



(12) 发明专利

(10) 授权公告号 CN 102067548 B

(45) 授权公告日 2014. 06. 18

(21) 申请号 200980123613. 1

H04L 12/24(2006. 01)

(22) 申请日 2009. 05. 08

(56) 对比文件

(30) 优先权数据

12/117, 139 2008. 05. 08 US

CN 1586055 A, 2005. 02. 23, 说明书第 1 页第 30-32 行、第 2 页第 2-6、25-26 行、第 6 页第 30-32 行、第 7 页第 1-11 行以及第 13 页第 1-12 行.

(85) PCT国际申请进入国家阶段日

2010. 12. 23

US 20070028001 A1, 2007. 02. 01, 全文.

(86) PCT国际申请的申请数据

PCT/CA2009/000630 2009. 05. 08

审查员 张小倩

(87) PCT国际申请的公布数据

W02009/135309 EN 2009. 11. 12

(73) 专利权人 黑莓有限公司

地址 加拿大安大略省沃特卢市

(72) 发明人 N·米斯特里 W·丁 姚忠辉

(74) 专利代理机构 中科专利商标代理有限责任

公司 11021

代理人 王波波

(51) Int. Cl.

H04L 29/06(2006. 01)

H04L 29/08(2006. 01)

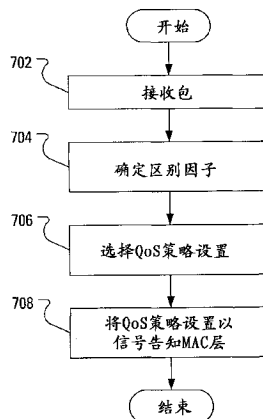
权利要求书2页 说明书7页 附图7页

(54) 发明名称

用于应用的网络感知适配器

(57) 摘要

来自各种应用的数据业务流可由作为用于过滤的更大平台的一部分的网络感知适配器处理, 该网络感知适配器将流传递给智能网络部件。与给定的数据流相关联的区别因子可以被确定, 并且用作为该给定数据流选择 QoS 策略设置的基础。然后, 可以将所选择的 QoS 策略设置以信号告知媒介访问控制层, 以利用有效且最佳的 (从技术和商业观点都是) 的嵌入的下层有效媒介类型的 QoS 功能来执行该应用的通信。



1. 一种在数据流进行到媒介访问控制 MAC 层之前适配数据流的方法,所述 MAC 层已经被适配以接收 QoS 策略设置,所述方法包括:

接收包,其中所述包与数据流相关联;

确定与所述数据流相关联的区别因子,所述区别因子包括与所述数据流相关联的媒介类型的标识;

基于所述区别因子,为所述数据流选择 QoS 策略设置;以及

将所述 QoS 策略设置以信号告知与所述包相关联的所述 MAC 层,

其中,所述信号告知包括:

做一个所述包的克隆;

对所述克隆进行修改以形成修改后的克隆,所述修改后的克隆包含所选 QoS 策略设置的信号表示;并且

将所述克隆注入到所述 MAC 层中。

2. 如权利要求 1 所述的方法,其中所述 QoS 策略设置包括服务质量设置。

3. 如权利要求 1 所述的方法,其中所述 QoS 策略设置包括用户体验设置。

4. 如权利要求 1 所述的方法,其中所述数据流是多个因特网协议 IP 包的流,并且确定所述区别因子包括检查所述多个 IP 包中的给定包的头部域。

5. 如权利要求 4 所述的方法,其中所述头部域具有关于所述给定包的源地址的指示。

6. 如权利要求 4 所述的方法,其中所述头部域具有关于所述给定包的目的地地址的指示。

7. 如权利要求 4 所述的方法,其中所述头部域具有关于所述给定包的源端口的指示。

8. 如权利要求 4 所述的方法,其中所述头部域具有关于所述给定包的目的地端口的指示。

9. 如权利要求 1 所述的方法,其中所述区别因子包括与所述数据流相关联的应用的标识。

10. 如权利要求 9 所述的方法,其中所述应用与优先级相关联,并且为所述数据流选择所述 QoS 策略设置包括,基于所述优先级和所述 QoS 策略设置之间的关联进行选择。

11. 如权利要求 9 所述的方法,其中所述应用包括视频应用。

12. 如权利要求 9 所述的方法,其中所述应用包括语音应用。

13. 如权利要求 9 所述的方法,其中所述应用包括游戏应用。

14. 如权利要求 9 所述的方法,其中所述应用包括协作应用。

15. 如权利要求 9 所述的方法,其中所述应用包括即时消息应用。

16. 如权利要求 1 所述的方法,其中所述区别因子包括与所述数据流相关联的订户的标识。

17. 如权利要求 1 所述的方法,其中所述媒介类型包括基于 IEEE802.3 标准的有线连网技术。

18. 如权利要求 1 所述的方法,其中所述媒介类型包括基于 IEEE802.11 标准的无线连网技术。

19. 如权利要求 1 所述的方法,其中所述媒介类型包括基于 IEEE802.16 标准的有线连网技术。

20. 如权利要求 1 所述的方法,其中所述媒介类型包括基于源自 3GPP 长期演进项目的标准的无线连网技术。

21. 一种在数据流进行到媒介访问控制 MAC 层之前适配数据流的设备,所述 MAC 层已经被适配以接收 QoS 策略设置,所述设备包括:

用于接收包的装置,其中所述包与数据流相关;

用于确定与所述数据流相关联的区别因子的装置,所述区别因子包括与所述数据流相关联的媒介类型的标识;

用于基于所述区别因子,为所述数据流选择 QoS 策略设置的装置;以及

用于将所述 QoS 策略设置以信号告知与所述包相关联的所述 MAC 层的装置;

其中,所述用于信号告知的装置还包括:

做一个所述包的克隆的装置;

对所述克隆进行修改以形成修改后的克隆的装置,所述修改后的克隆包含所选 QoS 策略设置的信号表示;以及

将所述克隆注入到所述 MAC 层中的装置。

用于应用的网络感知适配器

技术领域

[0001] 本申请一般涉及处理由应用产生的业务流,更具体地,涉及用于将流传递给智能网络部件的网络感知适配器。

背景技术

[0002] 在联网设备的应用层执行的应用传统上通过简单的媒介访问控制 (MAC) 层在该联网设备所连接到的网络上通信。此外,因为对于不同的连网协议 (networking protocol), MAC 层不同 (即,不同的传输媒介),应用被开发时,要考虑到设备连接到的网络中使用的特定连网协议对应的 MAC 层的特性。

[0003] 与较新的连网协议相关的 MAC 层增加了允许应用指定由应用产生的特定数据业务流所需要的服务质量 (QoS) 的复杂性。简单的 MAC 层常常缺少允许应用为特定数据业务流指定所需的 QoS 的复杂性。因此,大多数的应用没有被提供,或没有被要求具有,使应用充分利用网络技术下层传输所必需的一定程度的智能。

[0004] 由于不具有一定程度的智能,这样的应用不能充分利用嵌入在一些新的连网技术 (例如 WiMAX) 中的 QoS 功能的优势,也不能处理这类新的连网技术的一些独特的技术挑战。不幸的是,当在新的应用中建立利用新的连网技术的优势所必需的一定程度的智能时,为执行新的应用所设计的设备的复杂度增加了。因此,所增加的复杂度可能抬高设备的费用,并且可能对专用集成电路技术的物理限度造成困难。

发明内容

[0005] 数据业务流可以由作为用于过滤的更大平台的一部分的网络感知适配器处理,该网络感知适配器将流传递给智能网络部件。与给定的数据流相关联的区别因子 (distinguishing factor) 可以被确定,并且用作为该给定数据流选择 QoS 策略设置的基础。然后,将所选择的 QoS 策略设置以信号告知媒介访问控制层。

[0006] 依照本发明的一个方面,提供了一种在数据流进行到媒介访问控制 (MAC) 层之前适配 (adapt) 数据流的方法, MAC 已经被适配以接收 QoS 策略设置。该方法包括:接收包 (packet), 其中该包与数据流相关;确定与数据流相关联的区别因子;基于该区别因子,为数据流选择 QoS 策略设置;以及将 QoS 策略设置以信号告知与该包相关联的 MAC 层。在本发明的其他方面,提供用于实现此方法的计算设备,以及提供用于使处理器适合于实现此方法的计算机可读介质。

[0007] 通过结合附图回顾以下对本发明的特定实施例的描述,本发明的其他方面和特征对于本领域技术人员将会容易明白。

附图说明

[0008] 现在参考附图,附图通过举例的方式示出了本发明的实施例,其中:

[0009] 图 1 示出了可以连接到数据通信网络的计算设备的部件的互相连接;

[0010] 图 2 示出了一种使用 Windows 管理工具 (Windows ManagementInstrumentation) 的通信的逻辑结构;

[0011] 图 3 示出了一种使用 Windows 过滤平台 (Windows FilteringPlatform) 的通信的逻辑结构;

[0012] 图 4 示出了依照本发明的一些方面,包含网络感知适配器、过滤器引擎和 QoS 管理器的 Windows 过滤平台。

[0013] 图 5 示出了依照本发明的一个方面,图 4 中的过滤器引擎的示例操作方法中的步骤。

[0014] 图 6 示出了依照本发明的一个方面,图 4 中的 QoS 管理器的示例操作方法中的步骤;以及

[0015] 图 7 示出了依照本发明的一个方面,图 4 中的网络感知适配器的示例操作方法中的步骤。

具体实施方式

[0016] 图 1 示出了可以以有线方式或无线方式连接到数据通信网络 108 的计算设备 100 的部件的互相连接。被示出的计算设备 100 的部件包括:微处理器 116,和连接到微处理器 116 的存储设备 120、随机存取存储器 (RAM) 118 和网络接口卡 (NIC) 104。NIC 104 允许计算设备 100 连接到数据通信网络 108。

[0017] 微处理器 116 在存储的程序的控制下工作,代码被存储在存储设备 120 中。如图 1 中所描述的,在工作时,RAM 118 存储包括操作系统程序或代码模块 136 的程序,例如已知的微软 Windows™ 操作系统。诸如 Windows 的操作系统一般将 RAM 118 空间分为两部分,即用户空间 140 和受限访问空间,后者诸如是内核空间 138 或它的功能等价物。RAM 118 进一步地存储通常以标号 142 标识的、一般驻留在用户空间 140 中的软件应用,和一般驻留在内核空间 138 中的驱动程序 144。

[0018] 图 2 示出了使用 Windows 管理工具 (WMI) 的通信的逻辑结构,其中通信是在应用 202 和 NIC 104 之间进行。WMI 是提供操作系统接口的已知的 Windows 驱动程序模型 (Windows Driver Model) 的扩展集,装置化的部件通过这些接口提供信息和通知。WMI 是源自分布式管理任务组 (DMTF) 的基于 Web 的企业管理 (WBEM) 标准和公共信息模型 (CIM) 标准的一种微软的实现。

[0019] NIC 104 需要选自图 1 的驱动程序 144 的驱动程序 206。驱动程序 206,包括 MAC 层,与内核空间中的网络驱动程序接口规范 (NDIS) 驱动程序 208 通信。NDIS 驱动程序 208 实现适用于由微软和 3Com 公司共同开发的 NIC 的 API。NDIS 是逻辑链路控制 (LLC),它形成了 OSI 数据链路层 (7 层中的第 2 层) 的上子层 (upper sublayer),并且充当 OSI 数据链路层和网络层 (7 层中的第 3 层) 之间的接口。下子层 (lower sublayer) 是 MAC 设备驱动程序。NDIS 是一个函数库,常被称为“封套 (wrapper)”,其隐藏了下层的 NIC 104 的硬件的复杂性,并且充当层 3 网络协议驱动程序和硬件级 MAC 驱动程序的标准接口。

[0020] NDIS 驱动程序 208 与也在内核空间中的 WMI 模块 210 通信。模块 WMI 210 与用户空间中的 WMI API 212 通信。WMI API 212 与应用 202 通信。

[0021] 图 3 示出了使用 Windows 过滤平台的通信的逻辑结构,其中,该通信是在应用 302

和 NIC 104 之间进行。

[0022] 和图 2 示出的使用 WMI 的结构一样,如图 3 示出的使用 WFP 的结构包括与驱动程序 206 通信的 NIC 104。驱动程序 206 又与 NDIS 驱动程序 208 通信。与图 2 的 WMI 情形不同,NDIS 驱动程序 208 与 Windows 过滤平台 (WFP) 310 通信。WFP 310 跨内核空间 138 和用户空间 140,并与用户空间 140 中的 WFP API 312 通信。WFP API 312 与应用 302 通信。

[0023] 微软 Windows 操作系统的 Vista™ 版本中包括一种 WFP API。该 WFP API 允许应用捆绑到 (tie into) Windows Vista 和 WindowServer 2008 中新的网络堆栈的包处理和过滤流水线中。WFP 提供包含集成通信的特性。此外,WFP 可被配置为以每一应用为基础来调用处理逻辑。

[0024] 当计算设备 100 连接到的网络是传输通信协议 (TCP) 和因特网协议 (IP) 网络时,网络堆栈 440 可被称为 TCP/IP 堆栈。

[0025] 根据图 4, WFP 310 可被实现为:过滤引擎、网络堆栈 440、和一组调出 (callout) 模块。网络堆栈 440 可包括多个“垫片 (shim)”。垫片将包的内部结构作为属性揭示出来。对不同的层的协议,存在不同的垫片。在工作时,过滤引擎将接收到的包中的数据与指定的一组规则相比较。默认认为 WFP 310 包含一组内置的垫片。此外,其他协议的垫片可利用 WFP API 312 向过滤引擎注册。这组内置的垫片包括:应用层执行 (Application Layer Enforcement, ALE) 垫片 404;流垫片 408;传输层模块 (TLM) 垫片 410;以及网络层模块 (NLM) 垫片 414。图 4 中示出的已经注册的垫片包括:传输驱动程序接口 Winsock 内核 (TDI WSK) 垫片 406;转发层垫片 412;以及 NDIS 层垫片 416。

[0026] 提供基本过滤能力的过滤引擎跨内核空间 138 和用户空间 140 两者。如图 4 所示出的那样,过滤引擎位于内核空间 138 中的部分被称为“过滤器引擎”402,过滤引擎位于用户空间 140 中的部分被称为“基础过滤引擎”418。

[0027] 过滤引擎将给定的包中的数据与过滤规则相比较,其中该包中的数据已经被垫片所揭示。基于数据与一个或多个规则之间的匹配,过滤引擎可以要么允许该给定的包通过,要么阻止该给定的包通过。如果另外的操作有必要,可通过使用调出模块来执行该另外的操作。以每一应用为基础来运用过滤规则。

[0028] 基础过滤引擎 418 是管理过滤引擎的模块。基础过滤引擎 418 接受过滤规则并强制执行特定于应用的安全模型。基础过滤引擎 418 还为 WFP 310 维护统计数据,并维护 WFP 310 的状态日志。

[0029] 调出模块是由过滤驱动程序揭示的回调 (callback) 函数。过滤驱动程序被用于提供不同于默认过滤能力的过滤能力,其中在默认过滤能力中,要么阻止要么允许包通过。在过滤器规则的注册过程中,调出模块可被指定。当具有与过滤器规则相匹配的数据的包被接收到时,过滤器引擎 402 通过一组调出 API 420 来调用相关联的调出模块。该相关联的调出模块然后执行某个特定的过滤能力。

[0030] 调出模块 -- 其可以在所有的层被注册 -- 扩充了 WFP 310 的能力。每个调出模块有唯一的全局唯一标识符 (GUID)。调出模块可被用于深度检查 (Deep Inspection),也就是,执行对网络数据的复杂检查以确定哪些数据要被:阻止;允许;或传递给另一个过滤器。调出模块可被用于包修改 (Packet Modification),其可以包括对作为流的一部分被接收的包的头部或净荷的修改,以及将修改过的包注回流中。调出模块的其他用途包括流修

改 (Stream Modification) 和数据记录 (Data Logging)。

[0031] 图 4 中示出的示例调出模块包括:反病毒调出模块 422、家长控制 (parental control) 调出模块 424、入侵检测系统 (IDS) 调出模块 426、和网络地址转换 (NAT) 调出模块 428。

[0032] 在用户空间 140 中,基础过滤引擎 418 与 WFP API 312 通信,WFP API 312 又与诸如应用 302、基于操作系统的防火墙 450 和其他的应用 452 (其可以是进一步的防火墙) 等应用通信。WFP API 312 还与 QoS 管理器 432 通信,来实现本发明的一些方面。

[0033] 总的来看,QoS 管理器 432 配置网络感知适配器调出模块 430 来区分来自应用 302、基于 OS 的防火墙 450 和其他应用 452 的流的优先次序。如本领域的普通技术人员将会清楚的,图 4 所示出的应用 302、450、452 只不过是例子,并且可能有更多或更少或不同的应用为计算设备 100 (图 1) 所需。

[0034] 在执行上述区分优先次序时,网络感知适配器调出模块 430 与驱动程序 206 中的 MAC 层 (未示出) 交互。此外,网络感知适配器调出模块 430 与应用 302、450、452 以及与策略交互。如本领域普通人员将会清楚的,各种关于处理来自应用 302、450、452 的流的策略可由计算设备 100 从中央目录接收。

[0035] 适配器调出模块 430 的作用是智能地利用信令以通过使用策略和 / 或服务特征实现端到端的 QoS。网络感知适配器调出模块 430 能进一步:管理 QoS 和频谱效率的权衡 (trade off);管理终端操作响应和功率保留的权衡;以及管理其他可被嵌入的网络智能,诸如基于位置的服务 (LBS) 和组播广播服务 (MBS)。

[0036] 在操作中,过滤器引擎 402 被组织为一组过滤器层,每个层都有相关联的优先级。给定的流可能被阻止,即使有更高优先级的过滤器已经允许了该给定的流。过滤器的结构允许多个动作在同一个流上被执行。过滤器引擎 402 中的层被分为子层。在子层内,过滤器按权重顺序被评价 (evaluate)。在第一次匹配 (允许 / 阻止) 时,评价停止。当流和过滤器之间发生匹配时,有关的调出模块被通知。如果被通知的调出返回继续,则下一个匹配过滤器被评价。特别地,流在每个子层被评价。

[0037] 在操作中,关于即将从应用 302 到来的流,QoS 管理器 432 传输指令给过滤器引擎 402 来为网络感知适配器调出模块 430 初始化调出驱动程序。特别地,该指令可指定卸载 (unload) 函数。一旦接收到 (图 5 的步骤 502) 初始化调出驱动程序的指令,过滤器引擎 402 就可以创建设备对象,并注册该网络感知适配器调出模块 430。

[0038] QoS 管理器 432 的操作可参照图 6 进一步加以考虑。最初,QoS 管理器 432 向过滤器引擎 402 打开会话 (步骤 602)。QoS 管理器 432 然后指示过滤器引擎 402 给过滤器引擎 402 的结构增加子层 (步骤 604),并指示过滤器引擎 402 给刚增加的子层增加过滤器 (步骤 606)。

[0039] 响应于接收到步骤 606 中要增加过滤器的指令,过滤器引擎 402 可增加所要求的过滤器,并执行通知函数 (图 5 的步骤 504),其通常被称为“Notifyfn()”,来向网络感知适配器调出模块 430 表明增加了过滤器。

[0040] 当来自已经为其设立了过滤器的应用 302 的流的包到达网络堆栈 440 时,这些包由过滤器引擎 402 处理。

[0041] 特别地,过滤器引擎 402 可执行分类函数 (图 5 的步骤 506),其通常被称为

“Classifyfn()”，来向网络感知适配器调出模块 430 表明在网络堆栈 440 中来自自己已经为其设立了过滤器的应用 302 的流的包的到来。

[0042] 图 7 示出了在网络感知适配器调出模块 430 处适配流的示例方法的步骤。

[0043] 特别地，网络感知适配器调出模块 430 最初接收（步骤 702）流的包，并确定（步骤 704）与该流相关联的区别因子。随后，网络感知适配器调出模块 430 基于该区别因子为数据流选择（步骤 706）QoS 策略设置。最后，网络感知适配器调出模块 430 修改流数据来将所选的 QoS 策略设置以信号告知（步骤 708）驱动程序 206 中的 MAC 层（未示出）。

[0044] 在包级别，为了修改流数据来将所选的 QoS 策略设置以信号告知（步骤 708）MAC 层，网络感知适配器调出模块 430 可以为在步骤 702 中接收到的包做一个克隆。网络感知适配器调出模块 430 然后可以修改该克隆以形成一个修改过的克隆，该被修改过的克隆包含所选 QoS 策略的信号表示。网络感知适配器调出模块 430 然后可以用 Classifyfn() 将该克隆注入到网络堆栈 440 中。

[0045] 在这里采用术语“QoS 策略设置”是为了比已知的术语“服务质量”包含更多涵义。术语“服务质量”已知是包括针对比特率、延迟、抖动、包错误率的设置。除了这些设置之外，术语“QoS 策略设置”被用在这里来进一步包括用户优先级或用户组优先级，等等。

[0046] 区别因子可以是，例如，作为流的起源的应用 302 的标识、与流相关联的订户的标识、或流中数据的媒介类型的标识。

[0047] 当数据流是多个因特网协议（IP）包的流时，确定区别因子（步骤 704）可包括检查这多个 IP 包当中的给定包的头部域。在这种情况下，区别因子可以是，例如，源 IP 地址、目的地 IP 地址、源端口、或目的地端口。

[0048] 当区别因子是应用 302 的标识时，QoS 策略设置的选择（图 7 的步骤 706）可以基于与应用 302 相关联的优先级。另外，应用 302 的标识可以与应用的类型相关联，并且 QoS 策略设置的选择（图 7 的步骤 706）可以基于与应用 302 相关联的类型。示例的应用类型包括视频、语音、游戏、协作、和即时消息。

[0049] 区别因子可以是，例如，与作为流的起源的应用 302 相关联的订户的标识。

[0050] 可替换地，区别因子可以是，例如，网络接口卡 104 和数据通信网络 108（参见图 1）之间的链路的媒介类型的标识。媒介类型可被定义为，例如，由电气和电子工程师协会（IEEE）标准所定义的涉及 IEEE802.3 的有线以太网，诸如：10Mbit/s 以太网；快速以太网；G 比特以太网；或 10G 比特以太网。

[0051] 另外，媒介类型还可以被定义为，例如，“WiFi”，指的是由诸如 IEEE 802.11a、IEEE 802.11b、IEEE 802.11g 或 IEEE 802.11n 等电气和电子工程师协会（IEEE）标准所定义的无线网。

[0052] 此外，媒介类型可被定义为，例如，“WiMAX”（微波访问全球互通），指的是由 IEEE 802.16-2004 标准和 IEEE 802.16e-2004 修订所定义的无线网。

[0053] 媒介类型也可以被定义为，例如，“长期演进（LTE）”。长期演进是第三代合作伙伴项目（3GPP，参见 www.3gpp.org）中的一个项目的名称，用以完善通用移动通信系统（Universal Mobile Telecommunications System）移动电话标准（该标准也由 3GPP 持有），来应对将来的需求。

[0054] 当被过滤器引擎 402 和网络感知适配器调出模块 430 处理的流终止时，过滤器引

引擎 402 执行流删除函数（图 5 的步骤 508），通常被称为“Flowdeletefn()”，来向网络感知适配器调出模块 430 表明，流已经终止了。

[0055] 此外，QoS 管理器 432 可删除（图 6 的步骤 608）过滤器。

[0056] 响应于步骤 608 中过滤器的删除，过滤器引擎 402 可执行通知函数来向网络感知适配器调出模块 430 表明过滤器的删除。

[0057] 此外，QoS 管理器 432 还可删除（图 6 的步骤 610）子层，并关闭（步骤 612）在步骤 602 中打开的会话。响应于会话的关闭，过滤器引擎 402 可卸载（图 5 的步骤 510）调出驱动函数。

[0058] 在计算设备 100 工作时，用户可使用用户接口来选择要执行的应用。计算设备 100 可以是，例如，便携式（即，笔记本）计算机，而且所选的应用可以是视频传输应用。此外，网络接口卡 104 和数据通信网络 108 之间的连接可以是 WiMAX 连接。相应地，驱动程序 206 中的 MAC 层是 WiMAX MAC 层。已知 WiMAX MAC 层用于接受对 QoS 参数的指定。QoS 管理器 432 配置过滤器引擎 402 中的过滤器来向网络感知适配器 430 发送来自视频传输应用的流中的包。此外，一旦确定流中的数据是视频，网络感知适配器 430 就选择 200kbit/s 的带宽，并修改流中的包来以信号告知 WiMAX MAC 层，该流应该接收 200kbit/s 的带宽。

[0059] 在 WiMAX MAC 层，流中被修改过的包被接收，而且上述信令被处理。根据该处理，WiMAX MAC 层可通过数据通信网络 108 保留至视频流目的地的 200kbit/s 的带宽。WiMAX MAC 层然后可通过数据通信网络 108 来安排视频流向该目的地的传输。

[0060] 在可替换情况中，计算设备 100 的用户可指定 QoS 策略参数。这些参数可以是详细的，并且可以是常见的 QoS 参数，诸如比特率、延迟、抖动和丢包率，其可以被直接转换为嵌入在下层媒介类型中的 QoS。然而，并不是所有的媒介类型都支持所配置的所有或任意 QoS 参数。在这种情况下，网络感知适配器 430 可基于媒介类型进行映射。

[0061] 前述提供了一种适配数据流的方法，从而可以将所选的 QoS 策略设置以信号告知 MAC 层。然而，至今，还没有考虑如何处理当前网络条件不允许将 QoS 策略设置以信号告知的情况。在这种情况下，网络感知适配器 430 将尽最大努力来调度与 QoS 策略设置相关联的流中的业务。可替换地，尤其当 QoS 策略设置与全有或全无方式 (all-or-nothing approach) 相关联时，网络感知适配器 430 可以拒绝该流，并通知起源应用并随之通知用户。进一步可替换地，尤其当 QoS 策略设置与涉及“可接受降级 (degradation is acceptable)”方式的 QoS 策略相关联时，网络感知适配器 430 可用降级后的 QoS 设置来调度流中的业务。对某些网络媒介类型，它们的 QoS 状态（例如，可用带宽）可以被监控并反馈给网络感知适配器 430（通过 QoS 管理器 432）。

[0062] 有利地，当按照本发明的一个方面执行时，网络感知适配器 430 可被看作在应用层和 MAC 之间斡旋，从而减少了对在应用层执行的应用的复杂性的需要。网络感知适配器 430 可被安排为能够在 MAC 层实现对 QoS 参数的指定。当正确使用时，指定的 QoS 参数可用于提高频谱效率，并方便对数据通信网络 108 中固有的复杂性的管理。

[0063] 有利地，应用可按照与网络感知适配器 430 交互而不是与为特定网络技术特别设计的 MAC 层交互的预期来开发。

[0064] 本领域的普通技术人员应该清楚，尽管本发明的各方面是在已知的 Windows 过滤平台 (WFP) 情形下给出的，但是 WFP 的使用对于本发明各方面的工作并不是本质的，并且仅

仅充当本发明各方面可在其中实现的环境的一个示例。

[0065] 上述的本申请的实施例只是为了充当例子。对特定实施例的替换、修改和变化可由本领域技术人员不脱离本申请的范围来实现，该范围由这里所附的权利要求限定。

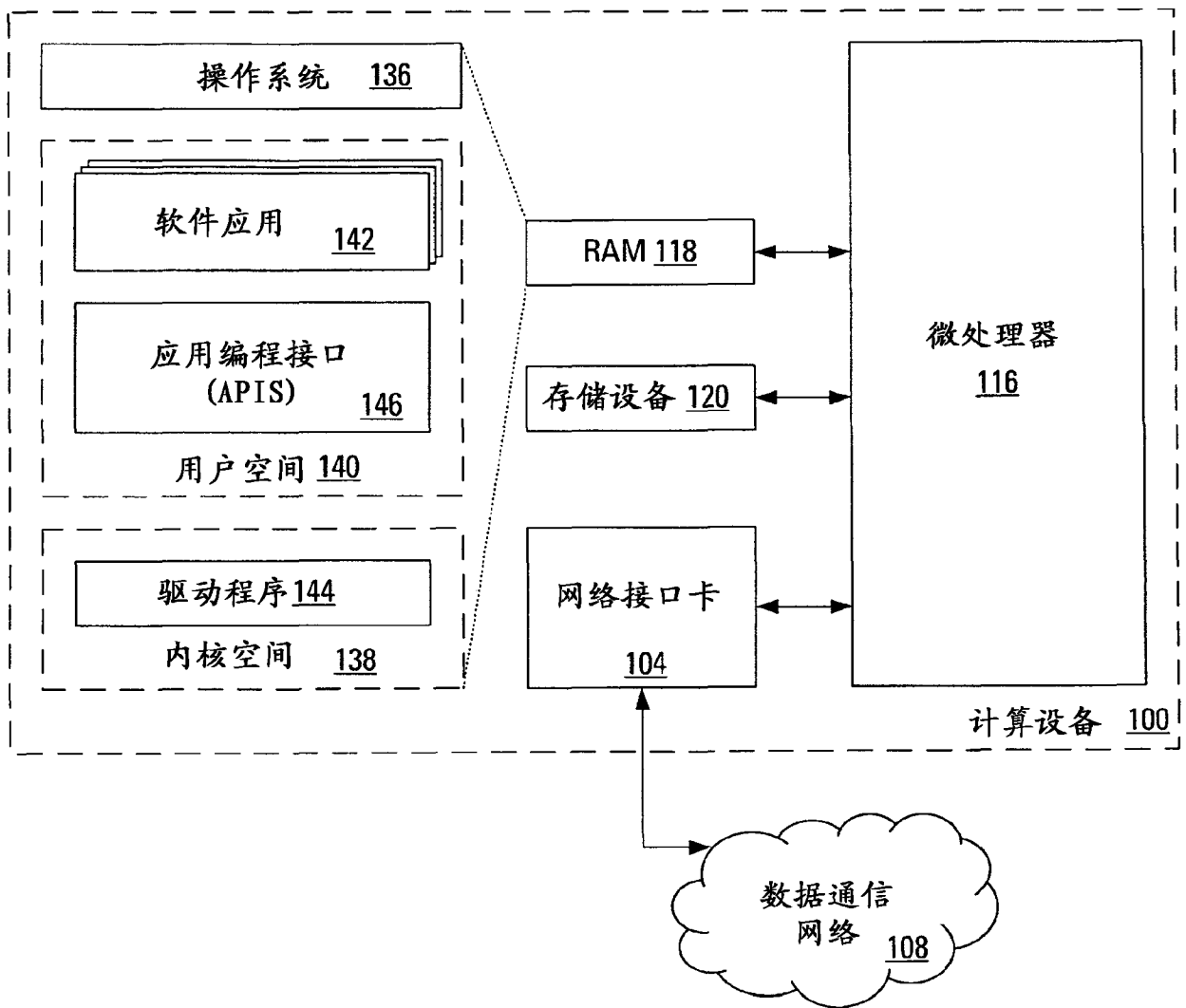


图 1

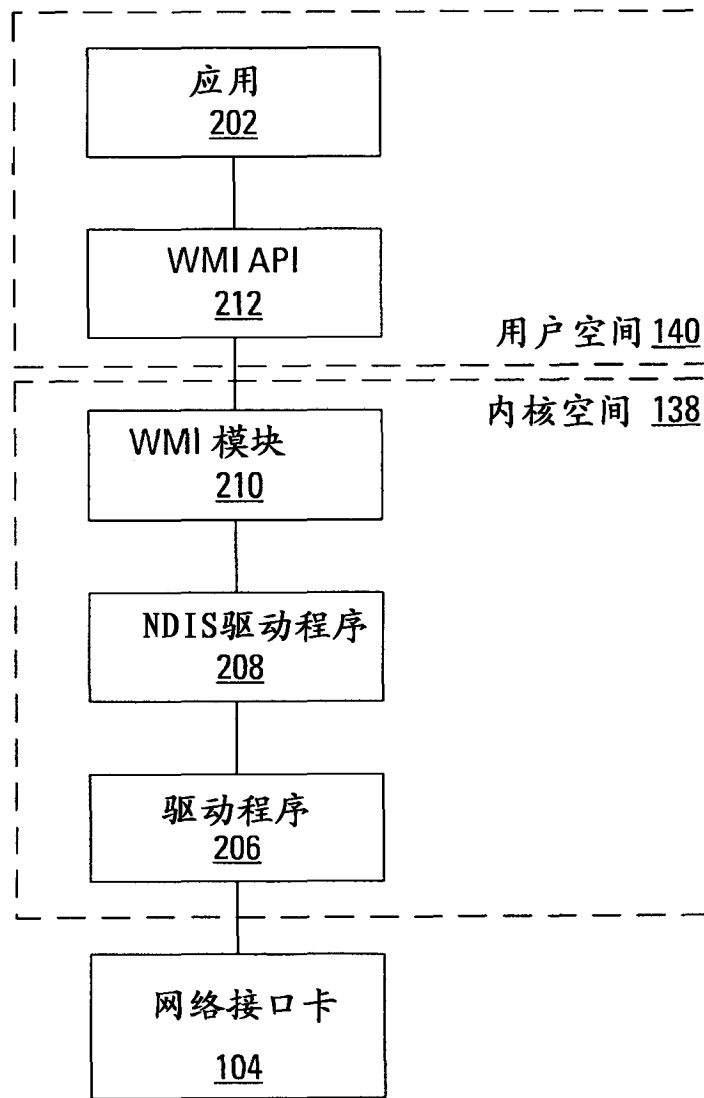


图 2

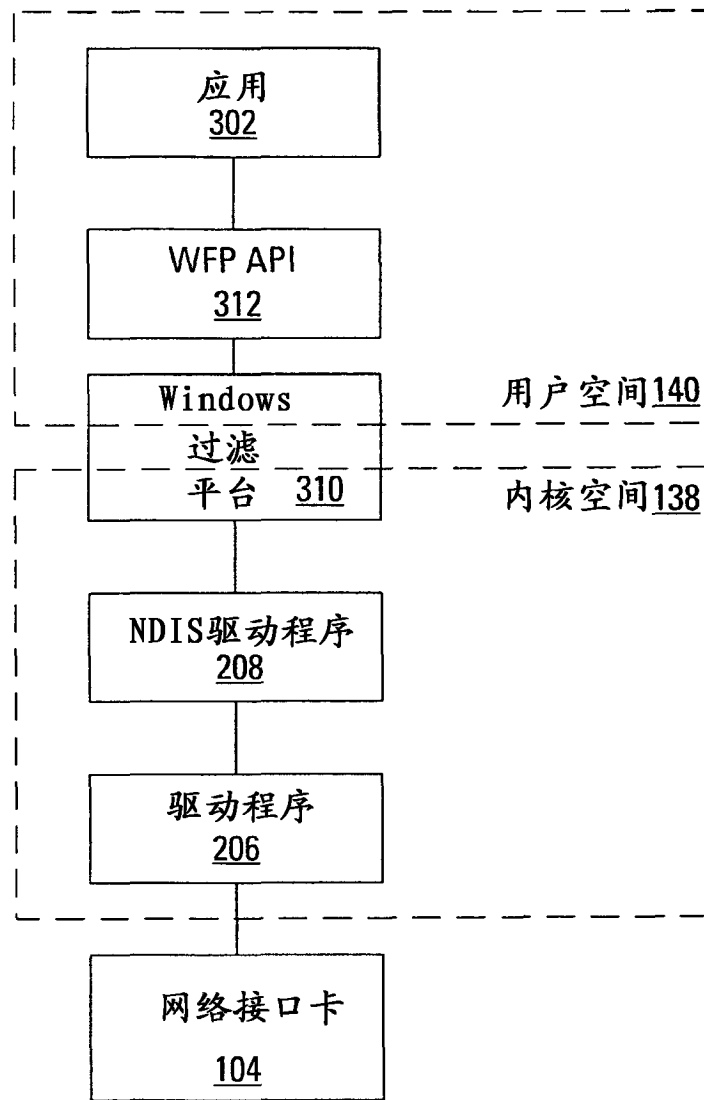


图 3

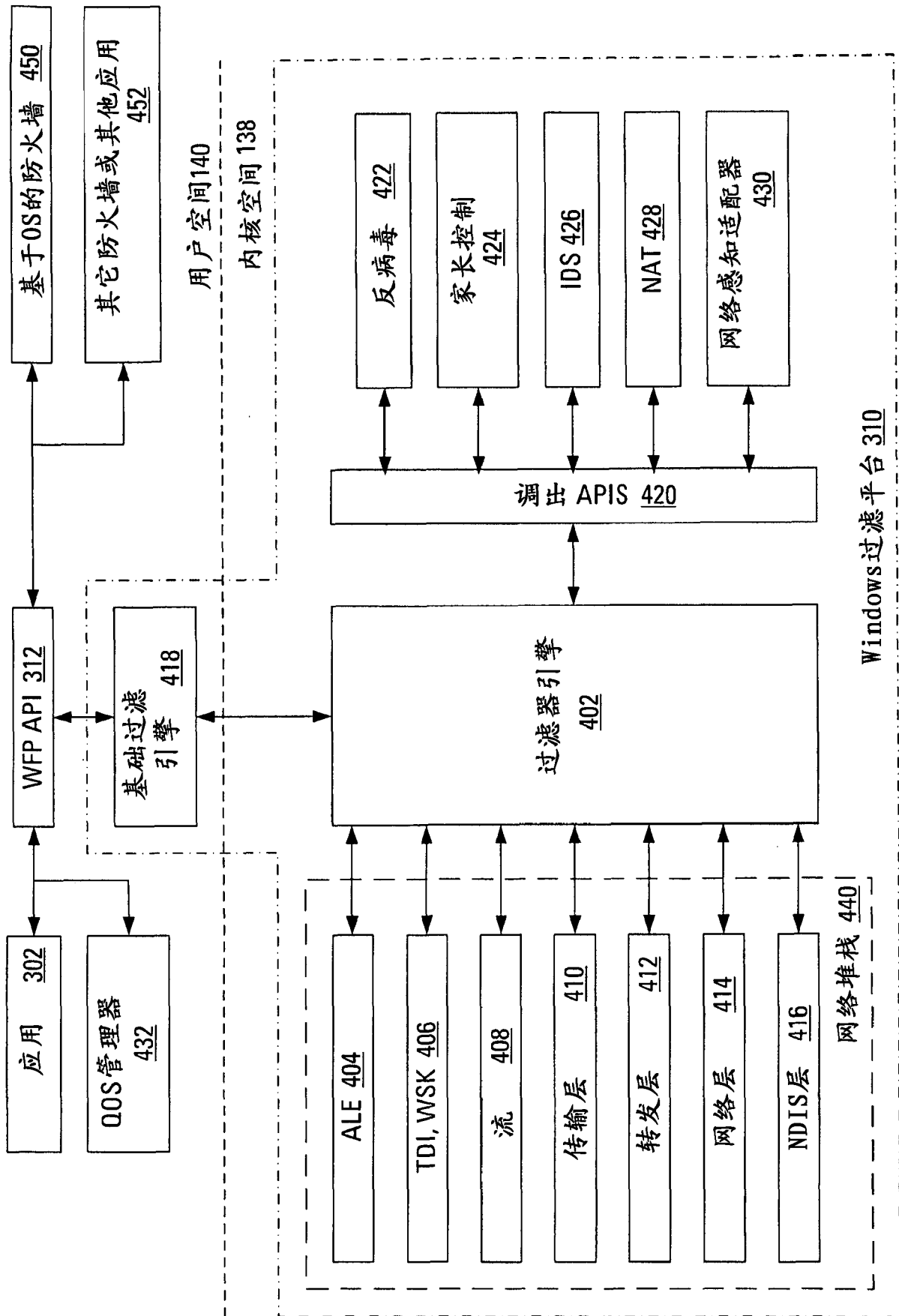


图 4

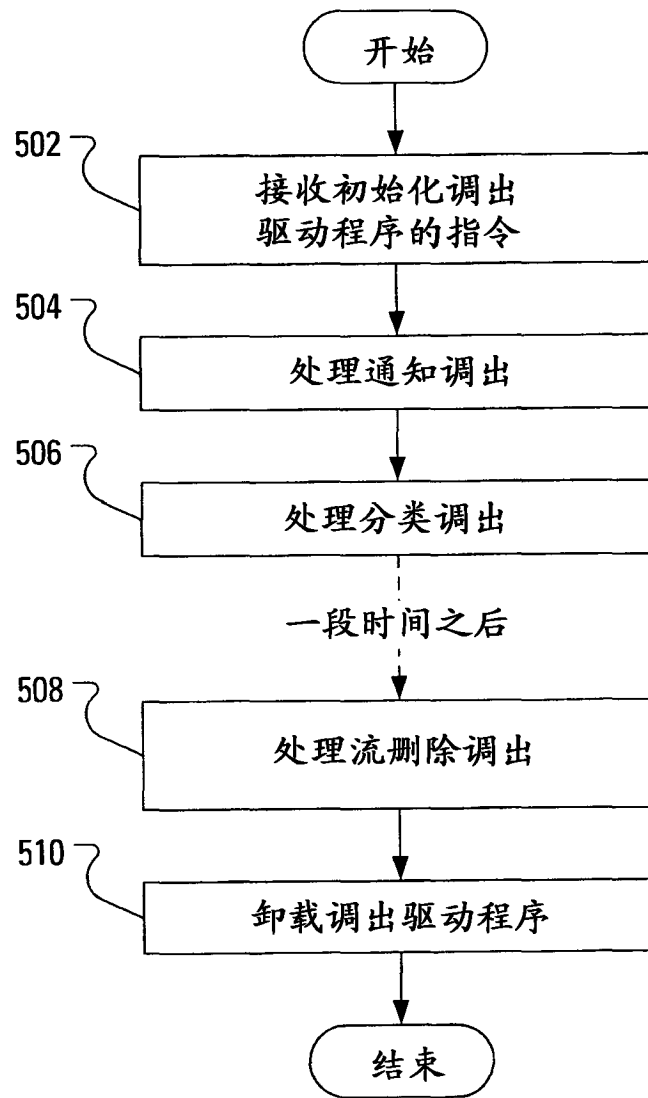


图 5

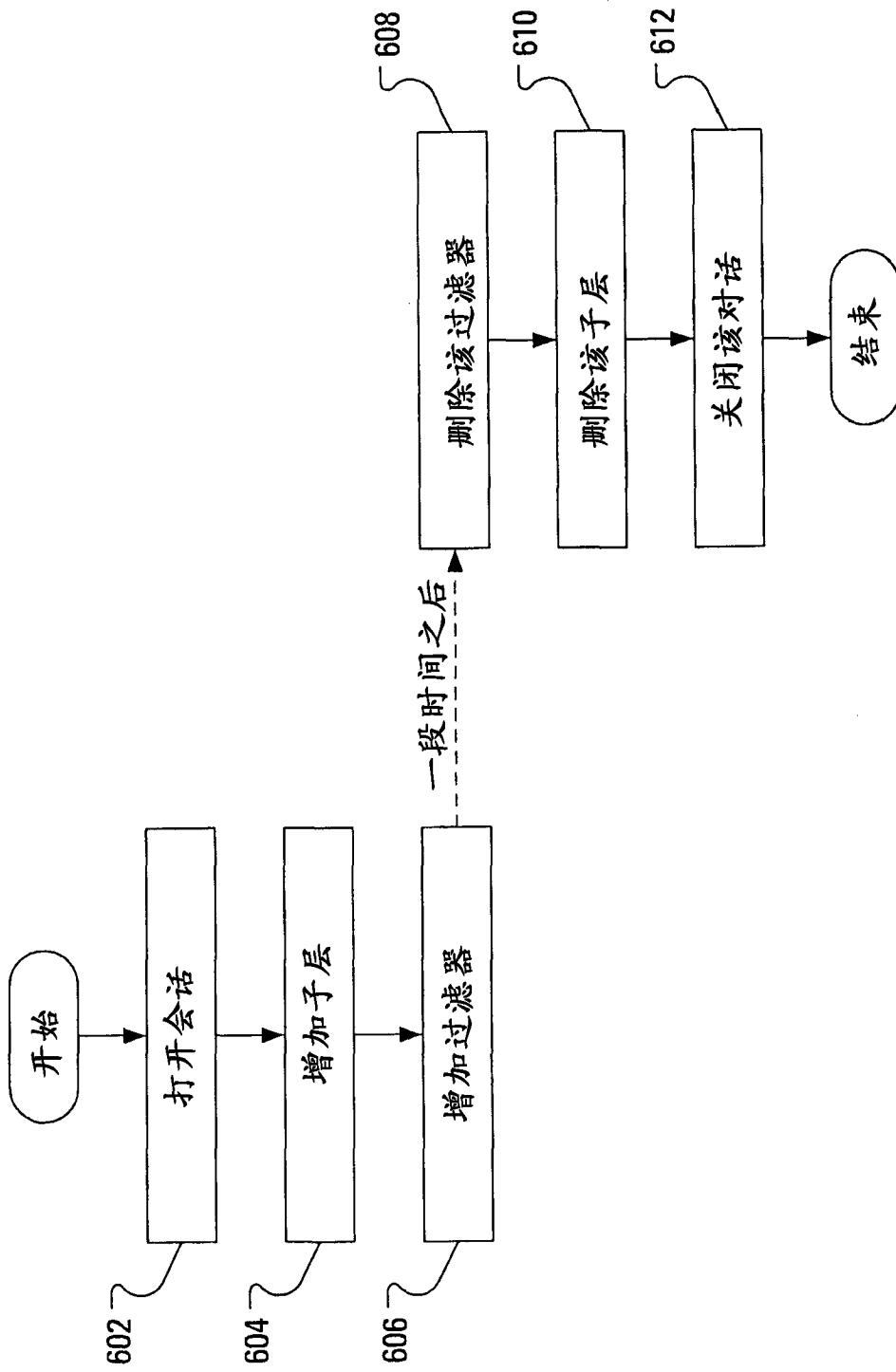


图 6

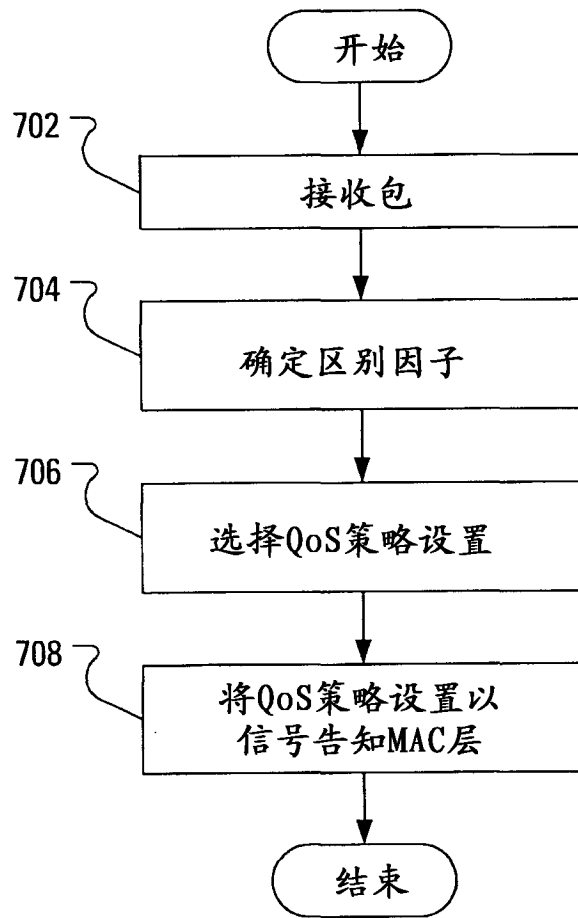


图 7