



# 转发表设计

路由器实验团队

2020年10月

# 主要内容

## Contents

- 转发表功能
- 转发表数据结构
- 转发表性能
- 实现思路



# 转发表功能

- IP分组转发的依据是转发表
  - Forwarding Table , 也称FIB , Forwarding Information Base
- 每个表项包含网络前缀及下一跳信息
  - 网络前缀 : IP地址前缀、前缀长度或子网掩码
  - 下一跳信息 : 下一跳接口、下一跳IP地址
- 转发表的操作
  - 插入、更新、删除 : 软件更新转发表时使用 , 精确匹配
  - 查询 : 转发时查询转发表 , 最长前缀匹配
    - 性能要求高 , 影响转发速率
- ✓ 编程作业中已练习过基本操作



# 转发表功能

- 转发时，若转发表中无法查到对应表项
  - 生成一个ICMP Destination Unreachable ( Type 3 , Code 0 ) 报文并发送给源IP地址 ( 本实验中可选 )
    - 需要 ( 再次 ) 查询转发表，获得源IP地址的下一跳信息
    - 该报文TTL可根据实现选取，一般为64或255
    - 包含原IP分组头部及至少8字节有效载荷
    - 若该IP分组已经为ICMP错误报文，则不再发送



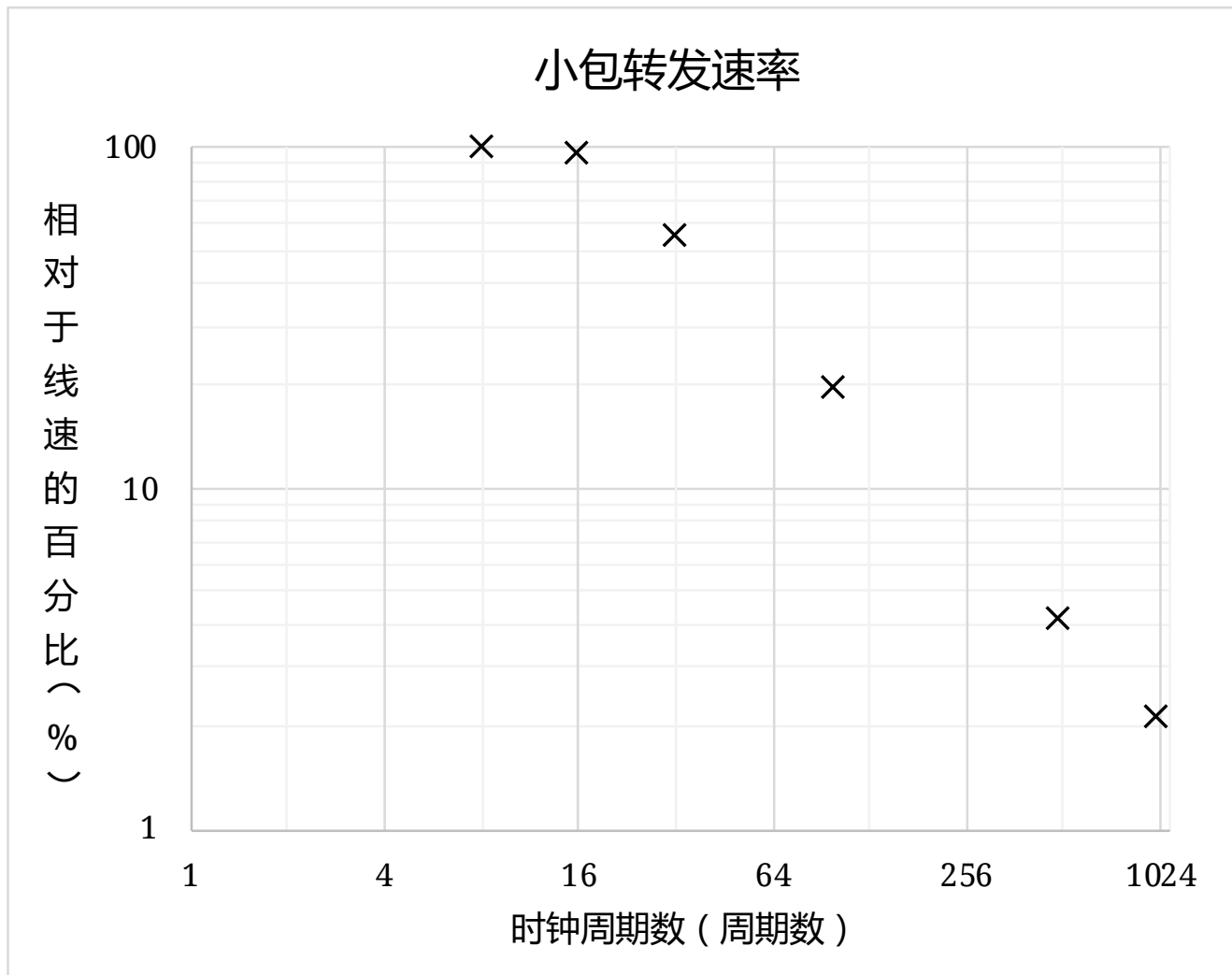
# 转发表数据结构

- 转发表需要基于硬件实现，优化性能
- 蛮力查找
  - 实现简单，性能不佳
- 基于树的数据结构
  - Trie：将IP地址前缀视作字符串，实现最长前缀匹配
  - 压缩Trie：减少树的深度，优化存储空间和性能
- 其他高级数据结构与算法
  - Luleå、Poptrie、.....
- 同时，需要方便软件修改转发表



# 转发表性能

仅供参考





# 实现思路

- 如何在实验板 ( XC7A200T ) 上存储转发表？
  - BRAM :  $365 \times 36 \text{Kib} = 1642.5 \text{KiB}$  , 最少1周期延迟
  - LUTRAM :  $2888 \text{Kib} = 361 \text{KiB}$  , 组合逻辑
  - SRAM :  $2 \times 2 \times 16 \text{Mib} = 8192 \text{KiB}$  ,  $\sim 10 \text{ns}$
  - 容量越大, 延迟越大
- 每条表项约98 bits (  $32+32+32+2$  )
  - 若蛮力存储, 1000条约11.96KiB
- 存储后, 可实现相应数据结构
- 需要为CPU上的软件提供访问接口 ( 后续介绍 )
  - 提供访问和配置寄存器? 共享内存, 暴露存储?



# 本周任务

---

- 进一步完善转发逻辑
- 设计并实现转发表





谢谢