

认知无线电网络频谱分配与协作集划分算法^{*}

杨 威¹⁺, 班冬松¹, 梁维发², 窦文华¹

¹(国防科学技术大学 计算机学院, 湖南 长沙 410073)

²(Research School of Computer Science, Australian National University, Canberra, ACT 0200, Australia)

Algorithms for Joint Spectrum Allocation and Cooperation Set Partition in Cognitive Radio Networks

YANG Wei¹⁺, BAN Dong-Song¹, LIANG Wei-Fa², DOU Wen-Hua¹

¹(School of Computer, National University of Defense Technology, Changsha 410073, China)

²(Research School of Computer Science, Australian National University, Canberra, ACT 0200, Australia)

+ Corresponding author: E-mail: weiyang@nudt.edu.cn

Yang W, Ban DS, Liang WF, Dou WH. Algorithms for joint spectrum allocation and cooperation set partition in cognitive radio networks. *Journal of Software*, 2012, 23(1): 122–139. <http://www.jos.org.cn/1000-9825/4077.htm>

Abstract: The coexistence of multi-primary users and multi-secondary users in cooperative cognitive radio networks motivate the study to propose a joint spectrum allocation and cooperation set partition problem, which so far has not been addressed before. The problem is formulated as a 0-1 integer non-linear programming model. Due to its NP-hardness, the study proposes a suboptimal Centralized Genetic Algorithm (CGA) to show its convergence by modeling it as a homogeneous finite Markov chain. The study then extends CGA to a fully Distributed Genetic Algorithm (DGA) that consists of two phases. The core techniques include a minimum dominate set based cluster partition, a spectrum pre-allocation algorithm in phase 1, and an inter-cluster cooperation set negotiation and cluster fitness refinement algorithm in phase 2. A Fast-Convergent DGA (FDGA) is also devised to reduce the system configuration time. Extensive experiments by simulations demonstrate that in terms of the fitness that reflects the performance of the proposed algorithms: (1) CGA is shown to perform as well as 92% of the optimal solution by brutal search under small network sizes; (2) As the network size increases, due to the massive search space CGA has to deal, DGA and FDGA instead outperform CGA with 20% on average when achieving the same algorithm termination condition; (3) FDGA delivers similar results as DGA while reducing the configuration time significantly, which is more suitable for large-scale networks.

Key words: cooperative cognitive radio network; spectrum allocation; cooperation set partition; distributed genetic algorithm; homogenous finite Markov chain

摘 要: 针对协作认知无线网络中较为复杂的多主用户与多次级用户共存场景,提出联合频谱分配与协作集划分问题,并将该问题形式化描述为整数 0-1 非线性规划问题,证明其是 NP-hard 的.首先,设计了集中式的遗传算法

* 基金项目: 国家自然科学基金(60603061, 60903223)

收稿时间: 2010-10-18; 定稿时间: 2011-07-01

CGA(centralized genetic algorithm)对问题求解,对该算法进行齐次有限马尔可夫链建模并对其全局收敛性进行了分析;随后,提出了一种包含两阶段的分布式遗传算法 DGA(distributed genetic algorithm),包括基于最小支配集的分簇与频谱预分配阶段和簇间协作集协商与簇内适应值精化阶段.此外,还提出一种快速收敛的 DGA 算法(fast-convergent DGA,简称 FDGA)缩短分布式算法运行时间.仿真实验结果表明,根据能反映出算法性能的适应值结果对各算法进行比较:(1) 小规模网络下 CGA 获得的解平均为通过穷举算法得到的最优值的 92%;(2) 随着网络规模的扩大,由于 CGA 搜索空间增大,DGA,FDGA 在达到相同停机条件时获得的适应值比 CGA 提高约 20%;(3) 与 DGA 相比,FDGA 虽能得到与 DGA 相近的结果,但却大大缩短了算法收敛的时间,更适应于大规模网络应用.

关键词: 协作认知无线网络;频谱分配;协作集划分;分布式遗传算法;有限齐次马尔可夫链

中图法分类号: TP393

文献标识码: A

认知无线电(cognitive radio,简称 CR)最早由 Joseph Mitola III 在其博士论文中提出^[1],网络中的次级用户(secondary user,简称 SU)利用 CR 技术可以感知网络中的频谱空洞(spectrum hole),并采用动态频谱接入方式,使用原来只有主用户(primary user,简称 PU)才能使用的频谱资源,从而能提高频谱利用率,缓解频谱资源短缺的问题.CR 的提出激发了人们对认知无线网络(cognitive radio network,简称 CRN)的大量研究工作,其中最基本的问题是 CRN 的频谱共享模型及不同模型下的资源优化问题.总体来说,频谱共享模型可分为两类^[2]:

(1) Underlay 模型.SU 可与 PU 同时使用同一频谱资源,但必须依干扰温度模型来控制 SU 的发送功率,以避免对 PU 的干扰;

(2) Overlay 模型.该模型不强调功率控制,SU 能在时间-频率两维空间寻找频谱空洞,并对可用频谱加以利用.

基于 Overlay 模型,Zhang 等人提出了协作认知无线网络(cooperative CRN,简称 CCRN)的概念^[3].在该网络中,多个 SU 不仅可以根据权重或出价分时使用 PU 的频谱,PU 也可利用信道条件较好的 SU 在物理层采用协作传输技术来提高 PU 链路的传输速率.现有研究基本均假设网络中存在单 PU 与多 SU 协作的简单情况,而在实际网络中,多 PU 与多 SU 往往是共存的,SU 的数目往往多于 PU 的数目.在这种更为复杂的情景中,每个 PU 可利用某些 SU 进行协作传输,每个 SU 均有使用频谱的需求并可为 PU 提供协作.那么,如何为每个 PU 分配频谱、如何指派 PU 与 SU 之间的协作关系,使得在规定的时段内,PU 与 SU 利益的分配满足一定(例如公平性)的要求?解决该问题并不是对单 PU、多 SU 共存时问题的扩展,而是一个更加复杂的组合优化问题.

针对提出的问题,本文进行了以下几方面工作:

- (1) 首先,针对 CCRN 中多 PU、多 SU 共存的情况,提出了 CCRN 联合频谱分配与协作集划分问题 JSA-CSP,给出了该问题的 0-1 非线性规划描述,并证明该问题是 NP-hard 的;
- (2) 提出一种集中式遗传算法 CGA(centralized genetic algorithm)对 JSA-CSP 求解,设计了 CGA 算法的基本要素,对算法进行了齐次有限马尔可夫链建模,并证明了 CGA 的全局收敛性;
- (3) 针对大规模无中控节点的 CCRN 网络,提出一种包含两个阶段的分布式遗传算法 DGA(distributed genetic algorithm),分别是基于最小支配集的 PU 节点分簇与频谱预分配阶段和簇间协作集协商与簇内适应值精化阶段.此外,还提出了一种快速收敛的算法 FDGA(fast-convergent DGA),以缩短分布式算法运行的时间;
- (4) 最后,通过仿真实验验证了本文提出的算法的有效性和收敛性.通过对能反映算法性能的适应值结果进行比较,得到如下结论:① 在小规模网络下,CGA 得到的适应值平均可达穷举算法得到的最优解的 92%;② 随着网络规模的扩大,由于 CGA 的搜索空间呈指数增长,DGA 和 FDGA 逐渐显示出性能优势,在达到相同停机条件时,获得的适应值平均比 CGA 提高约 20%;③ 特别地,与 DGA 相比,FDGA 不仅能得到与 DGA 相近的结果,而且还大大缩短了算法收敛的时间,更适合在大规模网络中应用.

本文第 1 节介绍 CCRN 的相关研究工作.第 2 节给出网络协作模型,提出并形式化描述 JSA-CSP(joint

spectrum allocation and cooperation set partition problem)问题.第3节对 JSA-CSP 问题进行分析.第4节给出集中式遗传算法 CGA,并证明其全局收敛性.第5节详述分布式算法 DGA 和 FDGA.第6节通过仿真实验对 CGA, DGA 和 FDGA 的算法性能进行比较.第7节总结全文.

1 相关工作

认知无线网络中的协作方式源于两种技术:协作中转技术和基于最大比例合并(maximum radio combining,简称 MRC)^[4]的协作传输技术.(1) 协作中转技术,即两个或多个 CR 节点之间存在一个中转节点,由于它的存在并通过对网络中节点的传输进行调度,能使网络资源利用率有所增加.Jia 等人^[5]提出了利用中转节点并对各节点传输进行调度,来解决 CRN 中节点由于地理位置不同而导致的节点可用频谱不均衡与链路流量的不均衡问题.Almasaeid 和 Kamal^[6]进一步提出在中转节点采用网络编码技术来减小 CRN 组播传输延迟.Xu 和 Li^[7]则利用中转节点进行 XOR 编码操作以降低频谱的使用数量,从而提高了基于 OFDMA 的 CRN 的频谱利用率.因此,如何指派中转节点对节点传输进行调度、功率控制及提供 QoS 保证等,以满足其优化目标,都是这类优化问题需要考虑的因素^[8].(2) 基于 MRC 的协作传输技术,也就是提供协作者与被协作者通过并发传输相同的数据,使被协作者对应的接收端信噪比提升,从而增大被协作者传输链路的传输速率的技术.本文的网络传输模型正基于这种技术.Simeone 等人^[9]首先提出 PU 可将频谱租赁给若干个 SU,当且仅当这些 SU 可以与 PU 采用 MRC 技术以增加 PU 的传输速率.Zhang 等人^[3]随后采用博弈论的思想建立了斯坦科尔伯格(Stackelberg)寡头竞争模型,即,PU 先决定它使用某频谱的时间,随后,多个 SU 决定它们的出价以获取该频谱的使用时间,双方都以最大化自身效用值为最终目的.注意到,以上研究都假设网络中仅存在单个 PU 与多个 SU 的情况,而没有考虑到 CCRN 中往往是多 PU 与多 SU 共存的问题.虽然 Xu 和 Li^[10]考虑了多 PU 多 SU 的情况,并提出一种灵活的频谱分配方式 FLEC,但他们假设一个 SU 仅能与一个 PU 协作,使得能使用基于分布式二部图匹配算法来进行分布式实现.与上述研究工作不同,本文假设 CCRN 中多 PU 与多 SU 共存,并且 SU 的数量可多于 PU 的数量,多个 SU 因此可与同一 PU 协作,网络中的每个 SU 根据权重或出价都有使用某 PU 频谱的权利.那么,如何为每个 PU 分配频谱、如何匹配 PU 与 SU 之间的协作关系,使得在规定的时间内,PU 与 SU 利益的分配满足一定要求(本文讨论公平性要求)呢?需要说明的是,若固定每个 PU 使用的频谱,则该问题演变为一个如何指派多 PU 与多 SU 协作关系的多指派问题^[11].但在文献[11]中,每个指派带来的利益是节点数量的线性函数,而在 CCRN 中,这种利益函数通常是非线性的.我们的工作则综合考虑对 PU 初始频谱的分配和 PU 与 SU 协作关系的指派.

2 模型与问题描述

考虑一个由 M 条 PU 通信链路 $\{PT_i \rightarrow PR_i\}_{i=1}^M$ 、 N 条 SU 通信链路 $\{ST_j \rightarrow SR_j\}_{j=1}^N$ 组成的协作认知无线网络,其中: $M < N$; PT_i, PR_i 为第 i 条 PU 通信链路的发送、接收节点, $1 \leq i \leq M$; ST_j, SR_j 为第 j 条 SU 通信链路的发送、接收节点, $1 \leq j \leq N$.令 $PT = \{PT_i | 1 \leq i \leq M\}$ 为 PU 发送节点集合, $ST = \{ST_j | 1 \leq j \leq N\}$ 为 SU 发送节点集合(若不特别指出,本文后续所指的主用户(PU)节点特指 PU 的发送节点,次级用户(SU)节点特指 SU 的发送节点).设空间连续频谱段被划分为 M 段互不重叠且带宽均等的可用子频谱(又称子通道)供这 M 条 PU 通信链路使用,用 k 表示频谱序号, $1 \leq k \leq M$; 用 $C_i = k$ 表示 PT_i 选择频谱 k 进行通信.若不加以特殊说明,变量 i, j, k 总在 $1 \leq i \leq M, 1 \leq j \leq N, 1 \leq k \leq M$ 范围内取值.下面介绍本文出现的主要符号:

PT_i, PR_i : 第 i 条 PU 链路的发送、接收节点;	C_i : 分配给 PT_i 的频谱序号;
ST_j, SR_j : 第 j 条 SU 链路的发送、接收节点;	R_j^s : ST_j 获得的平均速率;
$PT: \{PT_i 1 \leq i \leq M\}$ PU 发送节点集合;	D_i^p, D_j^s : PT_i, ST_j 的最大传输半径;
$ST: \{ST_j 1 \leq j \leq N\}$ SU 发送节点集合;	ω_j : ST_j 的权重或出价;
PCS_i : PT_i 的潜在协作集;	ϕ_i, η_i : PT_i 与 PCS_i 在协作过程中的时间参数;
RCS_i : PT_i 的真协作集;	P_i^p, P_j^s : PT_i 与 ST_j 的发送功率;

$\mathbf{A}:\{a_{ij}\}_{M \times N}$ 潜在协作矩阵;
 $\bar{\mathbf{A}}:\{\bar{a}_{ij}\}_{M \times N}$ 真协作矩阵;
 $\mathcal{Q}:(\theta_k)_{M \times M}$ 频谱分配矩阵;
 $R_i^p(RCS_i)_{RCS_i \neq \emptyset}:PT_i$ 进行协作传输时获得的平均速率;
 $R_i^p(RCS_i)_{RCS_i = \emptyset}:PT_i$ 未进行协作传输时获得的平均速率; N_0 :噪声功率谱密度,设为常数.

定义 1(潜在协作集,真协作集). 设在单位时间 T 内任取 $ST_j \in ST$,它根据其出价或权重成比例地获得已经分配给 $PT_i \in PT$ 的频谱 $C_i=k$ 的使用时间.这就形成使用频谱 k 的 SU 集合,称为 PT_i 的潜在协作集(potential cooperation set of PT_i ,简称 PCS_i).在 PCS_i 中, PT_i 又可以选择若干 SU 并采用 MRC 技术进行协作传输,以提高 PT_i 的传输速率.称这些参与协作的 SU 为 PT_i 的真协作集(real cooperation set of PT_i ,简称 RCS_i),即真正帮助了 PT_i .显然, $RCS_i \subseteq PCS_i$ 且 RCS_i 可为 $\emptyset$,即 PT_i 不采用协作传输方式.不失一般性,设单位时间 $T=1$.

定义 2(潜在协作矩阵,真协作矩阵). 定义网络的潜在协作矩阵为 $\mathbf{A}=(a_{ij})_{M \times N}$,若 ST_j 利用分配给 PT_i 的频谱进行传输,则 $a_{ij}=1$,反之, $a_{ij}=0$.因此, $PCS_i=\{ST_j|a_{ij}=1\}$.类似地,定义真协作矩阵 $\bar{\mathbf{A}}=(\bar{a}_{ij})_{M \times N}$,它标识 PT_i 是否选择 ST_j 进行协作传输.显然,任取 $i,j, \bar{a}_{ij} \leq a_{ij}$.因此, $RCS_i = \{ST_j | \bar{a}_{ij} = 1\}$.

定义 3(协作过程). 设 $C_i=k$ 为分配给 PT_i 的频谱, PCS_i, RCS_i 也已知.首先,将单位时间按比例被划分为 ϕ_i 和 $1-\phi_i$ 两个时间段.

(1) ϕ_i 阶段

1) 若 $RCS_i \neq \emptyset$,则 ϕ_i 段又被划分为 $\phi_i \gamma_i$ 和 $\phi_i(1-\gamma_i)$ 两个时间段, $0 \leq \gamma_i < 1$.

- $\phi_i \gamma_i$ 段: PT_i 使用频谱 k 将数据广播给 RCS_i 中 SU 节点.为了使 RCS_i 中所有节点都能成功接收数据,在 $\phi_i \gamma_i$ 段, PT_i 的发送速率 R_{i1}^p 应由 PT_i 到 RCS_i 中节点的通道增益中最小者决定.即

$$R_{i1}^p = \log_2 \left(1 + \frac{\arg \min_{ij} \bar{a}_{ij} g_{ij}^k P_i^p}{N_0} \right) \quad (1)$$

- $\phi_i(1-\gamma_i)$ 段: PT_i 与 RCS_i 中的所有 SU 节点使用频谱 k 再将数据传输给 PR_i, PR_i 接收数据时,在物理层采用 MRC 技术,因此得到 $\phi_i(1-\gamma_i)$ 段速率为

$$R_{i2}^p = \log_2 \left(1 + \frac{P_i^p g_{ii}^k}{N_0} + \sum_{j=1}^N \frac{\bar{a}_{ij} P_j^s g_{ji}^k}{N_0} \right) \quad (2)$$

因此,由公式(1)、公式(2)可知,当 $RCS_i \neq \emptyset$ 时, PT_i 在单位时间内的传输速率为

$$R_i^p(RCS_i)_{RCS_i \neq \emptyset} = \min\{\phi_i \gamma_i R_{i1}^p, \phi_i(1-\gamma_i) R_{i2}^p\} \quad (3)$$

2) 若 $RCS_i = \emptyset$,则在 ϕ_i 段内, PT_i 不与任何 SU 协作,直接向 PR_i 传输,传输速率为

$$R_i^p(RCS_i)_{RCS_i = \emptyset} = \phi_i \log_2 \left(1 + \frac{P_i^p g_{ii}^k}{N_0} \right) \quad (4)$$

显然,我们希望为 PT_i 找到一个 $RCS_i \subseteq PCS_i$ 且 $RCS_i \neq \emptyset$,使得 $\max R_i^p(RCS_i)_{RCS_i \neq \emptyset} > R_i^p(RCS_i)_{RCS_i = \emptyset}$,即希望 PT_i 与 RCS_i 中的节点进行协作传输获得的传输速率比不进行协作传输获得的传输速率要高.若找不到,则 PT_i 只能采取直接传输方式.综上, PT_i 在单位时间内的传输速率为

$$R_i^p = \max\{R_i^p(RCS_i)_{RCS_i \neq \emptyset}, R_i^p(RCS_i)_{RCS_i = \emptyset}\} \quad (5)$$

(2) $1-\phi_i$ 段: PT_i 的潜在协作集 PCS_i 中的每个 SU 根据自身权重或出价高低 ω_j 成比例地获取频谱 C_k 的使用时间,任取 $ST_j \in PCS_i$,它获取的时间为 $t_j = (1-\phi_i)\omega_j / \sum_{k=1}^N a_{ik}\omega_k$. 其中, ST_j 的传输速率为

$$R_j^s = (1-\phi_i) \frac{\omega_j}{\sum_{k=1}^N a_{ik}\omega_k} \log_2 \left(1 + \frac{P_j^s g_{jj}^k}{N_0} \right) \quad (6)$$

图 1 给出了当 PT_i 的 PCS_i 为 $\{ST_1, ST_2, ST_3\}$ 时的协作过程的实例.图 1(a)为 $RCS_i = \{ST_1, ST_2\}$ 时的情况;图 1(b)

为 $RCS_i = \emptyset$ 时的情况,这时, PT_i 将独自完成传输过程.

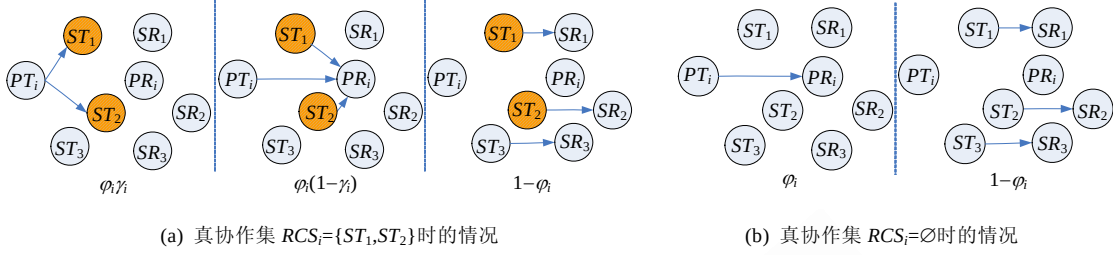


Fig.1 An instance of the cooperation process of PT_i with $PCS_i = \{ST_1, ST_2, ST_3\}$
图 1 PT_i 的潜在协作集为 $PCS_i = \{ST_1, ST_2, ST_3\}$ 时协作过程的实例

定义 4(频谱分配矩阵). 频谱分配矩阵为 $\Theta = (\theta_{ik})_{M \times M}$, 若给 PT_i 分配的频谱为 $C_i = k$, 则 $\theta_{ik} = 1$; 否则, $\theta_{ik} = 0$.

定义 5(联合频谱分配与协作集划分问题 JSA-CSP). 给定 PU 发送节点集合 PT 、SU 发送节点集合 ST 、PU 集合、SU 集合和空间频谱集合, 设网络中所有节点都参与协作过程. 联合频谱分配与协作集划分问题研究如何为每个 PU 分配频谱、如何匹配 PU 与 SU 节点之间的协作关系, 使得在单位时间段内, 它们之间利益的分配满足公平性的要求. 即, 找出潜在协作矩阵 $\mathbf{A}$ 、真协作矩阵 $\bar{\mathbf{A}}$ 、频谱分配矩阵 Θ 和时间优化参数向量 $\Phi = (\varphi_i)_{1 \times M}$ 与 $\mathbf{H} = (\gamma_i)_{1 \times M}$, 使得在单位时间内, 衡量资源分配是否公平的目标函数最大化. 其形式化描述如下:

Determine:

$$\mathbf{A} = (a_{ij})_{M \times N}, \bar{\mathbf{A}} = (\bar{a}_{ij})_{M \times N}, \Theta = (\theta_{ik})_{M \times M}, \Phi = (\varphi_i)_{1 \times M}, \mathbf{H} = (\gamma_i)_{1 \times M}.$$

Maximize:

$$\max(\sum_{i=1}^M \ln(R_i^p) + \sum_{j=1}^N \ln(R_j^s)) \quad (7)$$

Subject to:

$$N - M + 1 \geq \sum_{j=1}^N a_{ij} \geq 1 \quad (8)$$

$$\sum_{i=1}^M a_{ij} = 1 \quad (9)$$

$$\sum_{i=1}^M \theta_{ik} = 1 \quad (10)$$

$$\sum_{k=1}^M \theta_{ik} = 1 \quad (11)$$

$$a_{ij}, \bar{a}_{ij}, \theta_{ik} \in \{0, 1\} \quad (12)$$

$$\bar{a}_{ij} \leq a_{ij} \quad (13)$$

$$0 < \varphi_i < 1, 0 \leq \gamma_i < 1 \quad (14)$$

$$\forall i, k \in \{1, \dots, M\}, \forall j \in \{1, \dots, N\}.$$

注意到, 这是一个 0-1 非线性规划问题. 目标函数(7)表示资源分配的比例公平性指标最大化. 该函数的意义在于, 设存在 $\mathbf{A}, \bar{\mathbf{A}}, \Theta, \Phi, \mathbf{H}$ 及它们确定的 R_i^p, R_j^s , 使目标函数最大化, 若任取其他可行分配 $\mathbf{A}', \bar{\mathbf{A}}', \Theta', \Phi', \mathbf{H}'$ 及它们确定的 $R_i'^p, R_j'^s$, 则 $\sum_{i=1}^M (R_i'^p - R_i^p) / R_i^p + \sum_{j=1}^N (R_j'^s - R_j^s) / R_j^s \leq 0$. 即, 所有节点可获得的平均速率的比例变化之和为 0 或为负^[12]. 约束(8)表明, 每个 PU 至少要为一个 SU 提供频谱, 而最多可为 $N - M + 1$ 个 SU 次级用户提供频谱, 否则, 有些 PU 则未被利用. 约束(9)表明, 每个 SU 只能向一个 PU 提出频谱使用要求. 约束(8)、约束(9)进一步表明, 所有 PU 的潜在协作集须满足 $\bigcap_{i=1}^M PCS_i = \emptyset, \bigcup_{i=1}^M PCS_i = ST$. 约束(10)、约束(11)表明, 每个 PU 预先分配一个频谱, 每个频谱将被分配给一个 PU. 约束(12)和约束(14)分别为 $\mathbf{A}, \bar{\mathbf{A}}, \Theta, \Phi$ 中元素的取值约束. 约束(13)则表明, 每个 PU 的真协作集为潜在协作集的真子集.

3 问题分析

为简化起见,我们首先对问题进行分析.

定理 1. 给定 PT_i 的潜在协作集为 PCS_i , 分配给它的频谱为 $C_i=k$, 则存在时间复杂度为 $O(n^2)$ 的算法能确定真协作子集 RCS_i 及 ϕ_i, γ_i , 使函数最大化, 其中, $n=|PCS_i|$.

证明: 该定理的证明思想是: 先证任取一个 $RCS_i \subseteq PCS_i$, 存在 ϕ_i 和 γ_i , 使函数 f_i 在该 RCS_i 时最大化; 接着给出算法, 用于确定哪个 $RCS_i \subseteq PCS_i$ 能使 f_i 的取值最大化.

(1) 显然, 要么 $RCS_i \neq \emptyset$, 要么 $RCS_i = \emptyset$ 成立.

1) 若 $RCS_i \neq \emptyset$, 则 PT_i 利用频谱 k 与 RCS_i 中的节点进行协作传输. 根据公式(1)~公式(3)、公式(6), 有

$$f_i(C_i, PT_i, PCS_i) = \ln(\min\{\phi_i \gamma_i R_{i1}^p, \phi_i (1-\gamma_i) R_{i2}^p\}) + \sum_{ST_j \in PCS_i} \ln \left(\frac{(1-\phi_i) \omega_j}{\sum_{l=1}^N a_{il} \omega_l} \log_2 \left(1 + \frac{P_j^s g_{ij}^k}{N_0} \right) \right) \quad (1)$$

成立. 由于假设所有节点的发送功率都是常数, 因此, R_{i1}^p 和 R_{i2}^p 都为常数. 我们发现, $\phi_i \gamma_i R_{i1}^p$ 和 $\phi_i (1-\gamma_i) R_{i2}^p$ 分别为关于 γ_i 的增函数和减函数, 因此, 当 $\ln(\min\{\phi_i \gamma_i R_{i1}^p, \phi_i (1-\gamma_i) R_{i2}^p\})$ 取最大值时, 当且仅当 $\phi_i \gamma_i R_{i1}^p = \phi_i (1-\gamma_i) R_{i2}^p$ 成立, 即 $\gamma = \gamma_i^* = R_{i2}^p / (R_{i1}^p + R_{i2}^p)$. 任取

$ST_j \in PCS_i$, 为方便讨论, 记 $\Gamma_j = \frac{\omega_j}{\sum_{l=1}^N a_{il} \omega_l} \log_2 \left(1 + \frac{P_j^s g_{ij}^k}{N_0} \right)$ 为常数, 则有 $f_i = \ln(\gamma_i^* \phi_i R_{i1}^p) + \sum_{ST_j \in PCS_i} \ln((1-\phi_i) \Gamma_j)$. 注意

到, 这是一个关于 $\phi_i \in (0, 1)$ 的凹函数, 故当且仅当 $\phi_i = \phi_i^* = \left(\sum_{l=1}^N a_{il} \right) / \left(\sum_{l=1}^N a_{il} + 1 \right)$, f_i 达到最大值:

$$\ln(\gamma_i^* \phi_i^* R_{i1}^p) + \sum_{ST_j \in PCS_i} \ln((1-\phi_i^*) \Gamma_j).$$

2) 若 $RCS_i = \emptyset$, PT_i 不与任何次级节点合作, $\gamma_i = 0$. 由公式(4)和公式(6)可得, 若记 $\Gamma_i = \log_2 \left(1 + \frac{P_i^p g_{ii}^k}{N_0} \right)$ 为常数,

则有 $f_i = \ln(\phi_i \Gamma_i) + \sum_{ST_j \in PCS_i} \ln((1-\phi_i) \Gamma_j)$.

显然, 这是一个关于 $\phi_i \in (0, 1)$ 的凹函数, 当且仅当 $\phi_i = \phi_i^* = \left(\sum_{l=1}^N a_{il} \right) / \left(\sum_{l=1}^N a_{il} + 1 \right)$ 时, f_i 达到最大值:

$$\ln(\Gamma_i \phi_i^*) + \sum_{ST_j \in PCS_i} \ln((1-\phi_i^*) \Gamma_j).$$

因此, 由步骤 1) 和步骤 2) 可得:

$$\begin{cases} \gamma_i = \gamma_i^* = \frac{R_{i2}^p}{R_{i1}^p + R_{i2}^p}, \quad \phi_i = \phi_i^* = \frac{\sum_{l=1}^N a_{il}}{\sum_{l=1}^N a_{il} + 1}, & \text{if } RCS_i \neq \emptyset \\ \gamma_i = 0, \quad \phi_i = \phi_i^* = \frac{\sum_{l=1}^N a_{il}}{\sum_{l=1}^N a_{il} + 1}, & \text{if } RCS_i = \emptyset \end{cases}.$$

注意: 两种情况下, $\phi_i = \phi_i^*$ 的值相同.

(2) 接下来, 给出算法用于确定选择哪个 RCS_i 才能使 f_i 的最大化. 首先, 若 $RCS_i = \emptyset$, PT_i 可获得的最大平均速率为 $R_i^p(RCS_i)_{RCS_i=\emptyset} = \phi_i^* \Gamma_i$. 算法的思想是: 试图找到一个 $RCS_i \neq \emptyset$, 使得 $\max R_i^p(RCS_i)_{RCS_i \neq \emptyset} > \phi_i^* \Gamma_i$; 否则, 若找不到, 则只能令 $RCS_i = \emptyset$.

我们首先将在 PR_j 传输半径之外的节点排除掉, 获得一个集合 $PCS'_i = \{ST_j \mid ST_j \in PCS_i, d(ST_j, PR_i) \leq D_j^p\}$. 然后, 我们发现任取 $ST_j \in PCS'_i$, 若它先加入 RCS_i , 则 R_{i1} 便首先确定. 对于其他节点 $ST_l \in PCS'_i$, 当 $g_{il}^k > g_{ij}^k$ 时, 若 ST_l 加入 RCS_i , 则 R_{i1} 将保持不变. 我们便可将所有符合 $g_{il}^k > g_{ij}^k$ 条件的节点加入 RCS_i , 使 R_{i2} 最大化.

这样, 由 $R_i^p(RCS_i)_{RCS_i \neq \emptyset} = \phi_i^* R_{i1}^p R_{i2}^p / (R_{i1}^p + R_{i2}^p) = \phi_i^* / (1/R_{i1} + 1/R_{i2})$, 就可获得当 ST_j 首先加入 RCS_i 时的最大值. 通过以上方法, 就可以依次固定第 1 个加入 RCS_i 的节点, 寻找到能使 $R_i^p(RCS_i(g_{ij}^k))_{RCS_i \neq \emptyset}$ 最大的节点组合. 显然, $|RCS_i| \leq n = |PCS_i|$, 算法的时间复杂度为 $O(n^2)$, $n = |PCS_i|$ 为 PCS_i 中节点的数目.

综上,由步骤(1)和步骤(2)可得定理 1 成立. $\square$

定理 2. 给定潜在协作矩阵 $\mathbf{A}$ 和频谱分配矩阵 $\mathbf{O}$, 存在 $\bar{\mathbf{A}}, \Phi, \mathbf{H}$, 使得 JSA-CSP 的优化目标函数 $\sum_{i=1}^M (\ln(R_i^p) + \sum_{j=1}^N \ln(R_j^s)) = \sum_{i=1}^M f_i(C_i, PT_i, PCS_i)$ 最大化.

证明: 由于通过 $\mathbf{A}$ 可确定任意 PT_i 的 PCS_i , $\mathbf{O}$ 可确定 PT_i 使用的频谱 C_i , 而由定理 1, 存在 $RCS_i \subseteq PCS_i$ 及 ϕ_i, γ_i , 使得函数 $f_i(C_i, PT_i, PCS_i)$ 最大化, 则 $\bar{\mathbf{A}}, \Phi, \mathbf{H}$ 的对应元素便可确定. 因此, 定理 2 显然成立. $\square$

定理 3. JSA-CSP 是 NP-hard 的.

证明: 首先, 定义 JSA-CSP 优化问题所对应的判定问题; 然后, 通过限制法将判定问题归结为三维匹配 (3-dimensional matching, 简称 3DM) 问题. 由于后者为 NP-Complete 的^[13], 故 JSA-CSP 的判定问题为 NP-Complete 的; 最后, 由于若一个判定问题为 NP-Complete 的, 则对应的优化问题为 NP-hard 的^[13], 故 JSA-CSP 为 NP-hard 的.

(1) JSA-CSP 的判定问题: 设 C, PT, ST 分别是可用频谱、PU 发送节点、SU 发送节点集合, 其中,

$$M = |PT| = |C| \leq |ST| = N.$$

将 ST 划分为 M 个非空子集合, 记为 $ST' = \{ST(1), ST(2), \dots, ST(M)\}$, 定义三元组集合 $A \subseteq C \times PT \times ST'$, 其中,

$$A_i = (C_i, PT_i, ST(j)) \in A.$$

令 $ST(j) = PCS_j$ 为 PT_i 的潜在协作集. 由定理 1 可知, 三元组 A_i 的函数值为 $f'(A_i) = \max_{f_i(C_i, PT_i, PCS_i = ST(j))}$, 则 JSA-CSP 的判定问题为: A 中是否存在一个匹配 $A' \subseteq A$, 使得 $|A'| = |M|$, $\sum_{i=1}^M f'(A'_i) \geq \lambda$, 且 A' 中任何两三元组都没有相同分量. 其中, λ 为给定实数.

(2) 3DM 判定问题: 设 C, PT, ST 分别是 3 个集合, 且 $|C| = |PT| = |ST| = M$, 三元组集合定义为 $A \subseteq C \times PT \times ST$. 3DM 判定问题是: A 中是否存在一个匹配, 即是否存在 $A' \subseteq A$, 满足 $|A'| = |M|$, 且 A' 的任何两个三元组中都没有相同分量.

(3) 对 JSA-CSP 判定问题进行限制: ① 令 $|C| = |PT| = |ST| = M$; ② 由限制①, $ST' = ST$, 三元组中第 3 个元素只可取一个次级用户作为潜在协作节点; ③ 再令 λ 为负无穷, 则 JSA-CSP 判定问题转换为 3DM 判定问题, 即 3DM 判定问题为 JSA-CSP 判定问题的特例.

因此, JSA-CSP 的判定问题是 NP-Complete 的, JSA-CSP 是 NP-hard 的. $\square$

4 集中式遗传算法

由于 JSA-CSP 是 NP-hard 的, 我们提出采用遗传算法来解决这个 0-1 非线性规划问题. 经典的遗传算法通过随机方式产生若干个待求解问题的数字编码, 即染色体 (又称个体), 形成初始种群. 算法对所产生的每个染色体基于适应值函数算出的适应值进行评价, 淘汰低适应值的染色体, 选择高适应值的染色体参加遗传 (交叉、变异) 操作, 经遗传操作后的染色体集合形成下一代种群, 再对这个新种群进行下一轮进化.

根据 JSA-CSP 的特点, 本节我们首先对遗传算法的基本要素和流程进行设计, 提出一种集中式的遗传算法 CGA, 然后对 CGA 的全局收敛性进行分析.

4.1 算法描述

定义 6 (联合染色体, 种群). 定义联合染色体 (joint chromosome) 为 $JC = [\mathbf{A}, \mathbf{B}]_{M \times (N+1)}$. 如图 2 所示, 它包括两个子染色体, 其中: $\mathbf{A}_{M \times N}$ 为潜在协作矩阵; 子染色体 $\mathbf{B} = (\beta_i)_{M \times 1}$ 为频谱分配向量, 它是频谱分配矩阵 $\mathbf{O}_{M \times M}$ 对应的向量表示, β_i 为 $\mathbf{O}$ 第 i 行二进制串的十进制表示. 采用这样的编码, 目的是节省算法用于存储染色体的空间, 降低由于对矩阵操作带来的时间开销. 联合染色体的集合定义为种群 (colony), 用 CN 表示.

定义 7 (染色体屏蔽矩阵). 由于算法在运行过程中会对染色体的合法性进行检测, 即检查生成的染色体是否符合拓扑结构的约束, 故定义染色体屏蔽矩阵 (chromosome mask) 为 $\mathbf{CM} = [(m_{ij}^\alpha)_{M \times N}]$, 其中, $m_{ij}^\alpha = 1$ 当且仅当 PT_i 与 ST_j 之间的欧氏距离 $d(PT_i, ST_j) \leq D_i^p$, 且 ST_j 与 PR_i 之间的欧氏距离 $d(ST_j, PR_i) \leq D_j^s$ 时, D_i^p, D_j^s 分别为 PT_i 和 ST_j 的最大传输半径. $m_{ij}^\alpha = 1$ 表明 ST_j 可以成为 PT_i 的潜在协作节点, 否则 $m_{ij}^\alpha = 0$. 我们假设初始拓扑满足每个 PU 至少有一个不同的 SU 用户可以成为其潜在协作者.

定义 8 (适应值函数). 定义 JC 的适应值函数为

$$F(JC) = F([A, B]) = \sum_{i=1}^M \max f_i(C_i, PT_i, PCS_i) = \sum_{i=1}^M (\max(\ln(R_i^p) + \sum_{ST_j \in PCS_i} \ln(R_j^s))).$$

根据定理 1、定理 2, 可以计算出 JC 的最大适应值。

α_{11}	α_{12}	$\dots$	α_{1N}	β_1	Spectrum allocation vector
α_{21}	α_{22}	$\dots$	α_{2N}	β_2	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
α_{M1}	α_{M2}	$\dots$	α_{MN}	β_M	
Potential cooperation matrix					

Fig.2 Definition of a joint chromosome

图 2 联合染色体定义

在定义了 CGA 的基本要素后, 我们对其各步骤进行描述, 共包括 4 个步骤:

Step 1. 初始化种群.

算法首先产生一个包含 Q 个染色体的初始化种群 CN , 任取 $JC=[A, B]_{M \times (N+1)} \in CN$, 需满足:

- 1) $\text{BITAND}(CM, A) = A$
- 2) $\text{SUM}(A(i,:)) \geq 1, \forall i \in \{1, \dots, M\}$
- 3) $\text{SUM}(A(:, j)) = 1, \forall j \in \{1, \dots, N\}$
- 4) $B(i) \neq B(k), \forall i, k \in \{1, \dots, M\}, i \neq k$

(15)

这里, 我们借用 MATLAB 的数学表示, BITAND 为矩阵按位与操作, 该操作判断 A 是否满足拓扑结构约束. $\text{SUM}(A(i,:))$ 表示矩阵 A 的第 i 行元素之和, $\text{SUM}(A(:, j))$ 表示矩阵 A 的第 j 列元素之和. 这些约束分别根据约束(8)~约束(11)得到. 其中, 初始向量 B 的每一个元素可采用随机分配的方式完成, 初始矩阵 A 可采用如下方法由 CM 出发快速产生: (1) 初始的 A 是一个零矩阵. 将 CM 中每行按所拥有的 1 的个数来排序, 从 1 数量最少的行开始分配, 随机寻找该行为 1 的位置, 设找到一个为 1 的位置. 对应地, 若 A 中该位置所在列上没有其他 1, 则将 A 中该位置为 1, 这保证了约束(15)中的子约束 2) 和子约束 3); (2) 所有行都执行步骤(1)后, 注意到 A 中还有 $N-M$ 列未分配 1, 则从 CM 对应的这些列中随机选择一个 1 位置, 置 A 中对应位为 1, 这保证了约束(15)中的子约束 3).

Step 2. 混合交叉.

随机选中 CN 中两个联合染色体进行交叉, 交叉时随机地选择两个染色体的相同位置实行交换. 这个过程反映了随机信息交换, 目的在于产生新的基因组合, 也即产生新染色体. 我们采用的则是一种混合策略, 即在 A 中采用按列多点交叉, 在 B 中采用按行多点交叉后自交叉策略. 具体地说, 如图 3 所示, 若联合染色体 JC 与 JC' 被选中进行混合交叉操作:

(1) 设子染色体 A, A' 的随机交叉点为 L_1 . 采用 MATLAB 表示为

$$A=[A(:, 1:L_1-1), A(:, L_1), A(:, L_1+1:N)], A'=[A'(:, 1:L_1-1), A'(:, L_1), A'(:, L_1+1:N)].$$

交叉后得到: $\tilde{A}=[A(:, 1:L_1-1), A'(:, L_1), A(:, L_1+1:N)]$ 与 $\tilde{A}'=[A'(:, 1:L_1-1), A(:, L_1), A'(:, L_1+1:N)]$. 我们规定交叉后 $\tilde{A}$ 与 $\tilde{A}'$ 需要满足约束(15), 否则不进行交叉操作.

(2) 设子染色体 B, B' 的随机交叉点为 L_2 . 将它们分别表示为

$$B=[B(1:L_2-1), B(L_2), B(1:L_2-1)], B'=[B'(1:L_2-1), B'(L_2), B'(1:L_2-1)].$$

对应行的元素首先经过交叉后得到:

$$\tilde{B}=[B(1:L_2-1); B'(L_2); B(1:L_2-1)], \tilde{B}'=[B'(1:L_2-1); B(L_2); B(1:L_2-1)].$$

又注意到交叉后子染色体不满足约束条件(15)中的子约束 4), 因, 规定交叉后 $\tilde{B}$ 与 $\tilde{B}'$ 执行自交叉操作. 即,

若 $\tilde{\mathbf{B}}$ 有某元素 $\tilde{\mathbf{B}}(x) = \mathbf{B}'(L_2)$, 则 $\tilde{\mathbf{B}}(x) = \mathbf{B}(L_2)$. 对 $\tilde{\mathbf{B}}'$ 的操作同理.

CGA 将循环运行,直到得到一个包含 $2Q$ 个新联合染色体的新种群 CN' 为止.

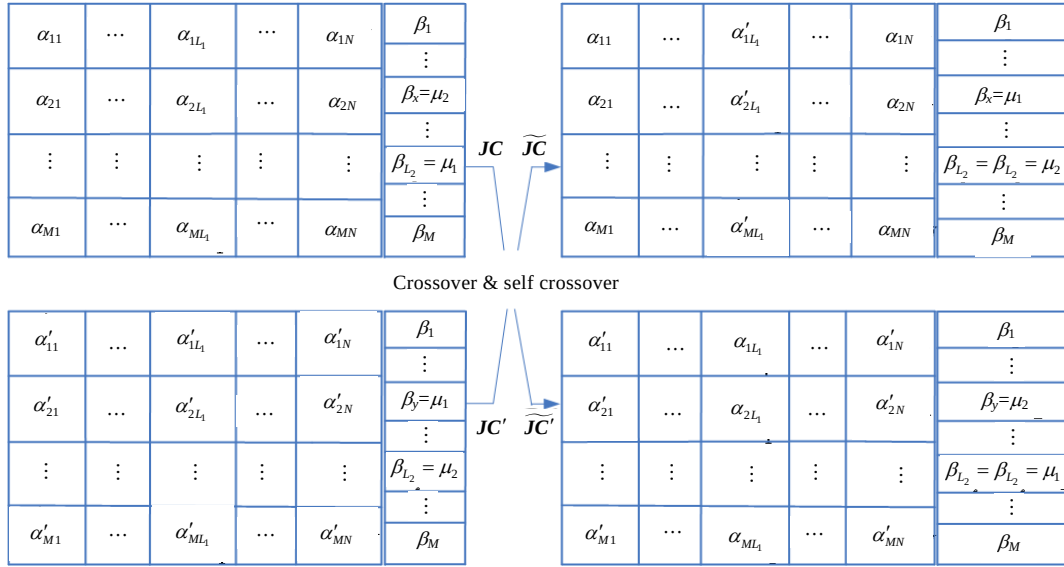


Fig.3 Mixed crossover process

图3 混合交叉过程

Step 3. 保护变异.

对 CN' 中的每个联合染色体 $JC=[\mathbf{A}, \mathbf{B}]$ 执行随机变异操作,根据生物遗传中基因变异原理,以变异概率 P_m 对子染色体 $\mathbf{A}$ 的每一列执行变异.将某些位由 0 变为 1,将某些位由 1 变为 0.若 $\mathbf{A}$ 所有列变异后不符合约束(15),则 $\mathbf{A}$ 保持不变;否则,对 $\mathbf{A}$ 进行变异.再以变异概率 P_m 对子染色体 $\mathbf{B}$ 的每行进行保护性变异操作,例如:第 y 行需执行变异操作,变异后得到的频谱为 $\tilde{\mathbf{B}}(y)$;若存在 $x \neq y$ 且 $\tilde{\mathbf{B}}(x) = \tilde{\mathbf{B}}(y)$, 则令 $\tilde{\mathbf{B}}(x) = \mathbf{B}(y)$. 这保证了约束(15)中子约束 4)的要求.

Step 4. 选择.

至此,算法得到经变异后的种群 $CN'(2Q$ 个联合染色体)和原始种群 $CN(Q$ 个联合染色体).根据适者生存原则,在这 $3Q$ 个联合染色体中,根据其适应值大小,选择出 Q 个适应值较高的形成新一轮循环的原始种群.当原始种群联合染色体的平均适应值变化率为 0 时算法结束,否则,转 Step 2.

4.2 CGA全局收敛性分析

定义 9(种群集合与种群适应值). 根据定理 3,优化问题的解空间共有 Ω 个,显然,可行解 $\Omega' < \Omega$. 这 Ω' 个可行解可组成 $C_{\Omega'}^Q$ 个不同的种群,任取种群 $CN_i, 1 \leq i \leq C_{\Omega'}^Q$, 其种群适应值定义为 $F'(CN_i) = \max\{F(JC) | JC \in CN_i\}$. 假设经计算得到的非重复种群适应值为 n 个,适应值相同的种群被划分为同一个种群集合,可被划分为 $S_1, S_2, \dots, S_n$ 个种群集合,种群集合的适应值定义为 $F''(S_i) = F'(CN_j), \forall CN_j \in S_i$. 显然, $\forall i \neq j, S_i \cap S_j = \emptyset$. 不妨设种群集合的适应值满足排序 $F''(S_1) > F''(S_2) > \dots > F''(S_n)$, 即最优解包含在种群集合 S_1 中.

定义 10(种群状态和状态转移矩阵). 令 $CN(k)$ 为 CGA 在 k 轮得到的种群,我们用一个长度为 n 的 0-1 向量 $\mathbf{p}^k$ 表示第 k 轮 $CN(k)$ 的状态, $\mathbf{p}^k(j) = 1$ 当且仅当 $CN(k) \in S_j, 1 \leq j \leq n$. 定义状态转移矩阵 $\mathbf{P} = (p_{ij})_{n \times n}, p_{ij}$ 为种群从属于 S_i 转移到属于 S_j 的概率,则种群在 CGA 的第 $k+1$ 轮的状态 $\mathbf{p}^{k+1} = \mathbf{p}^k \mathbf{P}$. 易知,任取 $i, \sum_{j=1}^n p_{ij} = 1$.

定义 11. 任取联合染色体 JC, JC' , 若 JC' 是由 JC 通过交叉和变异产生,则称 JC' 是 JC 通过杂交和变异可达的. 即 $\text{Prob}\{MC(JC) = JC'\} > 0, MC(JC)$ 表示由 JC_i 通过交叉(crossover)和变异(mutation)产生的染色体. $\text{Prob}\{\cdot\}$ 表示

事件 $\{\cdot\}$ 的概率.

引理 1. 对可行解中任意两个染色体 $J\mathbf{C}$ 和 $J\mathbf{C}'$ 是从 $J\mathbf{C}$ 通过交叉和变异可达的.

证明:事实上,设 $\widetilde{J\mathbf{C}}$ 是 $J\mathbf{C}$ 通过与其他染色体交叉而得到的,即 $C(J\mathbf{C}) = \widetilde{J\mathbf{C}}$, 则只要证明 $J\mathbf{C}'$ 是 $\widetilde{J\mathbf{C}}$ 通过变异可达的即可.若将 $\mathbf{B}$ 用 $\boldsymbol{\Theta}$ 表示,即 $\widetilde{J\mathbf{C}} = [\widetilde{\mathbf{A}}, \boldsymbol{\Theta}]$, $J\mathbf{C}' = [\mathbf{A}', \boldsymbol{\Theta}]$, 则可定义 $\widetilde{J\mathbf{C}}$ 与 $J\mathbf{C}'$ 的汉明距离 h 为它们不一致的列向量个数.不失一般性,设 $\widetilde{J\mathbf{C}}$ 与 $J\mathbf{C}'$ 的后 $M+N-h$ 列相同,则由变异操作的定义可知,由于 $P_m > 0$, $\text{Prob}\{M(\widetilde{J\mathbf{C}}) = J\mathbf{C}'\} = (P_m)^h (1/(N-1))^h (1-P_m)^{M+N-h} > 0$, 即 $J\mathbf{C}'$ 是从 $\widetilde{J\mathbf{C}}$ 通过变异可达的.因此, $J\mathbf{C}'$ 是从 $J\mathbf{C}$ 通过交叉和变异可达的. $\square$

引理 2. 在状态转移矩阵 $\mathbf{P} = (p_{ij})_{n \times n}$ 中,有 $p_{ij} > 0$ 当且仅当 $i \geq j$, $p_{ij} = 0$ 当且仅当 $i < j$.

证明:引理 2 说明,在 CGA 中,种群只能以大于 0 的概率从适应值较低的种群状态转移到适应值相同或适应值更高的种群状态.首先,由引理 1,可行解中任意两个联合染色体都是可以通过交叉和变异互相可达的,则任取种群状态 S_i 和 S_j , $i \neq j$, S_i 状态对应的种群是可以通过 CGA 对该种群中的联合染色体进行交叉变异,以概率 $p'_{ij} > 0$ 可达 S_j 状态对应的种群的.又由 CGA 的 Step 4 的选择过程可知,总是保存适应值最好的 Q 个染色体作为每一轮迭代的种群,并且上一轮的联合染色体不会参加下一轮的交叉和变异,种群只能从适应值较低的种群状态转移到适应值相同或适应值更高的种群状态.因此,存在种群状态之间转换是否可行的布尔矩阵 $\boldsymbol{\Delta} = (\delta_{ij})_{n \times n}$, $\delta_{ij} = 1$ 当且仅当 $i \geq j$, $\delta_{ij} = 0$ 当且仅当 $i < j$.而 $p_{ij} = p'_{ij} \times \delta_{ij}$, 因此有 $p_{ij} > 0$ 当且仅当 $i \geq j$, $p_{ij} = 0$ 当且仅当 $i < j$ 成立. $\square$

引理 3^[14]. 设 $\mathbf{P}$ 是一个 n 阶可约的随机矩阵,且是一个有限齐次马尔可夫链的状态转移矩阵,即

$$\mathbf{P} = \begin{bmatrix} \mathbf{F} & \mathbf{0} \\ \mathbf{R} & \mathbf{T} \end{bmatrix},$$

其中, $\mathbf{F}, \mathbf{T}$ 为方阵, $\mathbf{F}$ 为 $m(m \leq n)$ 阶本源矩阵; $\mathbf{R}, \mathbf{T}$ 为非 0 矩阵,则

$$\mathbf{P}^\infty = \lim_{k \rightarrow \infty} \mathbf{P}^k = \lim_{k \rightarrow \infty} \begin{pmatrix} \mathbf{F}^k & \mathbf{0} \\ \sum_{i=0}^{k-1} \mathbf{T}^i \mathbf{R} \mathbf{F}^{k-i} & \mathbf{T}^k \end{pmatrix} = \begin{pmatrix} \mathbf{F}^\infty & \mathbf{0} \\ \lim_{k \rightarrow \infty} \sum_{i=0}^{k-1} \mathbf{T}^i \mathbf{R} \mathbf{F}^{k-i} & \mathbf{0} \end{pmatrix}$$

是一个稳定随机矩阵,且 $\mathbf{P}^\infty = \mathbf{1}' \mathbf{p}^\infty$. 其中,无论初始分布 $\mathbf{p}^0$ 如何取值,总有 $\mathbf{p}^\infty = \mathbf{p}^0 \mathbf{P}^\infty$. $\mathbf{p}^\infty$ 满足:任取 $1 \leq i \leq m$, $\mathbf{p}^\infty(i) > 0$; 任取 $m \leq i \leq n$, $\mathbf{p}^\infty(i) = 0$. $\square$

定理 4. 算法 CGA 具有全局收敛性.

证明:设算法第 k 轮种群状态为 $\mathbf{p}^k$, 可以认为 $\mathbf{p}^k$ 是时齐的有限马尔可夫链上的第 k 个状态,由引理 2 可知,其马尔可夫状态转移矩阵可写为

$$\mathbf{P} = \begin{pmatrix} p_{11} & 0 & 0 & \cdots & 0 \\ p_{21} & p_{22} & 0 & \cdots & 0 \\ p_{31} & p_{32} & p_{33} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & p_{n3} & \cdots & p_{nn} \end{pmatrix} = \begin{pmatrix} \mathbf{F} & \mathbf{0} \\ \mathbf{R} & \mathbf{T} \end{pmatrix}.$$

由引理 3 可知, $\mathbf{F} = p_{11} = 1$, $\mathbf{R} = (p_{21}, p_{31}, \dots, p_{n1})^T$ 与 $\mathbf{T}$ 为非 0 矩阵.通过计算可得 $\mathbf{F}^\infty = 1$, $\lim_{k \rightarrow \infty} \sum_{i=0}^{k-1} \mathbf{T}^i \mathbf{R} \mathbf{F}^{k-i} = \overbrace{(1, 1, \dots, 1)}^{n-1}$, 则有,

$$\mathbf{P}^\infty = \lim_{k \rightarrow \infty} \mathbf{P}^k = \begin{pmatrix} \mathbf{F}^\infty & \mathbf{0} \\ \lim_{k \rightarrow \infty} \sum_{i=0}^{k-1} \mathbf{T}^i \mathbf{R} \mathbf{F}^{k-i} & \mathbf{0} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \cdots & 0 \end{pmatrix},$$

$$\mathbf{p}^\infty = \mathbf{p}^0 \mathbf{P}^\infty = (1, 0, 0, \dots, 0).$$

无论初始分布 p^0 如何取值, p^∞ 最终收敛到 $(1, 0, 0, \dots, 0)$, 即种群集合 S_1 对应的状态, 即算法 CGA 最终收敛到全局最优的种群状态. $\square$

5 分布式遗传算法

首先, 一方面随着网络规模的增大, CGA 的搜索空间呈指数级增长, 全局收敛时间变长, CGA 很难在规定的系统初始配置时间内得到较优化的解; 另一方面, CGA 需要 CCRN 中有一个中控节点来进行集中计算, 在诸如认知无线电 Ad Hoc 网络中^[15], 没有这样一个中控节点, 每个节点仅可获知有限跳内的邻居节点的信息. 因此, 需要设计分布式算法对 JSA-CSP 求解.

本文提出的分布式算法的思想是: 首先, 将 PT 集合划分成簇; 然后, 确定次级用户节点分别归属于哪个簇; 最后, 当次级节点都加入所确定的簇后, 簇内采用 CGA 算法求解. 由此衍生出需要解决的问题包括:

- (1) PT 集合如何根据网络拓扑划分成簇;
- (2) 频谱如何分配;
- (3) 如何确定次级用户节点分属于哪个簇.

为解决上述问题, 我们提出一种两阶段分布式遗传算法 DGA, 包括: 基于最小支配集的 PT 分簇与频谱预分配阶段和簇间潜在协作集协商与簇内适应值精化阶段.

5.1 阶段1(DGA-1): 基于最小支配集的 PT 分簇与频谱预分配阶段

定义 12(最小支配集). 用 $G=(PT, E)$ 表示集合 PT 中主用户构成的连通网络, $|PT|=M$, E 为边的集合. 网络的支配集 $DS \subset PT$ 定义为一组节点的集合, 任取 $PT_i \in PT, 1 \leq i \leq M$, 要么 $PT_i \in DS$, 要么 PT_i 的邻居节点 $N(P T_i) \in DS$. 最小支配集 DS^* 为节点数量最小的支配集^[16].

阶段 1 的思想是: 通过分布式最小支配集算法获得 PT 的最小支配集, 以最小支配集内的支配节点作为簇头. 每个初始簇由簇头、与簇头 1 跳相连的邻居主节点、簇内所有主节点传输半径内的次级节点组成. 之所以以支配集的方式分簇, 是因为簇内主节点都能一跳到达簇头, 不仅可以降低通信开销, 而且还有利于分布式实现. 更重要的是, 以最小的支配集的方式分簇, 簇的数量最小, 可降低簇间后续协调产生的通信开销, 也有利于簇内采用遗传算法进行簇内局部优化.

分布式最小支配集算法如下: 令 $|N^P(P T_i)|$ 表示 PT_i 的邻居主节点的数目. 如果 PT_i 的 $|N^P(P T_i)|$ 比 PT_i 两跳之内所有主节点的邻居数目都大, 则选择 PT_i 作为支配节点. 因此, 我们得到一种分布式最小支配集算法, 见表 1.

Table 1 Pseudocode description of DGA-1 (phase 1 of DGA)

表 1 DGA-1(DGA 算法第 1 阶段)的伪码描述

DGA-1
While PT_i has white neighbors Broadcast $ N^P(P T_i) $ to PT_i 's neighbors within 2 hops If $ N^P(P T_i) $ is the largest within 2 hops Set PT_i as black, set its 1 hop neighbors as grey end if end while for all $PT'_i \in DS^*$ Form cluster $CT_i = PT'_i \cup N^P(P T'_i) \cup N^S(N^P(P T'_i) \cup PT'_i)$ Broadcast $ CT_i $, calculate its operation rank and available spectrum Call CGA(CT_i) end for

初始情况下, 所有主节点被标记为白色. 若被选中为支配节点, 则被标记为黑色, 与支配节点连接的主节点标记为灰色. 最终, 所有被标记为黑色的节点形成最小支配集 DS^* . 令 $DS^* = \{PT'_i | 1 \leq i \leq l, l = |DS^*|\}$, 则最终形成 $CT_1, CT_2, \dots, CT_l$ 个簇, 其中, CT_i 包括簇头 $PT'_i, N^P(P T'_i)$ 及它们的邻居次级节点 $N^S(N^P(P T'_i) \cup PT'_i)$.

形成簇后, 每个簇头首先广播其所有节点数目, 数目多的簇首先通过 CGA 计算该簇占用的频谱, 并宣布独

占这些频谱;后运行的簇将不使用这些频谱.我们认为,簇内节点多的簇拥有优先使用频谱的权利,由于节点较多,可优化的空间较大,对于获取全局优化解的贡献较大,因此允许其优先分配频谱.

5.2 阶段2(DGA-2):簇间潜在协作集协商,簇内适应值精化阶段

经 DGA-1 后,簇与簇之间包含不同的主节点,但有可能包含相同的次级节点.称这些被包含在多个簇的次级节点为争议节点.因此,需要决定这些争议节点的归属.我们采用簇间潜在协作集协商算法来决定争议节点的归属.具体地说,在 DGA-2 中,每个簇每一轮驱除一个能使自身适应值增幅值最大的争议节点,最终使每个争议节点被唯一分配给一个簇.交互过程如下:DGA-1 结束后,每个簇头进行广播,所有次级节点获得其所属簇的消息.若次级节点被独占,则返回给簇头标识它被该簇独占;若为一个争议节点,则反馈给簇头,声明它是争议节点.每个簇头在得知该簇有争议节点后,开始计算驱除哪个争议节点能使这个簇的适应值增幅值最大,然后将驱除请求和计算出的适应值增幅值发送给该争议节点.这时存在 4 种情况,如图 4 所示:

- (1) 如图 4(a)所示.争议节点 ST_1 若属于两个簇(如簇 1 和簇 2),且只收到簇 1 的驱除请求,则其反馈给簇 1,表明同意并退出;同时反馈给簇 2,表示该节点从此属于簇 2;
- (2) 如图 4(b)所示.争议节点 ST_1 属于簇 1 和簇 2,若 ST_1 同时收到两簇的驱除请求,则 ST_1 独自计算退出哪个簇才能对整体的适应值增幅最大.如发现退出簇 1 最优,则反馈给簇 1,表明同意并退出;同时反馈给簇 2,表示该节点从此属于簇 2;
- (3) 如图 4(c)所示. ST_1 属于多于两个簇,例如属于簇 1、簇 2 和簇 3,若只收到簇 1 的驱除请求,则其反馈给簇 1,操作如情况(1);
- (4) 如图 4(d)所示. ST_1 属于多于两个簇,若只收到簇 1 和簇 2 的请求,则 ST_1 独自计算退出哪个簇才能对整体的适应值增幅最大.例如,退出簇 1 最优,则其反馈给簇 1,表明同意并退出;同时反馈给其他簇,表示这个节点仍然是争议节点.每个簇继续此过程,直到其簇内所有次级节点都成为非争议节点.

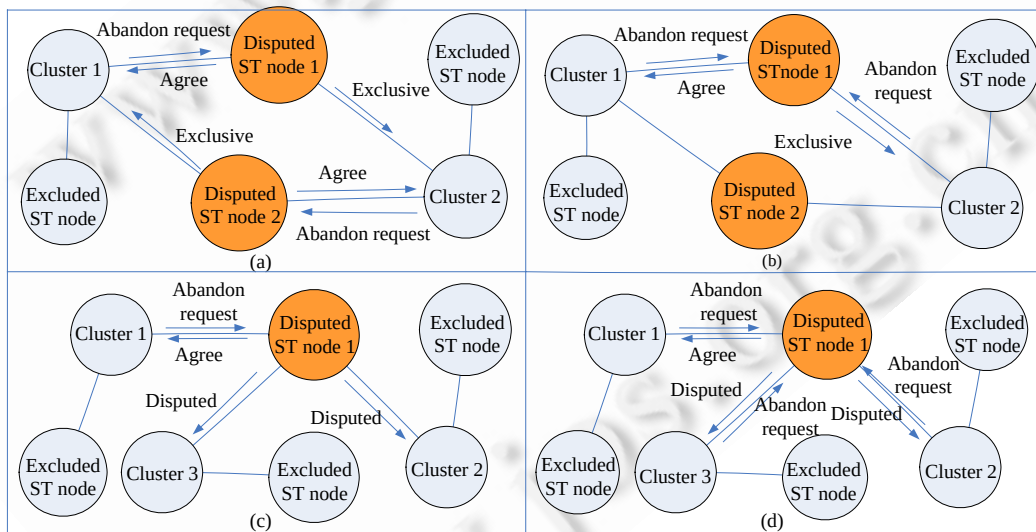


Fig.4 Negotiation process of secondary disputed nodes

图 4 次级争议节点的协商过程

每个簇在每一轮都对争议节点状态进行更新,确定争议节点的归属,使得网络中最终没有争议节点.至此,各簇彼此独立,在簇内采用 CGA 进一步优化协作集划分.我们称这一过程为簇内适应值精化过程.

DGA-2(DGA 算法第 2 阶段)的伪码描述见表 2.

Table 2 Pseudocode description of DGA-2 (phase 2 of DGA)**表 2** DGA-2(DGA 算法第 2 阶段)的伪码描述

DGA-2
Procedure of PT_i' (head of cluster CT_i) Broadcast Adhering requests to $ST_j \in N^S(N^P(PT_i') \cup PT_i')$ Receive message from If ST_j 's message is Disputed (indicate it is a disputed one) Mark it as disputed else if ST_j 's message is Exclusive (indicate it is a exclusive one) Mark it as exclusive else if ST_j 's message is Disagree Delete ST_j from the CT_i end While CT_i has disputed secondary users Select one (e.g. ST_k) eliminate which maximizes the increase of the CT_i 's fitness and send Abandon request to ST_k Receive message from ST_k if ST_k Agree Delete ST_k from the CT_i , $N^S(N^P(PT_i') \cup PT_i') = N^S(N^P(PT_i') \cup PT_i') - ST_k$ and update CT_i 's fitness else if ST_k Disputed (indicate it is still disputed by two or more clusters) Mark it as disputed else if ST_k Exclusive (indicate it is no longer a disputed one) Mark it as exclusive end if Receive message from other disputed secondary users (i.e. $ST_m \neq ST_k$) if ST_m Exclusive Mark it as exclusive else if ST_m Disputed Mark it as disputed end if end while Procedure of secondary user ST_k Receive Adhering requests from all the clusters If ST_k is the only neighbor of a primary user node of a cluster (i.e. CT_i) Send Exclusive to CT_i and broadcast Disagree to the other clusters else if ST_k is owned by only one cluster Send back Exclusive (indicate it is an exclusive one) else if ST_k is disputed by 2 or more clusters Broadcast Disputed (indicate that it is a disputed one) end if While ST_k is disputed If ST_k is disputed by two clusters if only one Abandon request received Send back Agree and send Exclusive to another cluster else if two Abandon requests received Select one cluster withdraw which can maximizes the increase of the fitness sum of the two clusters Send back Agree to this cluster and send Exclusive to another one end if else if ST_k is disputed by three or more clusters if only one Abandon request received Send back Agree and send Disputed to the others else if two or more Abandon requests received Select one cluster withdraw which can maximizes the increase of the fitness sum of all the clusters Send back Agree to this cluster and send Disputed to the others end if end if end while Fitness refinement procedure of CT_i Call [new fit, new cooperation relationship]= $CGA(CT_i)$ if new fit < current Fitness return [current fit, current cooperation relationship] else return [new fit, new cooperation relationship] end if

5.3 一种快速收敛的算法FDGA

由于在 DGA 中,所有簇串行地运行 CGA 以确定每个簇采用的频谱,这无疑增加了算法运行时间.因此,我们又提出一种快速收敛的 DGA 算法 FDGA.它与 DGA 的唯一区别在于,初始分簇形成之后,每个簇顺序地随机选取一个频谱并马上进入阶段 2,这样就以牺牲对频谱分配优化的方式为代价以节省频谱预分配的时间.这样,所有簇就有可能并行执行 CGA 算法,大大加快了算法收敛速度.

5.4 支持分布式算法的MAC协议及通信开销

为了支持分布式算法的实现,MAC 协议必须有效支持网络发现、时间同步,且需有效避免在算法进行分布式最小支配集计算与簇间潜在协作集协商过程中,由于节点之间通信在物理层产生的冲突.实际上,网络中往往仅存在一个通用控制信道.由于具体的 MAC 协议不是本文主要讨论的范畴,我们假设网络中节点时间精确同步,采用简单的固定时隙 TDMA 控制信道接入方式对通信开销进行估算.设所有 Beacon 消息长度都为 L ,都占用一个时隙.主节点个数为 M ,次级节点个数为 N 个.额外的通信开销主要包括 3 个部分:(1) 在分布式最小支配集算法中,每一轮,每个簇头利用两个时隙,即将自己的邻居节点数广播给两跳内的主节点,每个主节点因此得知其是否为簇头.对于一个连通拓扑,每轮至少确定 1 个簇头,因此至少确定 1 个簇头所支配的节点,故 DGA-1 最多进行 $M/2$ 轮.需要预留 $2M \times M/2 = M^2$ 个时隙;(2) 之后,在频谱预分配阶段,每个簇头用 M 个时隙进行广播簇内节点数目,然后再占用 M 个时隙广播其所占用的频谱,故整个过程需预留 $2M^2$ 个时隙;(3) 在潜在协作集协商过程中:① 每个簇头首先利用两个时隙广播 Adhering 消息给两跳内的次级节点,每个次级节点计算后进行广播以声明自身是否为争议节点并反馈回簇头.因此,需预留时隙 $2M+2N$ 个;② 每个簇头在分配给它的时隙内进行 Abandon 请求,占用 $2M$ 个时隙,每轮中,若次级节点为争议节点,则在分配给它的时隙内反馈信息,因此,后续为每个争议节点分配的时隙为 $2N$.设 DGA-2 进行轮数最多为 $Round$,共需预留 $(2M+2N) \times Round$ 个时隙.由步骤①和步骤②可得,潜在协作集协商阶段共需预留 $(2M+2N) \times (Round+1)$ 个时隙.注意到,有些簇在第 $Round$ 轮之前便结束协商,进入适应值精化阶段.综上,设通用控制信道的传输速率为 R ,可简单估算分布式算法整个过程中节点通信开销为 $T_{com} = [3M^2 + (2M+2N) \times (Round+1)] \times L/R$.在后续仿真实验中我们发现,与各簇进行遗传算法的计算时间(10^0 数量级)相比,额外通信开销的时间较短,通常在 10^{-1} s 的数量级.

6 性能评估

我们用 Matlab 设计仿真程序对 CGA, DGA 和 FDGA 进行性能评价.设主用户发送节点与次级用户发送节点分别均匀分布于 1×1 单位面积的方形区域,主用户接收节点与次级用户接收节点则随机分布在对应的发送节点最大传输半径内.用三元组 $NS(M, N, D)$ 表示网络规模,在网络规模较小的情况下,通过对比 CGA 得到的优化解与穷举算法(brutal search,简称 BS)得到的最优解来分析 CGA 算法的收敛性和有效性.当网络规模较大时,由于很难在短时间内通过 BS 得到最优解,则通过比较 CGA 与 DGA, FDGA 来分析分布式算法的性能.若不特殊说明,每个网络规模取 50 个实例,每个实例运行 50 次算法.仿真结果为 2 500 次结果的平均值.

仿真参数见表 3.

Table 3 Simulation parameters

表 3 仿真参数

PT 节点数量 M	[3,50]	ST_j 权重或出价 ω_j	[1,10]随机分布
ST 节点数量 N	[8,100]	变异概率 P_m	0.01
可用频谱数量	M	算法最大迭代次数 MAX	50
P_j^s / N_0	10	初始种群联合染色体个数 $ CN $	30
P_i^p / N_0	10	Beacon 信息长度 L	16bit
传输半径 $D_i^p = D_j^s = D$	[0.253,0.7]	通用控制信道传输速率 R	2Mbps
$g_{ij}^k, g_{ii}^k, g_{jj}^k, g_{ji}^k$	采用文献[17]中的参数设置,即 $g_{ij}^k = d_{ij}^{-\gamma} / \xi_{ij}^k$, d_{ij} 为任意节点之间的欧式距离; $\gamma=2$ 为路径损耗常数; ξ_{ij}^k 是高斯随机变量,符合标准正态分布		

6.1 CGA与BS比较

首先,通过给出一个 DEMO1 而导出 CGA 的运行过程.设网络规模为 $NS(4,10,0.7)$,随机产生该小规模网络的拓扑.图 5 为运行 CGA 得到的结果,圆形节点表示主发送节点,方形节点表示次级发送节点,圆括号中的标号为分配给主发送节点的频谱.节点间若有连线,则表示它们有潜在协作关系;若连线为实线,表明实线相连的次级节点为主节点的真协作集中的节点.在该拓扑下,如图 6 所示,CGA 共进行了 20 轮迭代,在种群的平均适应值达到停机条件后终止,最终的适应值为 3.1787. DEMO1 验证了 CGA 算法的有效性和收敛性,可得到为每个主节点所分配的频谱、潜在协作集和真协作集.

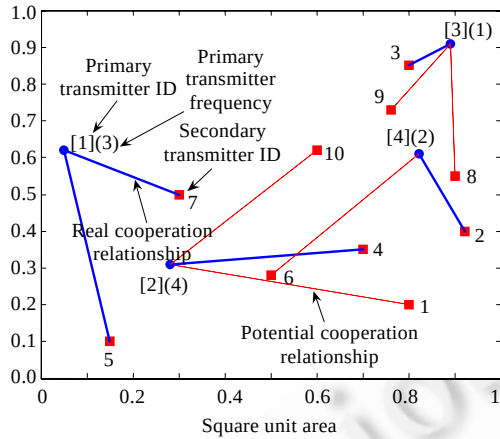


Fig.5 Topology result delivered by CGA to DEMO1

图 5 DEMO1 上运行 CGA 得到的拓扑结果

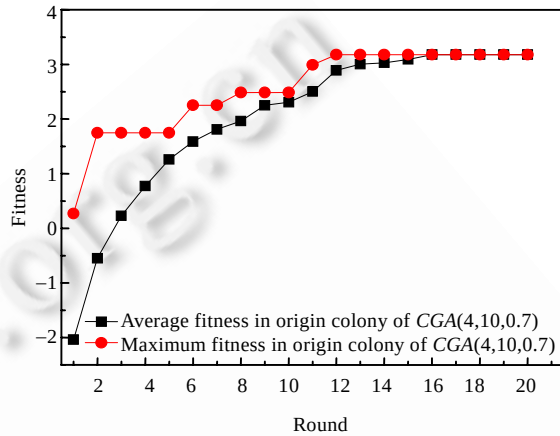


Fig.6 Evolution progress of CGA in DEMO1

图 6 DEMO1 中 CGA 的进化过程

接下来,令网络规模分别为 $(4,8,0.7)$, $(4,9,0.7)$, $(4,10,0.7)$, $(4,11,0.7)$, 比较在不同网络规模下 CGA 与 BS 得到的适应值平均值.图 7 的结果表明,在不同的小规模网络下,CGA 得到的平均值可达 BS 得到的最优适应值的 92%;而算法的平均运行时间较短,为 0.42s.而 BS 的平均运行时间为 203s(未在图中标注).这组实验验证了在小规模网络下 CGA 的有效性和收敛性.

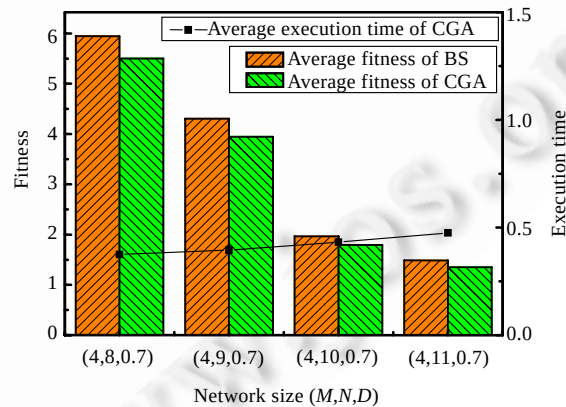


Fig.7 CGA vs. BS under various small-scale networks

图 7 不同小规模网络下 CGA 与 BS 比较

6.2 DGA,FDGA与CGA比较

当网络规模较大时,很难采用 BS 在短时间内就得到最优解.因此,为验证 DGA,FDGA 的有效性和收敛性,

我们将 DGA,FDGA 与集中式算法 CGA 进行比较.先通过一个 DEMO2 给出 DGA 运行步骤.在网络规模为 $NS(20,40,0.4)$ 下产生一个连通拓扑,如图 8 所示.由于 DGA 与 FDGA 仅在频谱的初始分配上有差异,其他过程相同,故仅验证 DGA.DGA 先利用分布式最小支配集算法产生 4 簇, $CT_1=\{2,5,11,16\}$, $CT_2=\{1,3,15,19\}$, $CT_3=\{4,10,18\}$ 和 $CT_4=\{6, 7,8,9,12,13,14,17,20\}$;然后,各簇对争议次级节点进行协商,每一轮适应值都进行更新.这个过程如图 9 所示,其中,1~13 轮为簇间协调过程,第 14 轮进行簇内适应值精化操作.总适应值最终收敛到 41.463 7,运行时间为 4.34s(图中略).最终得到的结果如图 10 所示.

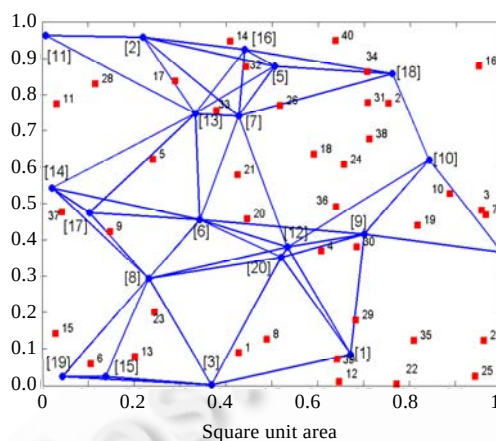


Fig.8 Initial topology of DEMO2 under $NS(20,40,0.4)$

图 8 网络规模为(20,40,0.4)时 DEMO2 的初始拓扑

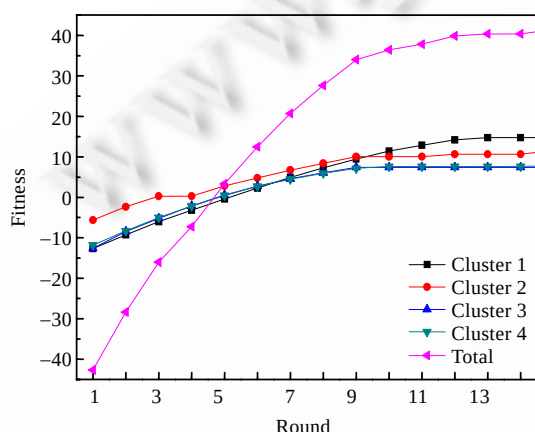


Fig.9 Evolution progress of DGA to DEMO2

图 9 DEMO2 中 DGA 的运行过程

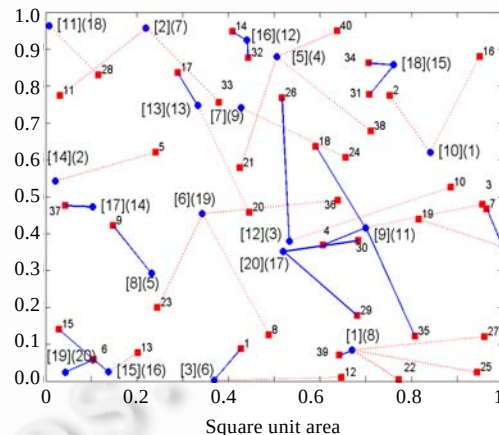


Fig.10 Result delivered by DGA to DEMO2

图 10 DEMO2 上运行 DGA 得到的结果

下面比较 3 种算法在不同网络规模下的性能.我们以 $NS(20,40,0.4)$ 中每个节点的传输半径内的平均节点数为基准,分别设定网络规模为 $(30,60,0.327)$, $(40,80,0.283)$, $(50,100,0.253)$, $(60,120,0.231)$.图 11 的结果说明,随着网络规模的扩大,DGA,FDGA 开始显示性能优势.与 CGA 得到的适应值相比,DGA 得到的适应值平均提高了 23.3%,FDGA 得到的适应值平均提高了 20%.这是由于随着网络规模的扩大,CGA 搜索空间呈指数级增长,收敛时间变长,很难在算法规定的停机条件时达到理想的结果.另外,随着网络规模的扩大,CGA,DGA 的运行时间有所延长,已经不适用大规模分布式系统;而 FDGA 随网络规模的扩大,执行时间仍然较短且增长缓慢.这是由于与 DGA 相比,FDGA 的频谱预分配方式进行了简化,从而大大缩短了频谱初始分配所消耗的时间.需要说明的

是,图 11 所示的运行时间已经包括了算法通信开销,而这个开销较小,是 10^{-1} 量级.例如,以网络规模为 (60,120,0.231) 为例,FDGA 第 2 阶段运行 20 轮,根据第 5.4 节中的额外通信时间开销的计算公式,得到额外通信开销为 0.146s;而这时的计算时间为 2.36s.图 11 所示结果为这两个数值的加和.

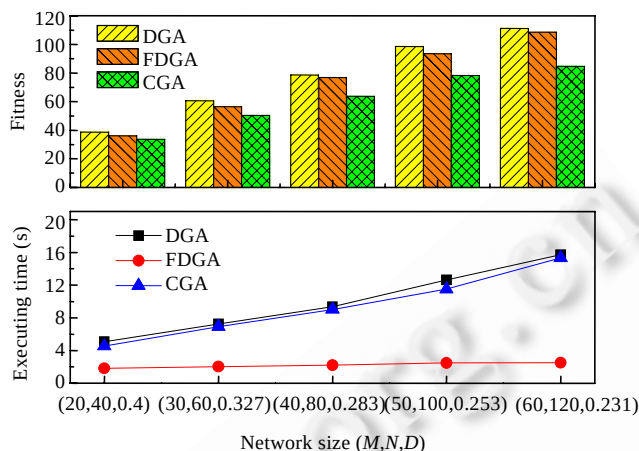


Fig.11 Fitness and execution time delivered by DGA, FDGA, and CGA under various network sizes

图 11 不同网络规模下 DGA,FDGA,BS 得到的适应值和算法运行时间的比较

7 结 论

本文针对协作认知无线网络中较为复杂的多主用户与多次级用户共存场景,研究了联合频谱分配与协作集划分问题,证明了其为 NP-hard 的.先后提出集中式遗传算法 CGA、分布式遗传算法 DGA 和一种快速收敛的分布式算法 FDGA.仿真实验验证了所提出算法的有效性和收敛性,其中,FDGA 算法能够有效求解提出的问题,收敛速度更快,更适合分布式环境应用.本文提出的算法也可用于解决这样一类问题,即网络中存在多个主设备、多个从设备、多个工作方式,每个主设备需要与多个从设备采用某种工作方式协同完成某项工作,不同的指派将依某一线性或非线性函数得出不同的工作产出.优化问题是如何寻求优化的指派,使工作产出最大化.

References:

- [1] Mitola J III. Cognitive radio: An integrated agent architecture for software defined radio [Ph.D. Thesis]. Stockholm: Royal Institute of Technology (KTH), 2000.
- [2] Goldsmith A, Jafar SA, Maric I, Srinivasa S. Breaking spectrum gridlock with cognitive radios: An information theoretic perspective. Proc. of the IEEE, 2009,97(5):894–914. [doi: 10.1109/JPROC.2009.2015717]
- [3] Zhang J, Zhang Q. Stackelberg game for utility-based cooperative cognitive radio networks. In: Proc. of the 10th Int'l Symp. on Mobile Ad Hoc Networking and Computing (Mobihoc). New Orleans: ACM Press, 2009. 23–32. <http://www.sigmobile.org/mobihoc/2009/organizers.html> [doi: 10.1145/1530748.1530753]
- [4] Laneman JN, Tse DNC, Wornell GW. Cooperative diversity in wireless networks: Efficient protocols and outage behavior. IEEE Trans. on Information Theory, 2004,50(12):3062–3080. [doi: 10.1109/TIT.2004.838089]
- [5] Jia J, Zhang J, Zhang Q. Cooperative relay for cognitive radio networks. In: Proc. of the 28th Int'l Conf. on Computer Communications (INFOCOM). Rio de Janeiro: IEEE Press, 2009. 2304–2312. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5062156 [doi: 10.1109/INFCOM.2009.5062156]
- [6] Almasaeid HM, Kamal AE. Assisted-Multicast scheduling in wireless cognitive mesh networks. In: Proc. of the 2010 Int'l Conf. on Communications (ICC). Cape Town: IEEE Press, 2010. 1–5. <http://www.ieee-icc.org/2010/> [doi: 10.1109/ICC.2010.5502790]

- [7] Xu H, Li BC. XOR-Assisted cooperative diversity in OFDMA wireless networks: Optimization framework and approximation algorithms. In: Proc. of the 28th Int'l Conf. on Computer Communications (INFOCOM). Rio de Janeiro: IEEE Press, 2009. 2141–2149. [doi: 10.1109/INFCOM.2009.5062138]
- [8] Jin J, Xu H, Li BC. Multicast scheduling with cooperation and network coding in cognitive radio networks. In: Proc. of the 29th Int'l Conf. on Computer Communications (INFOCOM). San Diego: IEEE Press, 2010. 1289–1297. <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5462203> [doi: 10.1109/INFCOM.2010.5461917]
- [9] Simeone O, Stanojev I, Savazzi S, Bar-Ness Y, Spagnolini U, Pickholtz R. Spectrum leasing to cooperating secondary ad hoc networks. IEEE Journal on Selected Areas in Communications, 2008,26(1):203–213. [doi: 10.1109/JSAC.2008.080118]
- [10] Xu H, Li BC. Efficient resource allocation with flexible channel cooperation in OFDMA cognitive radio networks. In: Proc. of the 29th Int'l Conf. on Computer Communications (INFOCOM). San Diego: IEEE Press, 2010. 561–569. [doi: 10.1109/INFCOM.2010.5462169]
- [11] Walkup DW, MacLaren MD. A multiple-assignment problem. Technical Report, Mathematical Note 347, Boeing Scientific Research Laboratory, 1964. <http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=AD0602672>
- [12] Peng CY, Zheng HT, Zhao BY. Utilization and fairness in spectrum assignment for opportunistic spectrum access. Mobile Network Applications, 2006,11(4):555–576. [doi: 10.1007/s11036-006-7322-y]
- [13] Xie Z. Network Algorithms and Complexity Theory. Changsha: National University of Defense Technology Press, 2003 (in Chinese).
- [14] Xing WX, Xie JX. Modern Optimization Algorithms. Beijing: Tsinghua University Press, 2005 (in Chinese).
- [15] Akyildiz IF, Lee WY, Chowdhury KR. CRAHNS: Cognitive radio ad hoc networks. Ad Hoc Networks, 2009,7(5):810–836. [doi: 10.1016/j.adhoc.2009.01.001]
- [16] Wang PJ, Alzoubi KM, Frieder O. Distributed construction of connected dominating set in wireless ad hoc networks. In: Proc. of the 21st Int'l Conf. on Computer Communications (INFOCOM). New York: IEEE Press, 2002. 1597–1604. <http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=7943> [doi: 10.1023/B:MONE.0000013625.87793.13]
- [17] Wu Y, Tsang DHK. Distributed power allocation algorithm for spectrum sharing cognitive radio networks with QoS Guarantee. In: Proc. of the 28th Int'l Conf. on Computer Communications (INFOCOM). Rio de Janeiro: IEEE Press, 2009. 981–989. [doi: 10.1109/INFCOM.2009.5062009]

附中文参考文献:

- [13] 谢政.网络算法与复杂性理论.长沙:国防科学技术大学出版社,2003.
- [14] 邢文训,谢金星.现代优化计算方法.北京:清华大学出版社,2005.



杨威(1982—),男,辽宁台安人,博士生,CCF 学生会员,主要研究领域为协作认知无线网络资源优化.



梁维发(1963—),男,博士,副教授,博士生导师,主要研究领域为图论,并行算法,无线网络.



班冬松(1982—),男,博士,CCF 学生会员,主要研究领域为无线传感器网络.



莫文华(1946—),男,教授,博士生导师,主要研究领域为高级计算机网络.