问题模型

 存在从基站i向用户j的链接，t对应时隙

 优化目标为最小化功率

 QoS限制（容量）（对每个用户）

 用户单天线：同时用户只能从单源接收或向他人发送（简化模型）

 t时刻j节点保存容量

  为j到j’’间信息的复用率



 功率限制，基站可同时向多个用户传输,中继只能同时向单一用户传输(单天线)

 基站能同时服务用户数量限制（单基站，多天线）

简化问题描述：

First, we decompose the problem into three sub-problems as follows:

①: Ignore the D2D constraints and assume that the stations are omnipotent: that is, stations can serve infinite number of users.

Assume that the base station use max power  in transmitting. Since the transmission rate  is a monotonous increasing function of  , we can find the biggest  to achieve a minimal power system.

 优化目标为最小化功率

 QoS限制（容量）

 单天线：同时用户只能从单源接收

容量



Denote the set of  in this sub-problem as , which implies that there exists a link  at time slot , denoted as  . In this case, for any user j, we can find the best (i.e. biggest) where that meets the requirement. Choose it and set  .

Then we will have an optimal solution for each user j (regardless of station capacity constraints and D2D constraints).

②: put station capacity (i.e. max number of users that can be served at any time slot) into consideration.

 功率限制，基站可同时向多个用户传输

系统总服务能力限制

This is equivalent to

 Given the above constraints

We propose a recursion method to acquire a good result from the optimal solution in ①

对未满足限制的时隙t，对该时隙中所有用户j，找到替换为其它时隙中次优解后，对当前系统影响最小的用户（容量变化最小）

for all timeslot  while  not met :

//找:替换为其它时隙中次优解后，对当前系统影响最小的边（容量变化最小）.去掉该边.

find , where , , 

set , 

//去掉该边后，寻找一组次优解满足QoS限制

while  not met :

find , where 

set  ,



③: put D2D into consideration:

We have already claimed a best result under station capacity constraint in a  subspace. This part we dig further into the D2D constraint in a  subspace.

we denote the edges we have in part ② as  with  . If we have a D2D approach, say  which is better than  powerwise, we choose the D2D approach and drop the regular edge. In order to do this, we developed an innovative method as follows.



for all user  :

set  as temporary set for all possible D2D approach to user ,

for all time slot  :

for all user/relay  :

if  and :

if  :

for all time slot  where and  and  :

if  and  :

set 

while  :

if  and  where ,,  , , :



else:



if :



set ,,,

set 

set  and 

set 

else :

break

D2D方法可以解决之前无法涉及的远距离通信问题，但前两步可能不收敛