1.100个计算节点，10/4个不同的服务，每个服务拆分为4个子任务A、B、C、D，其中C和D可进行计算重用。 4个player

1个服务：编解码，转码，渲染，分析

雪山，分析人: A,B,C1,D11

雪山，分析建筑:A,B,C1,D12

沙漠，分析人: A,B,C2,D21

沙漠，分析建筑:A,B,C2,D22

节点n 缓存了D21

s(t)={1,2,3,4}

s{t}=3 player4 选择

s{t+1}=4 player4 选择

……

A,B,C5,D52

15种缓存，每种缓存放在2个节点上。30节点有缓存。

2.为每个节点设置带宽和计算能力。正态分布，中心值为5GHz，范围为[1,10]

3.设置缓存情况。尽量在带宽和计算能力强的节点上放置缓存。带宽和计算能力在不同时隙是波动的

4.子任务计算量=50；检索时延=1； 子任务计算量=0.5GHz，检索时延0.05s

传输数据量也相应更改

4.奖励设计。

5.做决策的时候需要考虑 服务

5.维护重用指数

实验图：

· 算法收敛图。记录每一步的收益和累计平均收益。参考https://zhuanlan.zhihu.com/p/164390177

· 改变服务数量（服务数量=4或10）（本文算法，无计算重用的算法，无重用指数的算法，贪婪算法）

1.时延降低对比图

2.计算资源消耗对比图（later）

重用指数性能：检索成功率（later）

1.改变计算节点数量50，100，200.step\_rewad(横坐标：时间，纵坐标：奖励)

2.改变服务数量，4，8，12, step\_rewad (横坐标：时间，纵坐标：奖励)

3.对比算法收敛图，本文算法，无计算重用的算法，无重用指数的算法，贪婪算法(横坐标：时间，纵坐标：奖励)

4.对比算法时延图1（横坐标：节点数量，纵坐标：时延（s）） 取算法收敛后的最优值

5.对比算法时延图2（横坐标：服务数量，纵坐标：时延（s））

6.计算资源消耗量，对比本文方法和无重用的方法（横坐标：时间，纵坐标：计算资源消耗量）