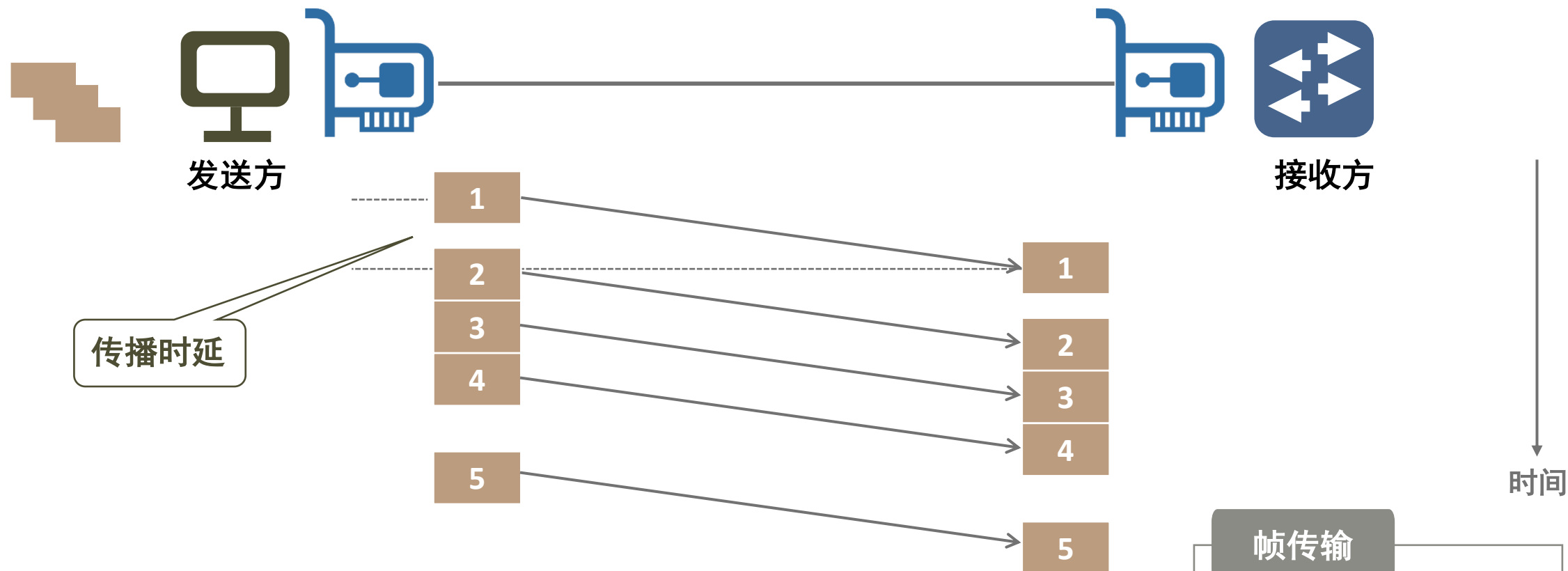


数据链路层的流 量控制



帧传输模型

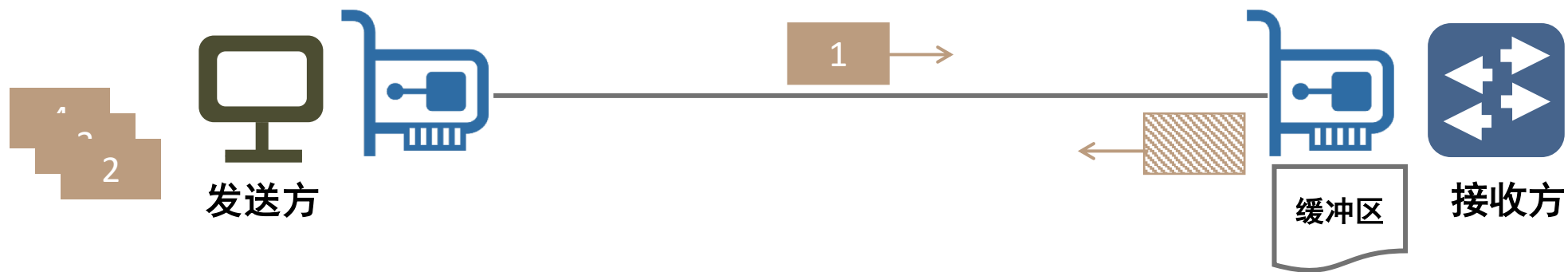


流量控制：用来确保发送实体发出的数据不会覆盖接收实体已接收的数据。



停一等流量控制

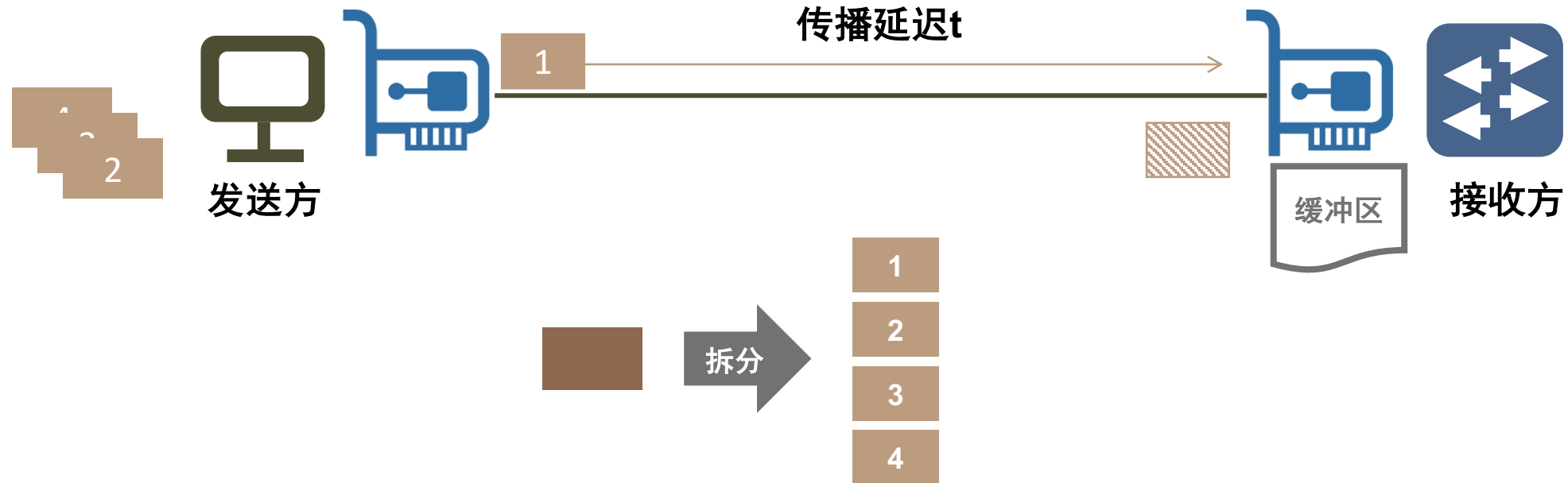
停等式控制：发送实体发送一个帧，接收实体收到后发回一个对该帧的确认，表示同意接收下一个帧；发送实体必须等待，收到确认后才能发送下一个帧。



**接收实体简单地用停止发送确认的方式来
阻止发送端的数据流**



停—等流量控制的特点



优点

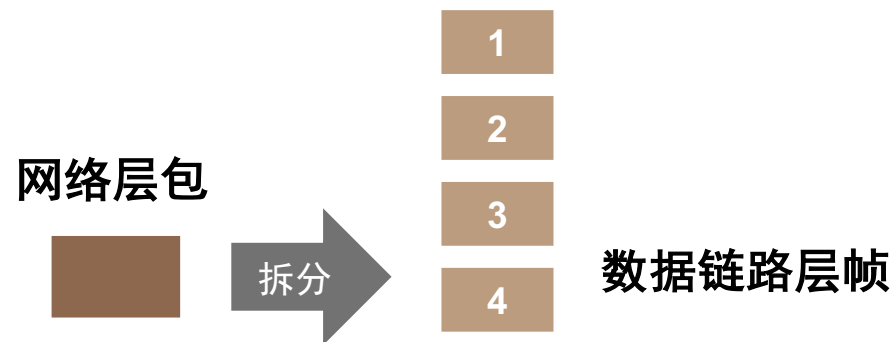
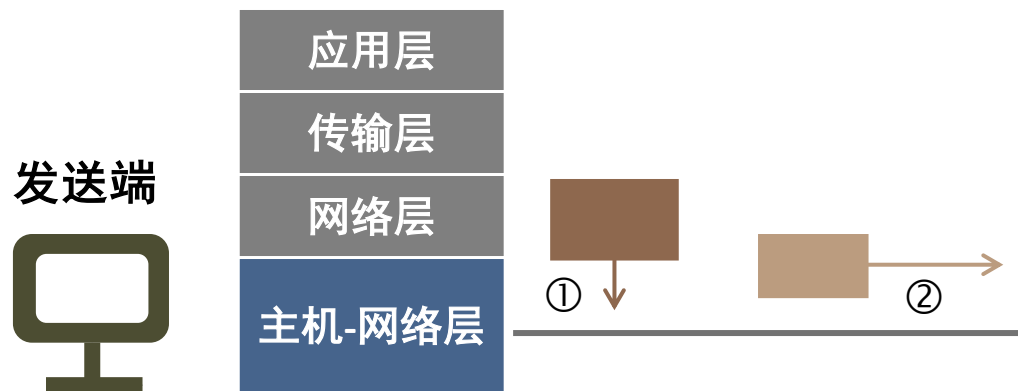
- 控制简单
- 适用于当包被分成数量不多的帧发送时

缺点

- 效率不高
- 当一个包用多个帧发送且链路传播时延大于发送时间时



为什么要分拆数据包？



?

为什么数据被分成
许多个较小帧发送？

可能的原因

- 接收缓冲区大小受到限制
- 传输数据愈长出错可能性愈大
- 一个节点不能占用信道时间过长



北京大学

包/分组

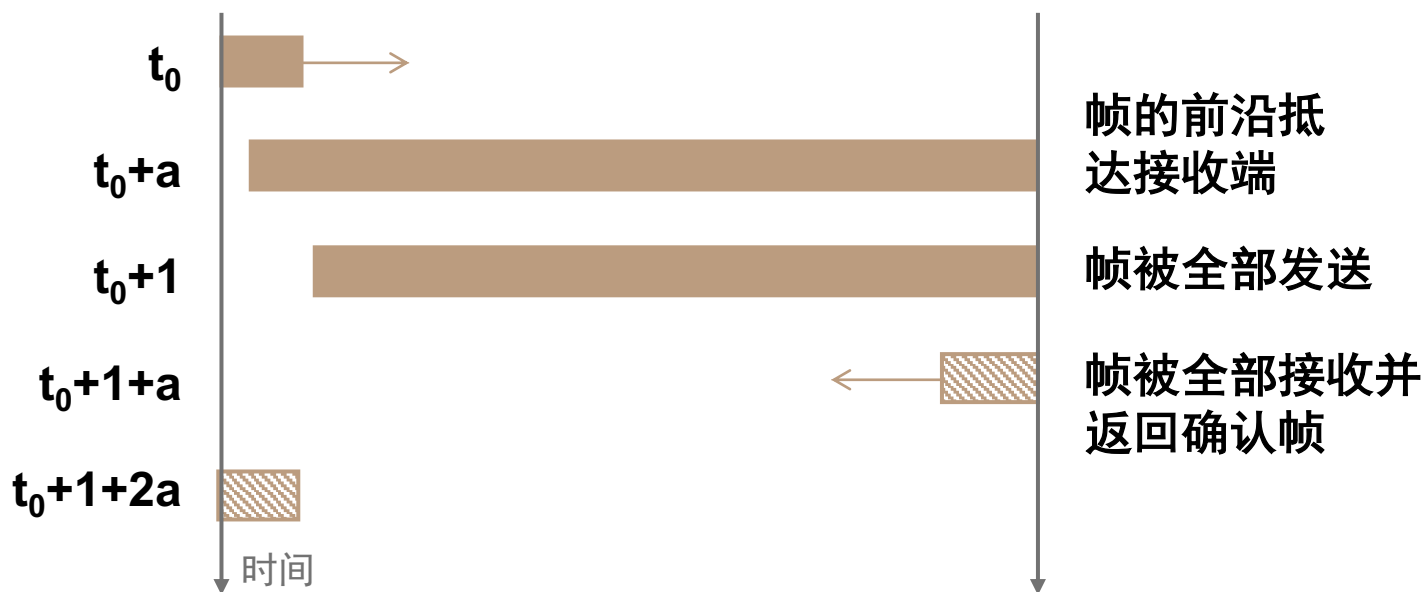
数据帧

停一等流量控制机制性能



●当传播时延 $<$ 发送时间

- 忽略确认帧的传输时间
- 信号的传播时延为 a
- 一帧的传输时间为1



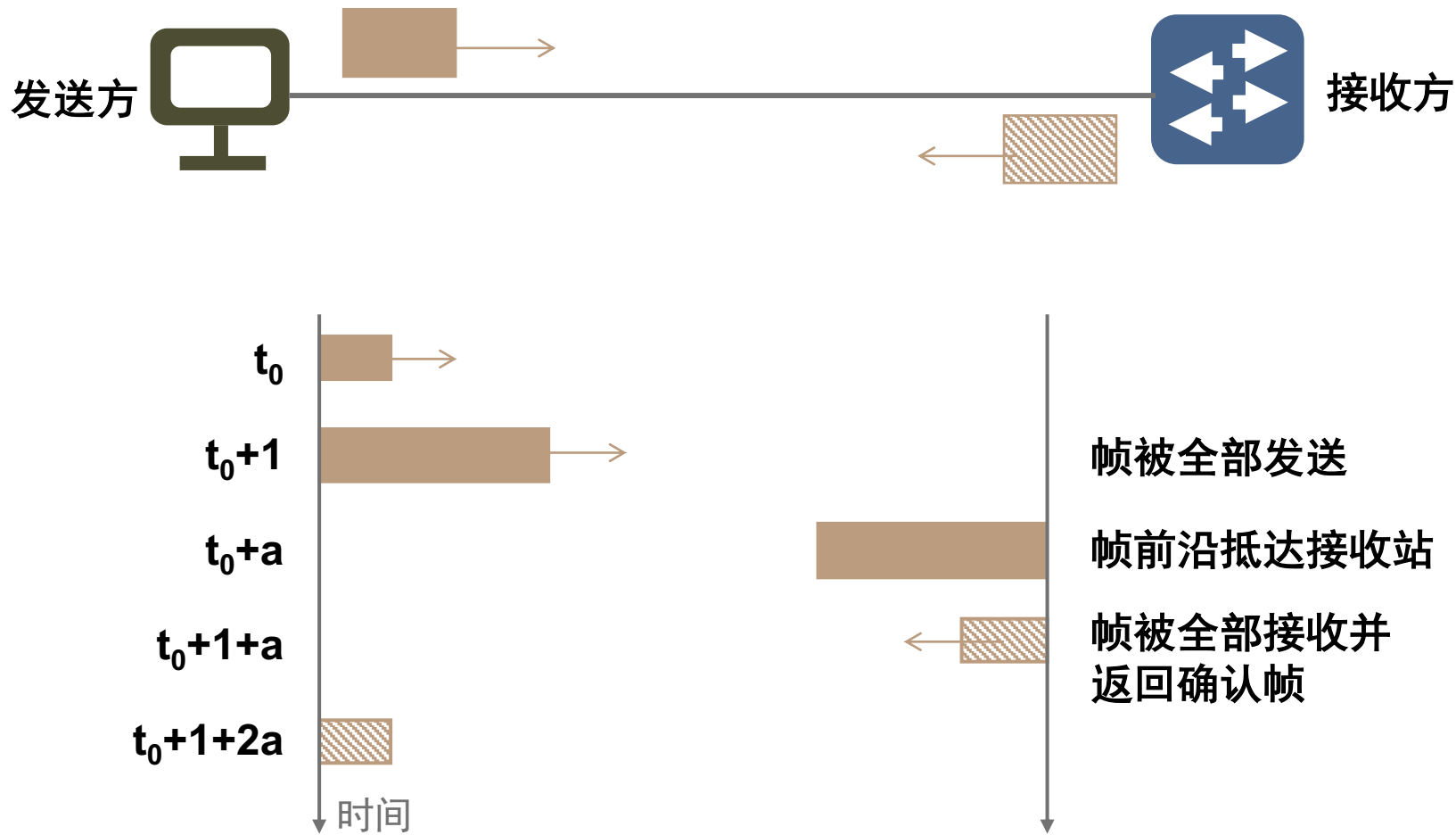
链路空白处即为链路空闲时间。

确认帧 (ACK)

数据帧



停一等流量控制机制性能（续）



●当传播时延>发送时间

- 忽略确认帧的传输时间
- 信号的传播时延为 a
- 一帧的传输时间为1

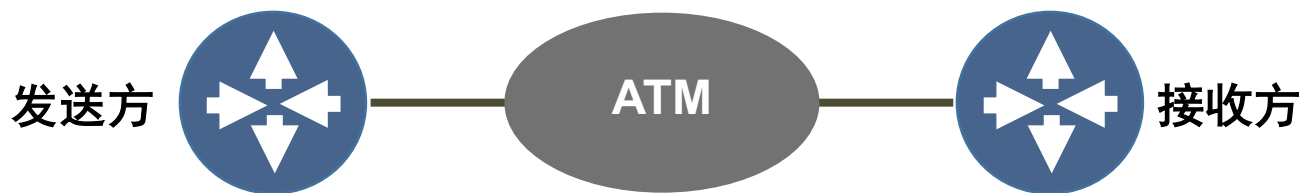
链路的大部分时间都
空闲！



长距离 vs. 短报文

示例1：在如图所示的网络中，两个网络设备通过ATM相连。

- 采用光纤作为传输介质
- 两端之间距离为1000km
- 数据帧长为53B = 424b
- 数据率为155.52Mbps



$$\text{发送时间} = 424 / (155.52 \times 10^6) = 2.7 \times 10^{-6} \text{ s}$$

$$\text{传播时延} = 10^6 / 3 \times 10^8 = 0.33 \times 10^{-2} \text{ s}$$

距离越长，帧越短，传输效率越低！

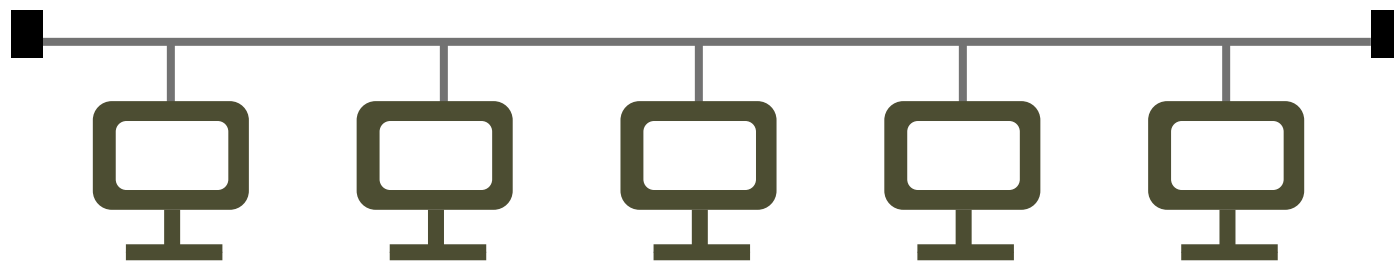
$$\text{线路传输效率} = \frac{\text{帧发送时间}}{2 \times \text{传播时延} + \text{帧发送时间}} = 0.004\%$$



高速率vs.近距离

示例2：在如图所示的广播网络中，5个节点共享一个传输介质。

- 两点之间的最大距离 100m；
- 数据速率100Mbps；
- 信号传播速率 2×10^8 m/s；
- 数据帧长1000b；



$$\text{发送时间 } t_{\text{frame}} = 1000 / (100 \times 10^6) = 10 \times 10^{-6} \text{ s}$$

$$\text{传播延迟 } t_{\text{prop}} = 100 / (2 \times 10^8) = 50 \times 10^{-6} \text{ s}$$

距离越短传播时延越小，链路利用率越高！

$$\text{线路传输效率} = \frac{\text{帧发送时间}}{2 \times \text{传播时延} + \text{帧发送时间}} = 99\%$$

