

大规模组网的集中式基站休眠算法^①

龙 恩^{②*} 万 溢^{③*} 刘 畅^{****} 田 霖^{****}

(* 重庆邮电大学 重庆 400065)

(** 移动计算与新型终端北京市重点实验室 北京 100080)

(*** 中国科学院计算技术研究所 北京 100190)

(**** 中国科学院大学 北京 100049)

摘 要 分析了通过基站休眠降低无线通信网络能效的集中式算法和分布式算法的原理和性能,在此基础上,针对集中式休眠算法随着网络规模的增大其计算复杂度将会异常巨大的问题,提出了一种面向大规模通信网络的集中式分簇算法。该算法首先在时域上运用多目标均衡优化对休眠时间段进行划分,随后在空域上对基站进行合理的分簇,最后通过粒子群优化算法进行了休眠组合的确定。仿真结果表明,该算法的计算复杂度要低于其他集中式算法,并且性能衰减可以忽略不计,整体的休眠部署更加合理。

关键词 能效, 基站休眠, 集中式, 计算复杂度, 分簇

0 引 言

近年来,随着移动用户以及终端的大幅增加,移动数据业务需求呈现出爆炸式的增长态势,无线通信网络也因此得到了飞速的发展^[1-3]。然而,庞大的移动通信网络在运维时却带来了严重的能耗问题^[4]。根据世界无线研究论坛(Wireless World Research Forum, WWRF)预测,2017 年移动用户数量将达到 70 亿,无线设备总量将达到 7 万亿,这就意味着移动通信网络的能耗压力将越来越严重。文献[5]的研究表明,移动通信网络中基站(base station, BS)单元上的能耗约占总能耗的 80%,因此,提高基站的能效将有助于减少整个移动通信网络的能耗。针对这种情况,本文进行了基站能耗研究,提出了一种面向大规模组网的集中式基站休眠算法,该算法不仅具有良好的降低网络能耗的性能,而且具有较低的计算复杂度。

1 相关研究

在对基站能耗问题的研究中,文献[6]提出了“潮汐效应”的概念。所谓潮汐效应,就是用户呈现出有规律的迁徙行为。例如,在工作时间段,大量用户会从住宅区域迁徙至商务区域,而在休息时间段,用户会从商务区域迁徙至住宅区域,这样就会导致用户在空域中的不均衡性。同时,文献[7]也表明,即使在负载的高峰期,移动通信网络中 90% 的数据业务是由 40% 的小区来承载,基站的任务分配极不均衡。

目前,基站休眠技术被认为是解决上述问题的最有效方法之一^[8-11]。基站休眠技术可以动态地关闭部分低负载基站,并由其相邻的基站进行补偿覆盖,以实现节能减排。网络将实时监控各个基站的负载情况,并在某些特定的休眠时间点,动态地关闭部分负载较低的基站,并由其相邻的一个或多个基

① 863 计划(2015AA01A705),国家自然科学基金(61201231)和国家科技重大专项(2014ZX03003004-003)资助项目。
② 男,1978 年生,博士,讲师;研究方向:5G 关键技术;E-mail: longken@cqupt.edu.cn
③ 通讯作者,E-mail: 307777327@qq.com
(收稿日期:2015-10-20)

站通过调整天线方向角、下倾角、发射功率对休眠基站进行补偿覆盖。

概括来说,基站休眠算法可以分为集中式算法和分布式算法两种。分布式算法通常基于预设的阈值进行判断^[10],在到达休眠时间点时,每个基站将根据其自身以及相邻基站的负载等状态信息判断是否休眠。然而,在分布式算法中,由于每个休眠基站只关注自身以及相邻基站的状态信息,无法获知整个网络的状态信息,所以只能部署局部最优的基站休眠方案。另一方面,集中式算法将由集中式管控模块收集网络中所有基站的负载信息,从网络全局的角度出发,通过集中式算法得到全局最优(或次优)的基站休眠部署方案,然后通知所有基站执行休眠、补偿或正常覆盖的命令。总体来说,集中式算法往往能够得到全局最优(或次优)解,因此,其性能要优于分布式算法^[9,12]。

然而,算法的计算复杂度是进行休眠部署时需要考虑的另一个问题。本文定义计算复杂度为运行一个算法所需进行迭代的总次数。例如分布式休眠算法中,每个基站只需要考虑自身以及相邻基站,复杂度比较低;而集中式算法需要整体考虑网络中所有基站的可能组合方式,复杂度比较高。目前,集中式接入网架构成为了学术界以及产业界的研究热点,比如 C-RAN^[6],以及我们提出的超级基站^[13],它主要包括三部分:由远端无线射频单元和天线组成的分布式无线网络;由高带宽低延迟的光传输网络连接远端无线射频单元;由高性能处理器和实时虚拟技术组成的集中式基带处理池。在超级基站中,几个基站为小规模基站,25 个左右的基站称为中规模基站,40 ~ 100 个左右的基站成为大规模基站。由于集中式架构将所有信息都集中到一起进行处理,因此,集中式接入网架构为集中式算法提供了良好的平台,集中式算法成为主要的发展趋势。

在针对集中式休眠算法的研究中,文献[9]就提出了一种集中式的算法,该算法要求遍历每个用户能够接入的基站,并且找寻到最优的接入基站,这样能够得到比较好的结果,但是用户如果比较多,那么该算法计算复杂度就会变得特别大。文献[11]结合能耗和服务质量(quality of service, QoS),提出

了一种自适应的集中式算法,但是该算法在基站比较少时性能较好,复杂度也较低,但是随着基站的增多,性能却将大大降低。在我们以前的研究中^[12],我们对于能耗效率与稳定性的均衡问题进行了研究,并且运用了快速穷举算法以及粒子群优化(particle swarm optimization, PSO)算法进行了仿真实验。但是我们发现,快速穷举算法以及粒子群算法等最优化算法在网络中基站规模比较小(14 个基站以下)时能够得到很好的运用,然而在基站规模比较大(40 个以上)时则会因为其太高的复杂度而很难得到理想的结果甚至难以计算。文献[14]也表明,将全部基站进行协作可以得到最佳的休眠决策,但是由于其高复杂度而不能在实际中实现。

针对上述问题,本文进行了面向大规模组网的集中式分簇休眠算法的研究,具体的研究内容包括:首先在时域上根据各个基站的负载趋势与负载趋势门限值,设计了一种通过复杂度和能耗的多目标优化寻找最优门限值的方法,运用此方法合理地将一天划分为多个时间段,让网络在每个时间段内分簇方式相同;其次在空域上根据每个基站的负载以及其邻区数量,设计了一个集中式架构下的分簇算法,在每个时间段起始时刻对无线网络进行分簇处理;最后对于每个簇,运用 PSO 算法得出其中最优的基站休眠组合方式。

2 系统模型

2.1 网络架构

如图 1 所示,若要在现有的分布式网络架构中实现集中式休眠算法,则必须在网络架构中添加集中式的管理模块。该模块必须具备收集所有基站信息(负载、功率、频谱等),并且可以对这些信息进行计算处理的能力。此外,还需要具备对所有基站进行状态(休眠或开启)改变的决策能力。而在集中式架构中,集中式休眠方式可以得到更方便的实现。集中式架构本身就拥有集中管理模块,它可以将无线资源(如频谱、时隙等)和硬件资源(如远端无线射频单元(remote radio head, RRH)、基带单元(base band unit, BBU)等)分配给各个虚拟基站。与此同

时,集中管理模块还能收集各类信息并进行集中管理。

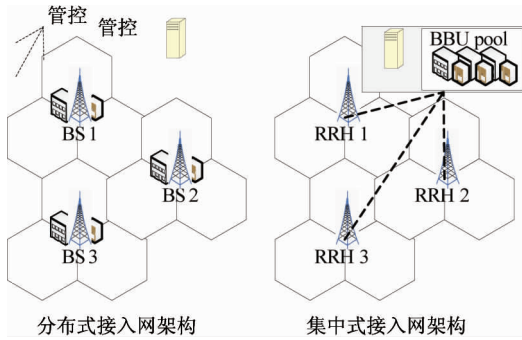


图1 集中式休眠机制的实现方法

2.2 能耗模型

假设集中式无线网络架构中有 M 个 RRH 和 N 个用户 (User Equipment, UE), 每个 RRH 有 N_{TRX} 根天线, 每个 RRH 的负载上限为 N_R , 文献[15]给出了其能耗模型:

$$P_{\text{in}} = \begin{cases} M \cdot N_{\text{TRX}} \cdot P_0 + \Delta P \cdot \sum_{i=1}^N P_{\text{out}_i}, & 0 < P_{\text{out}_i} < P_{\text{max}} \\ N_{\text{TRX}} \cdot P_{\text{sleep}}, & P_{\text{out}_i} = 0 \end{cases} \quad (1)$$

其中, P_{out_i} 表示满足 UE i 需求的功率。 P_0 表示在非休眠情况下每根天线的最小输出功率, ΔP 表示负载相关能耗的斜率。 P_{sleep} 表示 RRH 休眠状态下的能耗。

2.3 接入模型和优化模型

为了表示 RRH 与 UE 的连接关系, 定义一个连接矩阵 $X = (x_{n,m})_{N \times M}$, $x_{n,m} = \{0, 1\}$ 。当 UE n 接入 RRH m , $x_{n,m} = 1$, 反之 $x_{n,m} = 0$ 。假设 $P = (p_{m,n})_{M \times N}$ 表示 RRH 与 UE 的传输功率矩阵, $p_{m,n}$ 表示 RRH m 发送给 UE n 的传输功率, 则 $p_{m,n} = X \cdot P$ 。假设频率复用因子为 1, 系统带宽为 W , 所有接入 RRH m 的 UE 的数量 $\kappa m = \sum_{n=1}^N (x_{n,m})_{N \times M}$, 噪声功率为 σ^2 , 则 RRH m 分配给 UE n 的带宽为 $B_{m,n} = W/\kappa m$ 。RRH m 与 UE n 之间的信道增益只考虑路损, 设为 $PL_{m,n}$, 则 RRH m 与 UE n 之间的信干噪比为

$$\text{SINR}_{m,n} = \frac{PL_{m,n} \cdot P_{m,n}}{\sum_{j \neq m} PL_{j,n} \cdot P_{m,n} + \sigma^2} \quad (2)$$

RRH m 发送给 UE n 的传输速率为

$$R_{m,n} = B_{m,n} \cdot \log_2(1 + \text{SINR}_{m,n}) \quad (3)$$

为了得到集中式架构下最优的能耗所对应的休眠组合方式, 建立如下优化模型:

$$\min_{m,n} Z = P_{\text{in}} \quad (4)$$

$$\text{s. t. } x_{n,m} = \{0, 1\} \quad (5)$$

$$\sum_m x_{m,n} = 1 \quad (6)$$

$$R_{m,n} \geq \varepsilon n \quad (7)$$

$$d_{m,n} \leq 3 \times r \quad (8)$$

式(7)表示 RRH m 发送给 UE n 的传输数据的速率 $R_{m,n}$ 必须满足 UE n 最低的传输数据的速率要求 εn 。式(8)表示 RRH m 休眠以后, 原来接入 RRH m 的 UE 只能接入 RRH m 相邻的 RRH 中, $d_{m,n}$ 为 RRH m 到 UE n 的距离, r 为 RRH 的半径。

3 集中式休眠算法

3.1 穷举算法

为了解决上述优化问题, 通过穷举法可以得到最优解, 但是随着 RRH 数量的增多, 穷举法的复杂度会非常高甚至难以计算。

假设整个网络中有 M 个 RRH, 每个 RRH 有两个状态: 开启和关闭。UE n 可以接入的 RRH 数量为 V_N , 则此时穷举法的复杂度为 $2^M \cdot \prod_n V_N$, 这个复杂度是相当高的。由于用户接入离其最近的 RRH 可以得到最大的接收功率, 我们可以让每个用户只接入离其最近的 RRH。但是会有一种情况, 如果对于某个用户, 离其较远的 RRH 比离其最近的 RRH 提供了更大的带宽, 这时对该用户可能会有很小的损失。但是由于这种情况几乎不会发生, 因为我们对整个网络进行的集中规划, 会使开启的 RRH 得到充分的利用, 这样每个 RRH 提供的带宽不会有太大的区别。在上述用户接入情况下, 穷举算法的复杂度可以降低到 2^M 。这个数量在 M 比较小时, 复杂度并不高, 但是由于复杂度呈指数增加, 所以当 M 增大, 复杂度会变得非常高甚至导致没办法计算, 比如 $M=21$ 时 RRH 的组合方式就有 200 多万种。

由于穷举算法的复杂度过高, 所以本文提出了一种结合粒子群优化算法的分簇算法, 应用于 RRH

数量比较多的情况。

3.2 PSO 分簇算法

本文提出的分簇算法主要分为三个方面:(1)在时域上根据全网络的平均负载对全天进行时间段划分,每个时间段的起始时刻为分簇决策时间点(cluster point, CP)。(2)在空域上对每个 CP 运用分簇算法对 RRH 进行分簇,在同一时间段内分簇方式不改变。(3)对每个簇运用 PSO 进行求解。假设将整个网络划分为 a 个簇,第 i 个簇 CLu_i 运用的 PSO 的粒子群数目和完成的迭代次数分别为 b_i, k_i ,其中 $b_i = 2^{M_i} \times \tau$, M_i 为第 i 个簇的 RRH 数量, τ 为一个 $[0 \sim 1]$ 的小数。则此时的算法的复杂度为 $\sum_{i=1}^a b_i \times k_i$ 。随着 RRH 的增多,进行分簇划分的簇数量 a 会增加,这样就大大地降低了复杂度。

3.2.1 时间段划分方法

假设一天有 S 个切换触发时间点 (switching point, SP),在不同的 SP 时刻,RRH 的负载情况都不相同,相应地,休眠簇的数量和每个簇的大小也不相同。如果在每个 SP 都对 RRH 进行分簇,这样分簇操作就会过于频繁,而且可能会进行没有必要的分簇操作。所以我们根据每个 SP 的负载和负载趋势门限值,设计了一种通过能耗和复杂度的多目标优化的方法寻找最优负载趋势门限值,根据门限值来对全天 24 个小时进行时间段 (time divide, TD) 的划分。在时间段 TD_i 中,只是在 TD_i 开始的 SP 对 RRH 运用分簇算法进行分簇,在整个 TD_i 的 SP 保持 RRH 的分簇方式不变。

对于 TD 的划分根据以下两个条件进行:(1)对于 TD_i 中每个 SP,其负载与 TD_i 中的其他每个 SP 的负载的差值都必须小于负载趋势门限值 $G, G = \varphi \times Load_{\max}$,其中 $Load_{\max}$ 为网络满载时的负载值, φ 为 $[0 \sim 1]$ 的小数,表示 G 所占满载的比例。(2)对于 TD_i 中的所有 SP,必须保证所有 SP_{*i*} 是连续的,比如 SP₁、SP₃ 不能单独分到一个 TD 中,SP₁、SP₂、SP₃ 就可以分到一个 TD 中。

由于门限值 G 的不同,就会导致时间段的划分不同,从而得到的能耗和复杂度也不会相同。所以必须选择出最佳的门限 G 。在选择最佳的门限 G 时,还存在这样一个问题:对于分簇来说,对于同样

一个集中式网络,分簇得到的簇越多,就可能会导致能耗增加,这是由于分簇会导致一部分 RRH 的邻站减少(只有位于同一个簇且位置上相邻的 RRH 才能称为邻站),那么能够对其补偿的 RRH 减少,这样就可能导致本可休眠的 RRH 无法休眠,最后能耗会增加。这也是为什么穷举算法可以得到最优解,因为它相当于将整个网络划分为一个簇来进行计算。但是如果分簇越多,那么平均下来每个簇中的 RRH 数量就会减少,从而会使得每个簇进行集中式休眠算法的复杂度变低,这样也是穷举算法在 RRH 数量比较大的时候复杂度如此之大的原因。而对于不同的门限 G ,TD 的划分是不同的,那么就会有这样的情况:假设 SP₁ 时网络分为 3 个簇,SP₂ 分为 4 个簇,SP₃ 分为 2 个簇。如果 SP₁ 和 SP₂ 分为一个 TD,那么 SP₂ 就为 3 个簇,比原来少了,这样就会使能耗变低,但是复杂度变高了。如果 SP₂ 和 SP₃ 分为一个 TD,这样 SP₃ 就为 4 个簇,簇变多了,这样能耗可能变高,但是复杂度低了。这样在选择最佳的门限 G ,能耗与复杂度就会存在一个均衡。

针对上述均衡问题,可以建立如下模型:

$$\min_{m,n} C = \alpha \cdot A + \beta \cdot B \quad (9)$$

$$s. t. x_{n,m} = \{0,1\} \quad (10)$$

$$\sum_m x_{m,n} = 1 \quad (11)$$

$$R_{m,n} \geq \varepsilon n \quad (12)$$

$$d_{m,n} \leq 3 \times r \quad (13)$$

其中式(10)~(13)均与 2.3 节中的优化模型相同。优化目标(或式(9))中 α, β 为 $[0,1]$ 范围内的数,分别代表 A, B 的权重因子,且 $\alpha + \beta = 1$ 。 A 为全网络的归一化能耗,表示为

$$\min_{m,n} A = \frac{P_{in}}{\max P_{in}} \quad (14)$$

其中 P_{in} 为当前门限 G 的能耗, $\max P_{in}$ 为所有门限 G 中最大的能耗。

B 为全网络的归一化复杂度 (Complexity),表示为

$$\min_{m,n} B = \frac{Complexity}{\max Complexity} = \frac{\sum_{i=1}^a b_i \cdot k_i}{\max \sum_{i=1}^a b_i \cdot k_i} \quad (15)$$

其中 a 为簇的个数, b_i, k_i 为第 i 个簇的 PSO 粒子群数量和迭代次数。Complexity 为当前门限 G 的复杂度, $\max \text{Complexity}$ 为所有门限 G 中最大的复杂度。我们通过式(9)得到的最小的 C 所对应的门限 G 就为最优的门限。对于不同的 RRH 数量以及分布, 最优门限 G 都是不同的, 但是均可以用此方法得到。

3.2.2 分簇算法

在每个时间段的起始 TP, 网络是根据以下分簇算法进行 RRH 的分簇处理的。

分簇算法是根据每个 RRH 相邻的 RRH 数量 Nei 和负载 $Load$ 来进行分簇的。算法流程图见图 2。具体步骤如下:

(1) 初始化簇 CLu 的下标 $k = 0$ 。

(2) 判断是否存在 RRH_s 满足 $Nei_s > 0$ 并且 $RRH_s \notin Clu$ (处于不同簇的 RRH 即使相邻也不能称为邻站), 如果有则进行步骤(3), 如果没有则进行步骤(5)。

(3) 判断所有满足步骤(2)条件的 RRH_s 中是否有 RRH 满足 $Nei_s = 1$, 如果没有则进行步骤(4), 如果有则假设该 RRH_s 的唯一邻站为 RRH_i 并加入簇 CLu_k 中。遍历 RRH_i 的邻站, 每次选出邻站中 Load 最小的 RRH 假设为 RRH_j, 判断簇 k 中的负载总和 $Load_k$ 与 RRH_j 的负载 $Load_j$ 之和是否大于门限 T , 如果大于则不能将 RRH_j 加入簇 k 中, 如果小于门限 T , 则将 RRH_j 加入簇 CLu_k 中。遍历完成后 $k = k + 1$ 。返回步骤(2)。

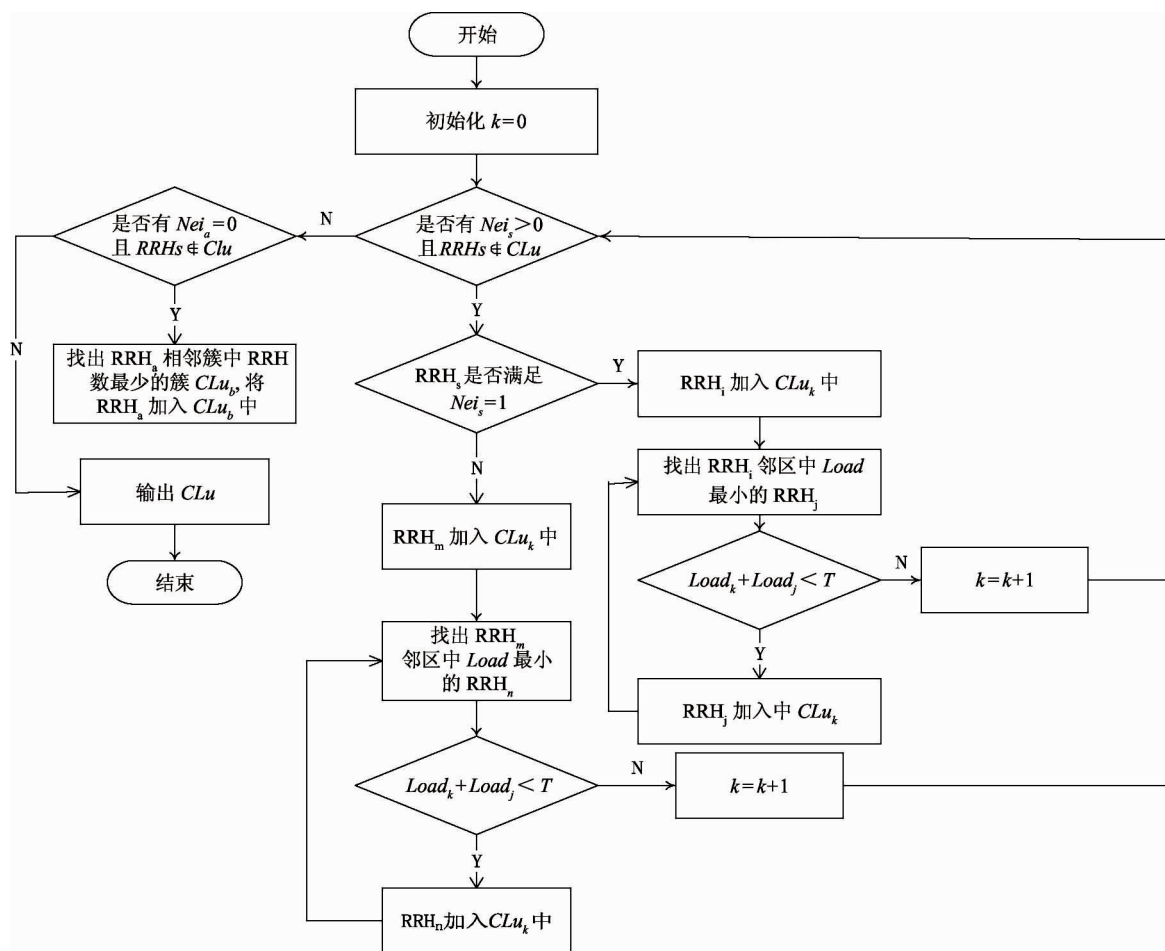


图 2 分簇算法流程图

(4) 选出所有满足步骤(2)条件的 RRH_s 中 Nei 最大的 RRH 并且假设为 RRH_m (如果邻站数目最多的 RRH 包含多个, 则优先对其邻站分布于无线网络

边缘的 RRH 进行分簇处理) 并加入簇 CLu_k 中。遍历 RRH_m 的邻站, 每次选出邻站中 Load 最小的 RRH 假设为 RRH_n, 判断簇 CLu_k 的负载总和 $Load_k$ 与

RRH_n 的负载 $Load_n$ 之和是否大于门限 T , 如果大于则不能将 RRH_n 加入簇 k 中, 如果小于则将 RRH_n 加入簇 CLu_k 中。完成遍历后 $k = k + 1$ 。然后返回步骤 (2)。

(5) 判断是否有 RRH_a 满足 $Nei_s = 0$ 并且 $RRH_a \notin Clu$ 。如果有进行步骤 (6), 如果没有则输出分簇集合 CLu , 算法结束。

(6) 选出满足步骤 (5) 条件的 RRH_a, 遍历其相邻的簇, 选出其中包含 RRH 数目最少的簇假设为 CLu_b , 将 RRH_a 加入簇 CLu_b 中。返回步骤 (5)。

3.2.3 PSO 算法

PSO 首先将解空间初始化为一群粒子, 每个粒子都有一个根据被优化的函数而确定的适应值, 都有自己的速度和位置以决定它的飞行距离和方向; 然后粒子根据自己迄今找到过的最好的位置和整个种群目前找到的最好位置来更新自己的速度和位置, 经过不断的迭代过程最终找到最优解^[16]。

在本文中, 我们根据本文中的优化问题对 PSO 算法进行了如下定义。我们定义每个 RRH 的开启关闭状态的组合为一个粒子, 并组成一个数组 $H = [h_i]$, 其中 $h_i = \{0, 1\}$ 分别表示 RRH_i 的关闭和开启状态。在初始化时, 我们对 RRH_i 进行 $[0, 1]$ 的随机初始化, 得到值 G_i , 如果 $G_i > 0.5$, 则 $h_i = 0$, 反之 $h_i = 1$ 。在速度方面, 对 RRH_i 进行 $[0, 1]$ 的随机初始化, 得到的值 V_i 就为 RRH_i 的初始速度。则 PSO 的位置更新模型就为

$$Gi(t + 1) = Gi(t) + Vi(t + 1) \tag{16}$$

$$Vi(t + 1) = \omega \cdot Vi(t) + c_1r_1[G_{ipbest} - Gi(t)] + c_2r_2[G_{gbest} - Gi(t)] \tag{17}$$

其中, $G_i(t + 1)$ 和 $G_i(t)$ 分别为一个粒子中 RRH_i 的 $t + 1$ 次和 t 次迭代的位置。 $V_i(t + 1)$ 和 $V_i(t)$ 分别为一个粒子中 RRH_i 的 $t + 1$ 次和 t 次迭代的速度。 G_{ipbest} 为 RRH_i 是到目前为止搜索到的最好的位置, G_{gbest} 为整个种群到目前为止搜索到的最好的位置。 c_1, r_1, c_2, r_2 是 PSO 中的参数, 具体数值参照文献[17]。 ω 是权重因子, 在改进的 PSO 算法中, 为了增大算法的搜索能力, 我们将其设为随机从 $\{-1, 0, 1\}$ 中选取。

在算法迭代循环中, 迭代停止的标志为达到了

最大迭代次数 S_{max} 或者在连续的 S_a 次迭代都得到相同的最好解。 S_a 是经过多次仿真实验得出的数值。

PSO 的算法流程图如图 3 所示。

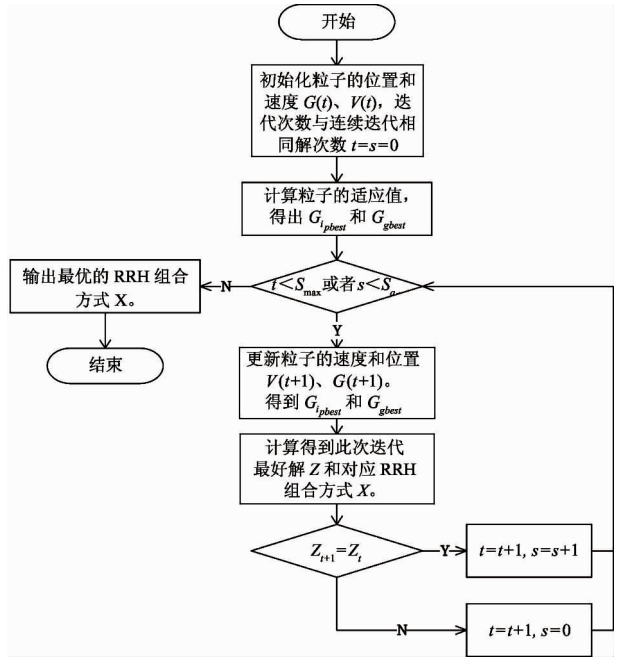


图 3 PSO 算法流程图

4 实验

本文的仿真是建立在 LTE 系统的基础之上, 能耗模型的参数参照文献[15], 更详细的参数见表 1。

表 1 仿真参数

参数	数值
M	14
W	20 (MHz)
N_R	50
ε	100 (kbps)
λ	0.4
r	200 (m)
S	24
σ^2	-174 (dB/Hz)
$PL(d)$	$137.5 + 35.2 \cdot \log_{10}(d)$
S_a	10
S_{max}	100
τ	0.25

首先,我们先研究了时间划分门限 G 所占满载的比例因子 φ 与划分最优目标 C 之间的关系,以此来寻找最优的时间划分门限值 G 。在这次仿真中假设有 14 个 RRH,其中包含 6 个商务区和 8 个住宅区, α 和 β 的取值均为 0.5,表示在此仿真中,能耗和复杂度是同等看重的,但是如果更看重能耗性能,可以适当加大 α 的值,减小 β ,反之更看重复杂度性能,则加大 β ,减小 α 。我们可以从图 4 看出, φ 的取值从 0 到 1,并且间隔是 0.05,可以表示 G 所占满载的比例是从 0 到 100%,并且以 5% 增长。随着 φ 的增大,表示门限 G 增大,这样就会使得时间区间变长以及时间区间数量的变小,直到将一天划分为一个时间段。在 φ 比较小时,由于时间区间段区间数量多,同一区间之间的负载都相差不大,可以得到的分簇比较适用于此时间段,所以 C 整体比较小。在 φ 比较大时,时间区间划分比较长,这样可能会使得负载相差比较大的时间点划分到了同一个时间段内,从而使得此时间段开始时进行的分簇不一定能适合整个时间段,导致 C 会比前面有很大的差别,比如 $\varphi = 0.5$ 前后。我们可以看出,曲线在 $\varphi = 0.1$ 时得到了最小的 C ,这说明在同等看重能耗和复杂度的情况下,选择满载的 10% 作为时间划分门限可以得到最优的结果。

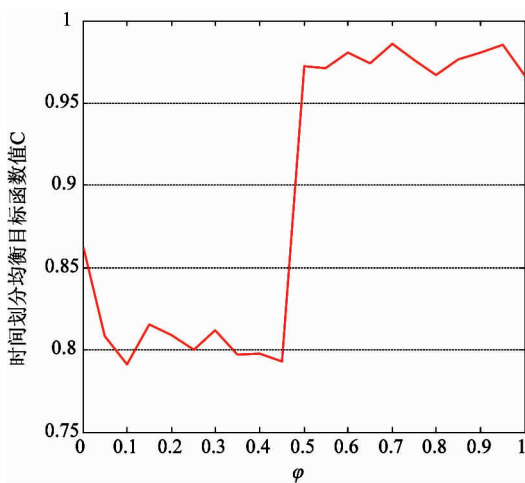


图 4 时间划分均衡函数 C 与 φ 的关系

图 5 表示文献[12]中运用的 PSO 算法、快速穷举算法和本文提出的分簇算法的能耗性能对比。在此仿真中假设有 14 个 RRH,其中包括 6 个商务区

和 8 个住宅区, $\varphi = 0.1$ 。可以从图中看出,在 1~9 点的低负载区域,3 种算法在能耗性能上一样,在 10~23 点的高负载区域,本文提出的分簇算法与 PSO 算法相差最大的值也仅占 3% 左右,一般相差 1% 到 2% 左右。而与最优的快速穷举算法最大相差也仅 6%,一般相差 1% 到 4%。可以说本文提出的分簇算法与 PSO 算法和最优结果快速穷举算法在能耗性能上差别不大,这就说明随着 RRH 的增多,分簇算法也能得到跟最优的结果相差不大的能耗性能,满足能耗性能的需求。

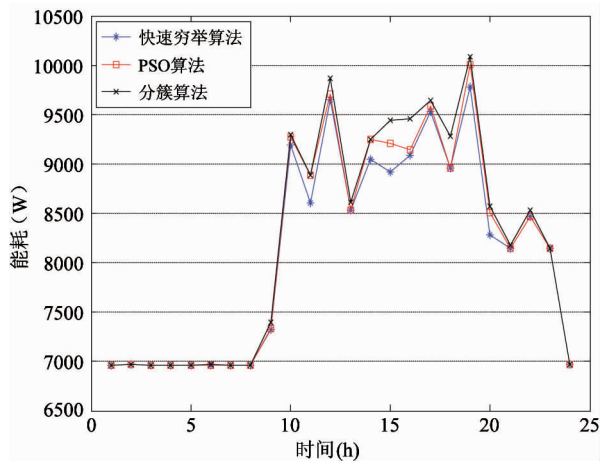


图 5 快速穷举算法、分簇算法和 PSO 算法的能耗性能

图 6 给出了分簇算法与 PSO 算法和快速穷举算法在每个时间点上的复杂度对比,图7给出了分簇算法与 PSO 算法和快速穷举算法每个时间点的平均复杂度的对比。假设也是 14 个 RRH,6 个商务

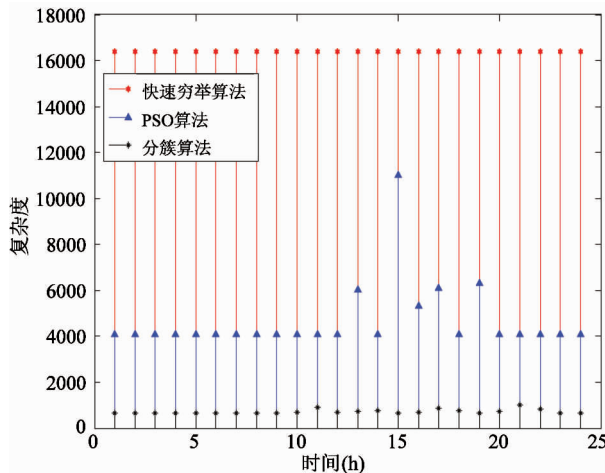


图 6 分簇算法、快速穷举算法和 PSO 算法的复杂度

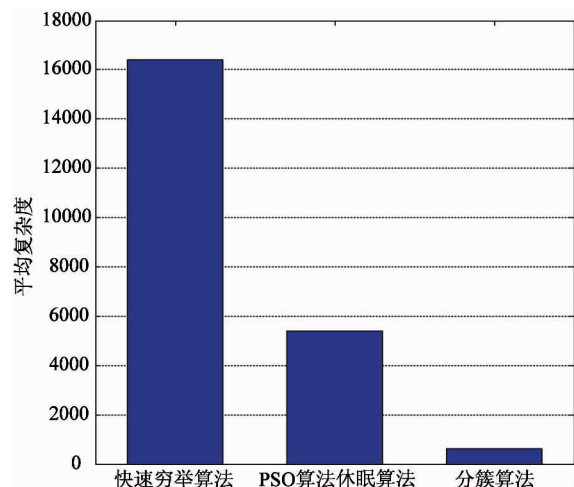


图 7 分簇算法、快速穷举算法和 PSO 算法的平均复杂度

区和 8 个住宅区, $\varphi = 0.1$ 。我们可以看出,在上述能耗性能相差并不是很大的情况下,分簇算法大大降低了复杂度,比 PSO 降低了 80%,比快速穷举算法降低了 90% 以上,而且随着 RRH 数量的增多,这个差距会更加的大。并且在 RRH 数量很大使得最优的穷举算法不能进行运用时,我们也可以运用复杂度很低的分簇算法来进行计算。

5 结 论

本文研究了一种面向大规模通信网络的集中式休眠算法。首先,在时域上构建了一个均衡的优化目标来进行时间段的划分,通过调整时间划分门限所占比例因子的值来得到最优的时间划分门限值。随后,基于时间段的划分,在每个时间段起始时刻设计了一种分簇休眠算法从空域上进行 RRH 的分簇。最后,对每个簇运用 PSO 算法来得到最优的 RRH 开关组合方案。本文对该分簇算法与 PSO 算法和快速穷举算法进行了仿真对比。仿真结果表明,本文提出的时间段划分方法以及分簇算法,在能耗性能上与 PSO 算法和最优的快速穷举算法相差不大,而复杂度比 PSO 算法和快速穷举算法得到了很大的降低。

参考文献

[1] Garcia V, Zhou Y, Shi J L. Coordinated multipoint transmission in dense cellular networks with user-centric adap-

tive clustering. *IEEE Transactions on Wireless Communications*, 2014, 13(8): 4297-4308

- [2] Zhou Y, Pan Z G. Impact of LPF mismatch on I/Q imbalance in direct conversion receivers. *IEEE Transactions on Wireless Communications*, 2011, 10(6): 1702-1708
- [3] Zhou Y, Wang J, Sawahashi M. Downlink transmission of broadband OFCDM systems——Part I: hybrid detection. *IEEE Transactions on Communications*, 2005, 53(4): 718-729
- [4] Fettweis G, Zimmermann E. ICT energy consumption-trends and challenges (2008). In: Proceedings of the 11th International Symposium on Wireless Personal Multimedia Communications, Lapland, Finland, 2008. 1-6
- [5] Fehske A, Fettweis G, Malmolin J, et al. The global footprint of mobile communications: The ecological and economic perspective. *IEEE Communications Magazine*, 2011, 49(8): 55-62
- [6] 黄宇红. C-RAN 无线接入网绿色演进白皮书. 北京: 中国移动通信研究院, 2010
- [7] Chiaraviglio L, Ciullo D, Meo M, et al. Energy-efficient management of UMTS access networks. In: Proceedings of the 21st IEEE International Teletraffic Congress, Paris, France, 2009. 1-8
- [8] LTE for UMTS-OFDMA and SC-FDMA Based Radio Access. John Wiley & Sons, 2009
- [9] Niu Z S, Wu Y Q, Gong J, et al. Cell zooming for cost-efficient green cellular networks. *IEEE Communications Magazine*, 2010. 48(11): 74-79
- [10] Guo W S, O' Farrell T. Dynamic cell expansion: traffic aware low energy cellular network. In: Proceedings of Vehicular Technology Conference, Quebec City, Canada, 2012. 1-5
- [11] Zhu Y, Kang T, Zhang T, et al. QoS-aware user association based on cell zooming for energy efficiency in cellular networks. In: Proceedings of the IEEE 24th International Symposium on Personal, Indoor and Mobile Radio Communications, London, UK, 2013. 6-10
- [12] Liu C, Wan Y, Tian L, et al. Base station sleeping control with energy-stability tradeoff in centralized radio access networks. In: Proceedings of the IEEE Global Communications Conference, San Diego, USA, 2015. 1-6
- [13] Zhai G W, Tian L, Zhou Y Q, et al. Load diversity based optimal processing resource allocation for super

- base stations in centralized radio access networks. *Science China Information Sciences*, 2014, 57(4): 1-12
- [14] Gong J, Zhou S, Niu Z S. A dynamic programming approach for base station sleeping in cellular networks. *Ieice Transactions on Communications*, 2012, 95(2): 551-562
- [15] Auer G, Giannini V, Desset C, et al. How much energy is needed to run a wireless network? *IEEE Wireless Communications*, 2011, 18(5): 40-49
- [16] Kennedy J. Particle swarm optimization. In: *Encyclopedia of Machine Learning*. Springer US, 2011. 760-766.
- [17] Shi Y, Eberhart R C. Empirical study of particle swarm optimization. In: *Proceedings of the 1999 Congress on Evolutionary Computation*, WASHINGTON, DC, USA, 1999, 3. 1945-1950

A base station sleep algorithm for large-scale centralized networks

Long Ken^{*}, Wan Yi^{*}, Liu Chang^{** ****}, Tian Lin^{**}

(^{*} Chongqing University of Posts and Telecommunications, Chongqing 400065)

(^{**} Beijing Key Laboratory of Mobile Computing and Pervasive Device, Beijing 100080)

(^{***} Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190)

(^{****} University of Chinese Academy of Sciences, Beijing 100049)

Abstract

The principle and performance of the centralized base station (BS) sleeping algorithm and the distributed BS sleeping algorithm for reducing the energy consumption of wireless communication networks were analyzed, and then, a centralized clustering algorithm (CCA) for BS sleeping was proposed for large scale wireless communication networks to solve the problem that the computational complexity of the centralized algorithm will become very large when the number of BSs increases. The CCA first formulates a bi-objective problem to divide the sleep period, and then the BSs are divided in clusters to obtain the combination of BSs' states through the particle swarm optimization (PSO). The simulation results show that the CCA significantly outperforms the PSO and fast exhaustive algorithms in the computational complexity while keeping the same performance in energy saving as these algorithms'.

Key words: energy efficiency, BS sleep, centralized, computational complexity, cluster