

doi:10.13582/j.cnki.1674-5884.2019.05.007

# “加法”思维与“减法”思维在“计算机网络”教学中的探索与实践

朱磊,周雷,武欣嵘,罗国明

(中国人民解放军陆军工程大学 通信工程学院,江苏 南京 210007)

**摘要:**系统思想是关于事物整体性、相互作用、演化发展的一种思维方法,常用于处理大型复杂问题。在计算机网络教学中,采用基于系统思想的“加法”思维和“减法”思维的分析方法,把难以理解的知识点看作一个整体进行分解,利用“减法”思维抓住关键核心问题进行求解,在此基础上,利用“加法”思维逐步考虑其余因素作用并改进问题求解方法,可以深刻理解问题的复杂性并最终完成复杂问题的求解。

**关键词:**系统思想;复杂问题;加法思维;减法思维

**中图分类号:**G642.0;TN919.2

**文献标志码:**A

**文章编号:**1674-5884(2019)05-0034-07

## 1 系统思想的基本观点

在工科教学中,系统思想是一件发掘问题和解决问题的利器,利用它既可以对复杂问题进行抽丝剥茧发现问题的本质,实现教学中的循循善诱;更能在思维层次和思维方式上,逐步培养出对复杂问题的分解与组合思想,实现学习过程中“鱼”和“渔”的双丰收。

系统思想是一种观念,认为世界是由相互关联、相互依赖、相互制约、相互作用的过程形成的统一整体<sup>[1]12</sup>。基于系统思想发展起来的处理复杂问题的基本观点主要包括以下几方面<sup>[2]2</sup>:(1)整体观点:系统内部的所有组成要素是一个整体,个体与个体之间、个体与整体之间是相互影响、相互制约的;(2)综合观点:系统表现出来的特点,是所有要素在其相互关系条件下共同作用的综合结果;(3)层次观点:处理复杂问题要区分和把握主要矛盾与次要矛盾,突出矛盾的主要方面;(4)价值观点:任何事务和问题始终都是多角度的,面面俱到不可取,体现系统的投入产出比是务实的朴素思想;(5)发展观点:用动态的、发展的观点去思考、研究、解决系统问题。如图1所示。



图1 系统思想的基本观点

综合运用上述基本观点,将系统思想的思维方式融入贯穿在课程教学中,不仅有利于知识学习和掌握,还为剖析复杂问题、完善解决思路提供了一种广泛适用的方法与思维方式<sup>[3-5]</sup>。

## 2 基于系统思想的“加法”思维与“减法”思维分析方法

无论是从学习的角度还是工程实践的角度,解决复杂问题都是一件非常困难的事。基于系统思想的“减法”思维教学法将复杂问题抽丝剥茧,

收稿日期:20190316

作者简介:朱磊(1973-),男,江苏南京人,教授,博士,主要从事网络空间安全研究。

退化分解为不同层次的、相互独立的若干部分,抽取核心问题进行解决。而基于系统思想的“加法”思维教学法则遵循对复杂问题认知的基本规律,以学习过程中认知的渐近性为主线,从解决核心问题开始,逐步添加相互作用的元素,拓展问题的复杂性,最终形成对复杂系统问题的深入理解,并形成最后的解决方法。

从本质上看,基于系统思想的“加法”思维和“减法”思维教学,是将系统思想的价值观点、层次观点、整体观点、综合观点和发展观点进行综合运用。首先从系统众多价值取向中,选取问题研究的核心目标;其次从核心目标出发,按照由主要到次要的顺序,逐步加入影响核心目标的元素,并分析加入元素本身及元素间的相互作用对核心目标的影响;最后在问题发展演化过程中,动态地考虑核心目标的影响因素和环境,得到对复杂问题有针对性的、客观的、动态的分析。

综合运用基于“加法”和“减法”的思维分析方法,将复杂问题递进式分解,以清晰的逻辑展示问题分析与求解过程。本文以通信网络运输层可靠数据传输协议(Reliable Data Transfer, RDT)的分析和设计过程为例,探讨如何综合运用基于“加法”和“减法”的思维分析方法进行教学。

### 3 可靠数据传输问题模型

分析网络数据的可靠传输问题:链路层可靠性仅提供相邻节点间的可靠传输,网络层采用了无连接的数据报服务(IP协议),尽最大努力传送,可能会出现节点丢包等行为。为了提供端到端可靠性保证的数据传输服务,必须在运输层实现可靠数据传输。可靠数据传输原理模型框架如图2和图3所示。

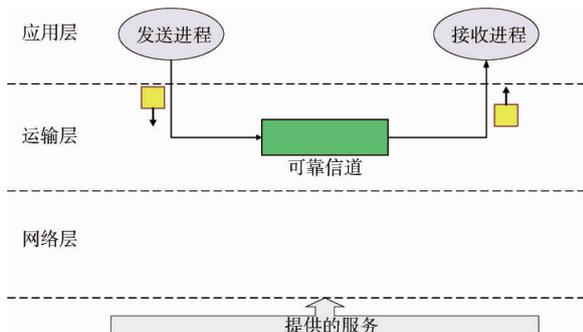


图2 理想的运输层可靠数据传输模型

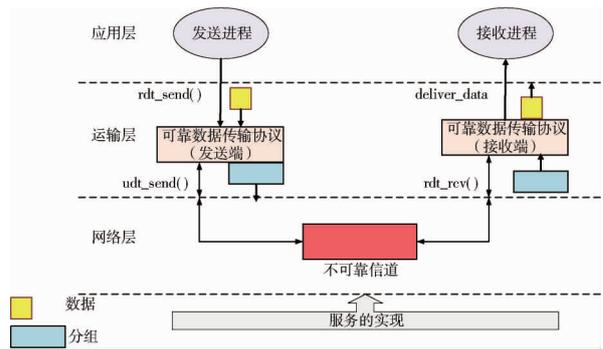


图3 现实的运输层可靠数据传输模型

图2为从应用层的视角,运输层为其提供可靠数据传输服务的模型,数据通过一条可靠通道进行传输,数据不错不丢不乱。图3为从运输层的视角实现可靠数据传输服务的模型,是本文利用“加法”思维和“减法”思维进行教学的内容。图中有关可靠数据传输协议的接口说明如下<sup>[6]5</sup>: (1) rdt\_send():通过 rdt\_send(),上层协议调用 rdt 协议的发送方,实现将数据交付给接收方; (2) udt\_send():通过 udt\_send(),rdt 协议调用下层协议在不可靠信道上进行分组传输; (3) rdt\_rcv():下层协议调用 rdt 协议的接收方,实现分组接收; (4) deliver\_data():由 rdt 协议调用,实现 rdt 接收方向上层提供可靠数据交付。

### 4 Rdt 协议设计方案探寻——“加法”和“减法”思维的运用

可靠数据传输是网络中最为重要的问题之一,也是网络类课程教学中的一个重点和难点问题。由于问题的复杂性,在教授和学习的过程中,如何将复杂的 rdt 协议版本演进的前因后果、问题措施、状态迁移过程、上下层之间交互等内容讲透彻、学明白,存在很大挑战。下面运用“加法”和“减法”思维,渐近地实现一个可靠数据传输协议的发送方和接收方。

运用“加法”和“减法”思维分析可靠数据传输协议时,“减法”的作用在于抓住主要矛盾,将多个元素相互作用的复杂问题,不断增加条件假设将其退化为易于分析、便于求解的理想问题;“加法”的作用在于从关键问题出发,不断弱化过于理想的假设条件,以逼近真实情况,并在多次迭代后求解后趋于实用,其分析思路如图4所示。

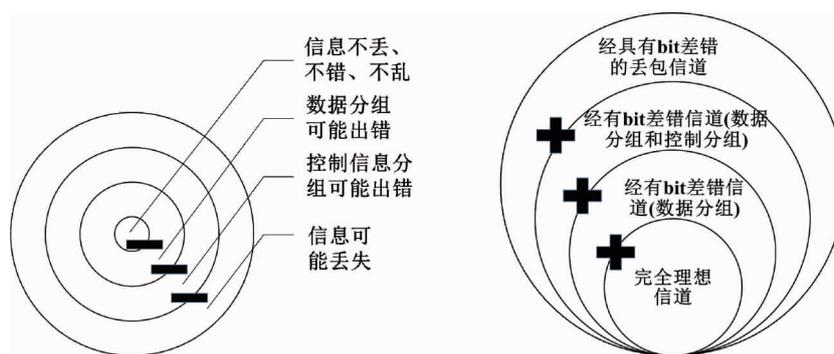


图4 基于“减法”“加法”的问题分析思路示意图

#### 4.1 基于完全理想信道的 rdt 1.0 协议

首先进行“减法”操作,抽象出核心问题,构造可靠数据传输协议。“减法”操作进行如下假设:(1)底层信道没有分组的丢失与超时,不会出现任何 bit 差错或者顺序错误;(2)接收方接收数据的速率与发送端发送数据的速率一致。

在理想信道状况下,rdt 协议发送方和接收方的 FSM 如图 5 所示。在发送端,引起状态变迁的事件由上层的过程调用产生;在接收端,引起状态变迁的事件由下层协议的过程调用产生。详细操作如下。

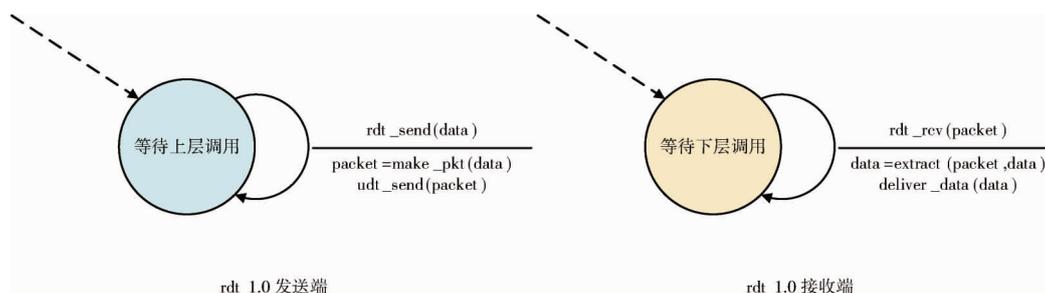


图5 理想信道条件下的 rdt 1.0 FSM

在发送端,上层调用产生 `rdt_send(data)` 事件,触发下列动作:(1)调用 `packet = make_pkt(data)`,生成一个包含该数据的分组;(2)调用 `udt_send(packet)`,将分组发送给下层信道;(3)返回等待状态,继续等待来自上层的数据发送请求。

在接收端,下层调用产生 `rdt_rcv(packet)` 事件,触发下列动作:(1)调用 `extract(packet, data)`,从分组中抽取数据;(2)调用 `deliver_data(data)`,将数据交付给上层;(3)返回等待状态,继续等待从下层信道接收分组。

通过以上讨论可知,经“减法”操作后的理想信道数据传输是可靠的:发送方和接收方不必担心数据出错;双方不需要提供任何反馈信息给对方;接收方能够以足够快的速率接收数据。

#### 4.2 经具有 bit 差错信道(仅数据分组受损)的 rdt 2.0 协议

在 rdt1.0 的基础上进行“加法”操作:考虑下

层信道模型可能出现数据分组 bit 受损情况,构造可靠数据传输协议。基于“加法”操作做如下假设:(1)数据分组在下层信道传输中可能出现 bit 受损;(2)接收方接收到的数据分组没有丢失与乱序。

由于数据在传输中可能出现 bit 差错,接收方收到数据后,要完成分组核实与确认工作:(1)若无差错,给发送方返回一条肯定确认(positive acknowledgement, ACK)控制分组;(2)若有差错,给发送方返回一条否定确认(negative acknowledgement, NAK)控制分组。

由于数据分组可能出错,发送方在发送完一个数据分组后,必须保留当前分组,等待接收方的确认,并完成以下工作:(1)若收到 ACK,变迁到等待发送状态;(2)若收到 NAK,则重新发送数据分组,并维持在等待确认状态。

由于信道引入 bit 差错,rdt2.0 对差错分组具

有重传能力,这种基于重传机制的可靠数据传输协议称为自动重传请求(Automatic Repeat reQuest, ARQ)协议<sup>[6,7]</sup>。ARQ协议需要拥有以下功能来处理 bit 错误:(1)差错检测:接收方能够检测到数据传输发生了错误(如校验和);(2)

接收方反馈:接收方提供明确的反馈信息给发送方(如 ACK 或 NAK);(3)重传机制:发送方重传出错的数据分组。

通过以上分析,得到“加法”操作后 rdt2.0 的发送方和接收方的 FSM 如图 6 所示。

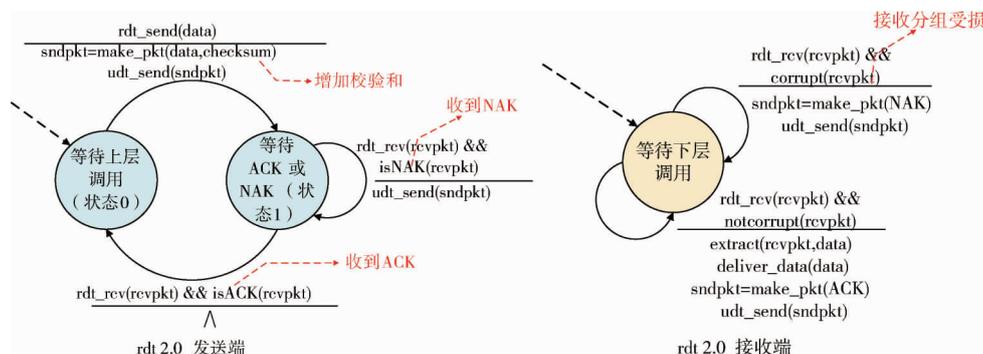


图 6 具有 bit 差错信道(仅数据分组受损)条件下的 rdt 2.0 协议

Rdt2.0 发送端 FSM 有 2 个状态,如图 6 左侧所示,分别是等待上层调用状态(状态 0)和等待确认状态(状态 1)。在状态 0,上层调用产生 rdt\_send(data) 事件,触发下列动作:(1)调用 sndpkt = make\_pkt(data, checksum),生成一个包含待发送数据和校验和的分组;(2)调用 udt\_send(packet),将分组发送到信道中;(3)完成(1)、(2)后,发送端的状态变迁为状态 1。

当发送端处于状态 1 时,下层调用产生 rdt\_rcv(rcvpkt) 事件,触发下列动作:(1)如果收到 ACK 分组(rdt\_rcv(rcvpkt) && isACK(rcvpkt)),说明接收端已成功接收数据分组,状态变迁到状态 0;(2)如果收到 NAK 分组(rdt\_rcv(rcvpkt) && isNAK(rcvpkt)),说明接收端接收到的数据分组出错,则调用 udt\_send(sndpkt) 重新发送该分组,发送端维持在状态 1。

在接收端 FSM 具有 1 个接收等待状态,如图 6 右侧所示,下层调用产生 rdt\_rcv(rcvpkt) 事件,触发下列动作:(1)如果收到的分组受损,即 rdt\_rcv(rcvpkt) && corrupt(rcvpkt),则返回 NAK;(2)如果收到的分组完好,即 rdt\_rcv(rcvpkt) && notcorrupt(rcvpkt),则从分组中抽取数据交付给上层后,返回 ACK。

通过给 rdt1.0 做“加法”操作升级的 rdt2.0 协议,其接收方只有一个状态。而发送方具有两个状态,当处于状态 1 时,只有接收到 ACK 并变迁到状态 0 时才能进行下一个数据分组的发送,所

以 rdt 2.0 也称为停等(stop-and-wait)协议。

#### 4.3 经具有 bit 差错信道(数据分组和控制分组都可能受损)的 rdt 2.1 协议

Rdt 2.0 的致命缺陷是没有考虑 ACK 和 NAK 可能受损。因此,在 rdt 2.0 的基础上,继续做“加法”操作,加入对受损 ACK 或 NAK 因素。在这种情况下,其基本假设同 rdt2.0。

在 rdt2.1 中,由于 ACK 和 NAK 都可能受损,发送端收到出错的控制分组后无法判断接收端是否正确收到数据分组,只有进行重发,而重发导致接收端收到重复的数据分组。解决方法是发送端对其发送的数据分组进行编号,将编号放在数据分组中一起发送。接收方接收到一个数据分组后,通过编号获知该分组是新的还是重传的数据分组。

因为 rdt 2.0 是一个简单的停等协议,只需采用 1 比特编号。发送方在打包数据分组时对数据分组进行 0、1 交叉编号,而接收端只需要发回对收到分组的确认(ACK 或 NAK),无需要编号。如图 7 所示,编号引入后,发送方 FSM 有 4 个状态:0 号分组和 1 号分组各有 2 个状态;接收方 FSM 也有 2 种状态:分别是等待 0 号分组状态和等待 1 号分组状态。

下面分析当 ACK 或 NAK 可能受损时的状态转移。发送方的状态转移如下:(1)发送方等待发送 0/1 号分组状态,发送 0/1 号分组,迁移到 0/1 号分组等待确认状态;(2)发送方在 0/1 号分组等待确认状态,收到 0/1 号分组正确的肯定确

认, 迁移到 1/0 号分组等待发送状态; (3) 发送方在 0/1 号分组等待确认状态, 收到 0/1 号分组出

错的确认(ACK 或 NAK), 重发当前数据分组, 并维持在 0/1 号分组等待确认状态。

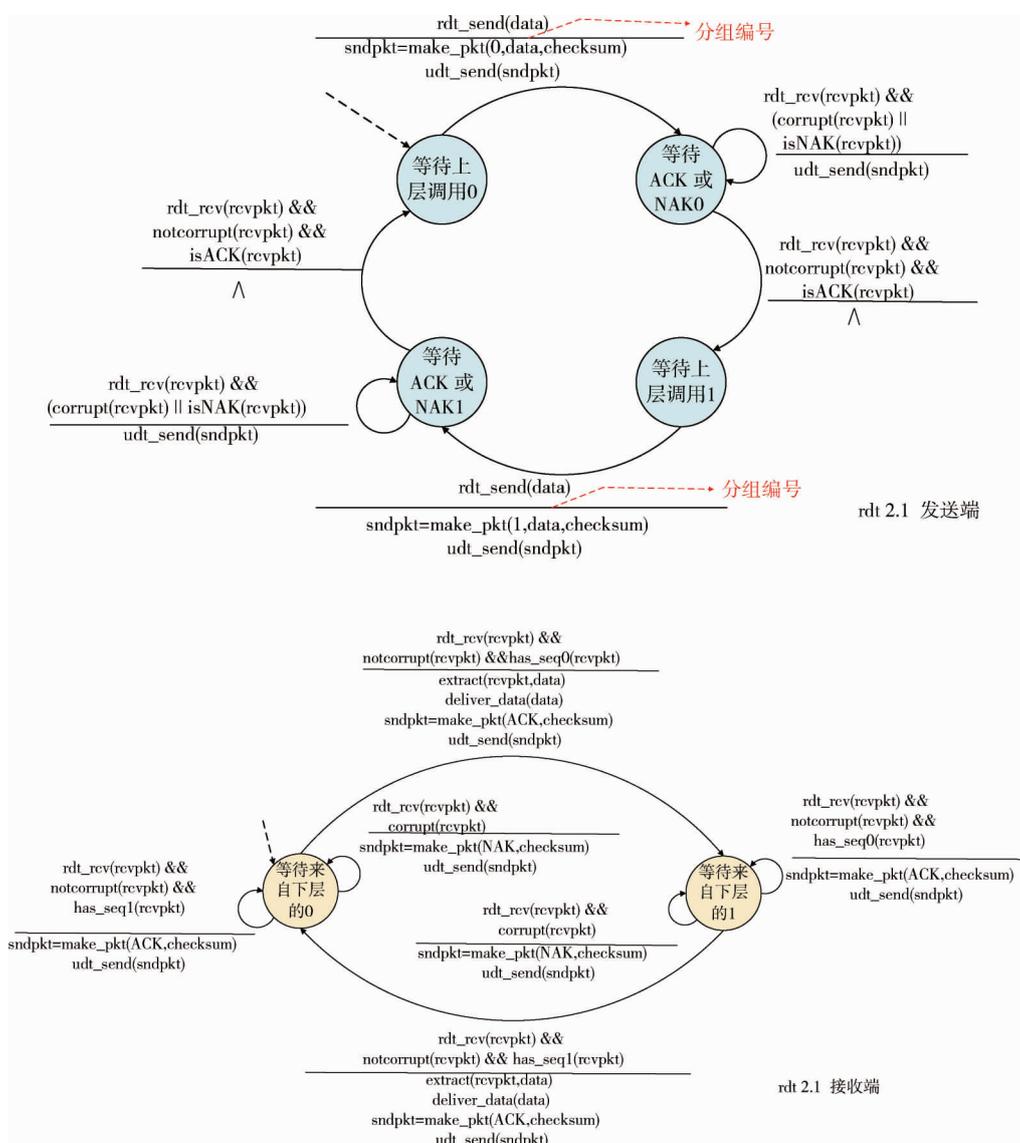


图 7 具有 bit 差错信道(数据分组和控制分组都可能受损)条件下的 rdt 2.1 FSM

由于 ACK 或 NAK 出错导致发送方重新发送数据, 接收方会收到重复的数据分组, 接收方的状态转换如下: (1) 接收方处于 0/1 号分组等待接收状态, 接收到正确的 1/0 号数据分组, 说明收到重复分组, 则拒绝分组, 并发送 ACK, 状态维持在 0/1 号分组等待接收状态; (2) 接收方处于 0/1 号分组等待接收状态, 接收到受损的数据分组, 发送 NAK, 状态维持在 0/1 号分组等待接收状态。

#### 4.4 只考虑肯定确认 ACK 的 rdt2.2 协议

在 rdt2.1 中, 接收方若收到的数据分组正确,

发送肯定确认 ACK; 若收到的数据分组出错, 则发送否定确认 NAK。对这种反馈机制做“减法”操作: 接收方不发 NAK, 只发送 ACK 达到与发送 NAK 相同的效果。这时接收方需要解决当收到错误的的数据分组时如何发送 ACK, 使得发送方能够得知数据分组传输出错的问题。

处理这个问题的方法是: 对确认报文引入编号, 当接收方收到出错的数据分组时, 丢弃分组并发送对上一个正确分组的肯定确认, 发送方收到非期待的确认编号分组, 即可判断接收数据出错。Rdt2.2FSM 如图 8 所示。

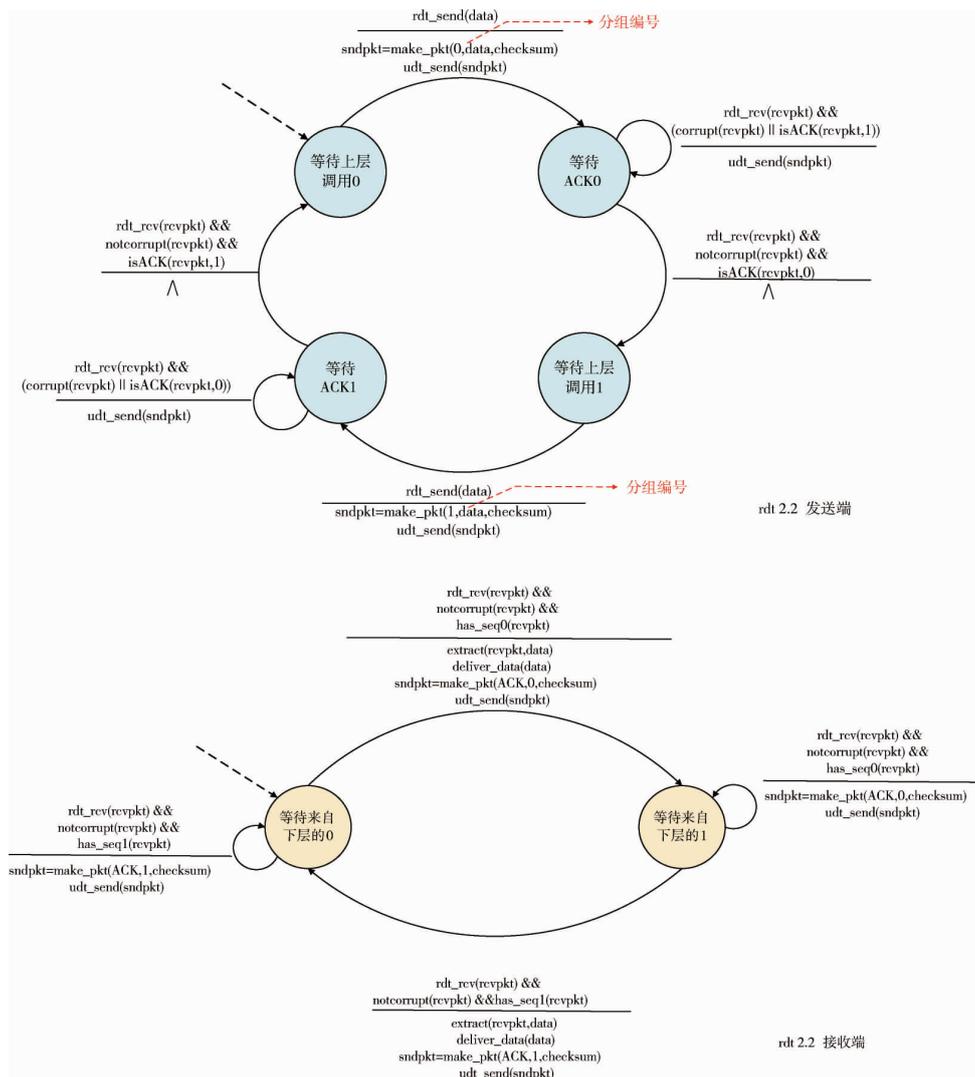


图 8 只发送 ACK 的 rdt 2.2 FSM

在 rdt2.2FSM 中,状态数与 rdt2.1 一致,但是每个状态的事件以及动作有所不同。接收方的事件及状态迁移如下:(1)接收方处于 0/1 号分组等待接收状态,接收到正确的 0/1 号分组,收下数据,交付上层,发送 ACK0/ACK1,状态迁移到 1/0 号分组等待接收状态;(2)接收方处于 0/1 号分组等待接收状态,接收到出错的分组或者接收到正确的 1/0 号数据分组,后者说明收到重复分组。两种情况均拒绝分组,并发送 ACK1/ACK0,状态维持在 0/1 号分组等待接收状态。

在发送方,需要处理对重复分组的 ACK 事件,事件及状态迁移如下:(1)发送方等待发送 0/1 号分组状态,发送 0/1 号分组,迁移到 0/1 号分组等待确认状态;(2)发送方在 0/1 号分组等待确认状态,收到 0/1 号分组的肯定确认 ACK0/ACK1,迁移到 1/0 号分组等待发送状态;(3)发送方在 0/1 号

分组肯定确认状态,收到确认出错或者对 1/0 号分组的肯定确认,后者说明接收方收到出错的数据分组或重复的数据分组。发送方重发当前数据分组,并维持在 0/1 号分组等待确认状态。

#### 4.5 经具有 bit 差错的丢包信道上的 rdt 3.0 协议

继续对 rdt2.2 做“加法”操作,将现实网络中存在的丢包现象加入到信道中。当信道出现丢包时,协议需要考虑:(1)如何检测丢包;(2)如何处理丢包。

无论是数据分组丢包还是控制分组丢包,发送端都收不到回复。处理丢包的方法是发送端发送完数据并等待“足够长”时间后未收到回复,则自动判断发生丢包,并重传数据。如果由于某些原因分组在信道上被延迟,数据分组和确认分组都没有丢失,发送端在“足够长”时间内没有收到

ACK,进行了重传操作,造成接收端收到重复的数据分组,发送端收到重复的ACK,这时可以用之前提到的分组序号解决。

发送端重传机制,需要一个倒计时器,倒计时到零时,中断发送端。发送端具有以下功能:(1)每发送一个分组(包括重传分组),便启动定时器;(2)响应定时器中断;(3)终止定时器。

rdt 3.0 协议是一个在不可靠信道上实现可靠数据传输的协议了,达到了一开始的设计要求。在 rdt 协议的设计中,通过对基于理想信道设计的 rdt1.0 不断地做“加法”操作,逐渐引入不可靠因素,把可靠数据传输协议的主体结构剖析清楚、层层递进,将一个原本复杂的协议设计过程针对性地展示分析问题、解决问题的思路。

通过以上讨论分析,总结可靠数据传输需要解决的问题。对应的解决机制如表 1 所示。

表 1 实现可靠数据传输的机制及作用

机制	作用
校验和	检测一个传输分组是否发生了 bit 错误
定时器	实现分组的超时重传
序号	对数据分组的编号,接收端可以判断分组是否失序,是否重复
确认 ACK/NAK	接收端通过确认分组通知发送端数据分组是否正确接收

## 5 结语

系统思想博大精深,可以指导我们解决复杂的系统问题。受系统思想的启发,结合在高校从

教 20 余年的教学经验,针对适用于问题牵引、启发式教学的知识内容,本文提出了一种基于系统思想的“加法”思维与“减法”思维相结合的教学设计方法,引导学生在学习过程中,通过“减法”抓住问题核心并找到解决问题的思路,通过“加法”深刻理解问题形成的复杂性,并将该方法用于工科的教学研究中。该方法为理想条件和现实状态之间逐步过渡提供了一种较好的解决方法,不仅有利于提升学习效率,更有助于培养分析问题、解决问题的思维素养。

## 参考文献:

- [1] 汪应洛.系统工程[M].第5版.北京:机械工业出版社,2015.
- [2] 谭跃进,陈英武,罗鹏程,等.系统工程原理[M].第2版.北京:科学出版社,2017.
- [3] 闫旭辉.基于系统思想方法论创造性地解决组织复杂性问题的研究[J].科技管理研究,2008(7):461-464.
- [4] 王海荣,谢章莲.浅析系统思想在高等教育管理中的应用[J].内蒙古农业大学学报(社会科学版),2007(5):248-252.
- [5] 谭跃进,郭波.钱学森系统工程创新教育思想的实践及启示——纪念钱学森发表《组织管理的技术——系统工程》四十周年[J].高等教育研究学报,2019(3):13-17.
- [6] James F Kurose, Keith W Ross.计算机网络-自顶向下的方法[M].第7版.陈鸣,译.北京:机械工业出版社,2018.
- [7] 谢钧,谢希仁.计算机网络教程[M].第5版.北京:人民邮电出版社,2018.

# Exploration and Practice of “Additive” and “Subtractive” Thinking Methods in the Teaching of Computer Network Course

ZHU Lei, ZHOU Lei, WU Xinrong, LUO Guoming

(Institute of Communication Engineering, The Army Engineering University of PLA, Nanjing 210007, China)

**Abstract:** Systematic thought is a thinking method of the entirety, interaction and evolution of things. It is often used to deal with large and complex problems. Based on the systematic thought, this paper proposes an “additive” and “subtractive” thinking and analyzing methods. Considering the difficult problem as a whole, it grasps the key of the problem using “subtractive” thinking method and gives the solution. Then it improves the solution using “additive” thinking method by considering the other factors and their interaction. In this way, complex problems can be fully understood and solutions to the problems will be found.

**Key words:** systematic thought, complex problem, additive thinking, subtractive thinking

(责任校对 蒋云霞)