



华中科技大学

HUAZHONG UNIVERSITY OF
SCIENCE AND TECHNOLOGY



Karma: Resource Allocation for Dynamic Demands

课程汇报PPT

汇报学生：邓振瀚 土木与水利工程学院

时间：2023.12.19



目 录

CONTENTS

01

研究动机

02

基于信用的资源分配

03

资源池使用策略

04

Karma效果评估

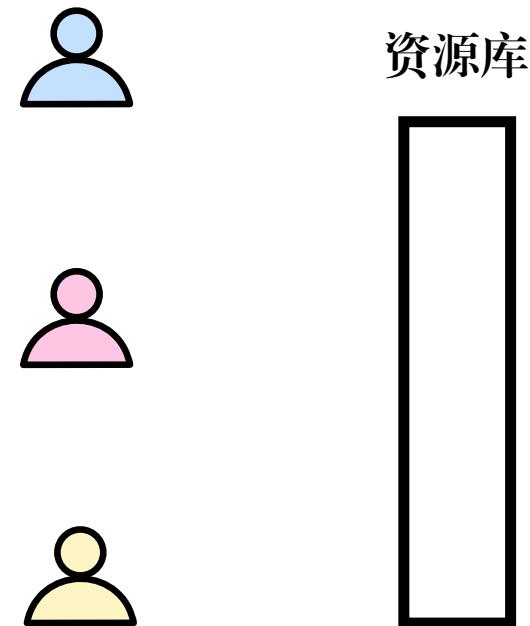


研究动机

Part1

1 研究动机

资源分配是计算机系统中的一个基本问题
涉及私有云和公共云、计算机网络、管理程序等



基于信用的
资源分配

资源池
使用策略

Karma
效果评估



1 研究动机

研究动机

基于信用的
资源分配

资源池
使用策略

Karma
效果评估



资源库



关键的理想属性：



Pareto efficiency: 高资源利用率
帕累托效率



Strategy-proofness: 谎报需求也不能受益
策略验证性



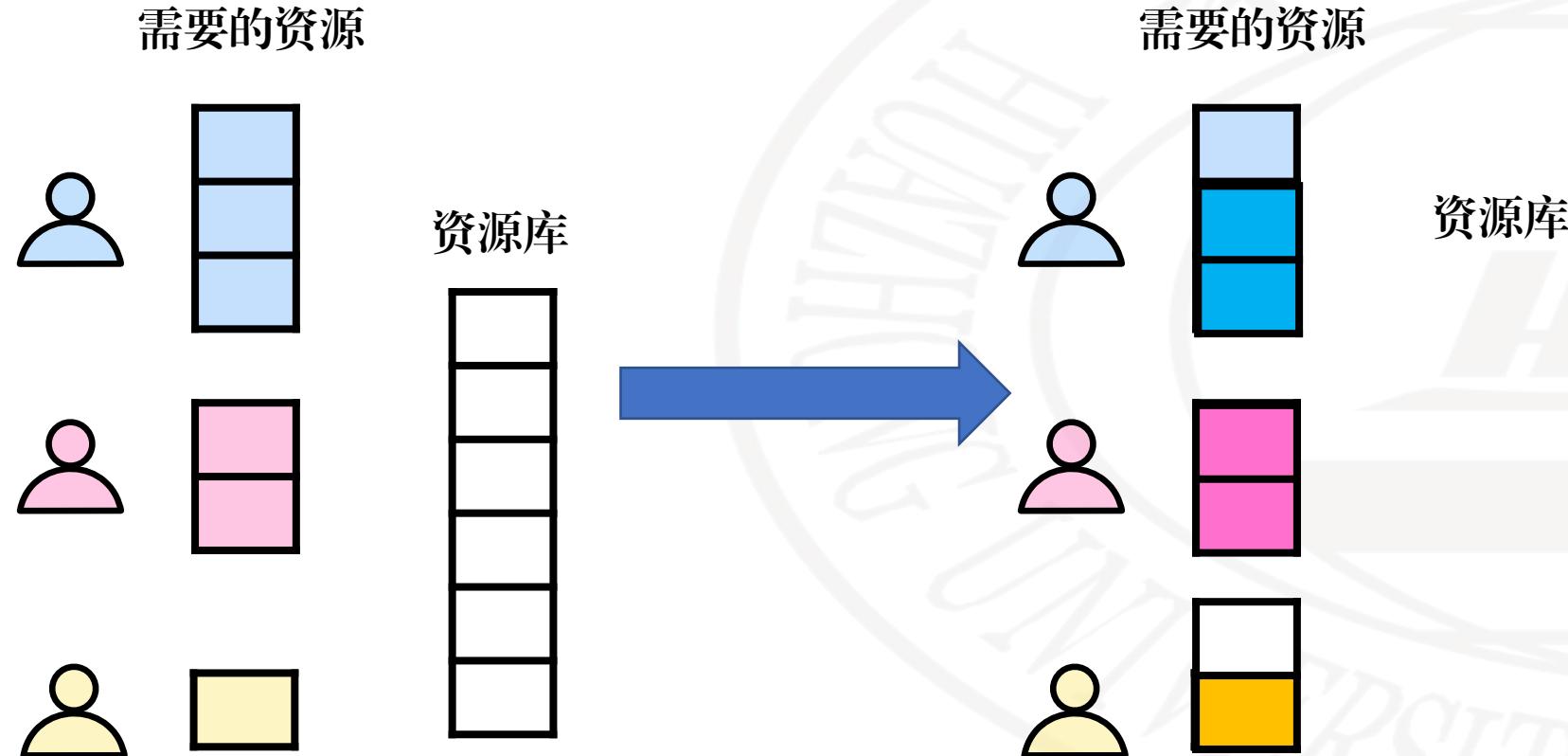
Fairness:
公平性

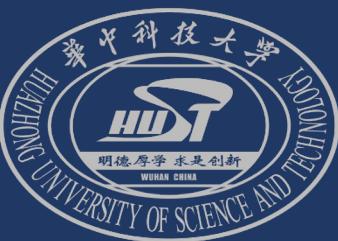


1 研究动机

资源分配现有的经典方法

一、 strict partitioning (严格分区)



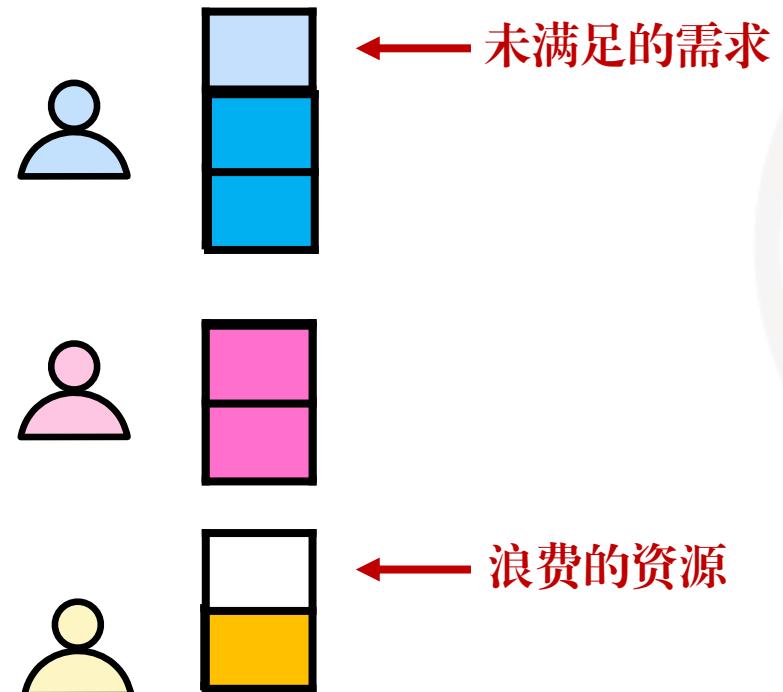


1 研究动机

资源分配现有的经典方法

一、 strict partitioning (严格分区)

需要的资源



关键的理想属性:



Pareto efficiency

帕累托效率



Strategy-proofness

策略验证性

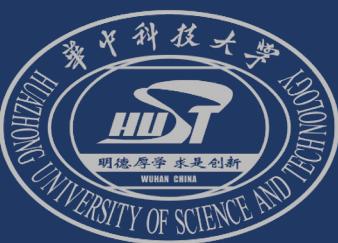


Fairness

公平性



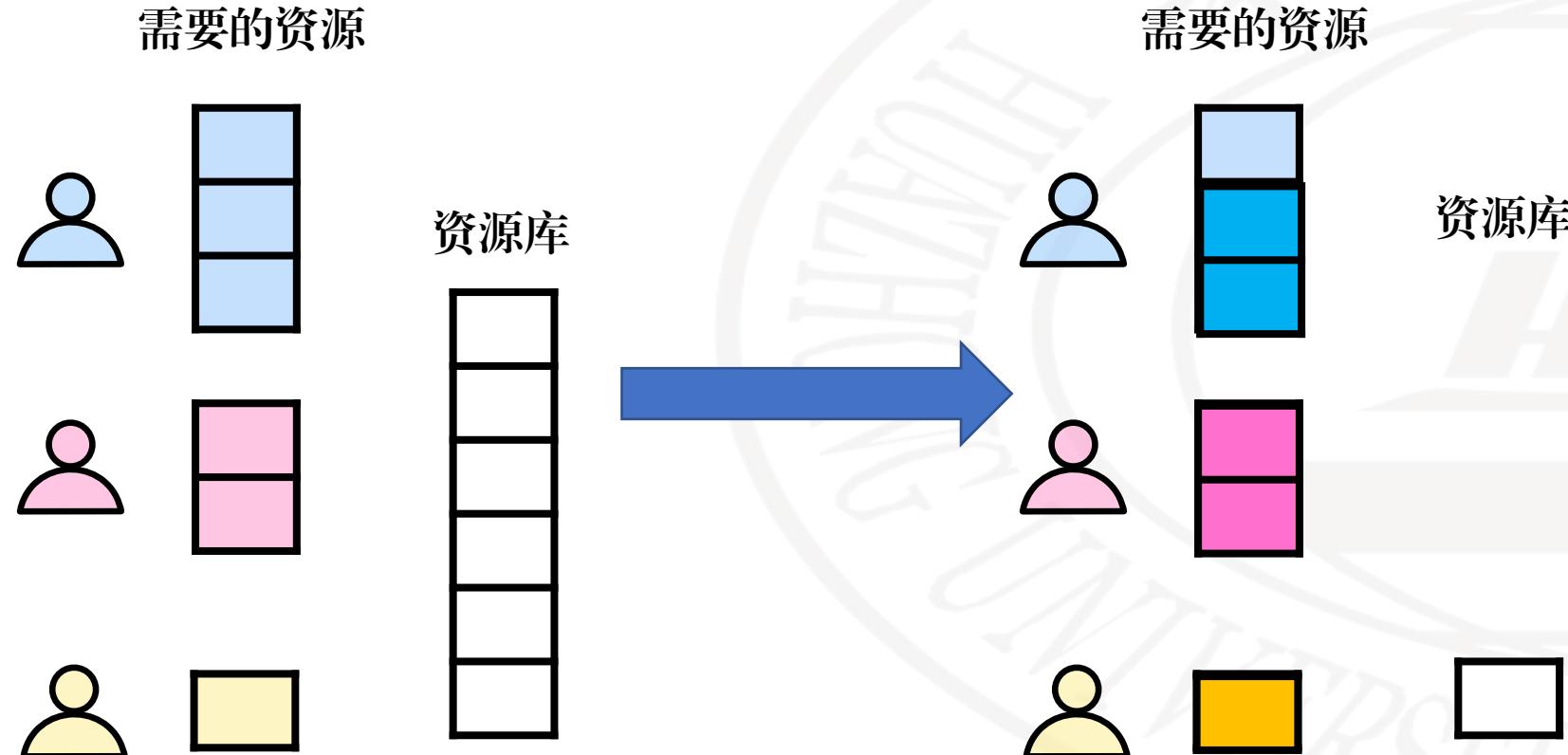
不能保证Pareto efficiency(帕累托效率)

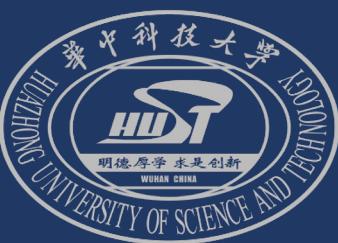


1 研究动机

资源分配现有的经典方法

二、 max-min fairness (最大最小公平性)



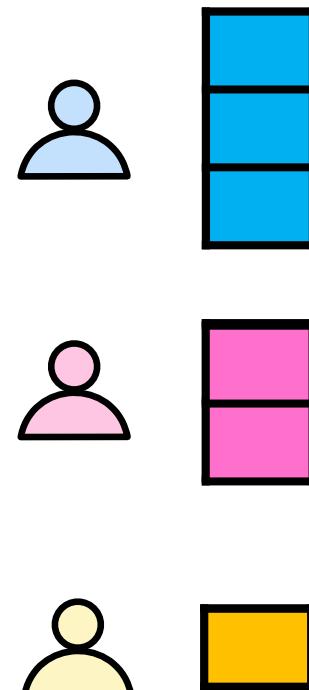


1 研究动机

资源分配现有的经典方法

二、 max-min fairness (最大最小公平性)

需要的资源



资源库

关键的理想属性:



Pareto efficiency

帕累托效率



Strategy-proofness

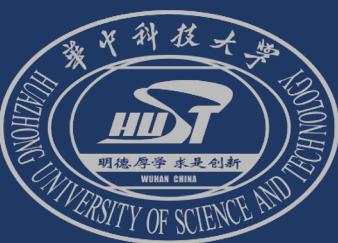
策略验证性



Fairness

公平性



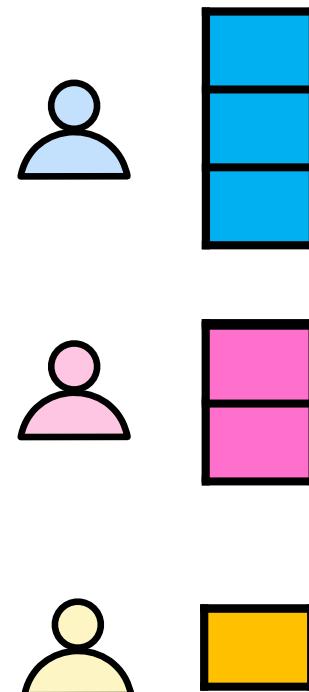


1 研究动机

资源分配现有的经典方法

二、 max-min fairness (最大最小公平性)

需要的资源



资源库

关键的理想属性:



Pareto efficiency

帕累托效率



Strategy-proofness

策略验证性



Fairness

公平性



基本假设: 用户的需求是静止的



1 研究动机

实际：用户的需求是动态变化的

在动态中应用max-min fairness会出现的问题

(1) 在 $t=0$ 时基于需求进行分配



Pareto efficiency

帕累托效率



Strategy-proofness

策略验证性



Fairness

公平性

(2) 周期性地进行分配



Pareto efficiency

帕累托效率



Strategy-proofness

策略验证性



Fairness

公平性



提出了一种新的动态用户需求
资源分配机制Karma



1 研究动机

实际：用户的需求是动态变化的

在动态中应用max-min fairness会出现的问题

(1) 在 $t=0$ 时基于需求进行分配



Pareto efficiency

帕累托效率



Strategy-proofness

策略验证性



Fairness

公平性

(2) 周期性地进行分配



Pareto efficiency

帕累托效率



Strategy-proofness

策略验证性



Fairness

公平性



提出了一种新的动态用户需求
资源分配机制Karma



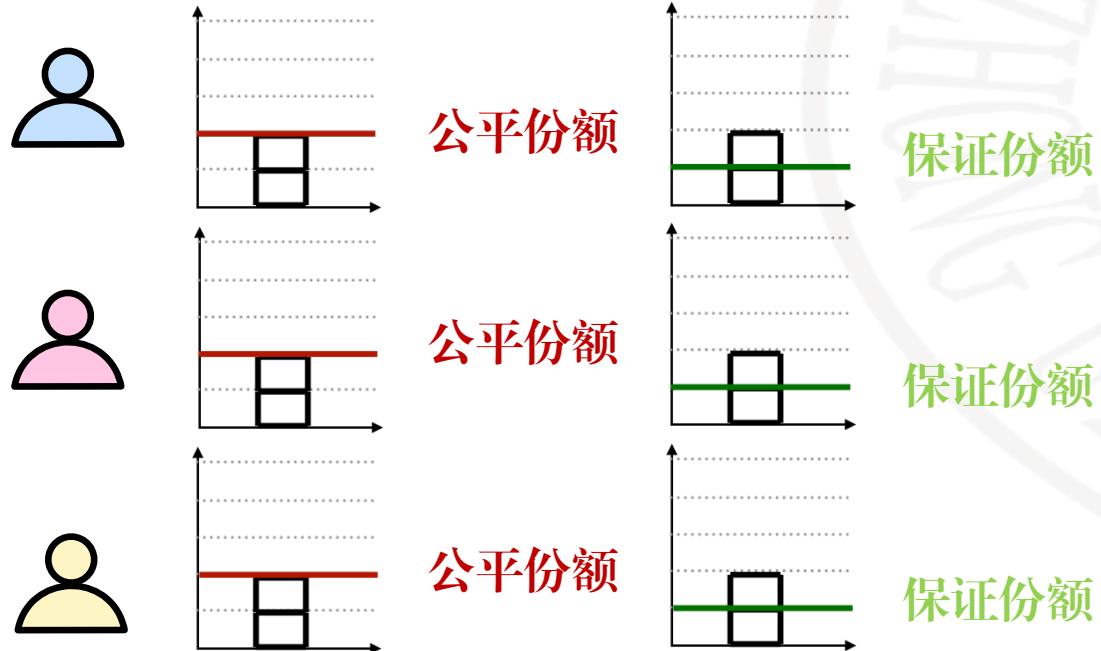
基于信用的资源分配

Part2

2 基于信用的资源分配

Karma的关键技术贡献是基于信用的资源分配算法：
在每个量子中，当用户捐赠其公平份额的一部分资源(例如，
如果他们的需求)时，他们会获得信用

Karma关键理念：捐赠切片和共享切片



共享：公平份额与保证份额之间的差额

捐赠：用户的需求低于其保证份额

捐赠切片：保证份额与需求之间的差额

**借用：用户的需求大于保证份额，需求
大于保证份额之外的部分，从系统的共
享或捐赠部分提供**

2 基于信用的资源分配

研究动机

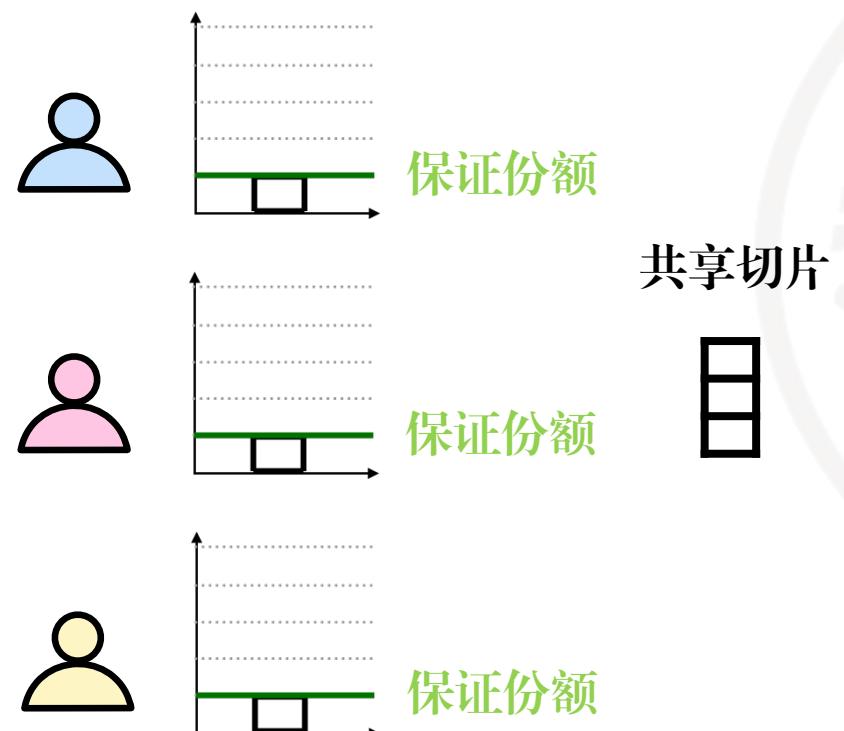
基于信用的 资源分配

资源池 使用策略

Karma 效果评估



Karma关键理念：捐赠切片和共享切片



共享：公平份额与保证份额之间的差额

捐赠：用户的需求低于其保证份额

捐赠量：保证份额与需求之间的差额

借用：需求大于保证份额

借用量：需求大于保证份额之外的部分，
从系统的共享或捐赠部分提供

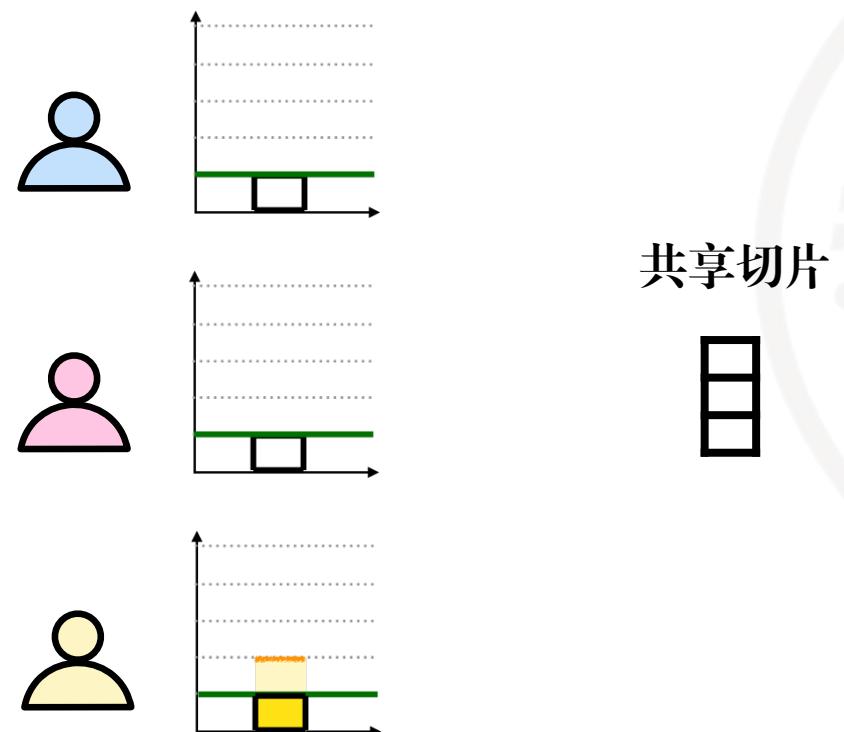
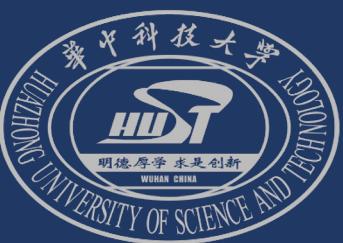
2 基于信用的资源分配

研究动机

基于信用的 资源分配

资源池 使用策略

Karma 效果评估



共享: 公平份额与保证份额之间的差额

捐赠: 用户的需求低于其保证份额

捐赠量: 保证份额与需求之间的差额

借用: 需求大于保证份额

借用量: 需求大于保证份额之外的部分，
从系统的共享或捐赠部分提供

2 基于信用的资源分配

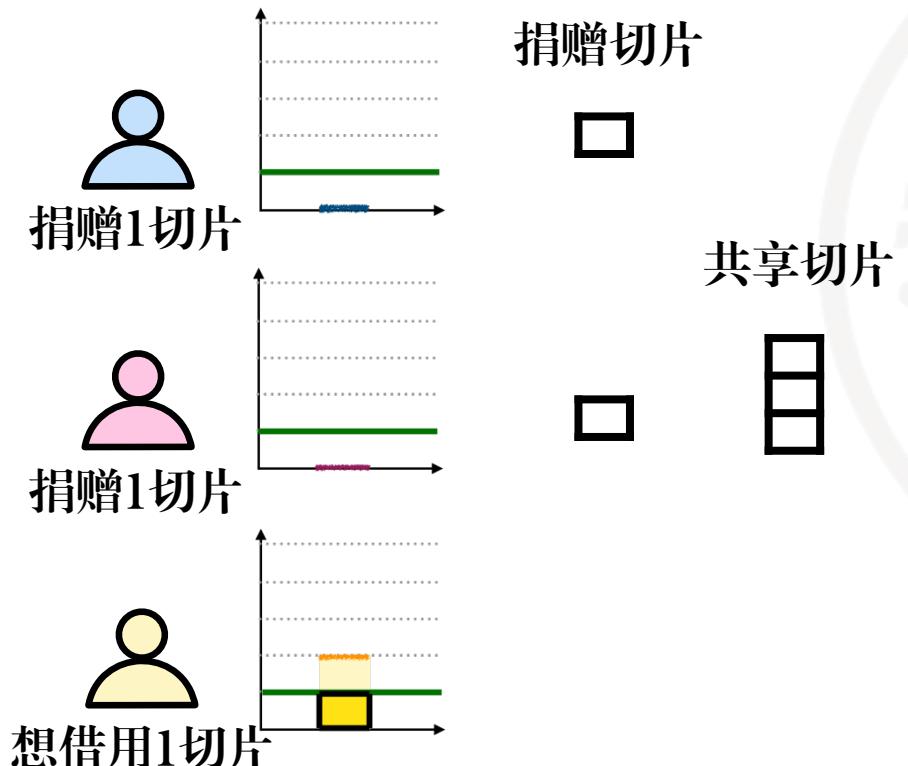
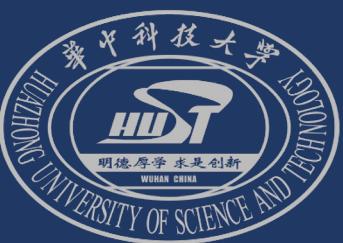
Karma关键理念：捐赠切片和共享切片

研究动机

基于信用的
资源分配

资源池
使用策略

Karma
效果评估



共享：公平份额与保证份额之间的差额

捐赠：用户的需求低于其保证份额

捐赠量：保证份额与需求之间的差额

借用：需求大于保证份额

借用量：需求大于保证份额之外的部分，
从系统的共享或捐赠部分提供

2 基于信用的资源分配

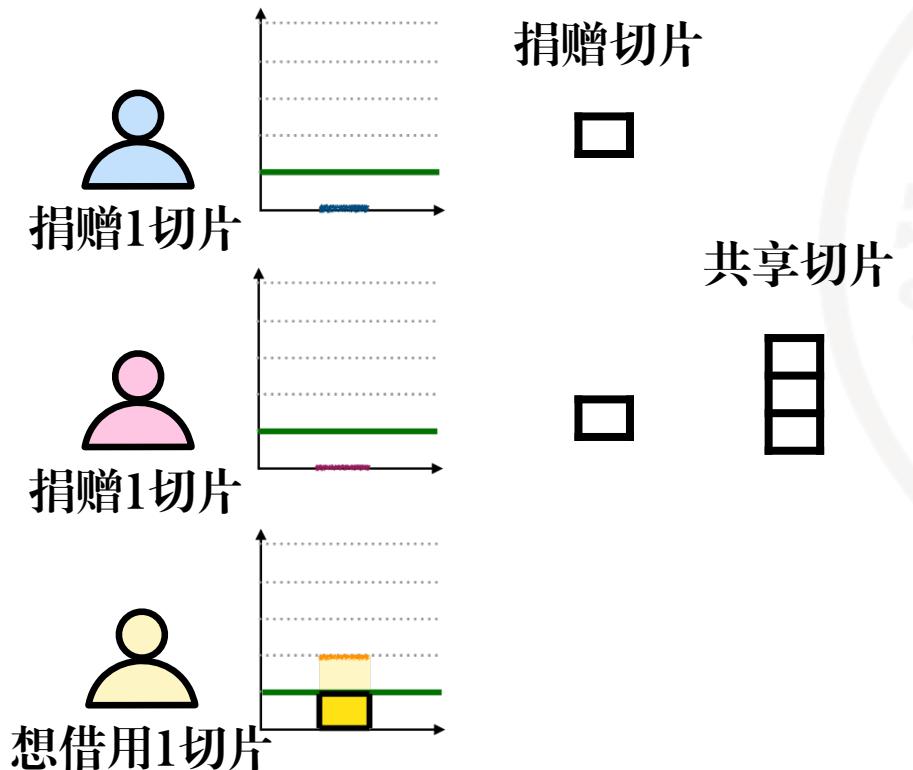
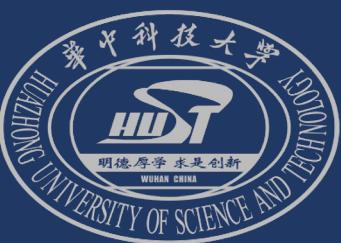
Karma关键理念：捐赠切片和共享切片

研究动机

基于信用的
资源分配

资源池
使用策略

Karma
效果评估



对于捐赠切片的：

每捐赠1切片+1信用积分

对于借用切片的：

每借用1切片-1信用积分

2 基于信用的资源分配

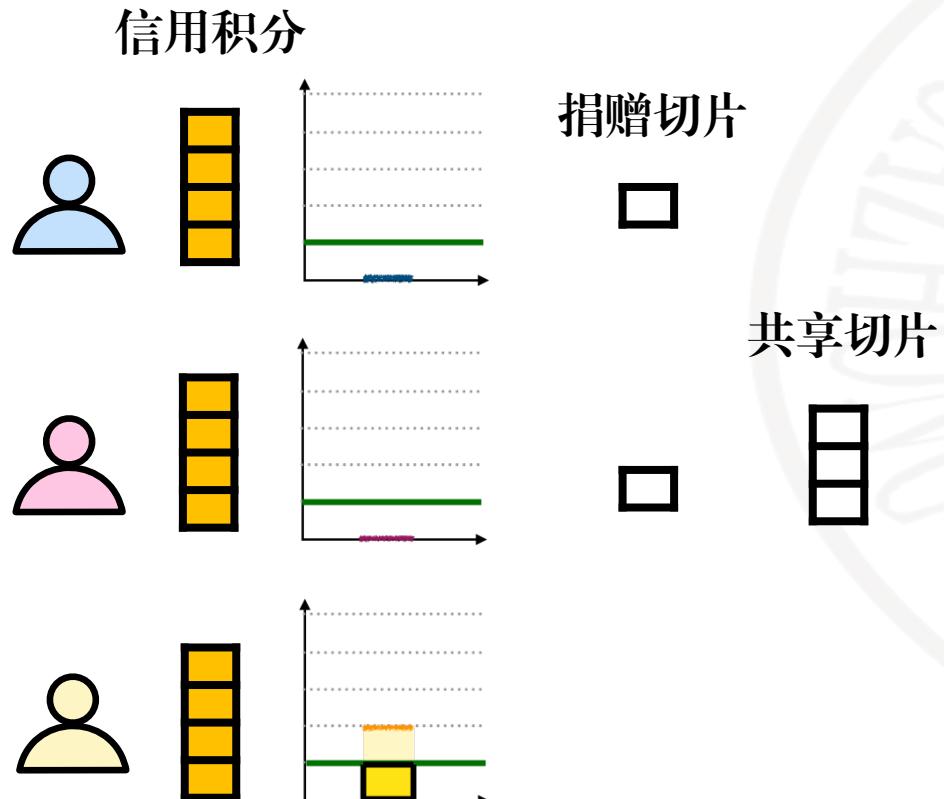
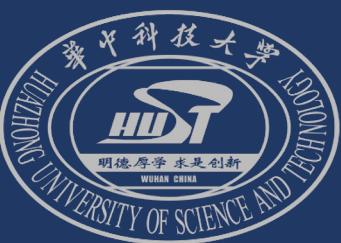
Karma关键理念：捐赠切片和共享切片

研究动机

基于信用的
资源分配

资源池
使用策略

Karma
效果评估



对于捐赠切片的：

每捐赠1切片+1信用积分

对于借用切片的：

每借用1切片-1信用积分

每个用户在 $t=0$ 时均被赋予初始信用积分

2 基于信用的资源分配

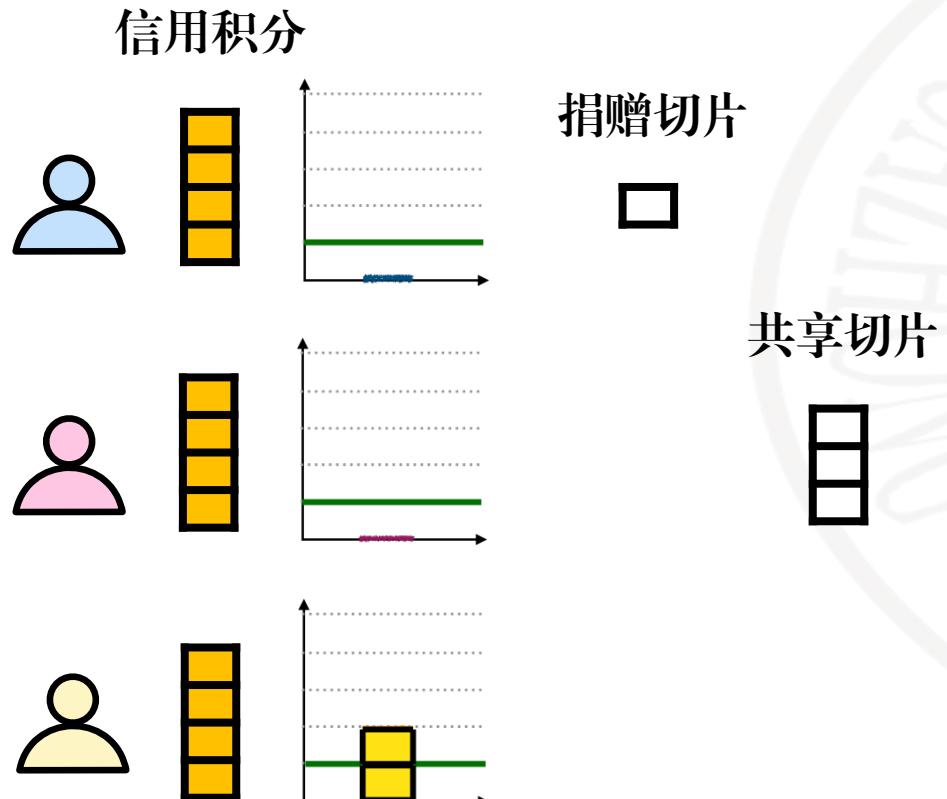
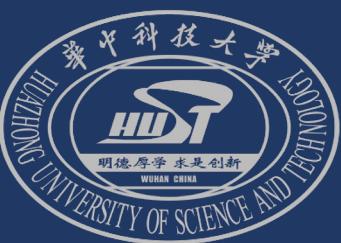
Karma关键理念：捐赠切片和共享切片

研究动机

基于信用的
资源分配

资源池
使用策略

Karma
效果评估



对于捐赠切片的：

每捐赠1切片+1信用积分

对于借用切片的：

每借用1切片-1信用积分

每个用户在 $t=0$ 时均被赋予初始信用积分

2 基于信用的资源分配

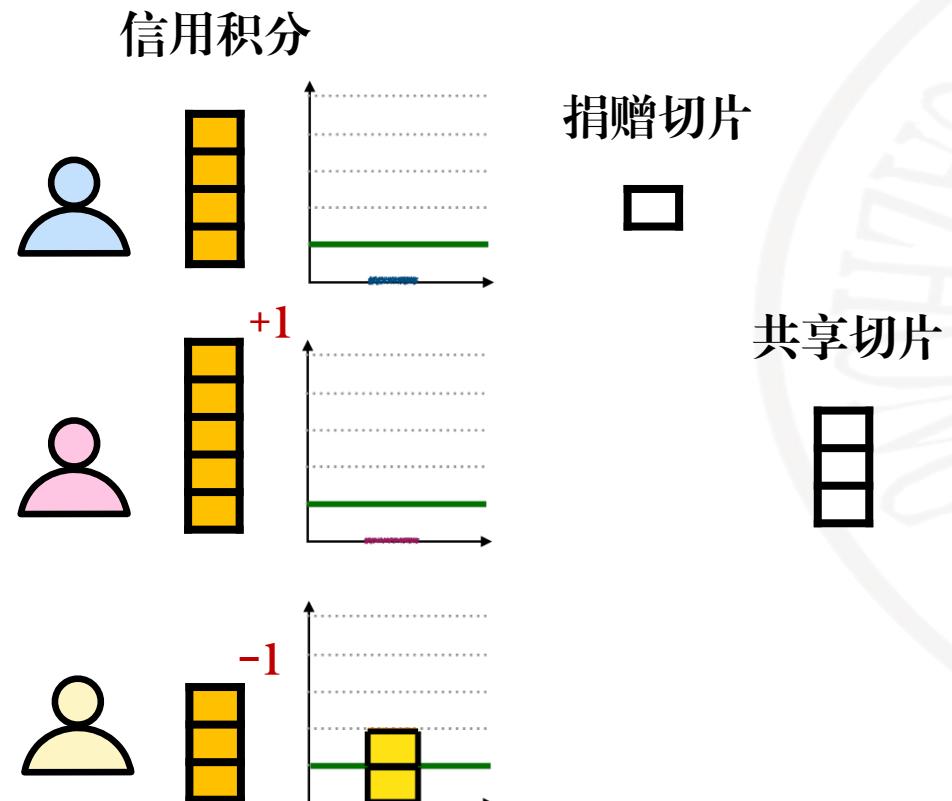
Karma关键理念：捐赠切片和共享切片

研究动机

基于信用的
资源分配

资源池
使用策略

Karma
效果评估



对于捐赠切片的：

每捐赠1切片+1信用积分

对于借用切片的：

每借用1切片-1信用积分

每个用户在t=0时均被赋予初始信用积分

2 基于信用的资源分配

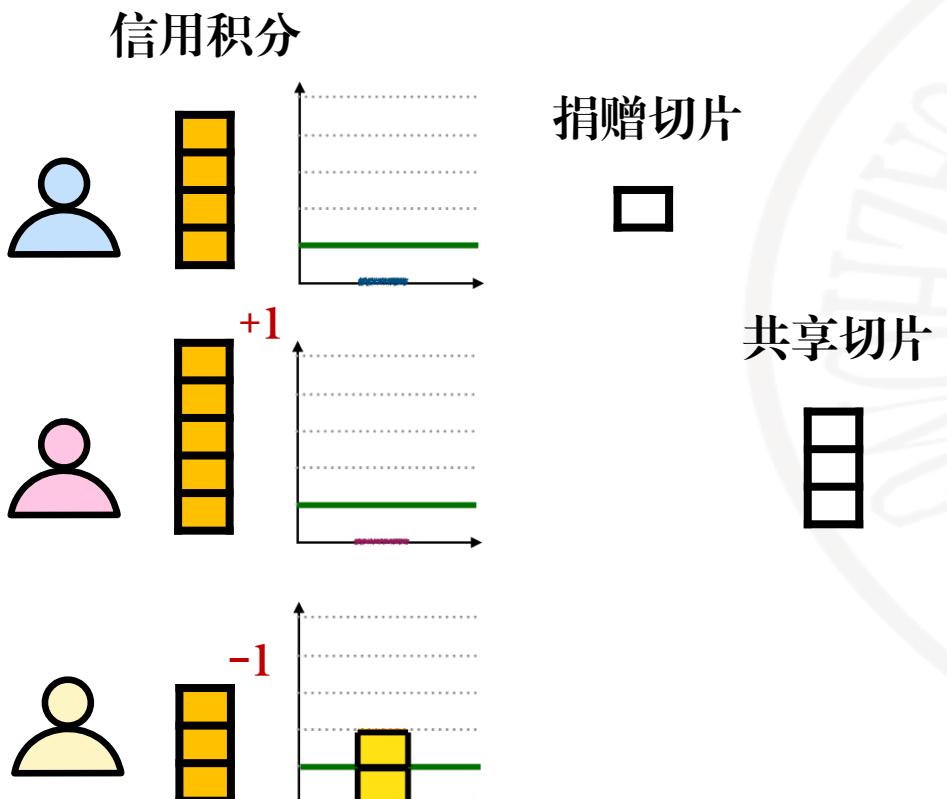
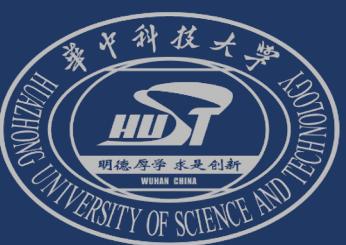
Karma关键理念：捐赠切片和共享切片

研究动机

基于信用的 资源分配

资源池 使用策略

Karma 效果评估



**如果用户的实际需求少于分配的资源，
用户程序主动按片为粒度贡献资源。当这
些资源片被分配给新的用户程序时，贡献
者获得信用点；**

**如果用户的实际需求超过分配的资源，
用户程序消耗信用点借用共享池或者贡献
池中的资源。如果没有信用点，用户程序
不能借用资源。**

2 基于信用的资源分配

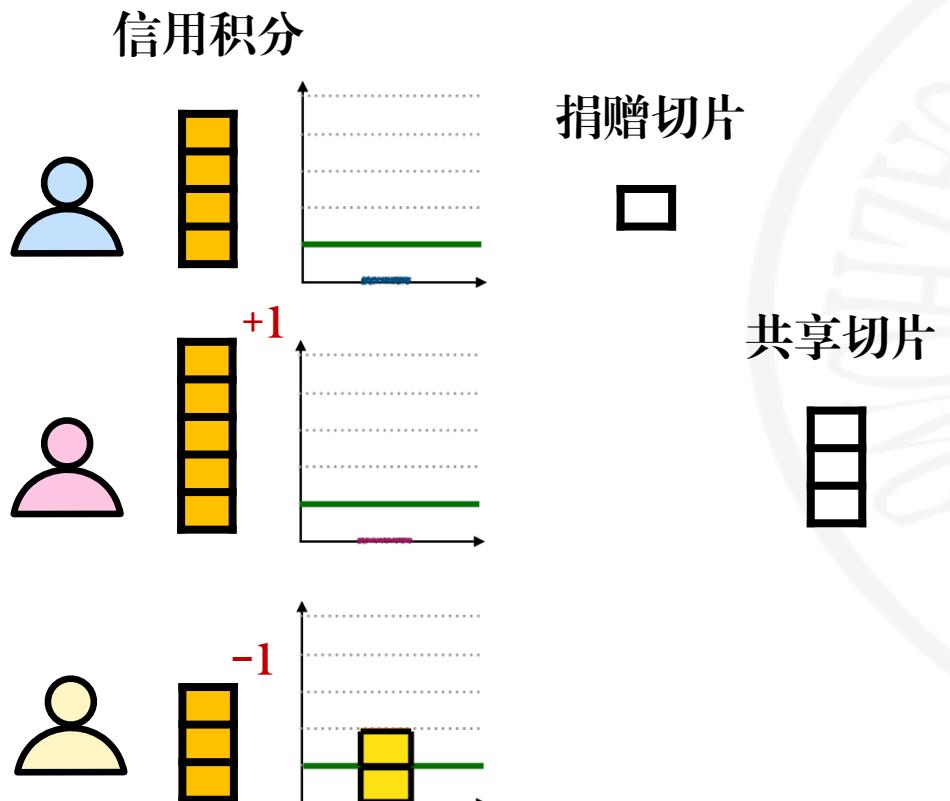
Karma关键理念：捐赠切片和共享切片

研究动机

基于信用的
资源分配

资源池
使用策略

Karma
效果评估

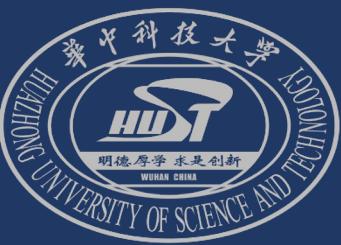


通过这套信用机制，Karma激励用户
程序主动贡献多余资源，确保公平分配，
也能通过扣除信用点限制用户无限制申请
资源。



资源池使用策略

Part3

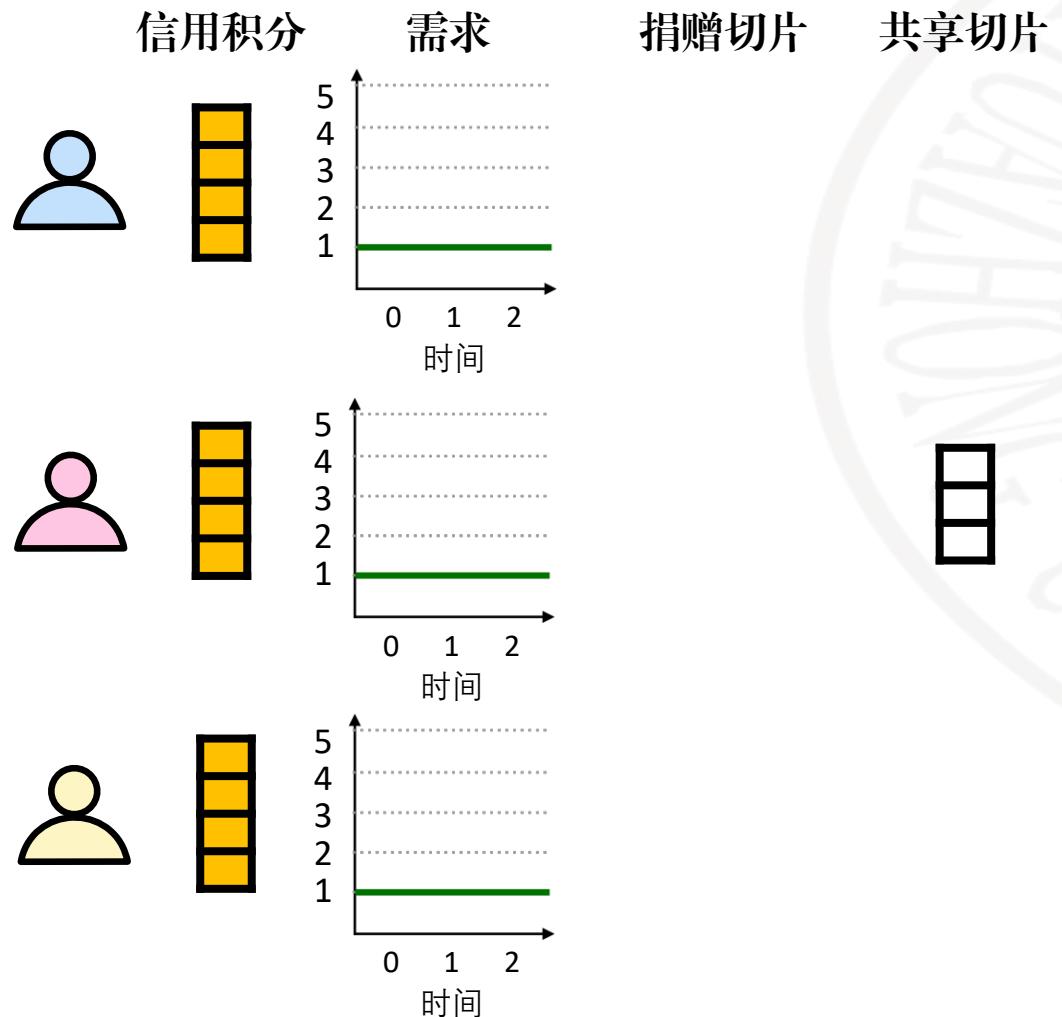


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

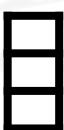
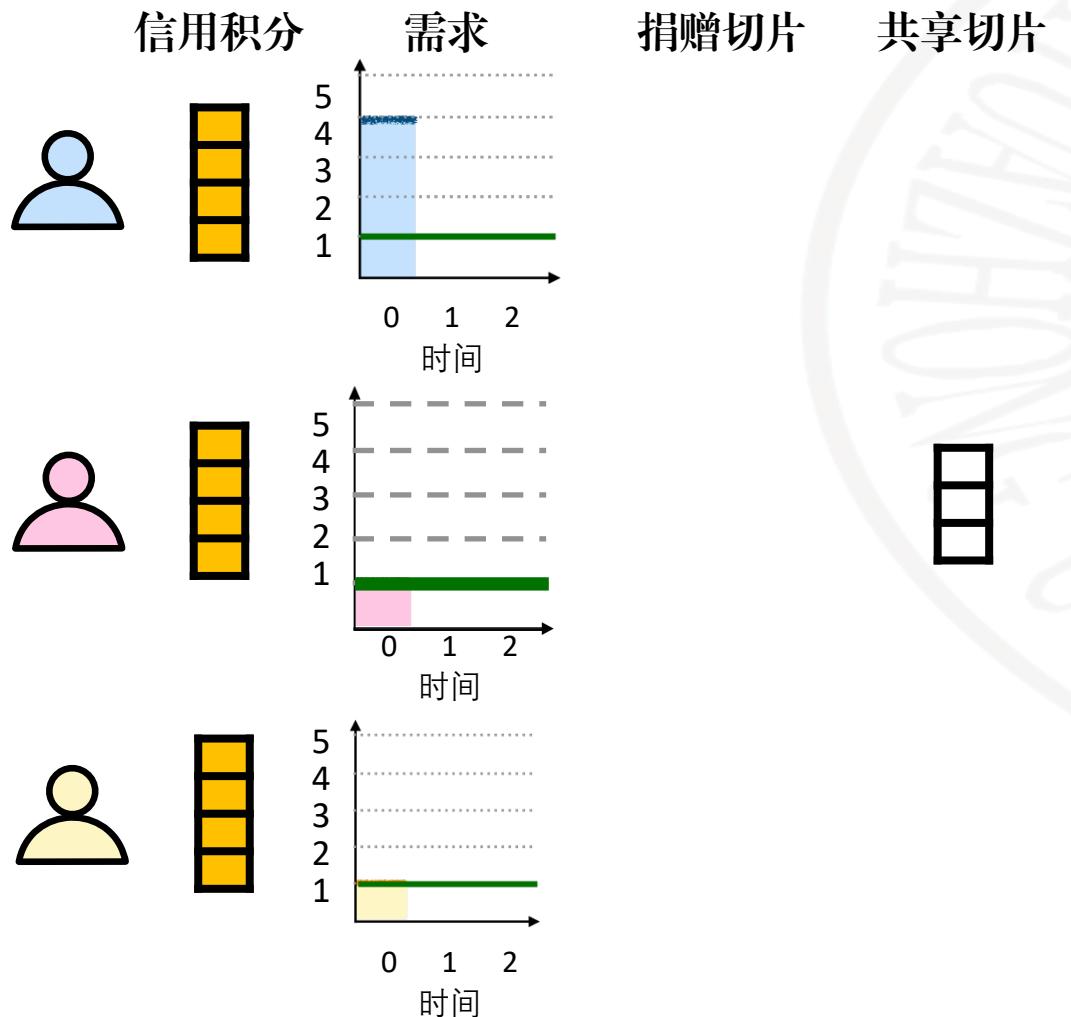


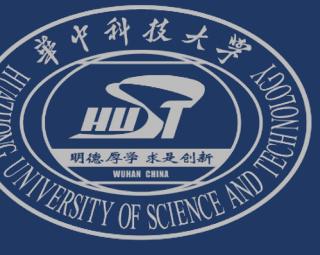
3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1





研究动机

基于信用的
资源分配

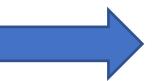
资源池
使用策略

Karma
效果评估

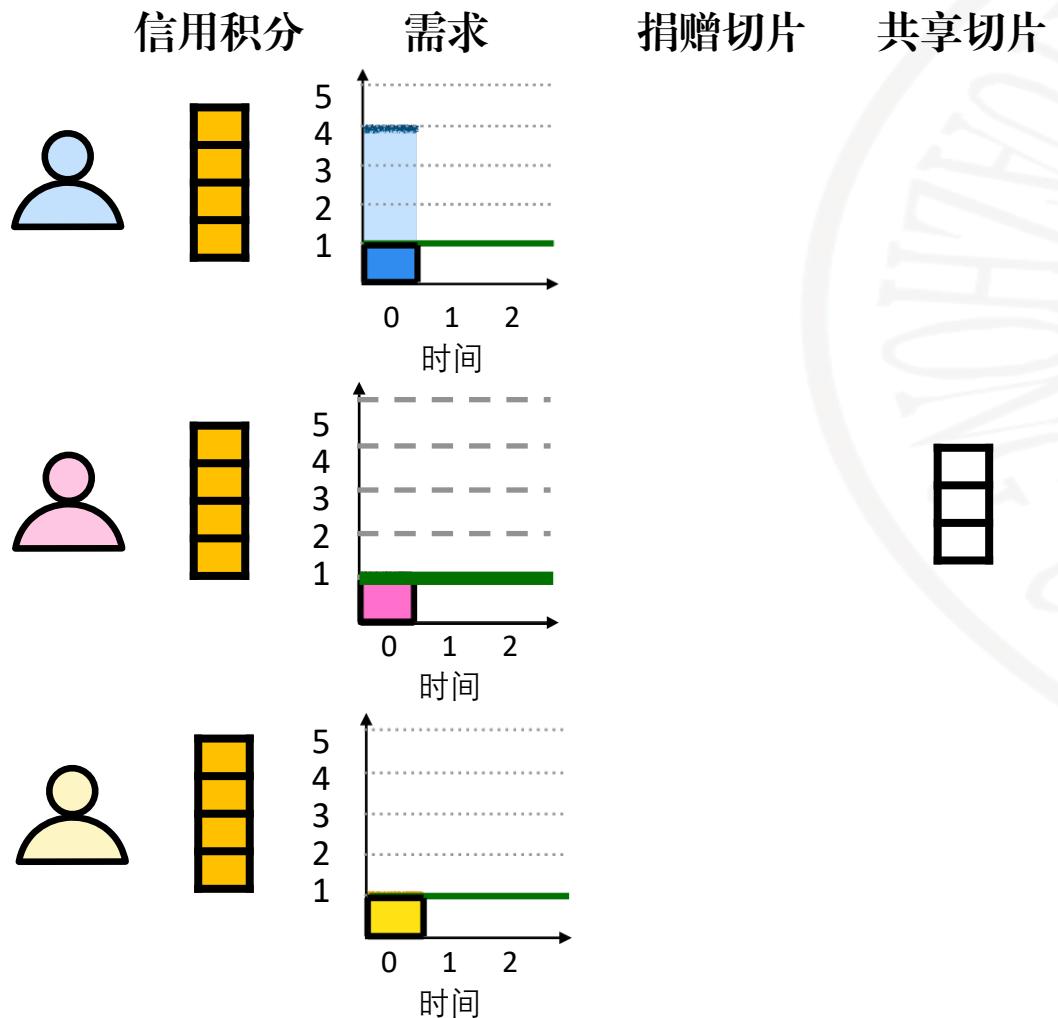
三

3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

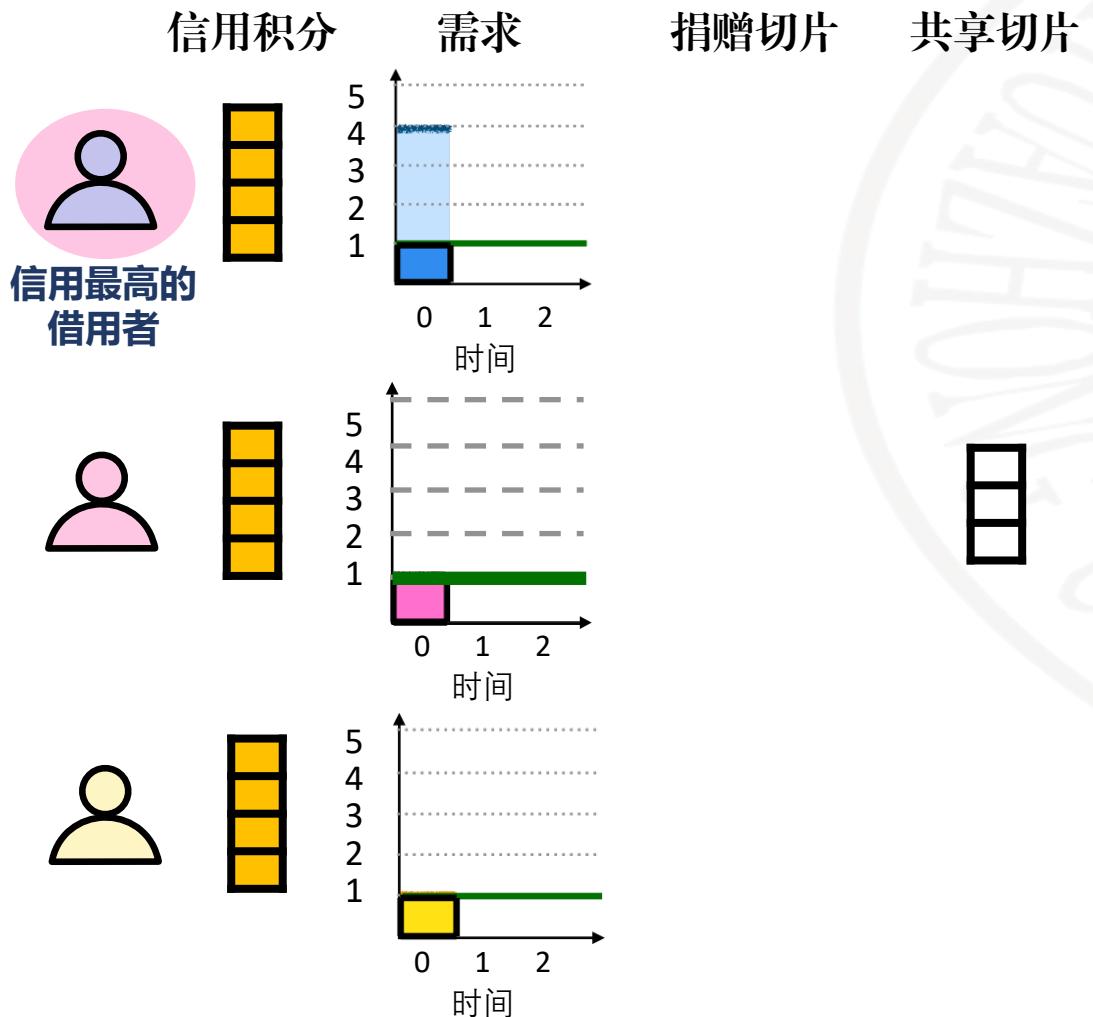


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



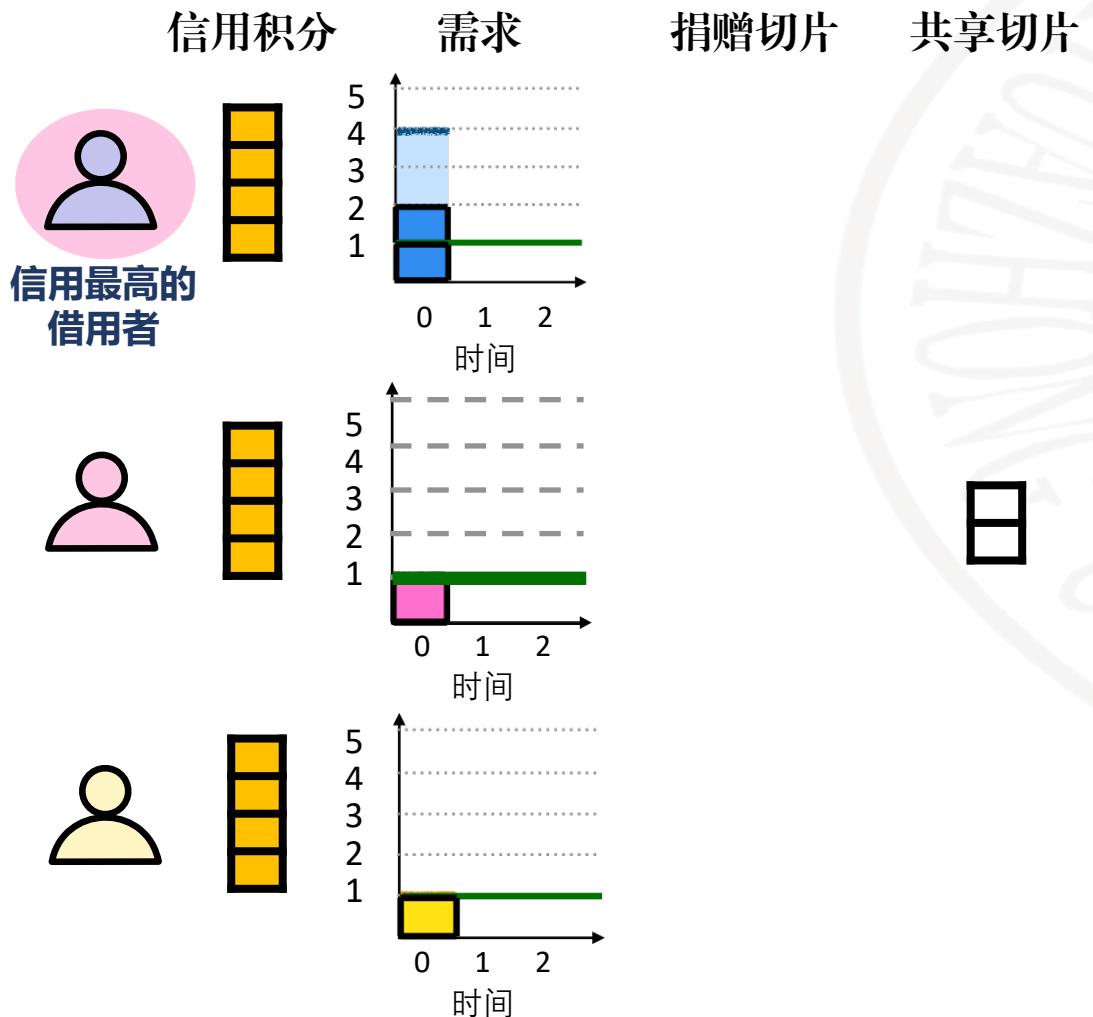


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者

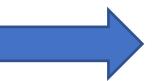


将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

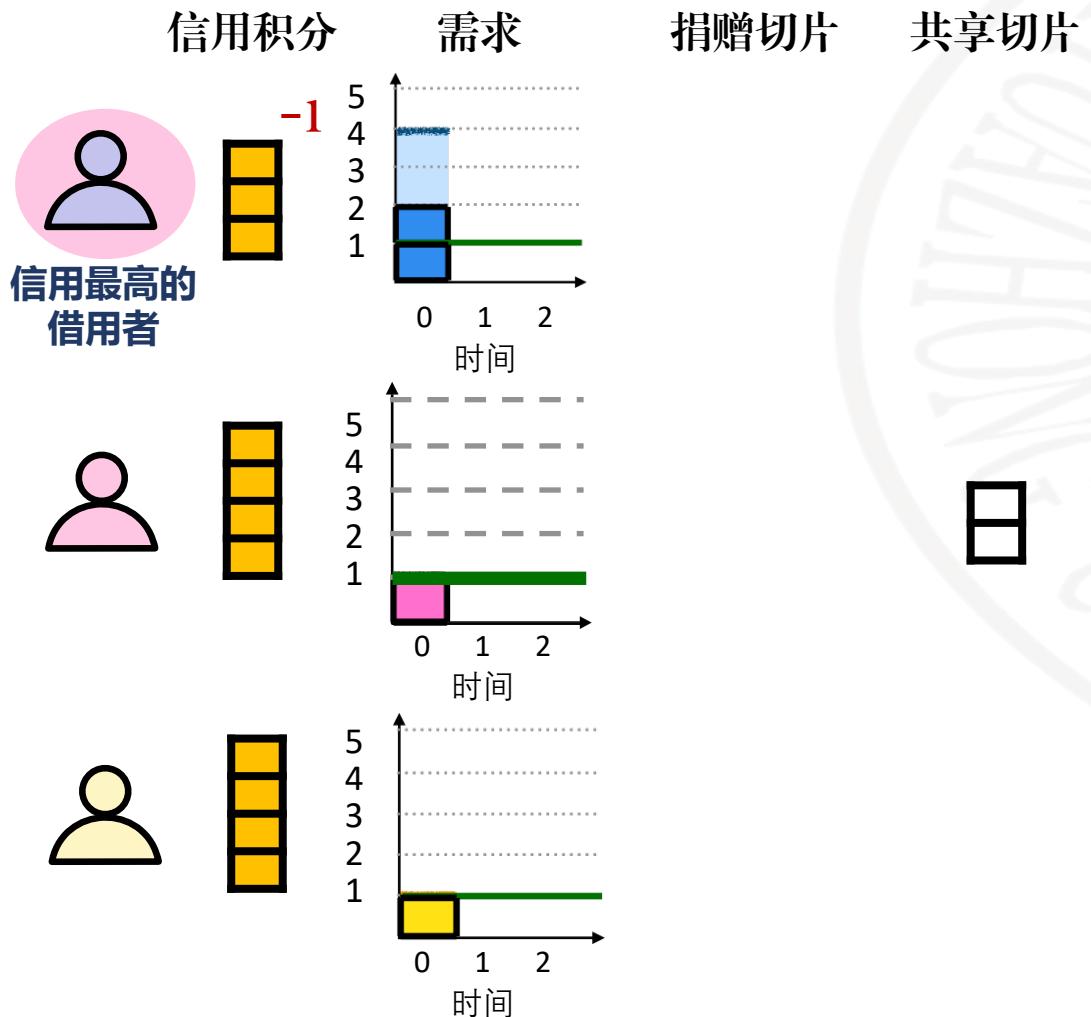


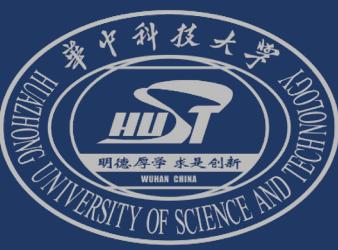
3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



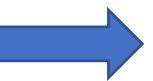
将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



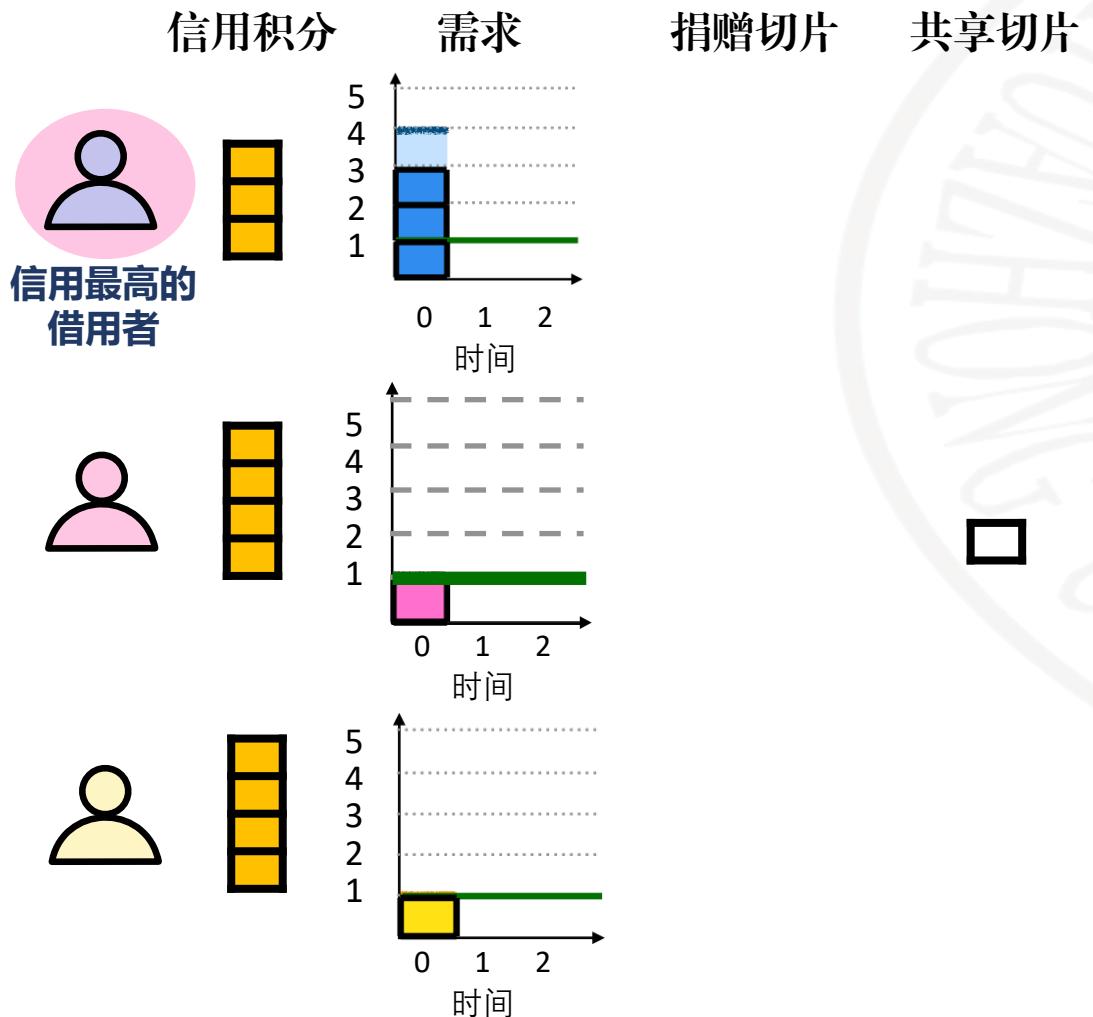


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

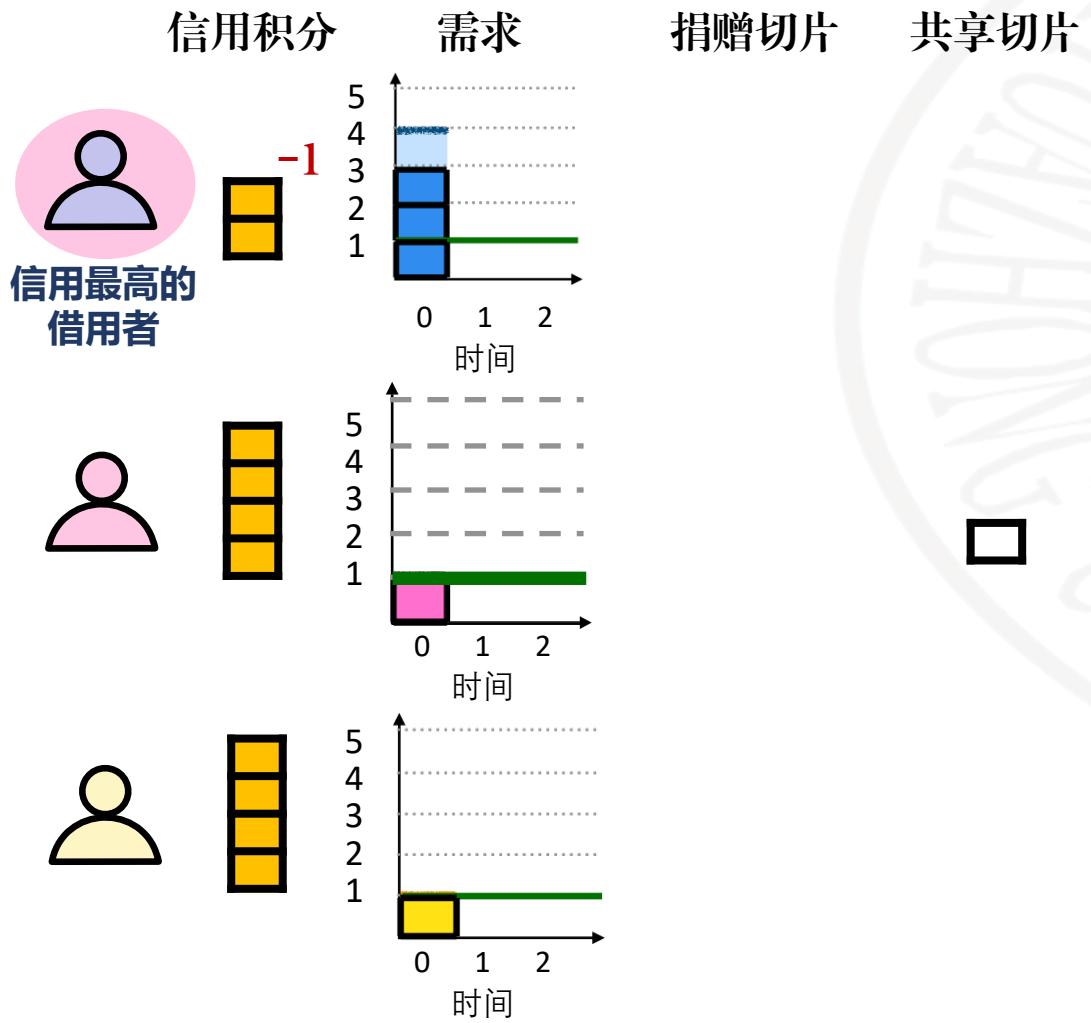


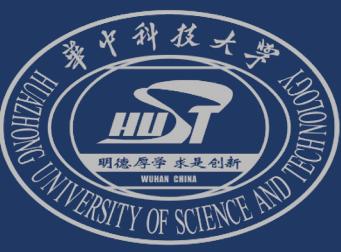
3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



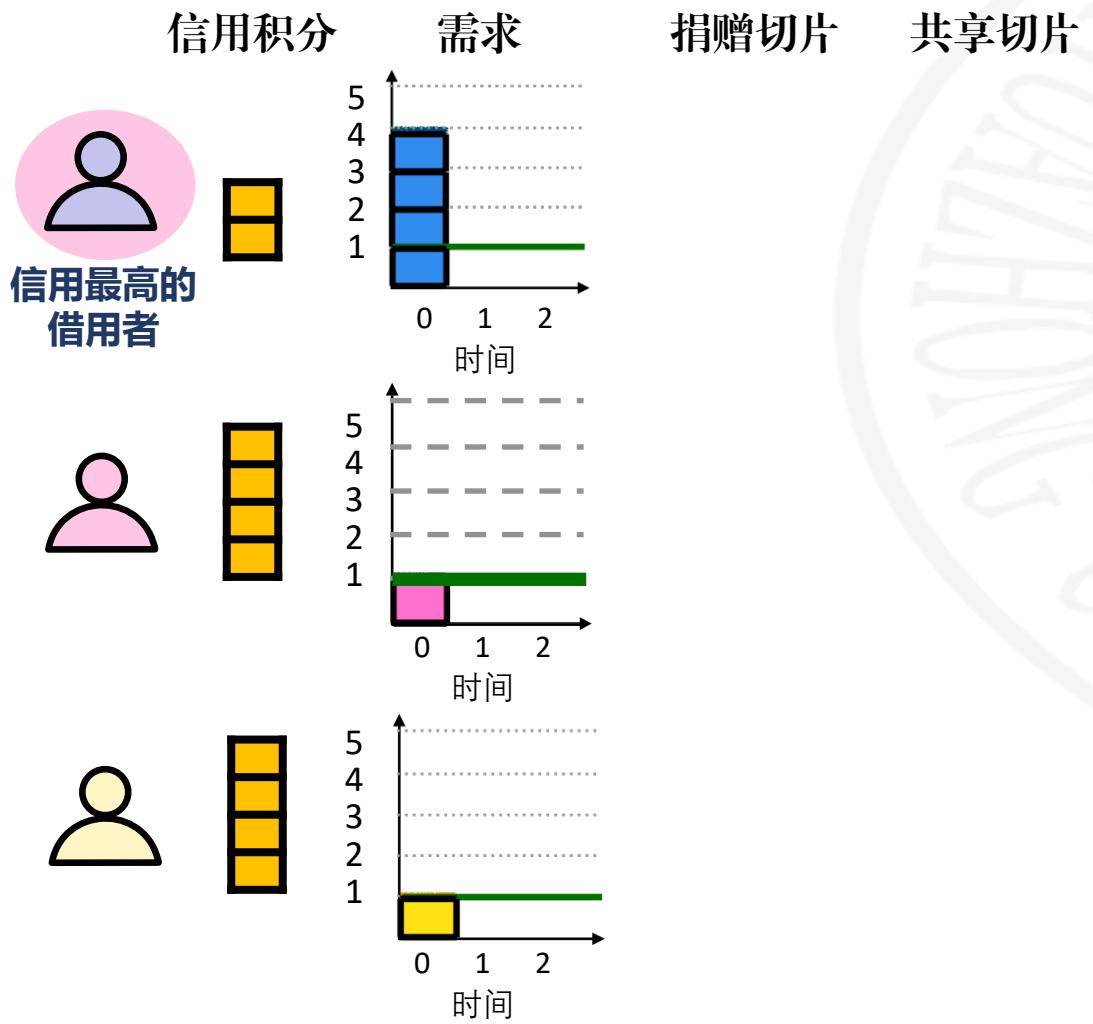


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



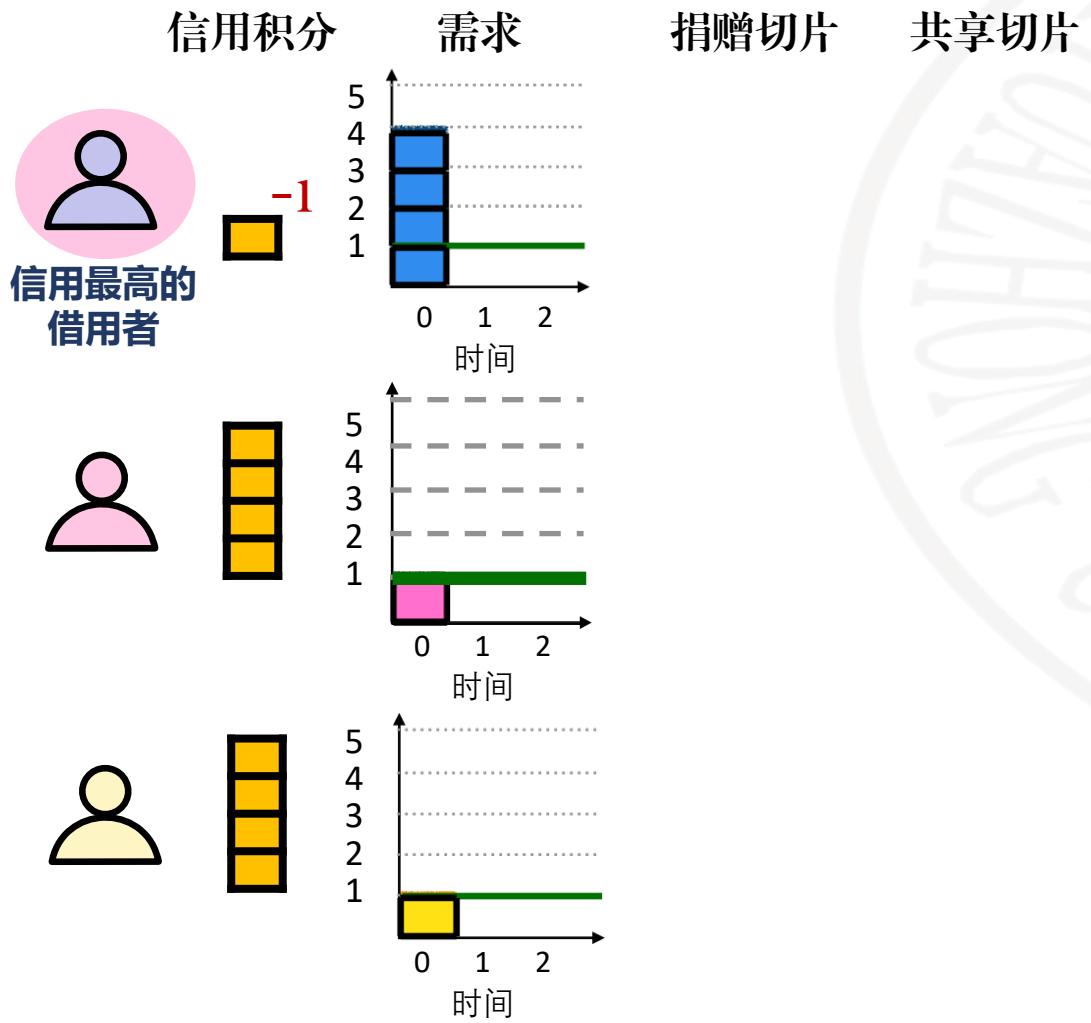


3 资源池使用策略

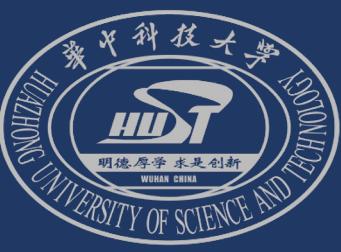
- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



研究动机

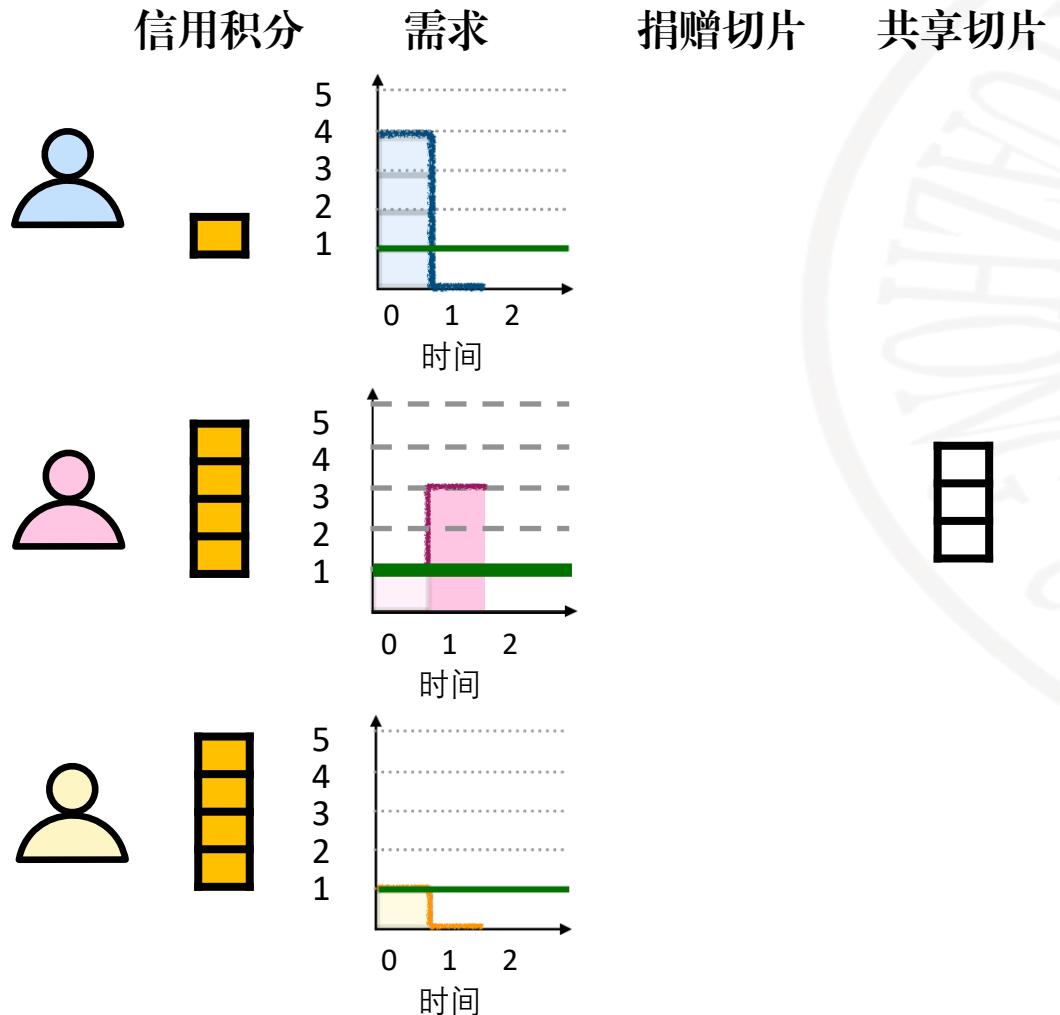
基于信用的
资源分配资源池
使用策略Karma
效果评估

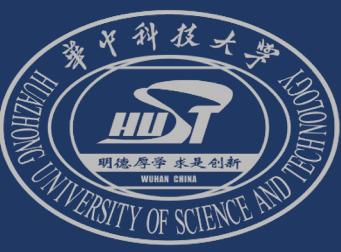
3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



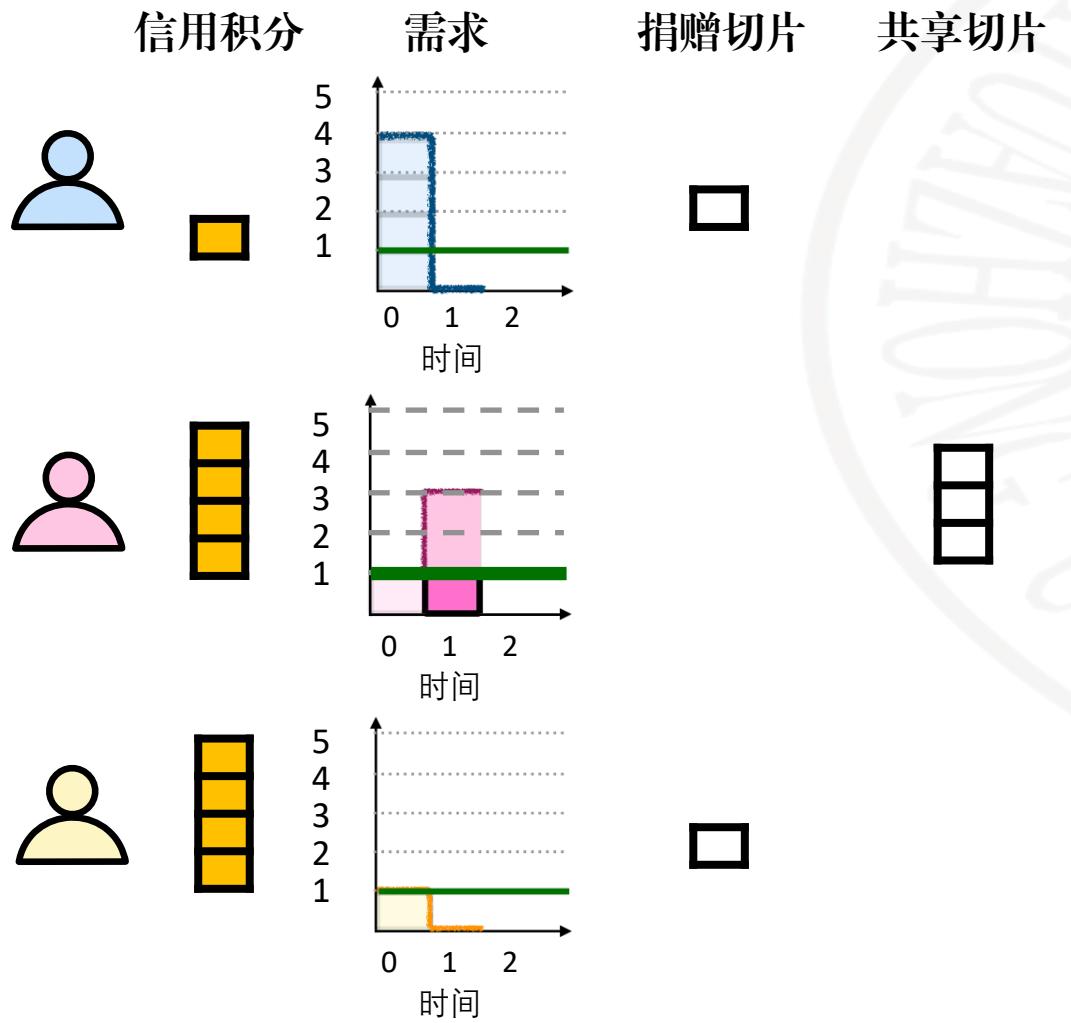


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

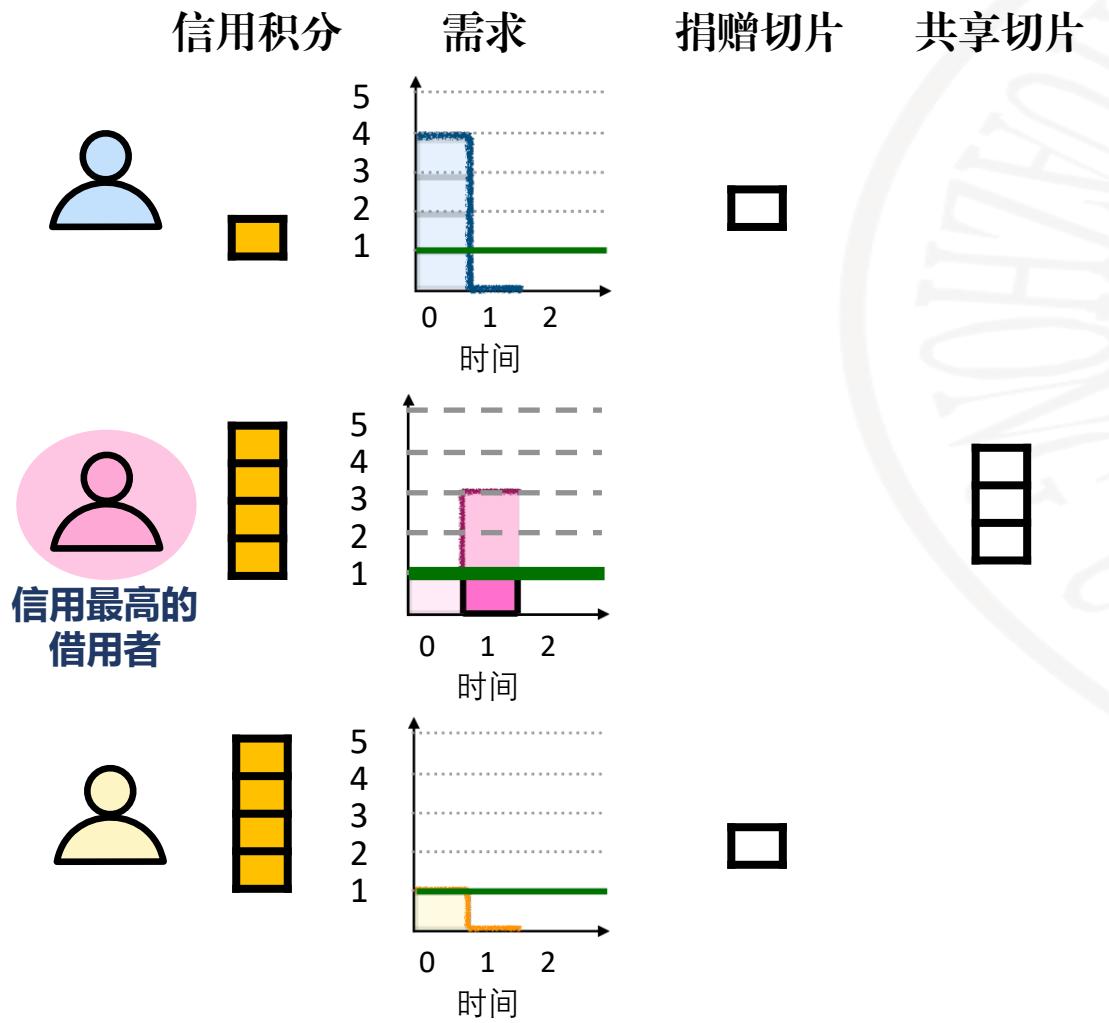


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

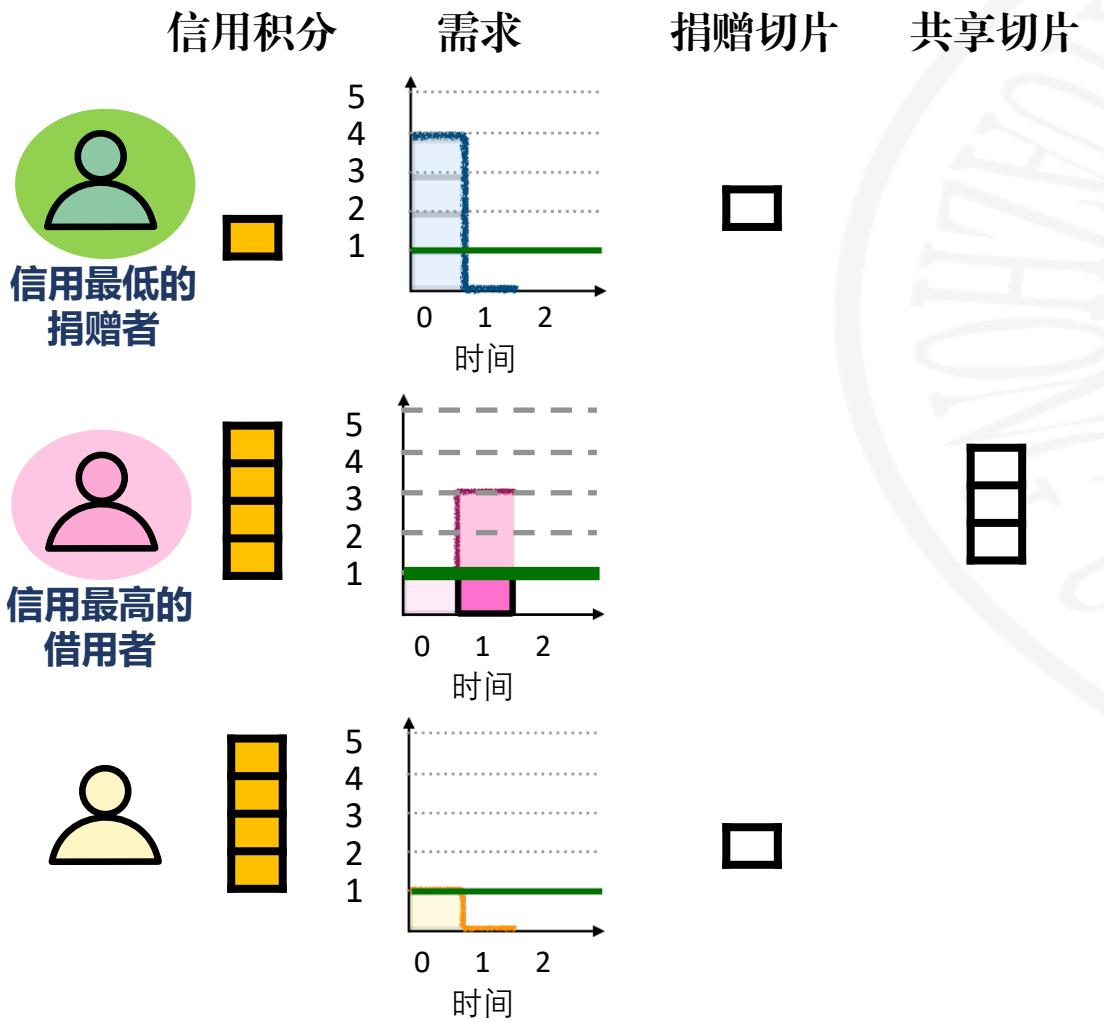


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

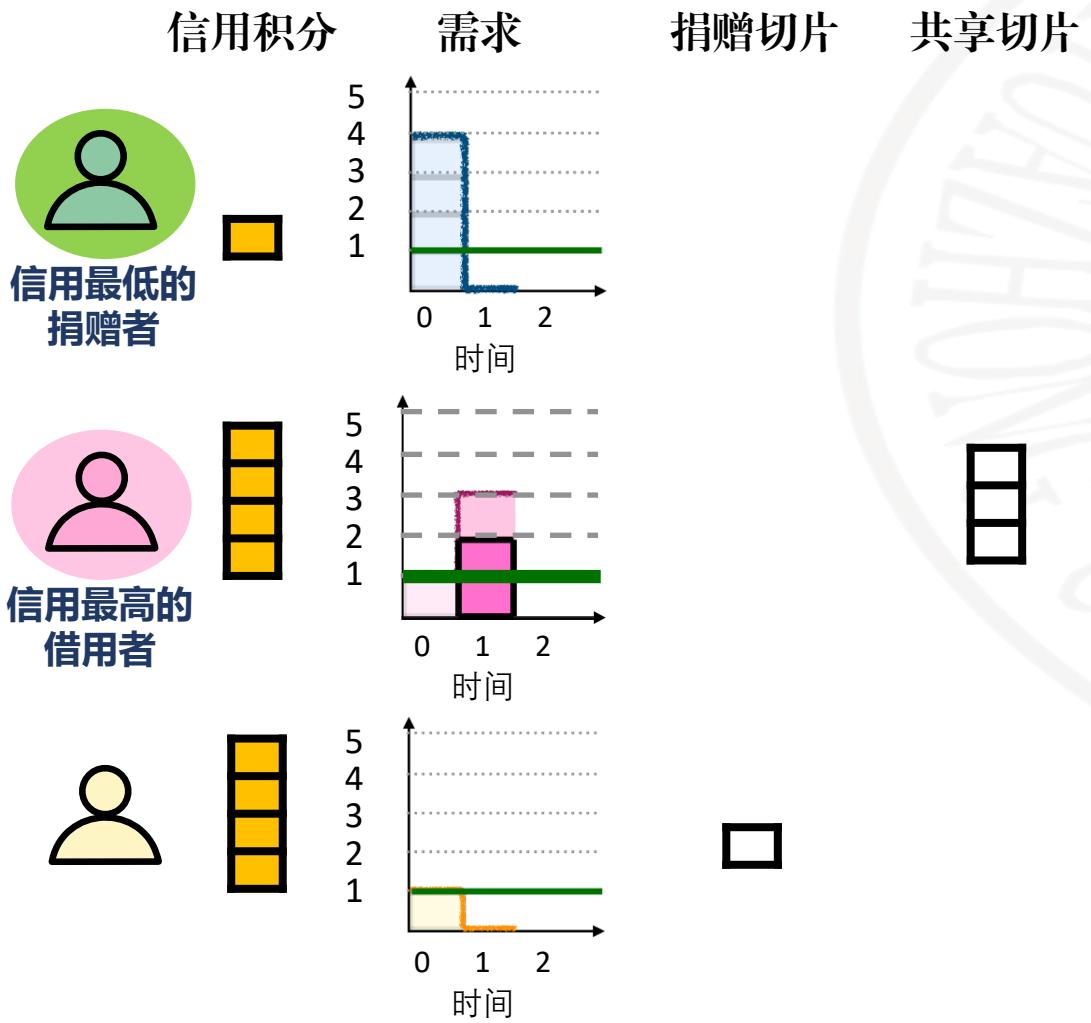


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



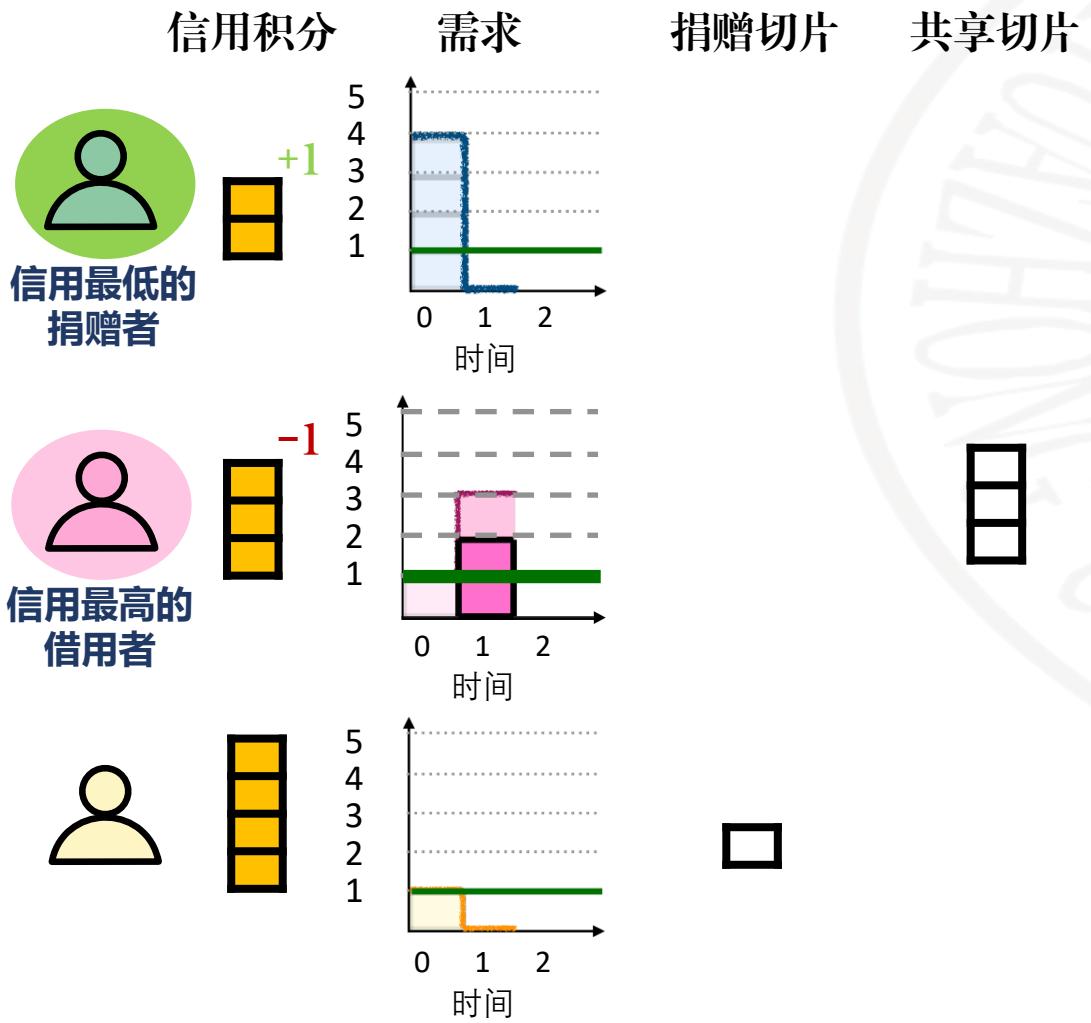


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

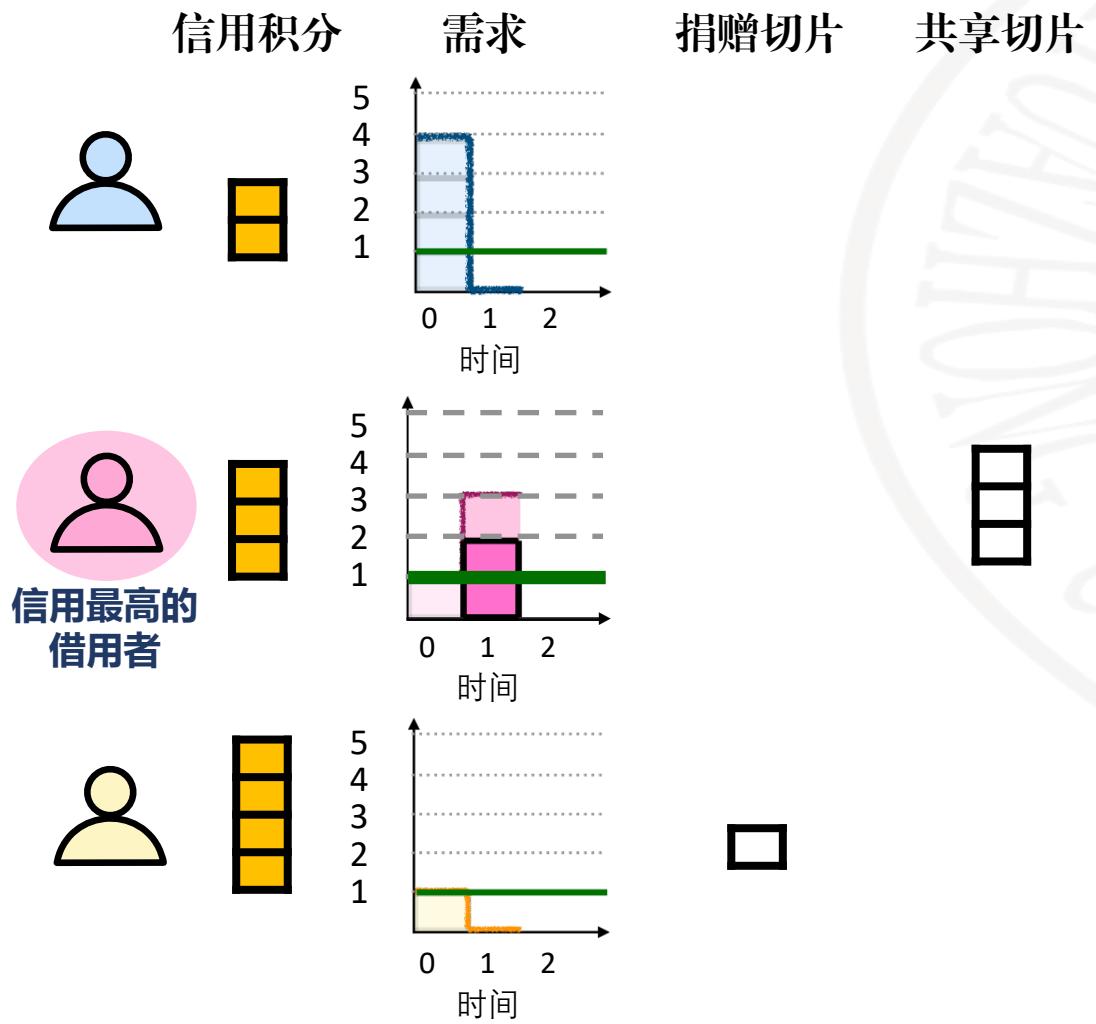


3 资源池使用策略

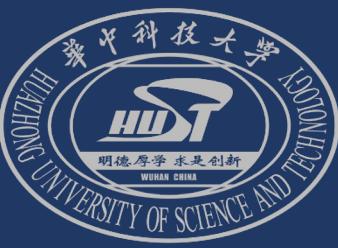
- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



研究动机

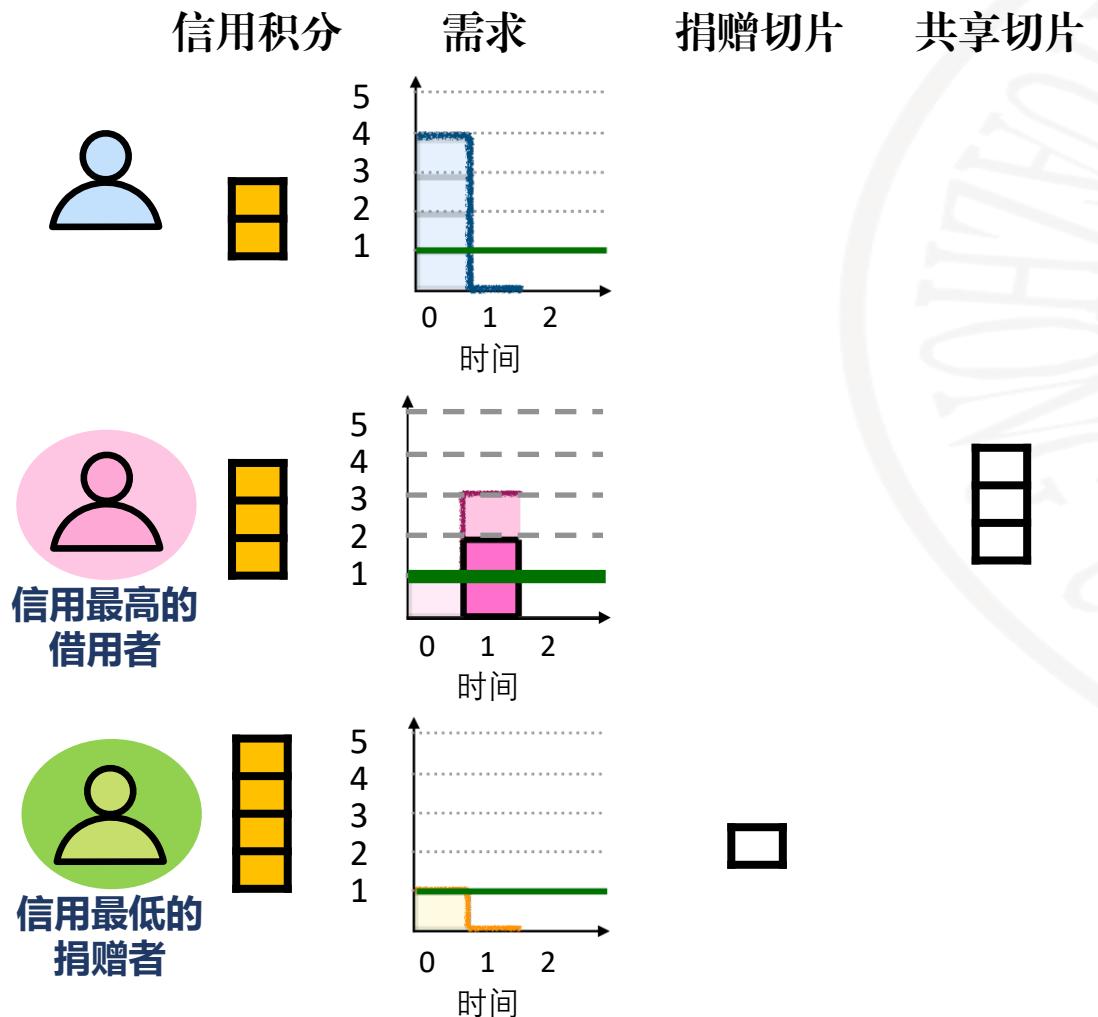
基于信用的
资源分配资源池
使用策略Karma
效果评估

3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

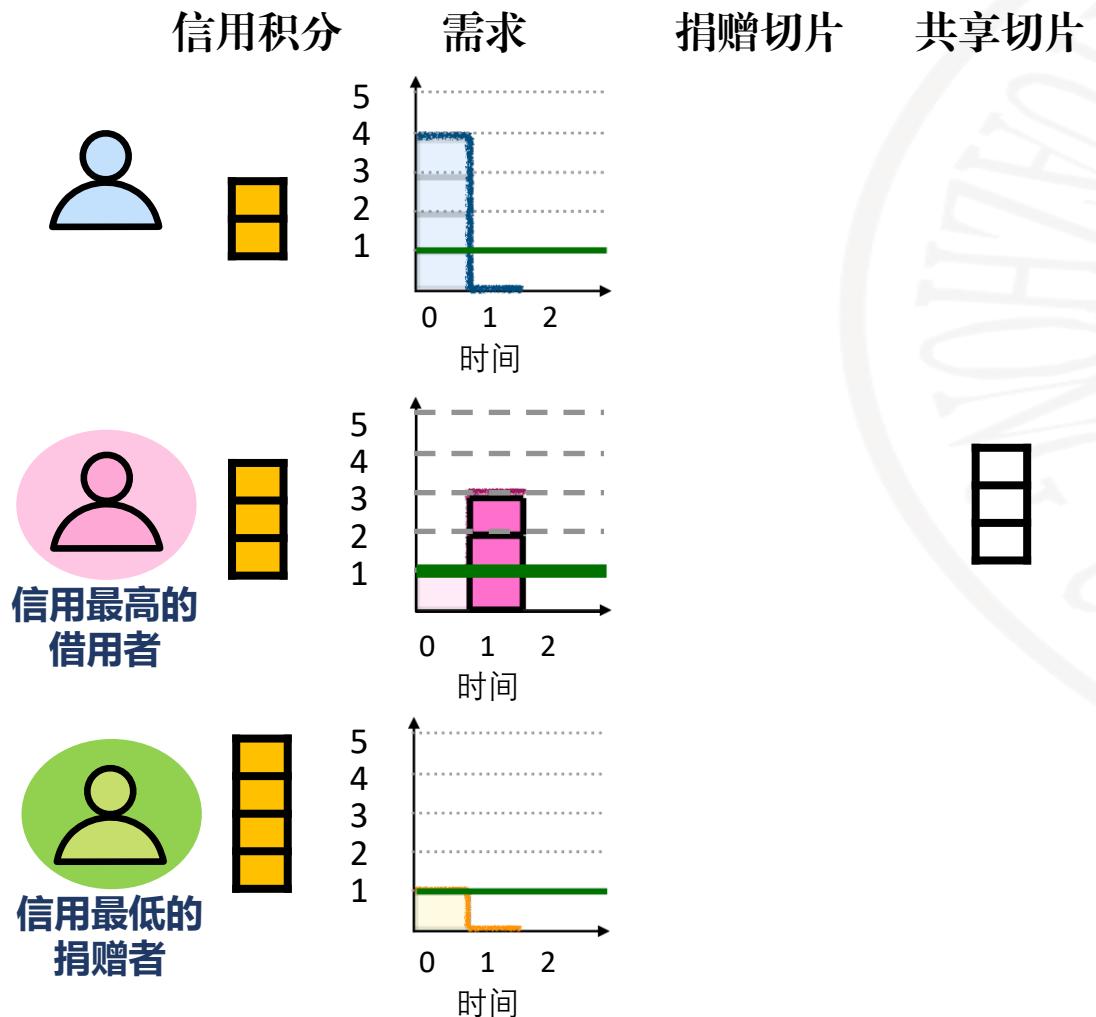


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

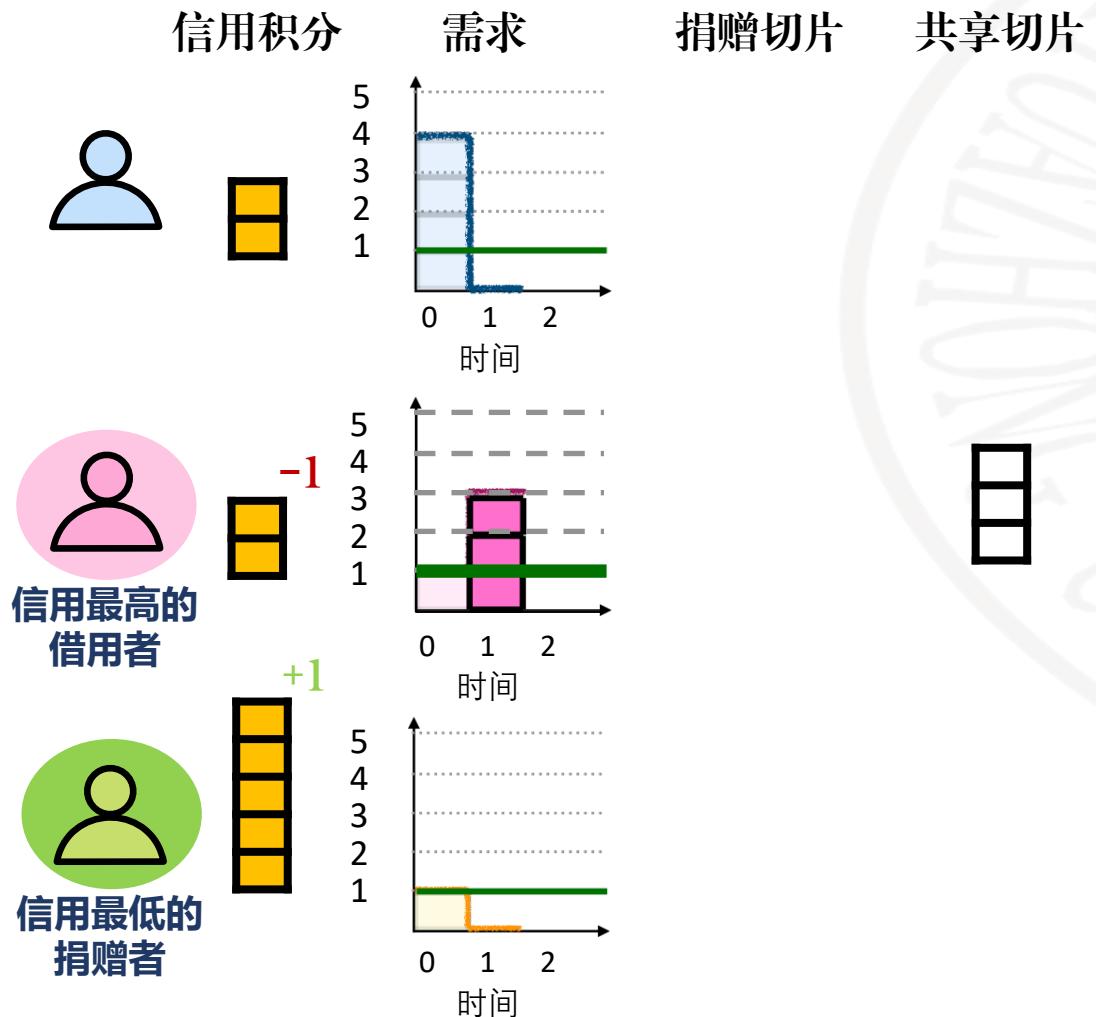


3 资源池使用策略

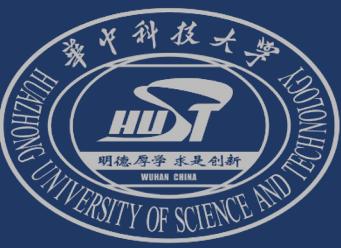
- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

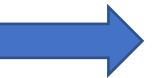


研究动机

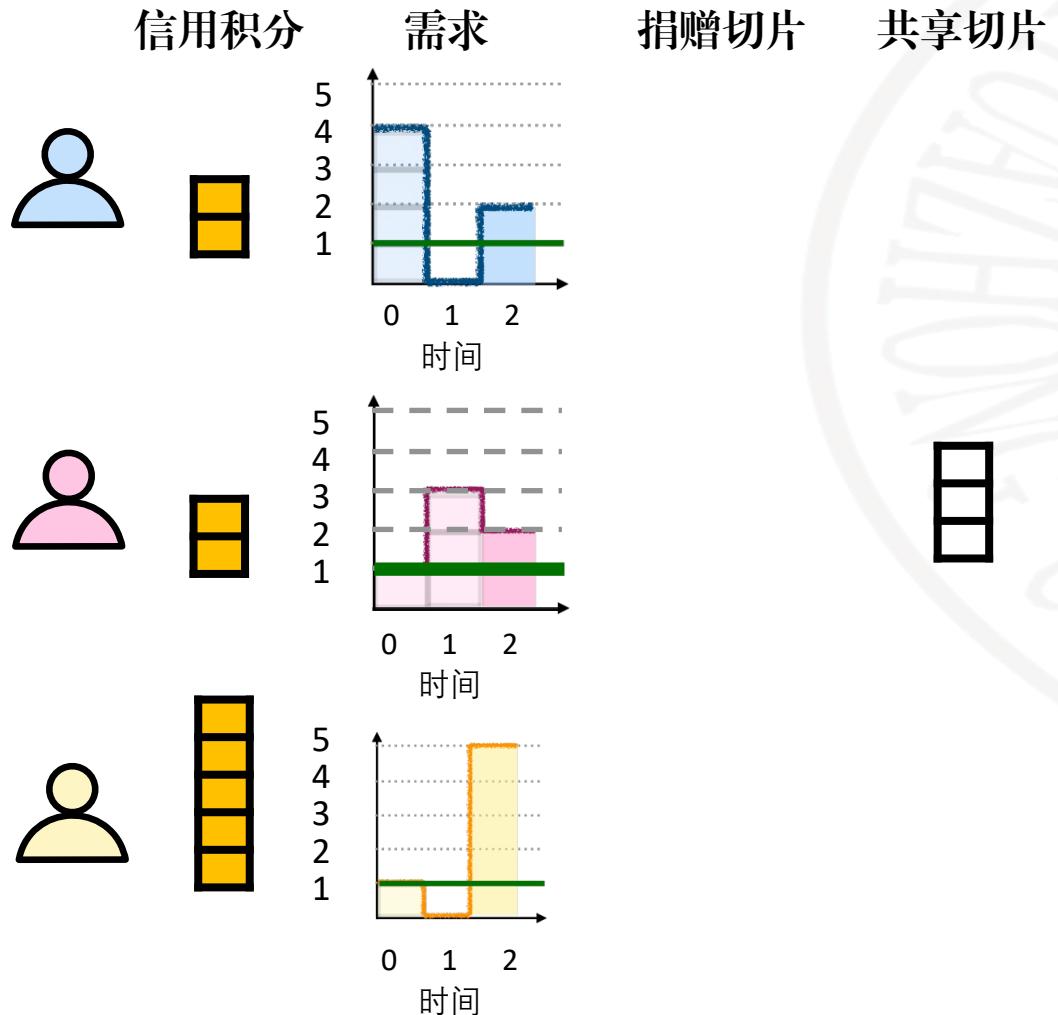
基于信用的
资源分配资源池
使用策略Karma
效果评估

3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者

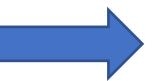


将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

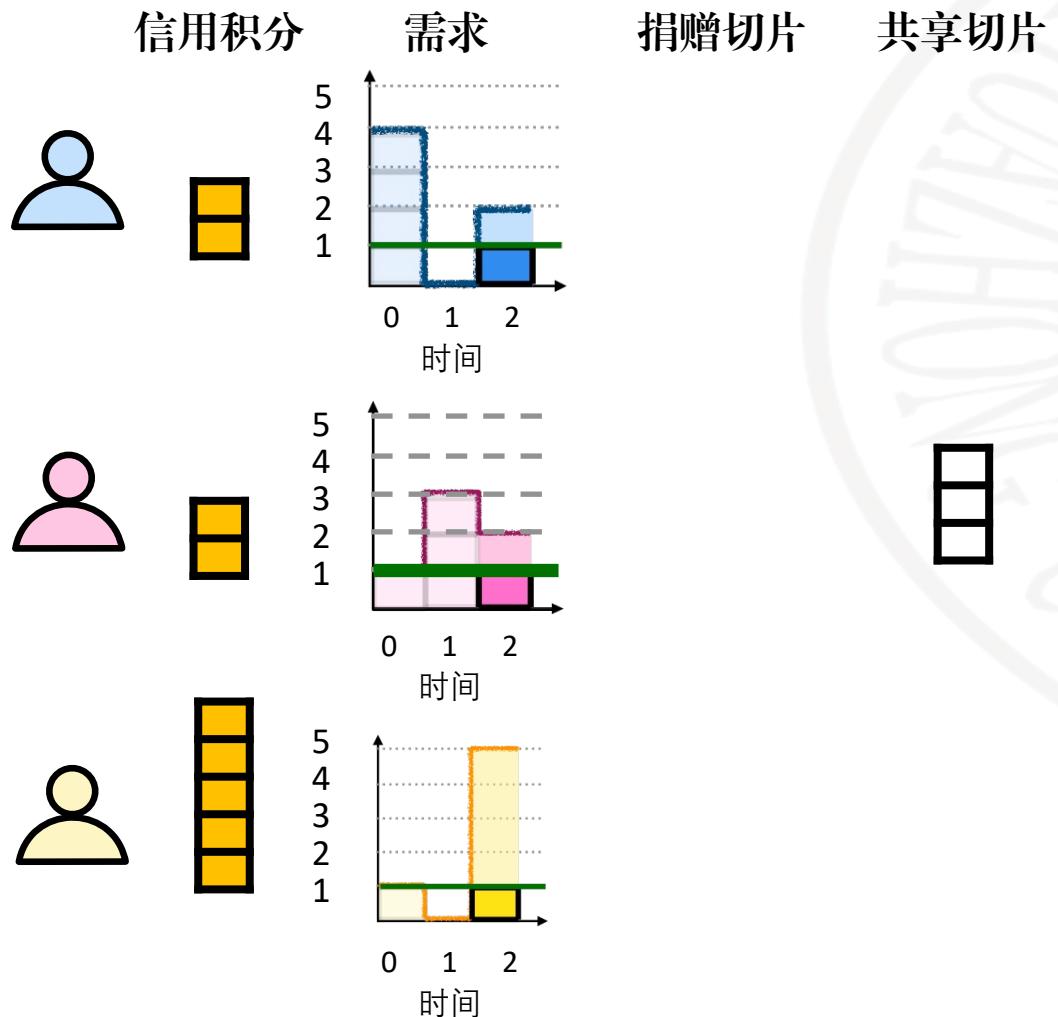


3 资源池使用策略

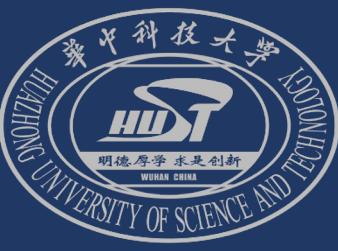
- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



研究动机

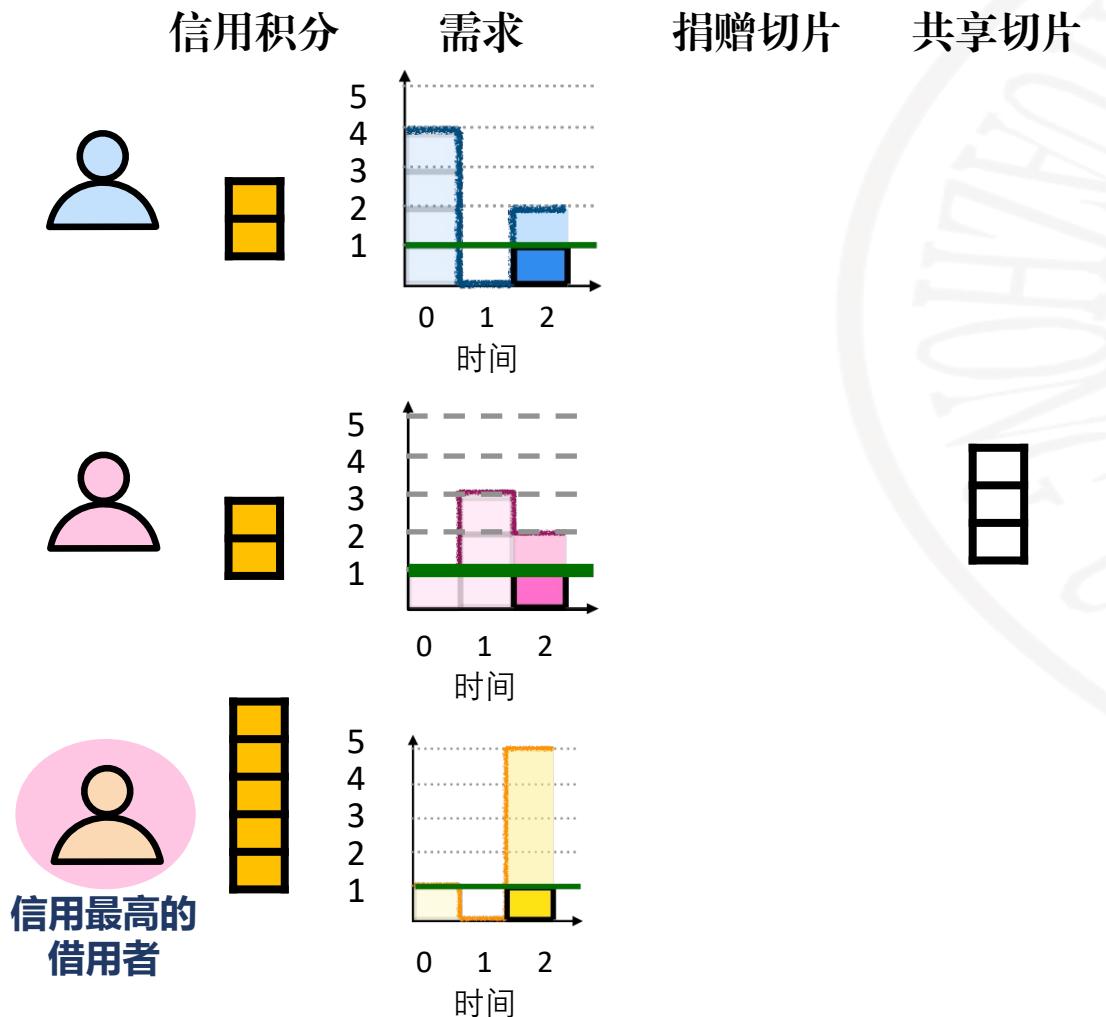
基于信用的
资源分配资源池
使用策略Karma
效果评估

3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



研究动机

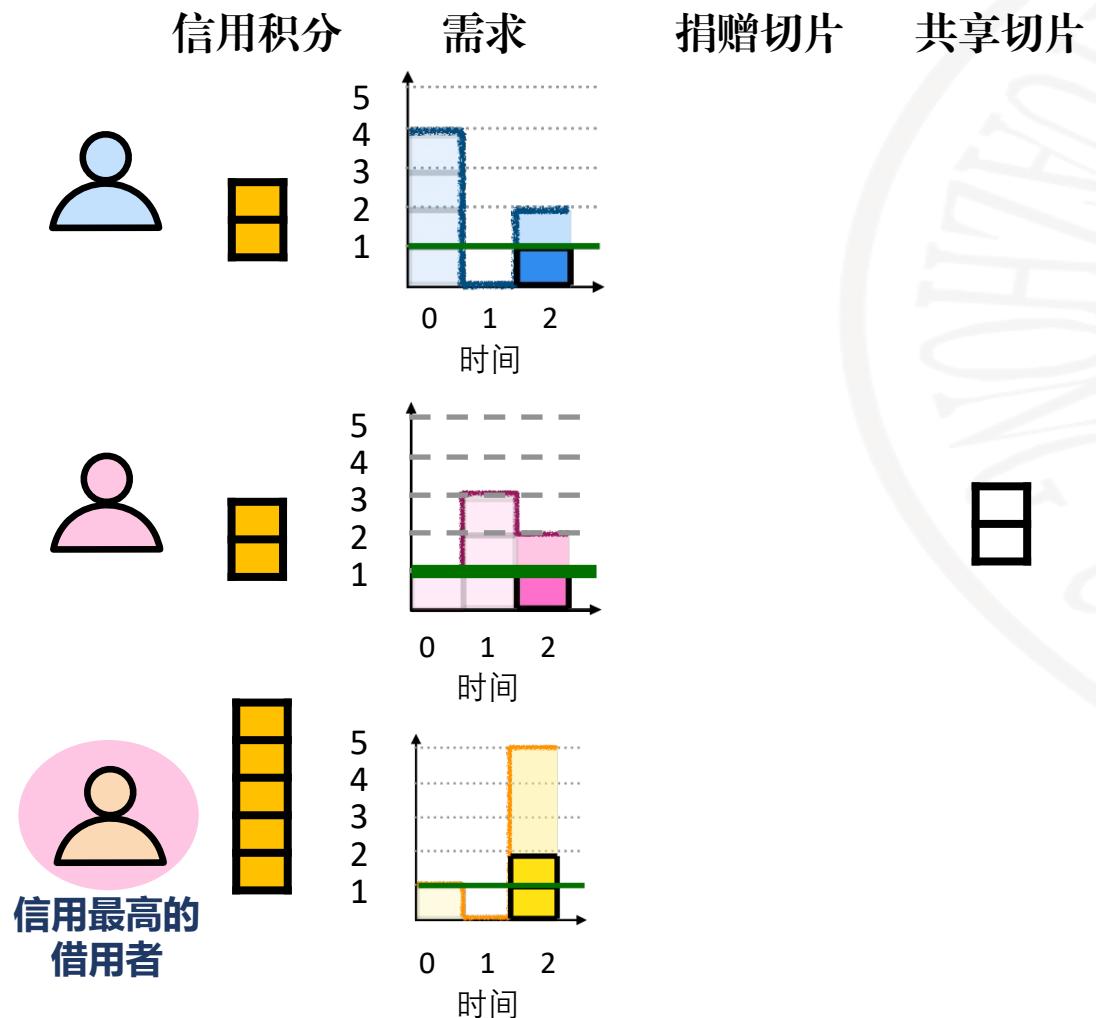
基于信用的
资源分配资源池
使用策略Karma
效果评估

3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



研究动机

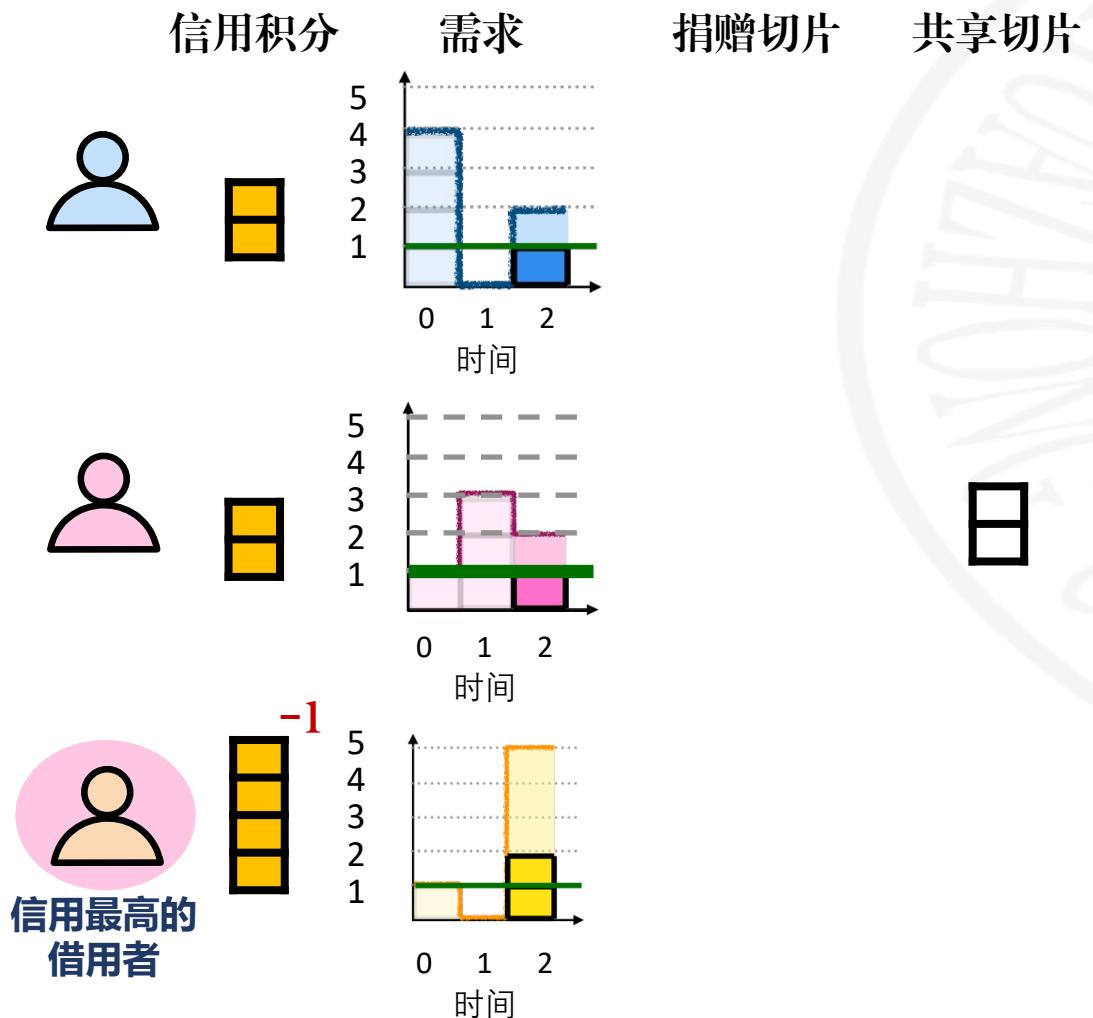
基于信用的
资源分配资源池
使用策略Karma
效果评估

3 资源池使用策略

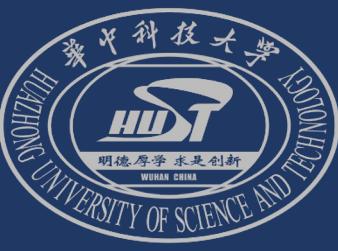
- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



研究动机

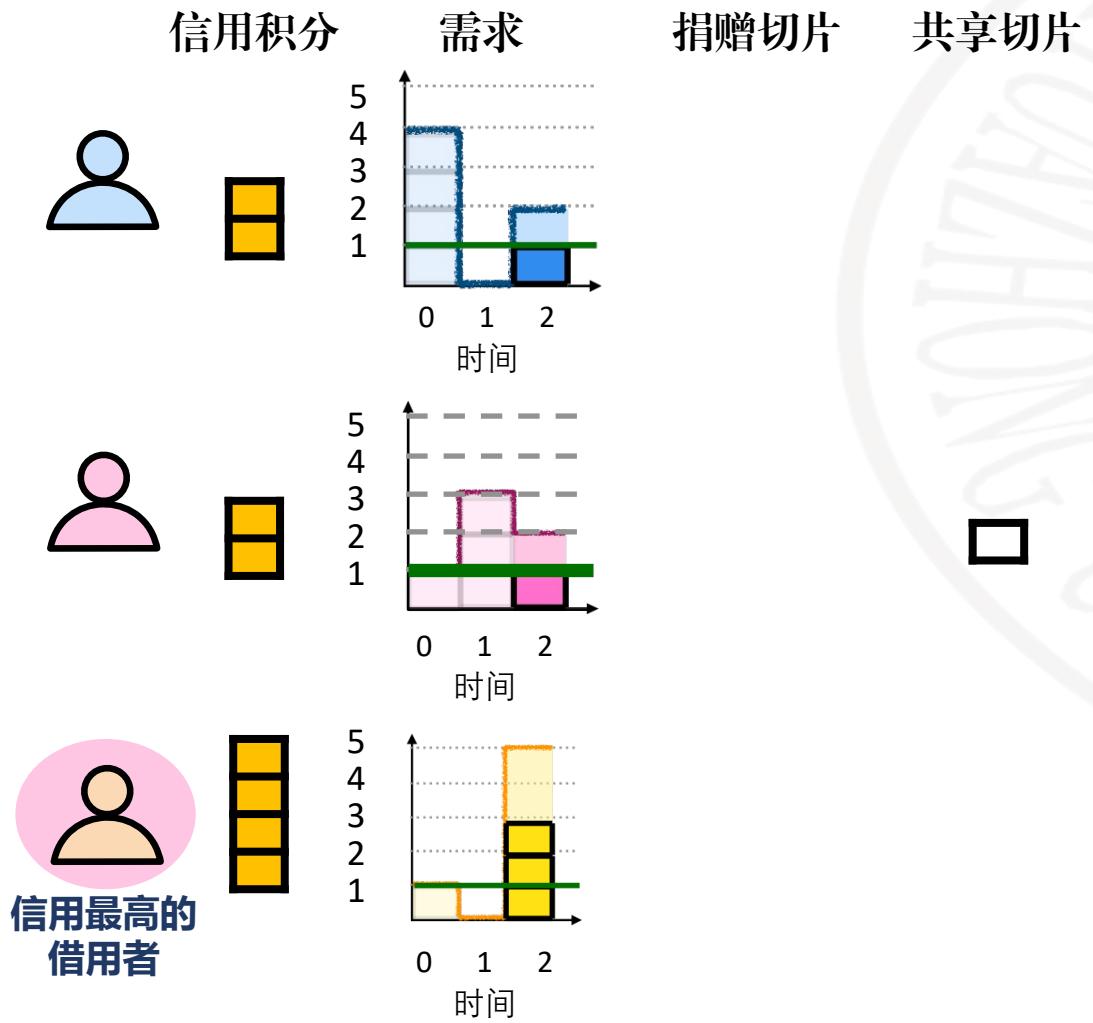
基于信用的
资源分配资源池
使用策略Karma
效果评估

3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



研究动机

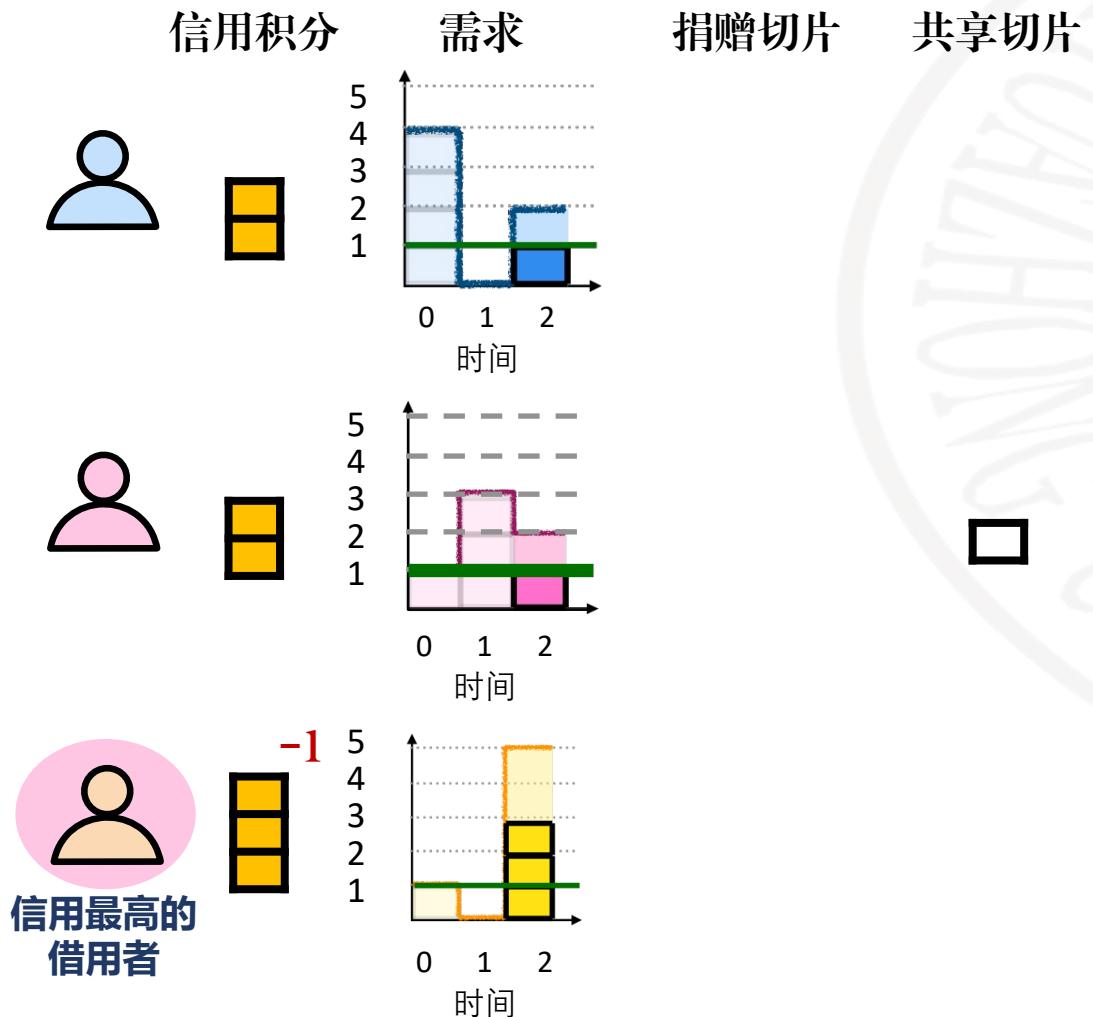
基于信用的
资源分配资源池
使用策略Karma
效果评估

3 资源池使用策略

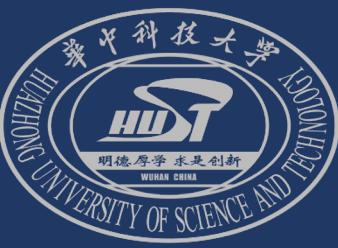
- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



研究动机

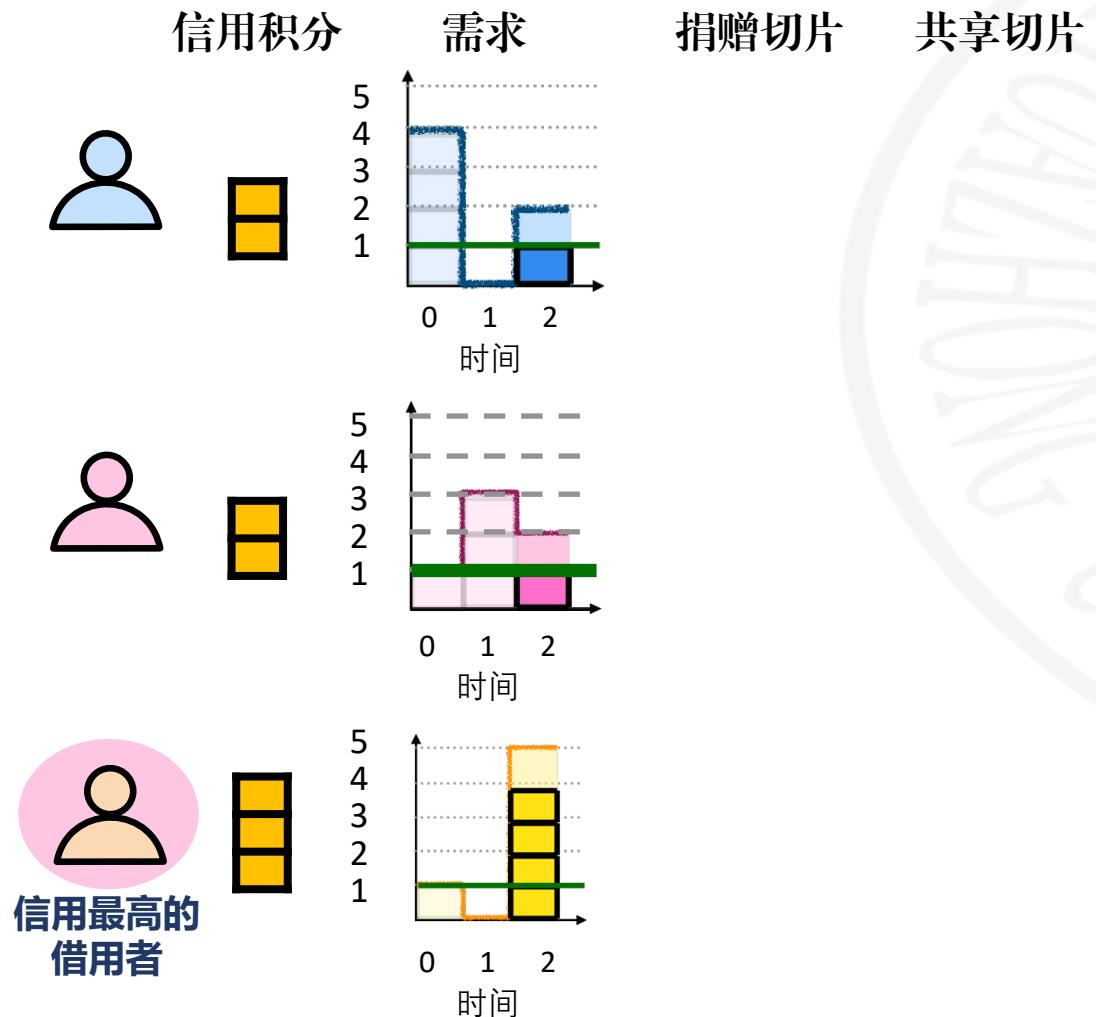
基于信用的
资源分配资源池
使用策略Karma
效果评估

3 资源池使用策略

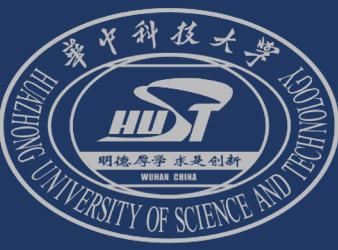
- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

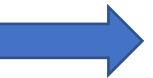


研究动机

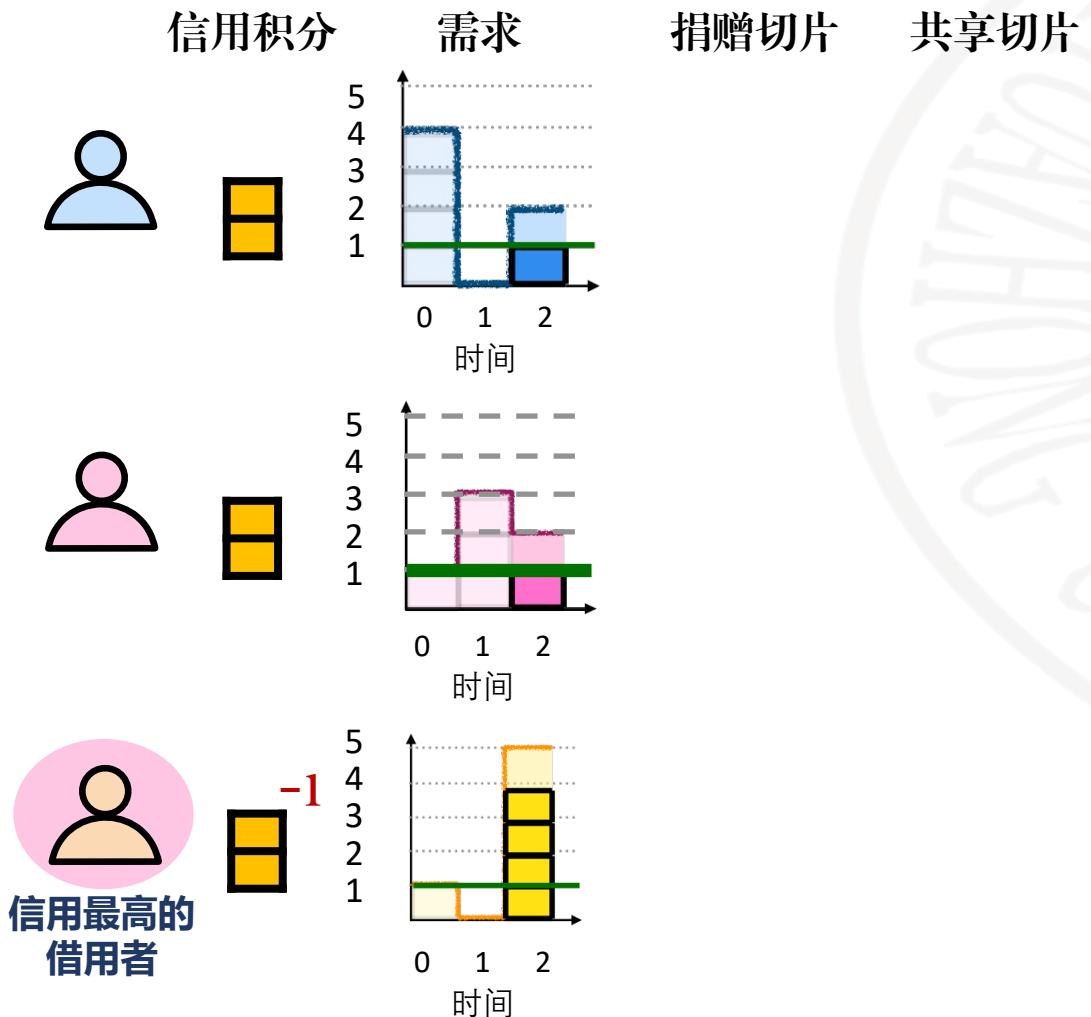
基于信用的
资源分配资源池
使用策略Karma
效果评估

3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



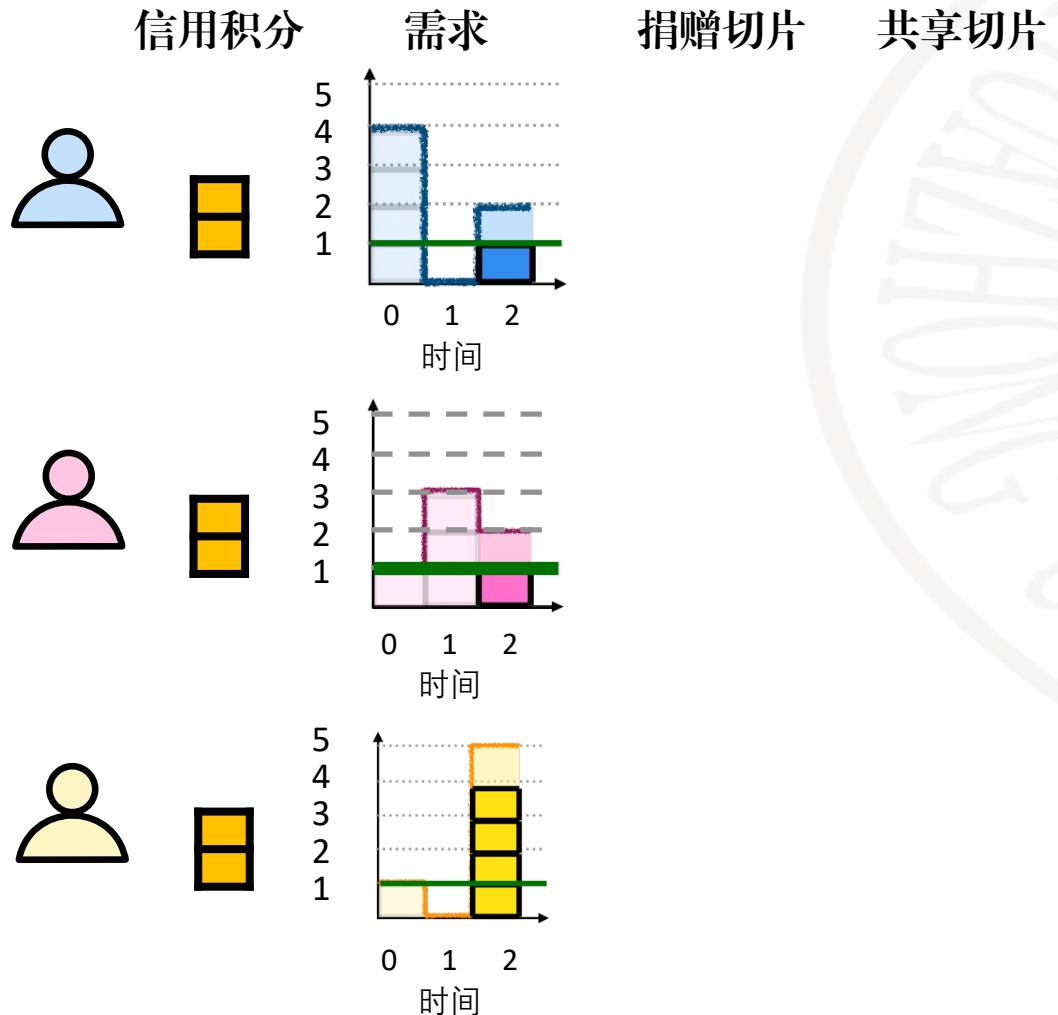


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者

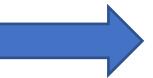


将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1

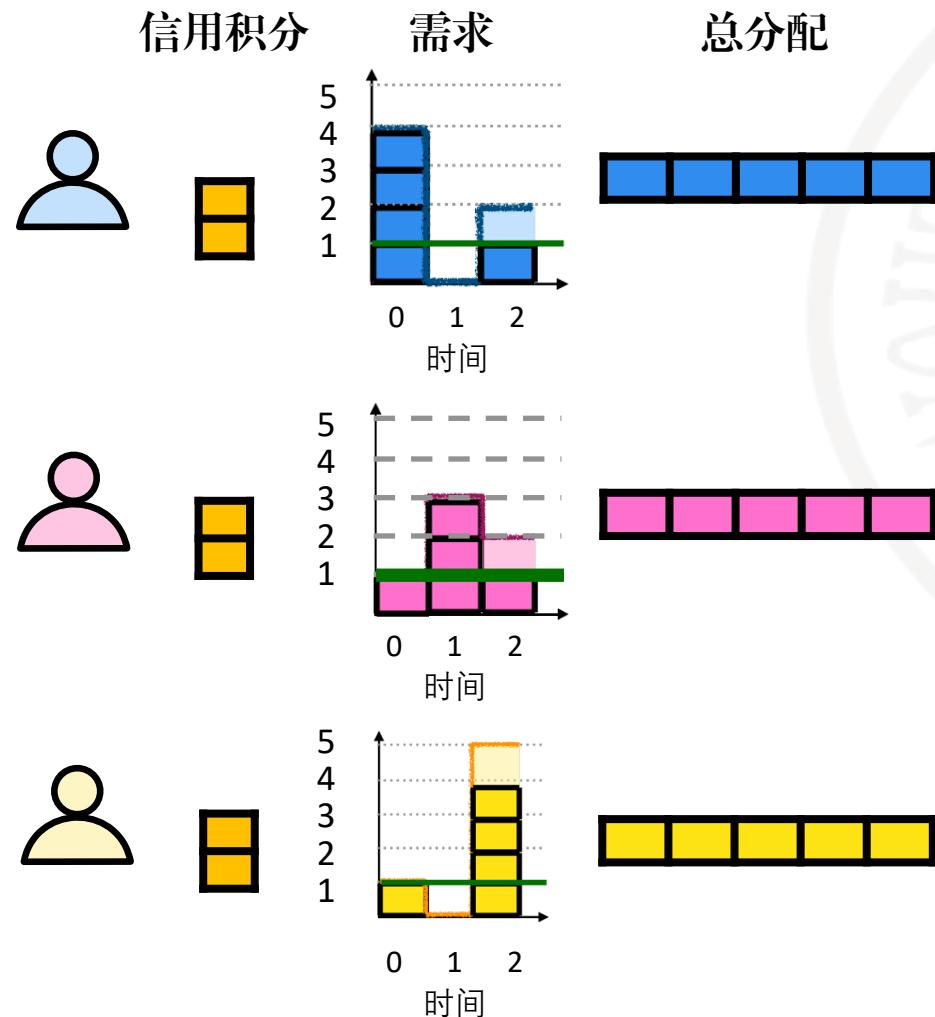


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



关键的理想属性：



Pareto efficiency
帕累托效率



Strategy-proofness
策略验证性

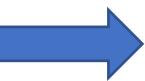


Fairness
公平性

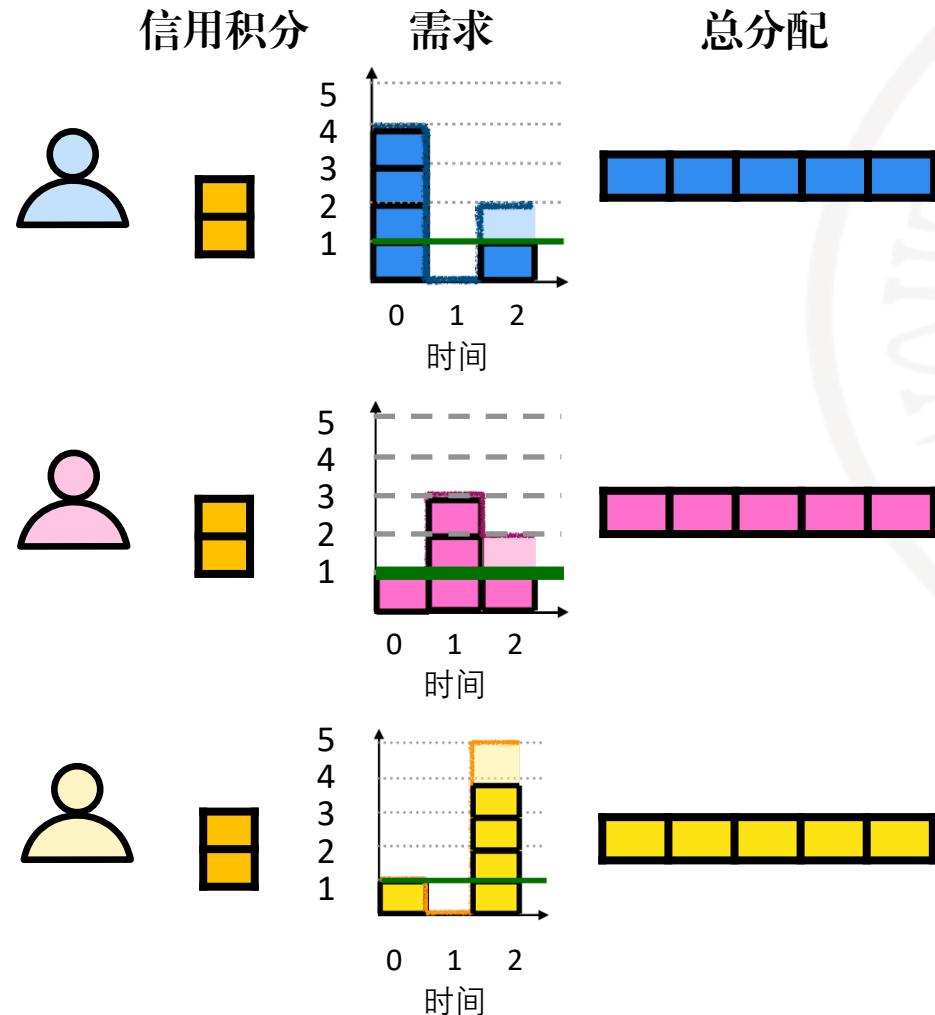


3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



对于资源分配，首先考虑供大于求的情况，此时Karma优先保证贡献的资源能被利用，最大化已分配资源的利用率。且为了保证公平，Karma会优先考虑使用信用点最少用户贡献的资源。只有当捐赠池被消耗完毕，才会考虑共享池的资源。

研究动机

基于信用的
资源分配

资源池
使用策略

Karma
效果评估

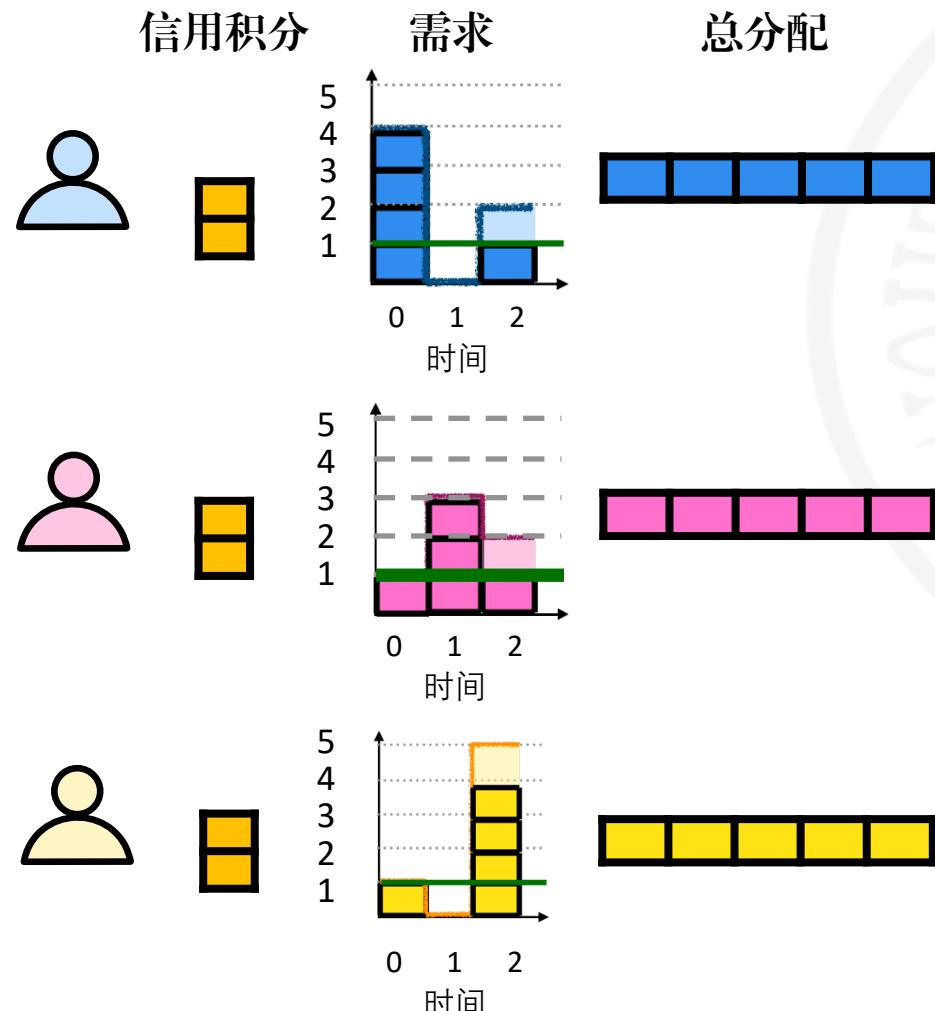
三

3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者

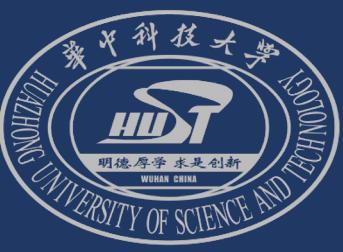


将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



其次，考虑求大于供的情况，同样出于公平考虑，Karma优先考虑为信用点数量最多（信用点多意味着借用的资源少）的用户分配资源。

研究动机

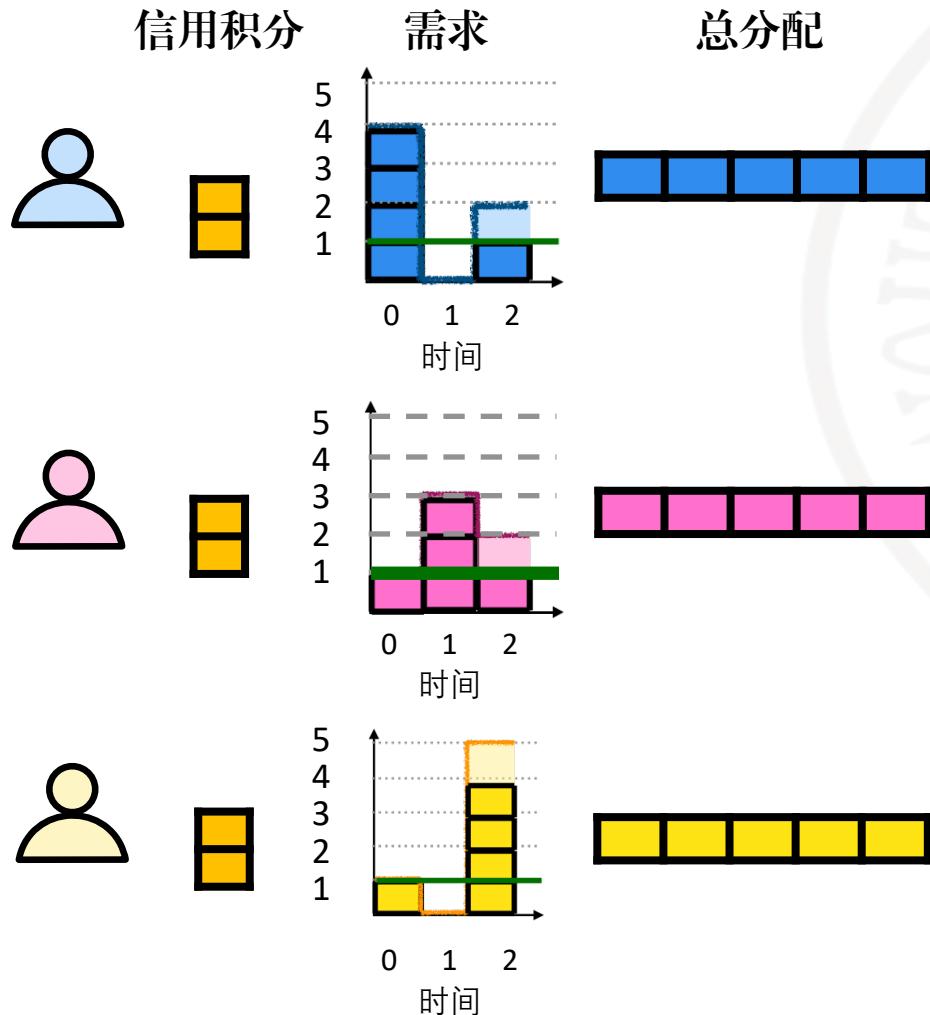
基于信用的
资源分配资源池
使用策略Karma
效果评估

3 资源池使用策略

- 挑选信用最高的借用者
- 挑选信用最低的捐赠者



将切片分给借用者：
借用者信用积分-1，捐赠者信用积分+1



通过上面两个优先级策略，Karma既能均衡系统中每个用户的信用点，也能均衡每个用户借用的额外资源。



Karma效果评估

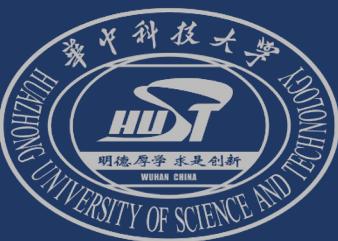
Part4

研究动机

基于信用的
资源分配

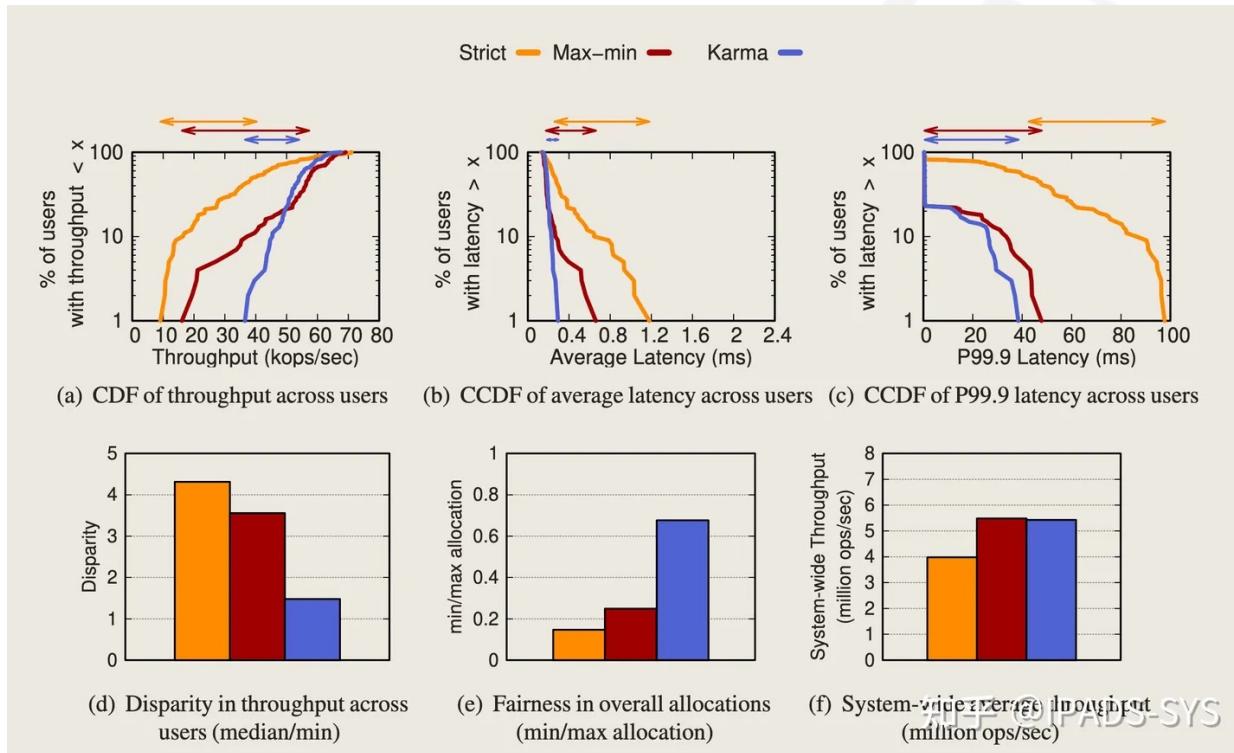
资源池
使用策略

Karma
效果评估



4 Karma效果评估

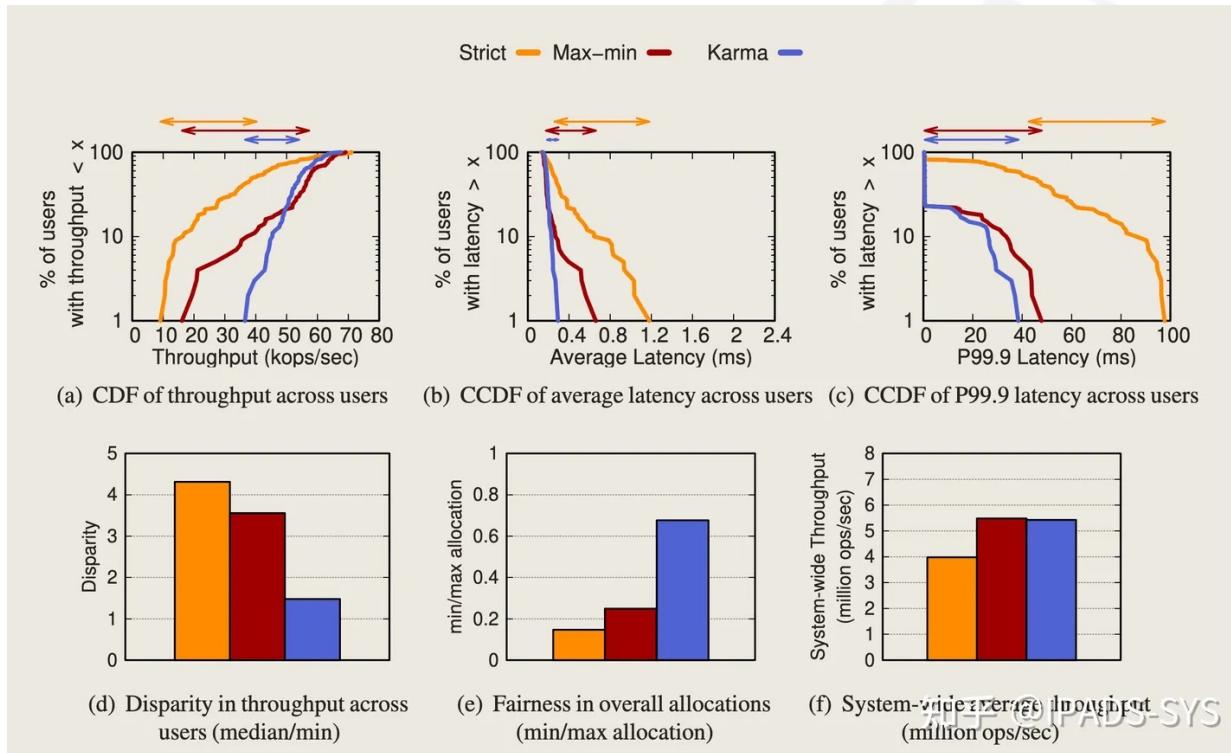
Karma的目标是在保证调度公平的情况下
提高系统吞吐、降低请求延迟
此外，Karma需要考虑用户请求动态变化的场景



从上排三张图来看，
Karma加持的Jiffy在绝大多数
情况下的吞吐都比现有方法更
高，平均延迟和尾延迟都比现
有方法低。

4 Karma效果评估

Karma的目标是在保证调度公平的情况下
提高系统吞吐、降低请求延迟
此外，Karma需要考虑用户请求动态变化的场景



从下排三图可知，Karma很好地完成了需求：公平调度。系统中用户程序的吞吐和资源分配都比其他方案更均衡，同时并没有显著降低系统的整体吞吐。



华中科技大学

HUAZHONG UNIVERSITY OF
SCIENCE AND TECHNOLOGY

Thanks For Listening