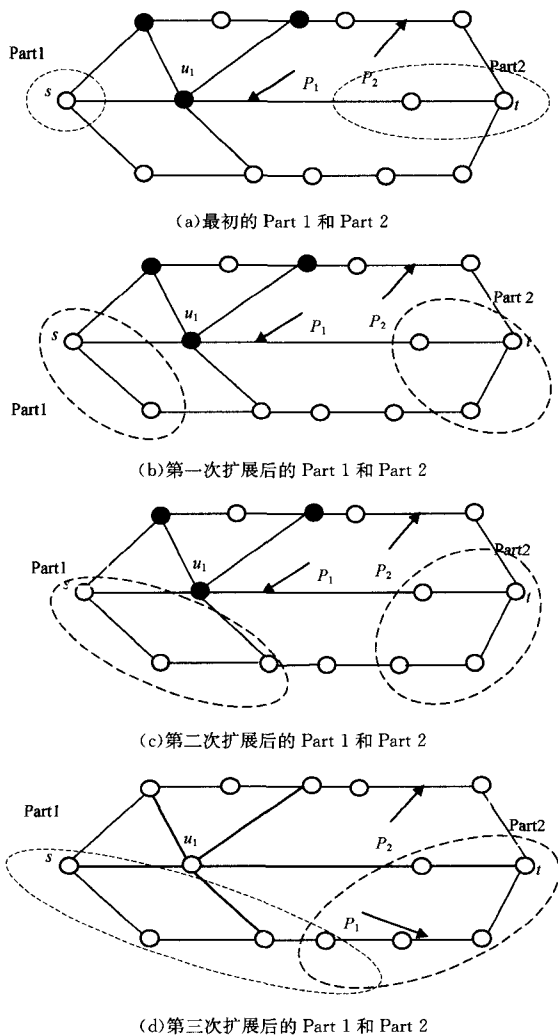


图 4 拟人算法在一个实例上的调整过程

4 模拟实验及分析

本节通过模拟实验来比较第 2 节中介绍的两种算法(即算法 1 和算法 2)和第 3 节的拟人算法(即算法 3)的性能,性能指标包括成功率、平均路长、平均运行时间这 3 方面。实验环境是 1.5GHz 主频的 PC 机,WIN7 下 C++ 编程。

实验中的图是随机产生的单位饼图(UDG),其方法如下:给定自然数 d 和 n ,令 $w = \sqrt{n\pi/d}$,在长为 w 的方形区域内随机产生 n 个点,且对点距不超过 1 的任意两点连边,由此得到一个 n 点的 UDG。由于方形区域面积为 w^2 ,因此区域内任意单位圆内的平均点数可近似看作 $n\pi/w^2$,这意味着该图的平均度点度为 $d = n\pi/w^2$ 。

对于每个随机产生的 UDG,随机选取其中的两点作为源点 s 和汇点 t ,当存在有两条点不相交的 $s-t$ 路径时,就作为一个问题实例(注:如果图中不存在两条点不相交的 $s-t$ 路径,必定也不存在两条非干扰不相交的 $s-t$ 路径)。实验中固定 $n=100$,依次对每个平均点度 $d=3,5,\dots,19$ 随机产生 100 个实例,各算法在这 100 个实例上运行时间的平均值作为算法在平均点度为 d 的图的实例上的平均运行时间;在这 100 个实例上的成功比率作为成功率,成功时每条路长的平均值作为平均路长。实验结果如图 5—图 7 所示。

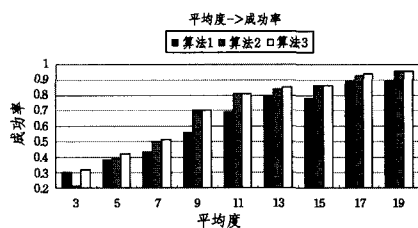


图 5 算法的成功率的比较结果

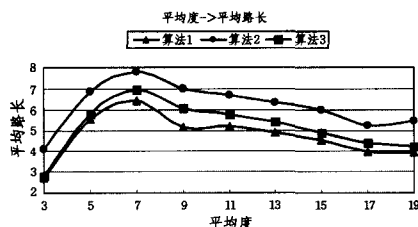


图 6 算法的平均路长的比较结果

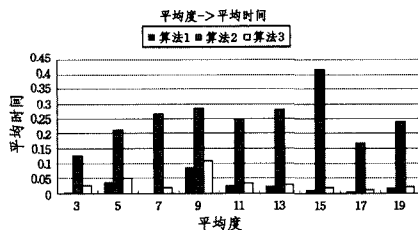


图 7 算法的运行时间的比较结果

从图 5—图 7 可看出,算法 1 的成功率最低,但其成功时的平均路长较短,运行时间最短;算法 2 的成功率很高,但其成功时的平均路长最长,且运行时间也最长;算法 3 的成功率最高,与算法 2 相差不大,但要高于算法 1 大约 10%,其成功时的平均路长要小于算法 2,运行时间大致与算法 1 相当,都远远低于算法 2。究其原因,算法 1 是一个贪心法,侧重于优先构造长度较短的路径,其时间复杂度为 $O(n^2)$,因此速度快,成功时平均路长短,但成功率不高;算法 2 可看作是算法 1 的某种穷举形式,时间复杂度高达 $O(n^6)$,因此速度慢,但成功率高,成功时平均路长较长;算法 3 尽管需要多次调整,但在实验中我们发现调整次数通常为 3—5 次,可看作常数,这样其时间复杂度可看作 $O(n^3)$,比较接近算法 1,由于其调整效果较好,成功率接近算法 2,同时由于调整时次数不多,成功时平均路长较短。因此,实验结果与理论分析是相吻合的。由此看来,本文提出的拟人算法兼顾了前两种算法的优点而避免了其缺陷,即在较短时间内仍以相对较高的概率返回一个解。

结束语 无向图中的任意给定的源点 s 和汇点 t 之间的两条非干扰不相交 $s-t$ 路径构造问题是 NP 难度的。本文提出了一种求解该问题的启发式算法(称之为拟人算法),从两条点不相交的 $s-t$ 路径开始,采用一种拟人策略逐步调整它们使其干扰程度逐渐减小,力图使得它们最终被调整成为两条非干扰不相交 $s-t$ 路径。模拟实验表明,从成功率、平均路长、平均运行时间等方面综合来看,拟人算法要优于现有的算法。拟人算法不需要知道无线网络中节点的几何位置表示,可适用于一般图,因此具有实用价值。未来的工作中将进一步优化拟人算法,并将其推广到求解更一般的 k -pairs 非干扰不相交路径问题上。

参考文献

- [1] Rabin M O. Efficient dispersal of information for security, load balancing, and fault tolerance[J]. Journal of the ACM, 1989, 36(2): 335-348
- [2] Hsu D F, Lyuu Y D. A graph-theoretical study of transmission delay and fault tolerance [J]. International Journal of Mini and Microcomputers, 1994, 16(1): 35-42
- [3] Kleinberg J. Approximation algorithms for disjoint paths problems[D]. MIT, Cambridge, MA, May 1996
- [4] Li C, McCormick T, Simich-Levi D. The complexity of finding two disjoint paths with min-max objective function [J]. Discrete Applied Mathematics, 1989, 26(1): 105-115
- [5] Xu D, Chen Y, Xiong Y, et al. On the complexity of and algorithms for finding the shortest path with a disjoint counterpart [J]. IEEE/ACM Transactions on Networking, 2006, 14(1): 147-158
- [6] 张品, 章坚武, 李乐民, 等. QoS 约束下的链路分离路径问题研究 [J]. 通信学报, 2006, 27(6): 36-42
- [7] 熊轲, 裴正定, 郭宇春, 等. 多约束最短链路分离路径精确算法 [J]. 软件学报, 2010, 21(7): 1744-1757
- [8] 方效林, 石胜飞, 李建中. 无线传感器网络一种不相交路径路由算法[J]. 计算机研究与发展, 2009, 46(12): 2053-2061
- [9] Voigt T, Dunkels A, Braun T. On-Demand construction of non-interfering multiple paths in wireless sensor networks[C]//Proceedings of the 2nd Workshop on Sensor Networks at Informatik 2005. Bonn, Germany, 2005: 150-154
- [10] Kawarabayashi K, Kobayashi Y. The induced disjoint paths problem[C]//Proceedings of the 13th Integer Programming and Combinatorial Optimization (LNCS5035). Bertinoro, Italy, Springer, 2008: 47-61
- [11] Zhang Ke-jia, Gao Hong, Li Jian-zhong. Finding multiple induced disjoint paths in general graphs[J]. Information Processing Letters, 2011, 111: 1022-1026
- [12] Teo J Y, Ha Y, Tham C K. Interference-Minimized Multipath Routing with Congestion Control in Wireless Sensor Network for High-Rate Streaming[J]. IEEE Transactions on Mobile Computing, 2008, 7: 1124-1137
- [13] Marathe M V, Breu H, Hunt H B, et al. Simple heuristics for unit disk graphs[J]. Networks, 1995, 25(2): 59-68
- [14] 黄文奇, 金人超. 求解 SAT 问题的拟物拟人算法—Solar [J]. 中国科学(E 辑), 1997, 27(2): 179-186
- [15] Huang Wen-qi, He Kun. A pure quasi-human algorithm for solving the cuboid packing problem[J]. Science in China Series F: Information Sciences, 2009, 52(1): 52-58
- [16] 黄文奇, 许如初. 近世计算理论导引—NP 难度问题的背景、前景以及其求解算法研究[M]. 北京: 科学出版社, 2004: 47-70

(上接第 66 页)

前高效能计算相应的问题求解中。但是在计算任务和体系结构通过重构实现异构特征匹配所造成的重构代价方面分析尚有不足, 如何保证以最小重构代价实现划分后异构特征的完全匹配是我们今后研究的重点。

参考文献

- [1] Freund R F. Optimal selection theory for superconcurrency [C]//Proceedings of Conference on Supercomputing. ACM: New York, 1989: 699-703
- [2] Khokhar A, Prasanna V K, Shaaban M E. Heterogeneous computing: challenges and opportunities [J]. Computer, 1993, 26(6): 18-27
- [3] Compton K, Hauch S. Reconfigurable computing: a survey of systems and software[J]. ACM Computing Surveys, 2002, 34(2): 171-210
- [4] DeHon A, Wawrzyniec J. Reconfigurable computing: what, why, and implications for design automation[C]//Proc of 36th Annual ACM/IEEE Design Automation Conference. Louisiana, USA: ACM Press, 1999: 610-615
- [5] 姜晶菲, 唐玉华, 刘福东, 等. 可重构异构系统结构研究[J]. 计算机技术与发展, 2009, 19(1): 25-28
- [6] Bruce H, Tamara G. Graph partitioning models for parallel computing[J]. Parallel Computing, 2000, 26: 1519-1534
- [7] 沈轶伟, 曾国荪. 异构计算中一种图的非均衡划分算法[J]. 计算机科学, 2006, 33(6): 260-263
- [8] Selvakumaran N, George K. Multiobjective hypergraph partitioning algorithms for cut and maximum subdomain-degree minimization[J]. IEEE Transactions on Computer Aided Design of Integrated Circuits and System, 2006, 25(3): 504-517
- [9] Jiang Y, Wang J. Temporal partitioning data flow graphs for dynamically reconfigurable computing[J]. IEEE Transactions on very Large Scale Integration System, 2007, 15(12): 1351-1361
- [10] Verbaauwhede I, Schaumont P. The happy marriage of architecture and application in next-generation reconfigurable systems [C]//Proceedings of the 1st conference on computing frontiers 2004. Ischia, Italy, 2004: 363-376
- [11] 郝水侠, 曾国荪, 谭一鸣. 计算任务与体系结构匹配的异构计算可扩展性分析[J]. 电子学报, 2010, 38(11): 2585-2589
- [12] Henkel J. An approach to automated hardware/software partitioning using a flexible granularity that is driven by high-level estimation techniques [J]. IEEE Transactions on very Large Scale Integration System, 2001, 9(2): 273-289
- [13] Fukushi M, Horiguchi S. A self-reconfigurable hardware architecture for mesh arrays using single/double vertical track switches[J]. IEEE Transactions on Instrumentation and Measurement, 2004, 53(2): 357-367
- [14] Neeb C, Wehn N. Designing efficient irregular networks for heterogeneous systems-on-chip[J]. Journal of Systems architecture, 2008, 54(2): 384-396
- [15] Kapre N, Dehon A. SPICE²: spatial processors interconnected for concurrent execution for accelerating the SPICE circuit simulator using an FPGA[J]. IEEE Transactions on Computer-aided Design of Integrated circuits and Systems, 2012, 31(1): 9-22
- [16] Senar M A, Ripoll A, et al. Clustering and reassignment based mapping strategy for message passing architectures[J]. Journal of Systems Architecture, 2003, 48(8-10): 267-283
- [17] Juhasz Z, Turns S J. A new heuristic for the process-processor mapping problems [C]//Proceedings of the Third Austrian-Hungarian Workshop on Distributed and parallel Systems. 2000: 91-94
- [18] 郝水侠, 曾国荪, 谭一鸣. 一种基于 DAG 图的异构可重构任务划分方法[J]. 同济大学学报, 2011, 39(11): 1693-1697
- [19] <http://helsinki.ee.princeton.edu/~vallerio/tgff/>