(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 115378843 B
(45)授权公告日 2023.12.22

(21)申请号 202210576647.3
(22)申请日 2016.07.21
(65)同一申请的已公布的文献号
申请公布号 CN 115378843 A
(43)申请公布日 2022.11.22
(30)优先权数据
62/195,488 2015.07.22 US
(62)分案原申请数据
201680051437.5 2016.07.21
(73)专利权人 动态网络服务股份有限公司
地址 美国新罕布什尔
(72)发明人 E·E·祖米杰沃斯基
T·L·泰星格尔 D·C·玛德利
(74)专利代理机构 中国贸促会专利商标事务所有限公司 11038
专利代理师 鲍进
(51)Int.Cl.
H04L 43/0823(2022.01)
H04L 43/0852(2022.01)
H04L 43/0864(2022.01)
H04L 61/4511(2022.01)
H04L 67/52(2022.01)
H04L 101/69(2022.01)
(56)对比文件
US 2003009594 A1,2003.01.09
US 9026145 B1,2015.05.05
US 2012131129 A1,2012.05.24
US 2008049776 A1,2008.02.28
US 2011282988 A1,2011.11.17
US 5675741 A,1997.10.07
审查员 包文韬

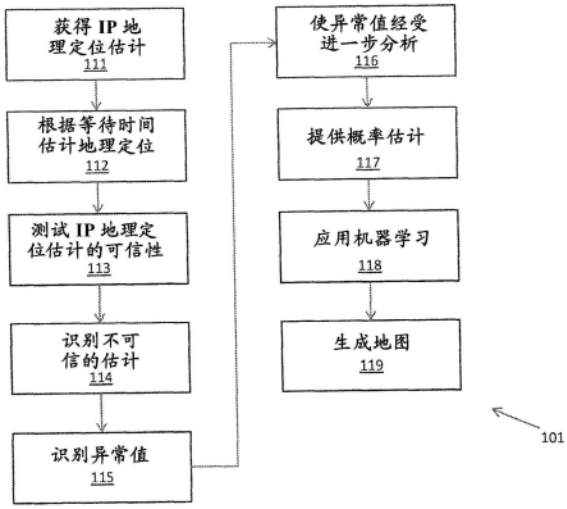
权利要求书3页 说明书32页 附图15页

(54)发明名称

使用跟踪路由进行地理定位的方法、系统和装置

(57)摘要

本公开涉及使用跟踪路由进行地理定位的方法、系统和装置。用于估计与特定互联网协议(IP)地址相关联的设备的地理定位(地理位置)的常规努力通常产生不幸的不准确结果。在许多情况下,估计的IP地理定位在错误的大洲上。本技术的实施例包括用于基于等待时间测量、域名服务器(DNS)信息和路由信息来识别和改善不正确估计的技术。例如,来自多个收集器的等待时间测量可以用于评估IP地理定位估计的可信性,并且在某些情况下,用于提高IP地理定位估计的准确性。DNS和路由信息可以用于证实估计的IP地理定位。结果产生的更准确的IP地理定位估计可以用于更有效地路由互联网流量、实施路由敏感信息的规则以及简化故障排除。



CN 115378843 B

1.一种包括指令的非瞬态计算机可读介质,所述指令在由一个或多个硬件处理器执行时使得执行操作,所述操作包括:

将数据分组传送到对应于第一设备的第一互联网协议(IP)地址;

在传送所述数据分组之后,接收指示所述数据分组未被递送的消息;

确定指示所述数据分组未被递送的消息是从与第一IP地址不同的第二IP地址发送的;

确定与第二IP地址对应的第二设备的地理位置;和

基于所述第二设备的地理位置,估计与第一IP地址对应的第一设备的地理位置。

2.根据权利要求1所述的非瞬态计算机可读介质,其中所述第一设备包括第一服务器,并且所述第二设备包括第二服务器。

3.根据权利要求1所述的非瞬态计算机可读介质,其中,将数据分组传送到第一IP地址包括:

选择与第一IP地址相关联的任意端口;和

将数据分组传送到任意端口。

4.根据权利要求1所述的非瞬态计算机可读介质,其中所述操作还包括:

通过以下操作来迭代地估计设备的地理位置:

(a)将第一数据分组传送到初始IP地址;

(b)从与初始IP地址不同的响应IP地址接收指示第一数据分组未被递送的消息;

(c)基于与响应IP地址相关联的第二设备的地理位置,估计与初始IP地址相关联的第一设备的地理位置;

(d)将第二数据分组传送到响应IP地址;和

(e)通过用响应IP地址迭代地替换初始IP地址,迭代地重复操作(a)到(d)。

5.根据权利要求1所述的非瞬态计算机可读介质,其中所述操作还包括:

基于相关IP地址集合中的每个IP地址识别所述相关IP地址集合,所述相关IP地址是以下中的至少一个:

(a)响应IP地址,基于将数据分组传送到包括在所述相关IP地址集合中的第一IP地址,从所述响应IP地址接收消息,以及

(b)初始IP地址,数据分组被发送到所述初始IP地址,使得从包括在所述相关IP地址集合中的第二IP地址接收消息。

6.根据权利要求5所述的非瞬态计算机可读介质,其中所述操作还包括:

向相关IP地址集合中的多个IP地址传送来自多个不同网络传感器的多个数据分组;

由所述多个不同网络传感器测量分别与所述多个IP地址相关联的多个等待时间;和

基于多个等待时间确定与相关IP地址集合相关联的特定服务器的地理位置。

7.根据权利要求5所述的非瞬态计算机可读介质,其中所述操作还包括:

反向解析相关IP地址集合中的多个IP地址;和

基于通过反向解析多个IP地址获得的位置信息来确定特定服务器的地理位置。

8.根据权利要求1所述的非瞬态计算机可读介质,其中,估计第一设备的地理位置包括确定:

第一设备和第二设备是同一服务器;和

第一IP地址和第二IP地址与同一服务器的不同接口相关联。

9.一种方法,包括:

将数据分组传送到对应于第一设备的第一互联网协议(IP)地址;

在传送所述数据分组之后,接收指示所述数据分组未被递送的消息;

确定指示所述数据分组未被递送的消息是从与第一IP地址不同的第二IP地址发送的;

确定与第二IP地址对应的第二设备的地理位置;和

基于所述第二设备的地理位置,估计与第一IP地址对应的第一设备的地理位置。

10.根据权利要求9所述的方法,其中所述第一设备包括第一服务器,并且所述第二设备包括第二服务器。

11.根据权利要求9所述的方法,其中,将数据分组传送到第一IP地址包括:

选择与第一IP地址相关联的任意端口;和

将数据分组传送到任意端口。

12.根据权利要求9所述的方法,还包括:

通过以下操作来迭代地估计设备的地理位置:

(a)将第一数据分组传送到初始IP地址;

(b)从与初始IP地址不同的响应IP地址接收指示第一数据分组未被递送的消息;

(c)基于与响应IP地址相关联的第二设备的地理位置,估计与初始IP地址相关联的第一设备的地理位置;

(d)将第二数据分组传送到响应IP地址;和

(e)通过用响应IP地址迭代地替换初始IP地址,迭代地重复操作(a)到(d)。

13.根据权利要求9所述的方法,还包括:

基于相关IP地址集合中的每个IP地址识别所述相关IP地址集合,所述相关IP地址是以下中的至少一个:

(a)响应IP地址,基于将数据分组传送到包括在所述相关IP地址集合中的第一IP地址,从所述响应IP地址接收消息,以及

(b)初始IP地址,数据分组被发送到所述初始IP地址,使得从包括在所述相关IP地址集合中的第二IP地址接收消息。

14.根据权利要求13所述的方法,还包括:

向相关IP地址集合中的多个IP地址传送来自多个不同网络传感器的多个数据分组;

由所述多个不同网络传感器测量分别与所述多个IP地址相关联的多个等待时间;和

基于多个等待时间确定与相关IP地址集合相关联的特定服务器的地理位置。

15.根据权利要求13所述的方法,还包括:

反向解析相关IP地址集合中的多个IP地址;和

基于通过反向解析多个IP地址获得的位置信息来确定特定服务器的地理位置。

16.根据权利要求9所述的方法,其中,估计第一设备的地理位置包括确定:

第一设备和第二设备是同一服务器;和

第一IP地址和第二IP地址与同一服务器的不同接口相关联。

17.一种系统,包括:

一个或多个处理器;以及

存储指令的存储器,所述指令在由一个或多个处理器执行时使得所述系统执行操作,

所述操作包括：

将数据分组传送到对应于第一设备的第一互联网协议(IP)地址；

在传送所述数据分组之后，接收指示所述数据分组未被递送的消息；

确定指示所述数据分组未被递送的消息是从与第一IP地址不同的第二IP地址发送的；

确定与第二IP地址对应的第二设备的地理位置；和

基于所述第二设备的地理位置，估计与第一IP地址对应的第一设备的地理位置。

18.根据权利要求17所述的系统，其中所述第一设备包括第一服务器，并且所述第二设备包括第二服务器。

19.根据权利要求17所述的系统，其中，将数据分组传送到第一IP地址包括：

选择与第一IP地址相关联的任意端口；和

将数据分组传送到任意端口。

20.根据权利要求17所述的系统，其中所述操作还包括：

通过以下操作来迭代地估计设备的地理位置：

(a)将第一数据分组传送到初始IP地址；

(b)从与初始IP地址不同的响应IP地址接收指示第一数据分组未被递送的消息；

(c)基于与响应IP地址相关联的第二设备的地理位置，估计与初始IP地址相关联的第一设备的地理位置；

(d)将第二数据分组传送到响应IP地址；和

(e)通过用响应IP地址迭代地替换初始IP地址，迭代地重复操作(a)到(d)。

使用跟踪路由进行地理定位的方法、系统和装置

[0001] 本申请是申请日为2016年7月21日、申请号为201680051437.5、发明名称为“使用跟踪路由进行地理定位的方法、系统和装置”的发明专利申请的分案申请。

[0002] 相关申请的交叉引用

[0003] 本申请根据35U.S.C.§119(e)要求于2015年7月22日提交的标题为“Methods,Systems,and Apparatus for Geographic Location Using Trace Routes”的美国申请No.62/195,488的优先权权益。上述申请通过引用被整体结合于此。

技术领域

[0004] 本发明公开涉及互联网技术领域,并且更具体地涉及使用跟踪路由进行地理定位的方法、系统和装置。

背景技术

[0005] 互联网协议(IP)地理定位(geographic location)或IP地理定位(geolocation)是推断或估计与特定IP地址相关联的设备的物理位置的实践。换句话说,IP地理定位是将IP地址固定到地球上具有期望特异性程度的位置的实践。用于估计或推断特定IP地址的地理定位的技术包括从(1)对应的互联网主机或本地网络节点的域名服务器(DNS)名称;(2)IP地址与跨已知地理定位分布的一组设备之间的等待时间测量结果;以及(3)部分IP到位置映射信息和边界网关协议(BGP)前缀信息的组合推断地理定位。关于这些技术的更多信息参见例如美国专利No.7,711,846,其通过引用被整体结合于此。

[0006] 不幸的是,IP地理定位估计往往不准确—并且有时非常不准确—因为它们基于对IP地址、路由协议和应用之间的逻辑关系而不是电缆、路由器、服务器、接入设备等之间的物理关系的观察。虽然逻辑关系通常与物理关系有关,但它们不一定联系在一起。例如,在互联网空间中彼此相邻的IP地址不一定在地理上彼此相邻,反之亦然:巴西和秘鲁在地理上彼此接壤,但在互联网空间中并不。此外,设备的物理位置的变化可能不一定对应于设备在互联网空间中的位置的变化,反之亦然。考虑经由BGP通告特定前缀的路由器。通过通告前缀,即使路由器在物理空间中移动,路由器也建立从逻辑的角度来看仍然固定的一个或多个逻辑互联网关系。

[0007] 此外,前缀不需要在一个地方。终端用户网络通常具有单个地理范围,但是基础设施IP地址(诸如在(包括路由器、交换机和防火墙的)广域网中使用的那些基础设施IP地址)可以分散在整个提供商的运营区域,其可以是全球范围的。因此,即使连续的基础设施IP地址作为单个前缀被路由到互联网的其余部分,它们可能在物理上位于远隔的城市,

[0008] 此外,用于推断或估计地理定位的网络信息可能不准确、不完整或者两者都有。前缀注册通常由最终用户自行报告,而无需由区域互联网注册机构进行有效性检查。DNS信息可能会引起误解;例如,与特定区域(例如,.uk)相关联的域不一定在那个区域中被托管。虽然互联网服务提供商通常在路由器接口名称中使用城市缩写,但命名约定因提供商而异,并不总是最新的。例如,路由器接口可以以它所附连到的光缆远端处的城市命名。类似地,

BGP信息可能是不确定的,特别是对于通告覆盖广泛地理区域(例如,大洲)的前缀的那些区域提供商。

[0009] 等待时间测量结果也可能是不精确的,这通常是由于人为增加测量时间而导致的延迟,这又导致对互联网节点之间的地理距离的增加的估计。这些延迟包括但不限于序列化延迟,即用于编码分组的时间;路由器处的排队延迟;以及等于总传播距离与传播速度的乘积(对于光纤中的光大约为200,000公里/秒)的传播延迟。如果通信介质(通常是光纤)在两点之间遵循曲折路径而不是直线路径,那么传播延迟将更高。实际上,许多光纤遵循沿着现有通行权的曲折路径。在其它情况下,由于地理约束(例如,丘陵和河流)、经济约束(例如,业主与互联网服务提供商之间缺乏商业关系)或两者兼而有之,光纤遵循曲折的路径。通常,等待时间越长,传播路径越有可能是迂回的并且可能导致对端点之间的距离人为增加的估计。

[0010] 不完整或不准确的网络信息和不精确的等待时间测量结果导致与IP地址的物理位置的估计相关联的不确定性程度随着地理定位估计的特异性程度而升高。例如,特定的IP地址的行星(地球)可以以非常高的置信度推断出来。当识别IP地址的大陆时,置信度往往下降。对于IP地址的国家,不确定性往往进一步增加,部分原因是每个国家的大小和边界不同。对大都市区/城市级别的IP地理定位的置信往往甚至更低,并且部分地取决于城市的位置和与其它城市的接近度。

发明内容

[0011] 发明人已经认识到,由于不正确的注册和/或DNS信息以及等待时间测量结果中的延迟,可用的IP地理定位数据往往不精确。此外,前缀可以重叠并且以复杂的方式相关,这使得精确的IP地理定位问题变得复杂。发明人还认识到,不精确的IP地理定位数据会对互联网流量管理和故障排除产生不利的影响。更具体而言,不精确的IP地理定位估计可能导致高DNS等待时间以及当流量实际上来自美国时,例如对欧洲资源的不准确的基于DNS的负载平衡。此外,不精确的IP地理定位估计可能导致关于网络问题的位置和原因的不准确结论,这又可能导致不正确、低效甚至徒劳的故障排除。

[0012] 本技术的实施例包括可以比其它IP地理定位技术更精确地实现的IP地理定位的方法和系统。一个示例包括定位可操作地耦合到互联网并具有IP地址的至少一个设备的方法。该方法包括从第三方自动获得基于设备的IP地址的设备的第一估计地理定位。它还包括从可操作地耦合到互联网的多个传感器中的每个传感器测量与到设备的IP地址的传输相关联的相应等待时间分布。(多个传感器中的每个传感器位于不同的地理定位)。处理器从测得的等待时间分布中选择至少一个等待时间,并且识别测量选定的等待时间的传感器。处理器基于等待时间估计从传感器到设备的最大可能地理距离,并将它与设备的第一估计地理定位与传感器的地理定位之间的距离进行比较。如果设备的第一估计地理定位不在距离传感器的地理定位的最大可能地理距离内,那么处理器基于最大可能地理距离和地理定位确定设备的第二估计地理定位。

[0013] 其它实施例包括估计可操作地耦合到互联网并具有IP地址的至少一个设备的地理定位的另一个种方法。该方法包括基于设备的IP地址从第一方自动获得设备的第一估计地理定位并且基于设备的IP地址从第二方自动获得设备的第二估计地理定位。处理器确定

第一估计地理定位和第二估计地理定位之间的距离。如果距离超过预定阈值，那么处理器从可操作地耦合到互联网的多个传感器中的每个传感器测量与到设备的IP地址的传输相关联的相应等待时间。(多个传感器中的每个传感器处于不同的地理定位)。处理器从相应等待时间中选择至少一个等待时间、识别测量选定的等待时间的传感器、并且至少部分地基于选定的等待时间估计从传感器到设备的最大可能地理距离。处理器然后基于从传感器的地理定位估计的最大可能地理距离确定设备的第三估计地理定位。

[0014] 可以使用新的和更新的地理定位估计来将分组路由到(一个或多个)设备和/或路由来自(一个或多个)设备的分组，以便减少分组等待时间和/或增加分组吞吐量。它们也可以用于将分组围绕特定地理区域路由、将分组避开特定地理区域路由、通过特定地理区域路由分组，例如，以遵守关于数据安全性的规则或法律。地理定位估计也可以用于选择互联网服务提供商(ISP)和解析域名系统(DNS)查询。

[0015] 本技术的实施例还包括用于估计互联网协议(IP)地址中的路由网络前缀的地理定位的方法和装置。为了估计路由网络前缀的地理定位，处理器或其它计算设备计算路由网络前缀的转接树(transit tree)。转接树表示到路由网络前缀的自治系统(AS)路径，并指示第一AS和第二AS之间的至少一个边缘。处理器基于第一AS和第二AS的地理定位来推断路由网络前缀的第一估计地理定位。在一些情况下，处理器可以将第一估计地理定位与从第三方获得的路由网络前缀的第二估计地理定位进行比较。如果第一估计地理定位和第二估计地理定位不匹配，那么处理器可以利用往返于路由网络前缀的传输的等待时间测量结果来验证第一估计地理定位。

[0016] 本技术的还有的另一个实施例包括用于估计具有第一IP地址的设备的地理定位的方法和装置。在这种情况下，连接到计算机网络(例如，互联网)的收集器或其它设备将分组传送到第一IP地址。响应于该分组，收集器从与第一IP地址不同的第二IP地址接收端口不可达消息。并且响应于端口不可达消息，收集器或耦合到收集器的另一个处理设备确定第二IP地址是第一IP地址的别名。因此，收集器或其它处理设备估计第一IP地址和第二IP地址的公共地理定位。

[0017] 应该认识到的是，前述概念和下面更详细讨论的附加概念的所有组合(假设这些概念不相互不一致)被认为是本文公开的发明性主题的一部分。特别地，出现在本公开结尾处的所要求保护的主题的所有组合被认为是本文公开的发明性主题的一部分。还应该认识到的是，也可能出现在通过引用结合的任何公开中的本文明确采用的术语应该被赋予与本文公开的特定概念最一致的含义。

附图说明

[0018] 本领域技术人员将理解的是，附图主要是用于说明性目的，并非旨在限制本文描述的发明性主题的范围。附图不一定按比例绘制；在一些情况下，本文公开的发明性主题的各个方面可能在附图中被夸大或放大以便于理解不同的特征。在附图中，相同的标号通常指代相同的特征(例如，功能上相似和/或结构上相似的元件)。

[0019] 图1A图示了用于估计与特定IP地址相关联的至少一个设备的地理定位的过程。

[0020] 图1B图示了用于估计与特定IP地址相关联的至少一个设备的地理定位的另一个过程。

[0021] 图1C是示出对使用如图1A和图1B中图示的那些过程所做的IP地址的地理定位估计的校正的地图。

[0022] 图2A图示了适合于执行图1A和图1B中示出的IP地理定位过程的地理定位系统的示例。

[0023] 图2B图示了包括物理和虚拟跟踪路由数据收集器和/或网络传感器的跟踪路由数据收集器的位置。

[0024] 图2C和图2D图示了全球分布式跟踪路由数据收集器和/或网络传感器的地理覆盖。

[0025] 图3A图示了示例地理定位服务器。

[0026] 图3B图示了示例跟踪路由收集器设备。

[0027] 图4图示了通过递增TTL来确定特定跟踪路由(在图2A上用黑箭头示出)的过程。

[0028] 图5图示了由三个跟踪路由数据收集器覆盖的区域的交集。

[0029] 图6A图示了经由多协议标签交换(MPLS)隧道的路由。

[0030] 图6B图示了估计与MPLS隧道相关联的IP地址的地理定位的过程。

[0031] 图7A图示了去别名和估计IP地址的地理定位的过程。

[0032] 图7B示出了使用图7A的过程识别和地理定位的别名IP地址的图形表示。

[0033] 图8A图示了转接树,其示出了由地理定位服务器生成的从其起源转接到互联网的核心的网络前缀的集合。

[0034] 图8B图示了使用如图8A中示出的转接树来估计路由网络前缀的地理定位的过程。

[0035] 图9图示了地理定位到单个区域的一组网络前缀随着时间的边界网关协议(BGP)路由稳定性。

[0036] 当结合附图考虑时,根据下面阐述的具体实施方式,本发明的特征和优点将变得更加明显。

具体实施方式

[0037] 下面是涉及使用跟踪路由和其它信息用于地理定位的发明性系统、方法和装置的各种概念及其实施例的更详细描述。应该认识到的是,以上介绍的和下面更详细讨论的各种概念可以以许多方式中的任何方式来实现,因为所公开的概念不限于任何特定的实现方式。具体实现和应用的示例主要是为了说明性的目的而提供。

[0038] 理论上,如果知道所有互联网性能数据,那么可以完美地路由互联网流量;但是,在实践中,并非所有的互联网性能数据都是已知的,因此可用的互联网性能数据和地理定位信息被用于引导互联网流量。例如,基于IP地址到数据中心的假定地理接近度,来自南美洲IP地址的数据请求可以路由到南美洲数据中心。但是南美洲IP地址可能具有彼此之间有限的连接性;例如,亚马逊雨林和安第斯山脉对连接性造成了主要的物理障碍。相反,南美洲的互联网流量通常通过迈阿密路由。因此,即使哥伦比亚数据中心在地理定位上更靠近巴西IP地址,但是将来自巴西IP地址的数据请求路由到迈阿密数据服务器而不是哥伦比亚数据中心实际上可能更高效。类似地,关于特定IP地址的地理定位的错误信息也可能导致不必要地增加等待时间、拥塞等的路由。

[0039] 可以使用关于IP地址的地理定位(地理位置)的更精确的估计来在互联网上更高

效地引导流量。例如，可以使用IP地址的地理定位来预测向IP地址传送分组和从IP地址传送分组的理论上有限的等待时间。地理定位数据可以绑定到全球等待时间地图，以便路由决策更可能接近理论的等待时间限制。地理定位数据也可以用于识别流量源和目的地的物理位置，其具有将提供足够的信息来做出智能路由决策同时还为源位置和目的地位置提供某种程度的匿名性的精确度。例如，地理定位信息可以用于将敏感流量在某些国家内路由或避开某些国家路由，例如，以便遵守出口法规或减少暴露于窃听。地理定位信息也可以用于故障排除网络问题和规划网络扩展。

[0040] IP地址的地理定位可以根据等待时间测量结果、DNS名称、路由信息或各种公共或私有来源(例如，公布的数据中心位置、工作板、商店位置等)进行估计。使用来自许多已知点的等待时间测量结果的三角测量提供了对位置的粗略估计，但是光纤不走直线、可能没有足够的地理定位上区分的点、测量点位置可能无法以足够的精确度得知、它们可能没有均匀/理想地分布、光的速度太快而无法进行短时间测量(光纤中1毫秒＝100公里)等。DNS名称信息(其通常包括与IP地址相关联的城市名称或机场代码)可以用于在算法上解析为ISP的基础设施的不同IP地址恢复的名称，但命名约定可能在ISP之间不同、可能不一致或者可能是错误的。此外，设备可能被移动，从而使得难以验证其位置。地理定位也可以根据针对特定服务提供商的服务区域所基于的路由信息进行估计，但往往只对区域性玩家在宏观级别上起作用。

[0041] 本文公开的技术涉及基于等待时间测量结果、DNS信息和路由的IP地理定位。但与许多其它技术不同，本技术的示例可以用于检测和校正由第三方提供的地理定位数据(包括商业地理定位信息)中的不一致性或错误。这种地理定位数据往往对于最终用户是正确的，但是对于基础设施却是错误的，这限制了其在路由和分析互联网流量方面的效用。

[0042] 识别和校正第三方IP地理定位估计中的错误

[0043] 图1A图示了用于从第三方IP地理定位数据改进与特定IP地址相关联的设备的地理定位估计的过程101。在这个示例中，过程从获得地理定位数据开始(方框111)，地理定位数据可以包括一个或多个IP地址的纬度和经度估计。该数据可以定期地、按需地、自动地或响应于用户的干预从第三方(诸如商业位置来源(例如，Neustar、MaxMind、Digital Envoy等))、前缀注册数据或其它公共或私有来源获得。

[0044] 在方框112中，处理器或其它合适的设备根据从全球许多点做出的等待时间测量来估计地理定位。例如，这些测量可以利用图2A-2D中图示的地理定位系统做出。处理器使用等待时间测量结果来测试地理定位数据的可信度(方框113)。如果等待时间测量结果指示特定地理定位估计是不可信的(例如，因为等待时间测量结果指示实际的位置比地理定位估计更靠近特定测量站点)，那么处理器可以丢弃地理定位估计(方框114)。

[0045] 处理器还可以使用如下面更详细描述的一种或多种合适的错误检测技术来识别地理定位数据和/或等待时间测量结果中的异常值(方框115)。如上所述，处理器可以使异常值经受使用等待时间、DNS命名、公共或私有来源的数据挖掘和/或路由信息的影响(方框116)。在方框117中，处理器基于等待时间、DNS命名、公共或私有来源和/或路由信息来提供差异IP地址的实际地理定位的概率估计。

[0046] 如果期望，处理器可以应用机器学习技术来提高差异识别的置信度(方框118)。换句话说，处理器可以利用来自连续测量的信息来降低置信区间。通常，更多的等待时间测量

产生更高的置信IP地理定位估计。处理器还可以生成显示原始地理定位估计、校正后的地理定位估计和/或校正本身的地图或其它表示(方框119)。

[0047] 以下示例图示了图1A中示出的过程如何可以用于评估和校正最终用户前缀195.160.236.0/22的商用IP地理定位估计。来自NH Hanover的收集器的最终用户前缀195.160.236.0/22中的IP地址195.160.236.1的跟踪路由数据如下：

[0048]

跳	探测1 等待时间	探测2 等待时间	探测3 等待时间	DNS名称（如果有的话）和跳IP地址
1	1 ms	1 ms	1 ms	dslrouter [1.254.254.1]
2	29 ms	26 ms	27 ms	10.20.10.1
3	28 ms	25 ms	25 ms	64.222.166.66
4	31 ms	28 ms	28 ms	POS3-0-0.GW3.BOS4.ALTER.NET [208.192.176.133]
5	38 ms	29 ms	28 ms	0.so-0-1-1.XL4.BOS4.ALTER.NET [152.63.22.174]
6	39 ms	38 ms	36 ms	0. so-7-0-0.XL4.NYC4.ALTER.NET [152.63.17.97]
7	42 ms	35 ms	36 ms	0.xe-5-l-0.BR2.NYC4.ALTER.NET [152.63.18.9]
8	39 ms	36 ms	36 ms	nyc-brdr-02.inet.qwest.net

[0049]

				[63.146.27.209]
9	45 ms	49 ms	42 ms	bst-edge-04.inet.qwest.net [67.14.30.26]
10	60 ms	54 ms	59 ms	63.239.36.122
11	56 ms	52 ms	53 ms	vor-b2.worldpath.net [64.140.193.24]
12	59 ms	54 ms		54 ms 195.160.236.1

[0050] 最终用户前缀195.160.236.0/22包含至少两个服务器IP地址(即,195.160.236.9和195.160.237.24);经由作为欧洲区域互联网注册机构的Réseaux IP Européens(RIPE)注册;并且自我报告为在英国。两个商业来源将这个IP前缀放在英国;另有四个商业来源将其放在英格兰的曼彻斯特;以及另一个商业来源将其放在NH,Laconia。

[0051] 终端用户前缀195.160.236.0/22由在迈阿密注册的Terrenap(AS 23148)通告,其中主要的互联网服务提供商包括Verizon、Hurricane Electric和XO。AS 23148发起140个前缀,其中113个看起来在美国,其它的在阿根廷、比利时、多米尼加共和国、西班牙和荷兰。但更具体的前缀195.160.236.0/24被不同地路由:它由被注册为在Portsmouth,NH的WorldPath(AS 3770)通告,其中主要的互联网服务提供商包括AT&T、Cogent和Century

Link。AS 3770发起46个其它前缀，所有这些前缀都看起来在美国。这表明这个前缀实际上在两个不同的地理定位，其中至少一个在美国，但其中任一个都不在英国。

[0052] 与最终用户前缀195.160.236.0/22相关联的DNS信息给出关于物理位置的一些附加线索，但不允许决定性地确定前缀的地理定位。跟踪路由测量产生与该前缀相关联的路由器的DNS名称。典型地，DNS名称包括指示最近机场或城市的以下三位数字的机场代码或城市缩写。在这种情况下，路由器DNS名称包含以下代码(后面是机场代码的解释)：BOS(美国波士顿)；NYC(美国纽约市)；BST(阿富汗Bost)；以及VOR(未定义)。因此，尽管DNS信息可以用于证实特定地理定位的其它证据，但是DNS信息本身并不一定提供精确的IP地理定位估计。

[0053] 等待时间测量可以设置前缀与一个或多个测量站点之间的距离的上限。在这种情况下，等待时间测量指示目标IP(195.160.236.1)距新罕布什尔州的汉诺威(Hanover，New Hampshire)不超过5300公里，并且距纽约市不超过1800公里(假设从DNS信息得出的NYC名称是正确的并且用于等待时间测量的分组遵循来往于目标IP地址的对称路径)。这些测量排除了英国作为目标IP地址的可能位置。

[0054] 但是，仅仅减去等待时间测量结果可能由于等待时间测量中的错误而不能产生对目标IP地址的地理定位的准确估计。错误的来源包括延迟(在以上背景部分中讨论)、去往和来自目标IP地址的路径的不对称性(即，测量分组遵循从测量设备到目标IP地址的一条路径以及从目标IP地址到测量设备的不同路径)以及将在下面更详细讨论的多协议标签交换(MPLS)。尽管如此，甚至单个等待时间测量可以用于丢弃不准确的IP地理定位估计。

[0055] 幸运的是，假设没有系统测量错误，那么综合许多等待时间测量可以减少IP地理定位估计中的不确定性。例如，在一对节点之间进行许多等待时间测量通常产生等待时间的分布。最短的等待时间可以产生对节点之间的距离的更准确的测量。通过对来自许多不同测量站点的目标IP进行更多等待时间和跟踪路由测量可以改善估计。

[0056] 在这种情况下，对最终用户前缀195.160.236.0/22中的这两个服务器进行更多的跟踪路由测量，进一步缩小了定位：

[0057] • 从Portsmouth，NH到195.160.236.9的跟踪路由是一跳：

[0058] • lgw-vip.ep.psm1.renesys.com(195.160.236.9)0.235ms 0.236ms 0.236ms

[0059] • 从佛罗里达州迈阿密到195.160.237.24的跟踪路由也是一跳：

[0060] • lmaster.ep.mia1.renesys.com(195.160.237.24)0.269ms 0.309ms 0.310ms

[0061] 这个前缀属于动态网络服务数据中心：195.160.236.0/24是从新罕布什尔州朴茨茅斯(Portsmouth，New Hampshire)通告的，并且195.160.236.0/22是从佛罗里达州迈阿密通告的。只要/24前缀可用，195.160.236.9就在朴茨茅斯并且195.160.237.24在佛罗里达州迈阿密。如果/24前缀消失，那么两个前缀(以及因此两个IP地址)都在迈阿密。

[0062] 图1B示出了用于基于计算机、路由器和其它设备的IP地址来估计它们的地理定位的另一个过程102。在步骤130中，地理定位服务器或其它处理器自动从一个或多个第三方服务获得一个或多个计算机、路由器等的地理定位估计。在一些情况下，地理定位服务器可以自动从由第三方运营的服务器下载或接收这些估计。如果地理定位服务器从多于一个来源接收给定IP地址的地理定位估计，那么它可以将地理定位估计彼此进行比较(步骤132)。如果估计不匹配—例如，它们相差太远(例如，相差超过100公里)，或者如果一个是通用的

(例如,“北美”),并且另一个是特定的(例如,“纽约,NY”)—那么地理定位服务器可以获得IP地址与收集器或传感器之间的传输的等待时间分布(步骤134)。

[0063] 例如,如下面详细描述的,地理定位服务器可以从与在IP地址和200个或更多个收集器之间传送分组相关联的往返时间(RTT)导出等待时间分布。地理定位服务器(或收集器)可以基于每个收集器的RTT测量结果确定RTT分布,并且然后基于RTT分布将等待时间估计到与RTT测量结果相关联的置信区间内。在一些情况下,地理定位服务器消除与多协议标签交换(MPLS)跳相关联的或者太短或太长(并且因此指示收集器和IP地址之间物理上不可能的距离)的RTT测量结果。地理定位服务器还可以识别等待时间中的错误的来源并且调整等待时间以解决这些错误。

[0064] 在步骤136中,地理定位服务器为被地理定位的每个IP地址从等待时间分布中选择一个或多个等待时间,然后在步骤138中识别对应的(一个或多个)收集器。接下来,地理定位服务器使用选定的等待时间来估计IP地址和(一个或多个)收集器之间的(一个或多个)距离。更具体而言,地理定位服务器可以使用等待时间测量结果和光纤中光的速度来估计IP地址和对应收集器之间的最大距离。如果地理定位服务器选择从围绕IP地址的收集器做出的三个或更多个短的等待时间,那么它可以使用比如图5所示的那些的三角测量技术更精确地估计IP地址的地理定位。(如果不是来自收集器的所有范围重叠,地理定位服务器可以将IP地址识别为任播IP地址,如下面更详细描述的)。

[0065] 在步骤142中,地理定位服务器确定第三方地理定位估计是否在由步骤140中估计的距离描绘的圆圈(或相交区域)内。如果是,那么地理定位服务器可以指示第三方地理定位估计精确到特定的距离范围内。如果不是,那么地理定位服务器根据收集器位置和距离计算生成IP地址的新的地理定位。这个新的估计可能落在由RTT测量分布设置的置信区间内,这会影响距离测量的不确定性。在一些情况下,变化是相当戏剧性的。例如,图1C示出了使用新的和旧的地理定位为一对IP地址生成的地图,其中一个IP地址从纽约市重新定位至塞内加尔达喀尔,并且另一个IP地址从法国巴黎重新定位至澳大利亚珀斯。

[0066] 新的地理定位估计可以以各种方式使用。例如,处理器可以使用新的地理定位估计来预测与作为路由表更新的一部分向IP地址传送分组或传送来自IP地址的分组相关联的等待时间。这些更新后的等待时间和路由表可以用于基于实际距离而不是网络中跳的数量来更高效地路由流量(例如,步骤146)。它们还可以用于规划何时以及何地安装附加的路由器(例如,在南美洲,以消除或减少经由迈阿密路由器发送流量的需要)。这可以减少总体等待时间和/或增加网络的某些部分中的分组吞吐量。

[0067] 新的地理定位估计也可以用于优选或避免某些地理区域。例如,用户可能优选地避开或围绕已知会造成安全性风险的国家或地区来路由敏感信息。用户可以不直接路由流量,而是可以选择互联网服务提供商(ISP)来基于ISP路由器的地理定位承载流量(步骤148)。用户还可以尝试通过基于ISP路由器的地理定位选择ISP来将流量路由通过某些国家,以便符合关于敏感信息的传输的法律、法规或政策。

[0068] 除了跳计数和等待时间之外或替代跳计数和等待时间,新的地理定位估计也可以用于基于地理定位来解析域名系统(DNS)查询(步骤150)。通过准确地知道用户的地理定位,用户查询的域可以被解析到托管所请求的上下文的最适当的数据中心,其中数据中心可以被选择为在地理上靠近,从而减少等待时间,或者出于先前提到的原因中的任何原因。

[0069] 示例地理定位系统

[0070] 图2A图示了适合于收集跟踪路由数据的地理定位系统200的示例，该跟踪路由数据可以用于例如根据图1A所示的过程101来识别和校正第三方地理定位数据中的错误。如果期望，可以将收集到的跟踪路由数据与DNS数据和从互联网服务提供商(ISP，例如，Sprint、AT&T等)收集到的路由数据组合。

[0071] 图2A所示的地理定位系统200包括地理定位服务器210，其耦合到地理定位数据库212、一个或多个客户端214以及跟踪路由收集器220的网络。虽然图2A仅示出了一个地理定位服务器210和数据库212，但是系统200可以包括和/或使用多个同步的地理定位服务器210和数据库212。当使用多个地理定位服务器210时，可以同步地理定位服务器210以便处理可以在多个数据库212上分布的数据。因此，数据库212可以被同步，并且因此可以使用有线和/或无线通信协议和/或技术通信。

[0072] 跟踪路由收集器220是驻留在它们各自的提供商的数据中心内的真实或虚拟机器，其中的每一个属于自治系统(AS)230或路由域。在操作中，跟踪路由收集器220测量与到它们自己的AS 230内和其它AS 230内的路由器240、目标计算设备250和边界网关协议(BGP)路由器260(也称为边界路由器260)的路由相关联的等待时间。

[0073] AS 230可以被认为是计算设备250的邮政编码—即，每个AS 230可以被描绘为基于ISP的并且不一定是地理范围的互联网的邻域。在每个AS 230内，存在实现AS 230的路由策略并维持到邻近AS 230中的BGP路由器260的连接的边界网关协议(BGP)路由器260(也称为边界路由器260)和其它路由器240。在申请时，全球互联网上的AS的数量超过54,000。

[0074] 更正式地，AS 230是具有单个明确定义的路由策略的IP网络的连接组，该路由策略由公共网络管理员(或管理员组)代表单个管理实体(诸如大学、商业企业、商业部门等)进行控制。AS 230中的给定IP网络内的节点共享相同的网络前缀，从而采用该前缀内的各个IP地址用于互联网连接。大多数自治系统230包括多个网络前缀。AS 230通过使用BGP(其是用于在TCP/IP网络中执行域间路由的外部网关协议(EGP))在边界路由器260之间交换路由消息来与其它AS 230共享路由信息。

[0075] 路由信息可以通过建立从边界路由器260到其BGP对等体之一的连接在AS 230内或在AS 230之间共享以便交换BGP更新。如本领域技术人员所理解的，在边界路由器260之间交换数据的过程被称为“对等(peering)”。在对等会话中，两个网络直接连接和交换数据。内部BGP对等会话涉及直接连接单个AS 230内的边界路由器260和内部路由器240。外部BGP对等会话涉及将相邻AS 230中的边界路由器260直接相互连接。

[0076] 图2A和图4图示了从跟踪路由收集器220到目的地计算机250a的跟踪路由测量。跟踪路由收集器220a使用互联网控制消息协议(ICMP)将第一分组发送到目的地计算机250a。跟踪路由收集器220a还为第一分组指定被称为“生存时间”(TTL)的跳限制值(hoplimit)，其等于1。当第一路由器240a接收到第一分组时，它将TTL递减(从1到0)。在处理TTL＝0的分组后，第一路由器240a向跟踪路由收集器220a返回“时间超时(Time Exceeded)”消息401a，而不是沿着到目的地计算机250a的路径将第一分组转发到下一个路由器。这使得跟踪路由收集器220a能够确定与到目标计算机250a的路径上的第一路由器240a的跳相关联的等待时间。跟踪路由收集器220a然后向目标计算机250a发送TTL＝2的第二分组。第二路由器260a返回另一个时间超时消息等等。随后的分组(包含TTL＝3至TTL＝6)从路由器260b、

260c、240b、260d和260e发出时间超时消息。当目的地计算机250a接收到TTL＝7的最终分组时，它将“回声应答(Echo Reply)”消息返回给跟踪路由收集器220a，使得跟踪路由收集器220a能够测量最后一跳的等待时间。

[0077] 除了由跟踪路由收集器220获得的跟踪路由数据之外，每个地理定位数据库212还可以包括其它数据，包括但不限于BGP UPDATE消息数据、路由注册表数据、域名服务器(DNS)数据、互联网网络数据、公共和私有来源的数据挖掘、和/或与任何或所有这些数据来源相关或从其导出的其它数据。这些数据可以从ISP和/或其它来源收集，并且可以用于提高地理定位估计准确性，如以上和下面所解释的。

[0078] 跟踪路由数据收集器的全球覆盖和分布

[0079] 图2B图示了图2A的全球分布式跟踪路由数据收集系统中的跟踪路由数据收集器的位置。该系统可以包括基于地理可访问性、人口密度、IP地址密度等分布的数十到数百或甚至数千个收集器(例如，300多个收集器)。图2B中的地图上的每个点表示不同的物理或虚拟跟踪路由数据收集器。

[0080] 图2C和图2D图示了如图2A和图2B所示的那个的全球分布式跟踪路由数据收集系统的地理覆盖。图2C和图2D中的阴影指示到互联网协议(IP)地址小区或组的中值等待时间。更具体而言，图2C和图2D中的每个四分之一度纬度-经度小区根据从最近的当前跟踪路由数据收集器到该小区中的所有IP的中值等待时间被加阴影。小区201在图2C和图2D中出现得越黑，跟踪路由收集器越靠近小区中的所有IP，并且因此地理定位估计的准确性越好。小区202在图2C和图2D中出现得越黑，跟踪路由数据收集器距该小区越远，并且可以认为地理定位估计的精确度越低。黑色指示0毫秒的等待时间，白色指示至少100毫秒的等待时间，并且灰色指示中间等待时间(例如，25毫秒)。

[0081] 图2C和图2D中的阴影可以帮助放置附加的收集器并且帮助对由跟踪路由收集器收集到的关于每个小区的数据进行加权。一种最佳的情景是其中小区是完全黑色。这种最佳情景将使得地理定位的估计具有100％的可靠性或准确性。

[0082] 地理定位服务器和跟踪路由数据收集器

[0083] 图3A图示了地理定位服务器110的框图，地理定位服务器110包括耦合到用户界面312的处理器318、通信接口319和存储可执行指令316的存储器314。这些可执行指令316包括用于执行地理定位服务器过程317的指令，当该指令由处理器318实现时，使处理器318分析以基于跟踪路由数据、网络前缀信息等来估计IP地址的地理定位。

[0084] 处理器318可以包括一个或多个高速数据处理单元以执行用于执行用户和/或系统生成的请求的程序组件。通常，这些高速数据处理单元包含各种专用处理单元，诸如但不限于：集成系统(总线)控制器、存储器管理控制单元、浮点单元、以及甚至专用处理子单元，比如图形处理单元、数字信号处理单元等等。此外，处理器318可以包括内部快速访问可寻址存储器，并且能够映射和寻址超出处理器本身的存储器；内部存储器可以包括但不限于：快速寄存器、各种级别的高速缓冲存储器(例如，级别1、2、3等)、RAM、ROM等。处理器318可以通过使用可经由指令地址访问的存储器地址空间来访问存储器314和可执行指令316，其中处理器318可以构造和解码所述指令地址以允许其访问通向具有存储器状态和/或可执行指令的特定存储器地址空间的电路路径。

[0085] 虽然通信接口319可以常规地接受、连接到多个接口适配器和/或与其通信，但是

不一定以适配卡的形式，诸如但不限于：输入输出(I/O)接口、存储接口、网络接口等等。例如，包括在通信接口319中的网络接口可以用于发送和接收来自图2A中的跟踪路由收集器设备320的信息。

[0086] 用户界面显示器312可以包括具有接受来自视频接口的信号的接口(例如，DVI电路系统和电缆)的基于阴极射线管(CRT)或液晶显示器(LCD)的监视器。可替代地，用户界面显示器312可以包括触摸屏和/或其它内容显示设备。视频接口合成由存储在存储器314中并由处理器318执行的可执行指令316生成的信息。可执行指令317包括具有一组指令以处理和分析从一个或多个跟踪路由收集器设备220获得的数据的地理定位服务器处理模块317。用户界面显示器312可以包括如由操作系统和/或操作环境提供的、与操作系统和/或操作环境一起提供的和/或在操作系统和/或操作环境之上提供的常规图形用户界面，所述操作系统和/或操作环境诸如Apple OS、Windows OS、Linux、基于Unix的OS等等。用户界面显示器312可以通过文本和/或图形设施允许程序组件和/或系统设施的显示、执行、交互、操纵和/或操作。用户界面显示器312提供用户可以通过其影响、与之交互和/或操作计算机系统的设施。用户界面显示器312可以与组件集合中的其它组件(包括其本身和/或类似设施)通信和/或向其传送。用户界面显示器312可以包含、传送、生成、获得和/或提供程序组件、系统、用户和/或数据通信、请求和/或响应。

[0087] 图3B图示了示例跟踪路由收集器设备220的框图。跟踪路由收集器设备220包括通信接口332和处理器324，比如分别在服务器110中的通信接口319和处理器318。跟踪路由收集器220还具有存储器326，其存储可执行指令328，包括用于从一个或多个目标计算设备(例如，图2A中的路由器240和目标计算设备250)收集跟踪路由数据的指令329。

[0088] 跟踪路由数据收集和跟踪路由数据

[0089] 图1和图4图示了跟踪路由数据系统的工作原理。为了执行跟踪路由，跟踪路由收集器220a使用互联网控制消息协议(ICMP)将第一分组发送到目的地计算机(250a)。跟踪路由收集器220a还为第一分组指定被称为“生存时间”(TTL)的跳限制值(hoplimit value)，其等于1。当第一路由器240a接收到第一分组时，它将TTL递减(从1到0)。在处理TTL＝0的分组后，第一路由器向跟踪路由收集器220a返回“时间超时(Time Exceeded)”消息401a，而不是沿着到目的地计算机250a的路径将第一分组转发到下一个路由器。这使得跟踪路由收集器220a能够确定与到目标计算机250a的路径上的第一路由器240a的跳相关联的等待时间。跟踪路由收集器220a然后向目标计算机250a发送TTL＝2的第二分组。第二路由器260a返回另一个时间超时消息等等。随后的分组(包含TTL＝3至TTL＝7)从路由器260b、260c、240b、260d和260e发出时间超时消息。当目的地计算机250a接收到TTL＝8的最终分组时，它将“回声应答(Echo Reply)”消息402返回给跟踪路由收集器220a，使得跟踪路由收集器220a能够测量最后一跳的等待时间。

[0090] 通过在每次它发送分组时增加TTL并且监视来自中间路由器的“超过TTL(TTL exceeded)”响应401a、401b、401c等，跟踪路由收集器设备220a发现到目的地计算机250a的路径上的连续跳和到目的地计算机250a的往返时间两者。收集到的“超过TTL”响应由跟踪路由收集器设备220a使用，以建立由ICMP分组遍历的路由器的列表，直到到达目标设备250a并返回ICMP回声应答402。

[0091] 收集到的跟踪路由数据包括用于跟踪路由中的每个设备的标识符，包括对应的跟

踪路由收集器设备220的标识符和/或IP地址。所包含的IP地址可以表示作为全球或本地计算机网络的一部分的路由器。跟踪路由数据还包括时间，该时间表示跟踪路由收集器设备220获得来自路由器的响应所花费的时间以及跟踪路由收集器设备220从目标计算设备获得ICMP回声应答所花费的时间。

[0092] 如果期望，由跟踪路由收集器设备220获得的跟踪路由数据可以由地理定位服务器110接收和处理，以在数据结构中生成如下面所示的中间人类可读格式：

[0093] tr_base_fields＝[('dcv',str),#数据版本

[0094] ('ts',int),#跟踪开始的时间戳

[0095] ('protocol',str),#[I]CMP,[U]DP,[T]CP

[0096] ('port',int),

[0097] ('collector_ip',str),

[0098] ('collector_external_ip',str),

[0099] ('collector_name',str),

[0100] ('target_ip',str),

[0101] ('halt_reason',str),#

[0102] [S]uccess,[L]oop,[U]nreachable,[G]ap

[0103] ('halt_data',int),#失败跟踪的附加信息

[0104] ('hoprecords',T5HopLi5t)]

[0105] 下面给出tr_base_fields数据格式下的跟踪路由数据的示例。每个字段在单独的行中列出以简化地理定位服务器过程的描述：

[0106] 1:T5

[0107] 2:1431005462

[0108] 3:I

[0109] 4:0

[0110] 5:192.170.146.138

[0111] 6:192.170.146.138

[0112] 7:vps01.nycl

[0113] 8:88.203.215.250

[0114] 9:S

[0115] 10:11

[0116] 11:q,0,1,0

[0117] 12:63.251.26.29,0.363,2,254

[0118] 13:74.217.167.75,1.297,3,252

[0119] 14:129.250.205.81,1.171,4,252

[0120] 15:129.250.4.148,1.614,5,250,576241

[0121] 16:129.250.3.181,87.140,6,250,519266

[0122] 17:129.250.4.54,112.258,7,247,16013

[0123] 18:129.250.3.25,114.446,8,248

[0124] 19:83.217.227.22,123.002,9,245

[0125] 20:212.39.70.174,125.613,10,245

[0126] 21:88.203.215.250,124.967,11,51

[0127] 用于地理定位的字段包括:2:时间戳(自从1970年1月1日以来的秒数,UNIX纪元);7:收集器名称(每个收集器的唯一标识符;这一个在纽约市);8:跟踪路由目标IP地址;以及11至21:跟踪路由跳数(取决于网络拓扑的可变数字)。

[0128] 每一跳包含逗号分隔的子列表,其具有:跳IP(如果没有接收到响应,那么为q);以毫秒为单位的往返时间(RTT);TTL;反向TTL;以及零个或多个MPLS标签。

[0129] 在一些实现中,地理定位服务器过程基于距跟踪路由数据收集器的等待时间,其是在每一跳中找到的RTT值。如以上示例所示,一个跟踪路由可以产生若干个响应跳,每个跳都具有IP和距收集器(在这种情况下,位于纽约市的收集器)的往返时间(RTT)。地理定位服务器过程将每个跟踪路由分解成各个(收集器城市、IP、RTT)元组或收集器边缘等待时间。

[0130] 在收集到的跟踪路由数据的这个示例中,考虑行12:63.251.26.29,0.363,2,254。看出IP地址63.251.26.29距收集器0.363毫秒(RTT)。在一些实现中,地理定位服务器过程可以不考虑跳中的第3或第4字段(TTL和反向TTL)。每个城市可以利用唯一的整数标识符。例如,纽约的geonameid是5128581。距位于纽约市的跟踪路由数据收集器设备的一跳可以被表示为元组:(5128581,63.251.26.29,0.363)。

[0131] 基于光纤约束中的光速,1毫秒RTT对应于沿着大圆或往返行程行进的大约100公里的最大可能距离。这意味着对于这一跳,在RTT为0.363毫秒的情况下,从NYC跟踪路由数据收集器到具有IP地址63.251.26.29的设备的最大距离为36.3公里(22.6英里)。在一些情况下,当离开数据中心时存在一些初始延迟时,有高的可能性该IP被并置在相同的数据中心处。鉴于这个证据,通过来自其它跟踪路由的附加测量结果进行强化,地理定位服务器过程可以基于跟踪路由数据收集器所位于的城市的纬度和经度以及由该收集器所覆盖的半径来改进IP地理定位。基于光速考虑的地理定位服务器过程分析强化了可以从地理定位服务器获取的推断。

[0132] 多协议标签交换(MPLS)和地理定位

[0133] 在一些情况下,跟踪路由跳可以在结尾处包含多协议标签交换(MPLS)标签,如以上在行15、16和17中示出的,其分别以MPLS标签576241、519266和16013结尾。为了等待时间测量和比较的目的,这些MPLS跳可以被丢弃,因为它们的RTT通常对应于MPLS隧道出口跳的RTT,并且因此将产生更大的可信度半径,如下面关于图6A更详细解释的。当考虑多次测量时,去除MPLS跳提供了更严格的可信度包络。因此,在一些实现中,通过减少统计上可信度半径来过滤MPLS跳以改善测量。

[0134] 丢弃来自示例跟踪路由数据的MPLS标签产生为地理定位生成的以下元组的列表,其中第一元素对应于纽约市收集器的geonameid(5128581)以及该跟踪路由的起源:

[0135] 5128581,63.251.26.29,0.363)

[0136] 5128581,74.217.167.75,1.297)

[0137] 5128581,129.250.205.81,1.171)

[0138] 5128581,129.250.3.25,114.446)

[0139] 5128581,83.217.227.22,123.002)

[0140] 5128581,212.39.70.174,125.613)

[0141] 5128581,88.203.215.250,124.967)

[0142] 值得注意的是,最后一跳中的IP地址是88.203.215.250,其与目标(字段8)相同。这意味着跟踪路由中的最终目标设备对由跟踪路由收集器设备执行的探测做出了响应或回应。

[0143] 使用重叠的边缘等待时间的地理定位

[0144] 在一些实现中,跟踪路由数据收集器边缘等待时间可以基于由图2B中示出的全球分布式跟踪路由数据收集器执行的跟踪路由。(在一些情况下,收集器每天可以从总共超过300个收集器执行超过500,000,000次测量)。地理定位服务器可以基于对于每个边缘的多次测量生成统计推断。每个跟踪路由数据收集器中嵌入的定时器可能会将噪声添加到被利用来限定从每个收集器到IP地址的可信度半径的观测的RTT的测量结果。为了考虑测量缺陷或噪声,地理定位服务器可以使用修改后的Thompson Tau测试来识别异常值,从而消除具有(潜在人为的)低RTT的异常值。作为计算上的便利,对于每个收集器边缘使用第25个百分位等待时间可以合理地消除异常值,并且因此可以代替修改后的Thompson Tau测试使用。第25个百分位是从这一点开始使用的,但这应该被视为可以消除异常值的多种方式之一,而不是视为定义本技术的方面。其它的可能性包括但不限于使用第5个、第10个或第15个百分位、中值、模(mode)或用于减少或消除异常值的任何其它合适的技术。

[0145] 图5图示了由三个跟踪路由数据收集器覆盖的区域的交集。在一些实现中,地理定位服务器过程可以生成从每个收集器到与目标计算机设备对应的IP地址的可信度半径。例如跟踪路由数据收集器设备501a、503a和505a。对于给定的IP地址,网络前缀和/或目标设备、由每个收集器覆盖的半径所限定的圆圈509、511和513的相交区域限定了其中IP被可信地地理定位的区域507a。那个区域内的城市是该IP的地理定位的候选。例如,在跟踪路由数据收集器501b、503b和505b当中,城市507b可以是取决于由收集器收集到的跟踪路由数据的候选城市。

[0146] 在一些情况下,圆圈可能不相交。在这种情况下,等待时间可能指示IP靠近两个或更多个收集器城市,即,比一对收集器之间的中间点更靠近每个收集器城市。这种情况被分类为地理不一致性,因为它指示具有相同IP的设备位于多于一个位置。这是任播网络的属性。地理定位服务器过程识别地理不一致性的实例并将对应的IP标记为任播。

[0147] 由于互联网提供商可以改变给定IP的位置,因此收集到的跟踪路由被不断地探测,以帮助确保对目标IP的测量是在目标的地理定位静止时进行的。

[0148] 下面基本上以PHP:超文本预处理器代码的形式提供了地理定位服务器过程的一些功能(包括探测目标设备、识别这些目标设备的城市以及识别任播IP地址)的伪代码表示的示例:

[0149]

```
# 对于跟踪路由中找到的每个 IP，构建（第 25 个百分位 RTT，
# 收集器城市）的排序列表
func ip_to_collector_latencies(traceroutes):
# 为每个唯一的（collector_city，IP）对构建 RTT 数组
rtt_dictionary = new dict
for 所有最近的跟踪路由:
    for 跟踪路由中的每一跳:
        if not MPLS（hop）:
            rtt_dictionary[(collector_city, IP)].append(RTT)
# 为每个 IP 构建第 25 个百分位 RTT，收集器城市对的列表
collector_latencies = new dict
for 每个 IP:
    for 收集器城市的每个逐对组合:
        if collector1-IP-latency + collector2-IP-latency <
            收集器之间最小的可能等待时间:
            将 IP 标记为任播
```

```
                    break
                if IP 不是任播:
                    rtt_collector_list = new list
                    for 每个 collector_city:
                        rtt_array = rtt_dictionary[(collector_city,IP)]
                        rtt25 = compute_25th_percentile_latency(rtt_array)
                        rtt_collector_list.append((rtt25, collector_city))
                    rtt_collector_list.sort()
                    collector_latencies[IP] = rtt_collector_list
        return collector_latencies

        # 当校正 IP 的第三方地理定位时,
        # 我们可以计算到每个收集器城市的最小可能的 RTT,
[0150]  # 并且与观察到的等待时间进行比较
        func is_geolocation_plausibile(IP, city):
            rtt_collector_list = collector_latencies[IP]
            for (rtt, collector_city) in rtt_collector_list:
                minpossrtt = minimum_rtt [(city, collector_city)]
                if rtt < minpossrtt:
                    return False # IP 被错误地理定位
            return True # IP 地理定位是可信的

        # 我们也可以创建所有可信的城市的列表以便
        # 自行确定 IP 地理定位
        func_geolocate_ip(IP):
            plausible_cities = create_list_of_all_cities()
            rtt_collector_list = collector_latencies [IP]
```

```
        for (rtt25, collector_city) in rtt_collector_list:
          for city in plausible_cities:
            minpossrtt = minimum_rtt[(city, collector_city)]
[0151]      if rtt25 < minpossrtt:
                # 基于光纤中的光速考虑 IP 不能在这个城市
                plausible_cities.remove(city)
        return plausible_cities
```

[0152] 在一些实现中,当等待时间太大而不能将选择减少到单个城市或大都市地区时,地理定位服务器将IP地址地理定位到更大的地理范围。例如,如果可信城市的最终列表全部位于相同的州或国家,那么地理定位服务器过程可以选择分配州范围或国家范围的地理定位。地理定位服务器过程的进一步实现确定其中可以地理定位IP的所有可信的网格小区。

[0153] 示例地理等待时间确定

[0154] 考虑由总部在南非的互联网提供商AS6637,MTN SA发起的IP地址41.181.245.81。

[0155] 下面是来自地理定位服务器过程的输出:

[0156] TLATSUMG2 1429865014 1430956773 41.181.245.81 0 0 0 -- 50 39 2265 342.4 London,GB 0.7 1478 2643743

[0157] (0.7,1478,2643743) (6.2,3,2641170) (10.4,51,2925533)

[0158] (20.3,46,2657896) (22.9,1,3173435) (24.4,28,2867714)

[0159] (32.0,2,3196359) (32.9,55,3067696) (38.9,1,756135)

[0160] (41.0,1,3099434) (46.5,1,649360) (61.3,56,250441)

[0161] (68.9,100,5101798) (69.0,3,5128581) (73.6,1,4930956)

[0162] (74.6,105,4744870) (75.5,2,4781530) (90.2,4,4887398)

[0163] (91.3,2,4180439) (97.2,1,108410) (98.3,113,4164138)

[0164] (103.3,2,4684888) (108.5,1,524901) (127.2,1,3598132)

[0165] (134.8,1,1277333) (136.1,46,5380748) (136.8,1,1174872)

[0166] (138.8,3,5391959) (139.2,1,5780993) (139.7,117,5809844)

[0167] (140.4,4,5368361) (149.2,10,3397277) (149.9,2,3369157)

[0168] (156.3,1,5392171) (194.2,1,3448439) (204.5,16,3435910)

[0169] (264.4,1,1835848) (340.3,1,2028462) (342.4,2,1581130)

[0170] 以上输出示例中的带下划线的字段是最近收集器的位置、到感兴趣的IP的等待时间、测量的数量以及收集器城市的geonameid。在这种情况下,最近的收集器在伦敦,并且从伦敦收集器到IP的等待时间为0.7毫秒。

[0171] 下划线字段下面的数据是来自每个收集器的具有第25个百分位等待时间的元组。每个逗号隔开的元组中有三个条目:

[0172] RTT(第25个百分位),测量的数量,geonameid

[0173] 这些元组按照等待时间排序,使得紧接着下划线字段之后的第一个元组具有最小

的等待时间。

[0174] 基于来自伦敦(2643743)收集器的1478次测量,元组(0.7,1478,2643743)是0.7毫秒的最接近的测量结果。这重复了先前三个字段中为了方便起见而严格分解的信息。

[0175] 元组(6.2,3,2641170)是从诺丁汉(2641170)收集器的6.2毫秒的第二近的测量结果。总共有39个收集器城市返回到该IP的测量结果。

[0176] 伦敦和开普敦之间的光纤中的最小可能的RTT等待时间是95.7毫秒。由于基于光速约束,IP距离伦敦跟踪路由收集器设备只有0.7毫秒,因此地理定位服务器过程可以确定性地确定IP地址41.181.245.81不位于南非,如由互联网提供商的身份可能暗示的。此外,考虑到其距伦敦收集器的低等待时间,地理定位服务器过程可以将该IP的地理定位分配给伦敦。为了做出这种校正,地理定位服务器过程首先确定其它跟踪路由数据收集器支持伦敦作为该IP的可信位置。

[0177] 例如,下面是从伦敦收集器到IP地址41.181.245.81的单个跟踪路由数据。在明确声明这个IP地址在伦敦之前,可以使用来自多个位置的更多的跟踪路由来提供附加的支持。

[0178] 跟踪路由到41.181.245.81(41.181.245.81),30跳最多,60字节

[0179] 分组1 20.ae1.edge-01-lin.as33517.net(80.231.219.189.)0.319ms 0.282ms 0.361ms 20.ae0.edge-02-lon.as33517.net(185.38.96.27)0.365ms 0.338ms 0.307ms 3xe-10-3-3.edge3.London15.Level3.net(212.187.193.189)0.358ms 0.285ms 0.338ms 4MTN-GROUP.edge3.London15.Level3.net(212.187.195.162)1.791 ms 1.830ms 1.872ms 5 41.181.190.190(41.181.190.190)2.034ms 2.083ms 2.060ms 6 41.181.245.81 (41.181.245.81)0.820ms 0.784ms 0.914ms

[0180] 反向域名系统(DNS)

[0181] 在一些实现中,地理定位服务器过程可以在可用时对IP使用反向DNS(rDNS)进一步增强推断。提供商经常在其路由器名称中根据它们定义的命名方案编码地理定位信息。在以上带下划线的跟踪路由示例中,80.231.219.189具有也加下划线的rDNS 20.ae1.edge-01-lon.as33517.net。在该标签中强调的片段“lon”指示该IP地址位于伦敦。地理定位服务器过程具有在可能的情况下从这些标签中提取地理定位信息的手动推导的基于规则的引擎。与受物理学约束的等待时间不同,这些标签由人类输入,并且如果没有得到正确的维护,那么会遭受错误。但是,将根据rDNS的地理定位与根据等待时间的地理定位组合可以产生更准确的IP地理定位。例如考虑69.252.112.58 se02.woburn.ma.boston.comcast.net。

[0182] 沃本(Woburn)位于波士顿以北大约15公里处。将地理定位服务器过程应用到这个IP地址产生以下结果:

[0183] TLATSUMG2 1430698623 1431022690 69.252.112.58 0 0 0--186 116 1042 307.5Boston,MA,US 8.5 6 4930956 8.5,6,4930956 9.2,11,5128581 9.4,7,5101798 10.2,6,5102076 12.8,5,5087168 14.4,16,4744870 15.1,6,4140963 15.7,5,4781530 16.4,5,5083221 16.7,5,5091383 24.9,14,6167865 29.1,12,4887398 35.8,13,4180439 36.8,16,4852832 51.4,3,5419384 52.5,14,4164138 54.4,19,4684888 63.4,6,4699066 66.9,6,5780993 76.3,18,5809844 77.6,5,6173331 77.8,13,5380748 77.8,70,2643743

78.6,20,5392171 79.2,9,5391959 80.2,4,3598132 80.9,26,5368361 81.9,2,5746545 83.8,7,2960316 84.6,11,2787416 84.8,5,2641170 85.3,6,3991164 87.5,12,2988507 88.8,3,2964574 90.7,6,4005539 94.3,35,2759794 95.1,34,2925533 98.55,4,3674962 98.6,5,2867714 100.05,12,2657896 100.2,6,2660646 101.3,14,756135 101.4,8,3173435 101.6,5,3621849 104.7,5,3117735 107.3,6,3067696 108.5,6,2761369 109.0,6,2673730 111.8,6,2520600 113.9,6,792680 113.9,6,2735943 114.45,4,3196359 116.1,5,3099434 116.9,10,3060972 119.0,6,683506 120.0,6,3054643 120.9,5,2267057 123.2,5,649360 123.9,7,2523920 126.8,5,625144 128.6,6,5856195 129.0,6,498817 132.2,5,3397277 132.3,14,745044 133.0,6,727011 135.6,5,250441 135.6,11,524901 135.9,6,726050 135.9,11,703448 138.7,6,293397 139.2,6,593116 143.25,4,323786 145.5,5,360630 148.3,6,3871336 149.9,5,587084 150.5,10,3448439 161.9,5,2548885 162.4,7,3451190 165.3,6,108410 165.8,3,616052 171.0,3,3461655 173.6,28,1850147 174.8,9,3435910 183.7,5,1526384 184.9,5,1528675 187.1,6,1853909 195.9,11,292223 196.1,15,1668399 198.6,12,1275339 204.9,5,1835848 207.25,8,1668341 216.4,4,1174872 217.45,4,1277333 219.9,9,1273294 220.9,21,2147714 222.3,4,160263 224.15,4,184745 224.7,6,3369157 224.7,30,1819729 225.0,10,1264527 230.55,4,1808926 234.8,6,285787 240.4,5,1176615 243.0,6,1701668 252.2,31,1880252 253.35,4,1816670 266.2,5,1784658 268.9,6,6930887 275.05,4,1797929 276.5,5,1821306 276.9,4,1609350 281.3,3,1581130 281.4,4,2063523 286.5,5,2028462 300.9,5,2158177 307.5,4,1735161

[0184] 波士顿是距离IP地址(69.252.112.58)8.5毫秒的最近跟踪路由数据收集器设备。但是rDNS提供了该IP地址在马萨诸塞州沃本(Woburn,MA)的线索。地理定位服务器过程验证沃本是否在该收集器等待时间分布允许的可信城市当中。因此,地理定位服务器过程能够做出对马萨诸塞州沃本而不是波士顿的更精确的地理定位分配,从而增强IP地理定位估计的准确性。

[0185] 此外,在一些实现中,一旦地理定位服务器使用等待时间被确认,地理定位服务器就尝试通过寻找相似的rDNS名称并通过自然语言处理为其分配地理定位来自动发现其自己的规则。它通过维护它已发现的所有rDNS名称的数据库来实现这一点。对于IPv4而言,这是通过对整个地址空间进行反向解析来完成的,截至撰写本文时,其产生超过12.4亿个rDNS条目。对于IPv6而言,考虑到空间的巨大性,rDNS条目经由从全球开放DNS递归捕获的被动DNS和根据路由的IPv6空间上的选择性rDNS解析来收集。通过指示服务记录它接收的查询以及它提供的答案两者(包括但不限于请求的时间戳、请求的域和记录类型、提供的答案以及对任何DNS查询和响应进行分类的其它属性),rDNS从开放式递归捕获。

[0186] 作为简单的示例,地理定位服务器可以(可能通过检查现有规则或通过考虑等待时间)发现在rDNS中包含'nyc'的字符串的存在强烈地与纽约市中的相关联的IP地址相关。当前在IPv4中有超过260万个这样的rDNS条目,其中许多条目目前没有被商业提供商地理定位到纽约市。

[0187] 考虑67.17.81.197(loop0.cs1.NYC2.NYC.gblx.net)。截至撰写本文时,三家不同的商业提供商可替代地将这个IP地址置于美国德克萨斯州的达拉斯、厄瓜多尔(没有城市)

或仅仅美国(没有城市)。等待时间与位于NYC的IP地址一致并且对于它位于德克萨斯州或厄瓜多尔,该等待时间是完全不可信的。目前在所有IPv4空间中有超过一百个.nyc.glbx.net rDNS记录,并且等待时间与位于纽约市的所有这些记录一致。还有许多与纽约市相关联的包含'nyc'的rDNS规则。因此,地理定位服务器能够发现rDNS中的字符串'nyc.gblx.net'强烈地与地理定位到纽约市的相关联IP地址相关,并且因此自动建立对应的规则并将其添加到基于规则的rDNS处理引擎。

[0188] 公共和私有数据来源

[0189] 在一些实现中,地理定位服务器从公共、私有和半私有来源识别可信的物理地址,用于由这里描述的技术进行进一步验证。这些来源可以包括但不限于数据中心作业的工作板和企业职位页面,以查找没有以其它方式发布的数据中心所宣称的物理地址;路由窥镜(looking glasses),其下拉菜单通常提供路由器位置;存储可以与IP地址绑定的位置,诸如具有多个位置的主要连锁店的位置;以及来自数十个全球和本地互联网路由注册机构(IRR)的前缀注册数据。

[0190] 作为以上技术的一个示例,考虑遍布119个国家超过36,000个的麦当劳餐厅。这些商店中的许多都被编号,并且数千个商店可以在以上提到的rDNS数据中找到。考虑以下两个IP地址及其rDNS条目:

[0191] • ip=206.59.233.82 fqdn=nmd.mcd 18734.mia.wayport.net

[0192] • ip=206.59.233.83 fqdn=eth 1-1.nmd.mcd 18734.mia.wayport.net

[0193] 这两个IP地址显然参考麦当劳的商店#18734。刮开公共网站这个商店位于19300palocka Blvd,Opa-Locka(Dade County),FL,33054,其距离迈阿密的中心大约12英里。以这种方式找到的可信的地理定位可以通过这里描述的其它方法进一步验证。

[0194] 任播示例

[0195] 当目标设备具有任播IP时,可以从地理定位服务器过程获得不同的示例。考虑下面示出的对于IP地址199.27.135.101的结果:

[0196] TLATSUMG2 1429747203 1430956517 199.27.135.101 7191 181.069922721 0.648613384403

[0197] 1850147=3448439,1850147=3448439,1835848=3448439,1835848=3448439,1880252=3674962,1880252=3674962,2147714=2520600,2147714=2520600,1853909=3448439,1853909=3448439,1850147=3461655,1850147=3461655,1819729=3448439,1819729=3448439,1835848=3461655,1835848=3461655,2147714=3117735,...,683506=1668399 187 116 3833 201.0 Chicago,IL,US 0.4 48 4887398 0.4,48,4887398 0.4,50,4164138 0.4,60,4744870 0.4,66,5809844 0.4,73,2147714 0.5,22,3067696 0.5,97,5368361 0.6,23,2520600 0.6,49,4180439 0.6,73,5392171 0.7,21,2673730 0.7,24,4781530 0.7,71,4684888 0.7,124,2925533 0.7,126,2759794 0.7,232,2643743 0.8,19,3117735 0.8,47,2988507 0.9,44,5128581 1.0,52,5380748 1.0,52,6167865 1.1,113,1850147 1.2,69,3448439 1.3,24,2761369 1.3,27,3173435 1.3,36,756135 1.4,25,4140963 1.4,27,5101798 1.5,20,1835848 1.6,111,1880252 1.8,118,1819729 1.9,23,5102076 1.9,37,5391959 4.6,19,6173331 5.2,35,3060972 5.3,23,4699066 5.4,13,3461655 5.6,14,3099434 5.6,21,4930956 5.8,36,2787416 6.0,

18,2641170 6.0,26,2960316 6.9,20,2867714 7.0,17,3196359 7.1,9,5746545 8.2,19,5083221 8.3,23,1853909 8.7,21,5091383 10.5,56,4852832 10.6,9,2964574 10.6,16,3674962 11.0,15,649360 11.9,24,2660646 12.7,20,498817 13.5,22,593116 13.9,43,2657896 14.6,18,2158177 16.15,20,3369157 17.7,21,3054643 19.0,29,1668341 19.9,24,792680 20.9,20,5780993 21.9,44,524901 22.0,21,3991164 22.4,40,703448 23.0,19,5419384 23.6,27,2523920 24.2,20,273594324.4,19,2267057 26.1,19,5087168 29.5,15,3598132 29.8,16,1609350 35.1,36,1264527 37.1,21,6930887 39.0,18,625144 39.4,57,1668399 41.2,20,727011 43.2,14,1277333 44.7,21,683506 44.8,16,1735161 45.2,24,726050 45.9,20,4005539 46.9,19,1821306 47.4,15,745044 47.8,16,323786 49.8,22,5856195 50.1,14,2063523 50.2,18,3397277 51.6,18,3621849 59.8,23,1701668 61.0,18,250441 61.0,23,293397 68.0,44,1275339 71.2,18,2548885 73.2,12,616052 74.9,21,587084 75.8,15,1528675 80.4,11,1581130 83.4,22,108410 84.9,37,1273294 93.0,18,1526384 105.6,47,292223 113.4,25,3451190 126.1,17,160263 132.3,15,1174872 141.7,15,360630

[0198] 以上示例中示出的第一行中的下划线值7191表示地理不一致收集器城市对的数量。虽然IP地址距4887398(芝加哥)为0.4毫秒,但是它也距分别为迈阿密(Miami)、阿什伯恩(Ashburn)、西雅图(Seattle)和悉尼(Sydney)的4164138、4744870、5809844和2147714为0.4毫秒。鉴于对于具有该IP地址的单个设备物理上不可能那么靠近这些城市(以及该结果中的若干其它城市)中的每一个,因此有可能推断出该IP地址是任播的实例并且可以被分配地理定位“地球”。

[0199] 确定地理定位三明治(sandwich)

[0200] 考虑作为从收集器IP地址开始并以目标IP地址结束的IP地址序列的跟踪路由。每个IP地址可以使用这里描述的一种或多种技术以及也可能其它技术来指定地理定位。IP地址序列还可以被细分为子序列或跳片段。在一些实现中,地理定位服务器使用三明治方法来考虑跟踪路由中的一些或全部这样的片段,其中片段的第一跳和最后一跳地理定位到同一国家,但是一个或多个中间跳地理定位到某个其它国家。

[0201] 三明治方法的基础是跟踪路由不可能离开一个国家,进入另一个国家,而只能返回到原来的国家。如果找到这样的片段,那么该片段中的IP地址中的一个或多个可能被错误地地理定位。可以使用这里描述的技术进一步分析片段中的每个IP地址。在一些实现中,地理定位服务器过程使用这种方法的一般化,从而查找片段中第一跳和最后一跳在距离上看似靠近,但是片段的跳之间的距离之和大的任何跟踪路由序列。即,从第一跳到最后一跳的净距离小于跳之间的距离之和。例如,当来自两个本地提供商的流量在某个远处位置进行交换时,这样的束发(hair-pinning)序列有时是合法的,因为对等的任何两个提供商在它们为商业或其它原因提供服务的所有市场内可能没有被物理连接。与任何潜在的错误地理定位一样,这种看似异常的地理定位应该通过这里描述的其它技术来验证。

[0202] 三明治地理定位示例

[0203] 考虑以下从纽约市收集器到澳大利亚的IP地址的跟踪路由。三明治被识别为序列114.31.199.242,114.31.199.59,114.31.199.58(跳13、14和15),因为这些跳分别地理定位到美国、新西兰和美国。值得注意的是,在这种情况下,RTT是没有帮助的,因为序列的每

一跳都在MPLS隧道中(由第四列中的MPLS标签指示),并且因此RTT反映在隧道末端的RTT,而不是在各个IP地址处的RTT。

[0204] 从vps01.nyc1到203.209.210.16,处于UTC 1434920785(2015-06-21-21:06:25)

[0205]

TTL	IP	RTT	MPLS	ASN	CC	City
0	192.170.146.138	0.0	--	29791	US	New York
2	63.251.26.29	0.338	--	13789	US	New York
3	216.52.95.70	0.794	--	13789	US	New York
4	38.88.194.85	0.897	--	174	US	New York
5	154.54.47.17	1.347	--	174	US	New York
6	154.54.27.157	13.565	--	174	US	Cleveland
7	154.54.44.85	20.738	--	174	US	Chicago
8	154.54.6.85	32.995	--	174	US	Kansas City
9	154.54.30.53	70.842	--	174	US	San Francisco
10	154.54.28.34	71.549	--	174	US	San Jose
11	154.54.1.162	72.421	--	174	US	San Jose
12	38.122.92.2	75.331	--	174	US	San Jose
13	114.31.199.242	241.947	16521	4826	US	San Jose
14	114.31.199.59	237.467	289900	4826	NZ	--
15	114.31.199.58	237.879	16721,16946	4826	US	San Jose
16	114.31.199.28	241.783	16946	4826	AU	Sydney
17	175.45.72.119	237.379	16955	4826	AU	Sydney
18	114.31.196.163	237.158	16741	4826	AU	Melbourne
19	114.31.196.38	237.027	--	4826	AU	Melbourne
20	175.45.117.218	252.021	--	4826	AU	Melbourne
21	203.209.196.49	249.522	--	23681	AU	Melbourne
22	203.209.196.61	252.346	3154	23681	AU	Melbourne
23	203.209.210.16	269.123	--	23681	AU	Melbourne

[0206] 但是,在三明治中的IP地址的反向DNS(在下面所示的名称中的“sjc”和“ca”)中存在地理定位线索,暗示加利福尼亚州圣何塞(San Jose,CA)是所有三个IP地址的地理定位。

[0207] 114.31.199.242 bundle-101.cor01.sjc01.ca.VOCUS.net

[0208] 114.31.199.59 bundle-100.cor02.sjc01.ca.VOCUS.net

[0209] 114.31.199.58 bundle-100.cor01.sjc01.ca.VOCUS.net

[0210] 从圣何塞中的位置的进一步测试显示114.31.199.59在圣何塞的2.157毫秒内,从而使得新西兰对于其地理定位不可信,因为由于光速约束,从圣何塞到新西兰惠灵顿(Wellington,NZ)的最小可能的RTT为107毫秒。

[0211] 从San Jose,CA到114.31.199.59的跟踪路由:

[0212] 到114.31.199.59(114.31.199.59)的跟踪路由,30跳最多;60字节分组

[0213] 1 v199.mag01.sjc01.atlas.cogentco.com(66.250.250.113)0.338ms0.347ms

[0214] 2 te0-4-0-1.ccr22.sjc01.atlas.cogentco.com(154.54.84.153)0.698ms te0-4-0-1.ccr21.sjc01.atlas.cogentco.com(66.28.4.157)0.628ms

[0215] 3 be2095.rcr21.b001848-1.sjc01.atlas.cogentco.com(154.54.3.138) 1.242ms 1.252ms

[0216] 4 38.122.92.2(38.122.92.2)1.128ms 38.122.93.2(38.122.93.2)1.120ms

[0217] 5 bundle-101.cor01.sjc01.ca.VOCUS.net(114.31.199.242)1.348ms1.166ms

[0218] 6 bundle-100.cor02.sjc01.ca.VOCUS.net(114.31.199.59)2.157ms*

[0219] 地理定位服务器得出结论,114.31.199.59距圣何塞2.157毫秒,这意味着真正的地理定位距离该圣何塞收集器不远于215公里。因此,新西兰不是可信的地理定位。地理定位服务器还检测用于这个IP地址的rDNS中嵌入的圣何塞的“sjc”机场代码,进一步支持将圣何塞作为其实际地理定位,并且服务器将这个IP地址的地理定位校正为加利福尼亚州的圣何塞。

[0220] 边缘等待时间地理定位

[0221] 在一些实现中,地理定位服务器使用边缘等待时间算法来推断相邻跳的地理定位。在检查跟踪路由中的倒数第二跳的地理定位以查看是否它通知最终目标的地理定位时,这尤其有用,特别是当最终目标是在其反向DNS中没有地理定位提示的通常被网络工程师称为眼球网络的最终用户网络时。在这种情况下,倒数第二跳可能是基础设施IP地址,例如数据中心路由器,其中数据中心的位置嵌入在其rDNS中。

[0222] 该方法通过在构造跟踪路由中且在MPLS隧道外部看到的相邻IP地址之间的等待时间差的分布来实现。在所有这些对的RTT之间计算中值等待时间差。计算出的等待时间差然后用于估计该对中的一个IP地址的地理定位,这里假设另一个IP地址的地理定位是正确的。然后,地理定位服务器将计算出的中值与每对相邻IP地址的假定地理定位的最小可能的RTT进行比较。如果前者小于后者,那么IP地址中的一个或两者的地理定位可能不正确。此外,在一些实现中,如果该对中的一个IP地址的地理定位得到很好的支持,那么地理定位服务器过程将另一个IP地址的地理定位估计为如由中值RTT和光纤中的给定光速限定的距离其邻居的半径内。

[0223] 倒数第二跳示例

[0224] Microsoft在怀俄明州Cheyenne附近的Laramie县具有数据中心。考虑从Cheyenne以东的新罕布什尔州朴茨茅斯(Portsmouth,New Hampshire)收集器的跟踪路由,以及从Cheyenne以西的华盛顿西雅图收集器的一个跟踪路由,如下所示。

[0225] Portsmouth到191.234.85.3[mpr＝27.69ms]:

[0226] 到191.234.85.3(191.234.85.3)的跟踪路由,30跳最多,60字节分组

[0227] 1 rtr01.psm1.renesys.com(10.200.0.7)0.193ms 0.189ms 0.188ms

[0228] 2 hsrp2.psm1.renesys.com(195.160.236.3)2.381ms 2.395ms 2.397ms

[0229] 3 ray-b2.worldpath.net(64.140.193.25)0.470ms 0.516ms 0.516ms

[0230] 4 bst-edge-05.inet.qwest.net(63.239.32.25)6.178ms 6.188ms 6.188ms

[0231] 5 nyc-edge-04.inet.qwest.net(205.171.30.62)8.228ms 8.238ms 8.238ms

[0232] 6 63.151.150.98(63.151.150.98)8.462ms 8.129ms 8.120ms

[0233] 7 ae0-0.nyc-96cbe-1b.ntwk.msn.net(207.46.38.113)8.038ms 8.104ms 8.257ms

[0234] 8 ae6-0.was02-96cbe-1c.ntwk.msn.net(191.234.84.142)14.889ms

14.797ms＊

[0235] 9＊＊＊

[0236] 10＊＊＊

[0237] 11＊＊＊

[0238] 12＊＊＊

[0239] 13＊＊＊

[0240] 14 ae7-0.den01-96cbe-1a.ntwk.msn.net(191.234.84.222)52.028ms 52.041ms 52.044ms

[0241] 15 ae8-0.cys01-96cbe-1a.ntwk.msn.net(191.234.80.191)56.479ms 56.490ms 54.767ms

[0242] 16 191.234.85.3(191.234.85.3)54.487ms 54.500ms 57.758ms

[0243] Seattle到191.234.85.3[mpr＝15.45ms]：

[0244] 到191.234.85.3(191.234.85.3)的跟踪路由,30跳最多,60字节分组

[0245] 1 173.208.32.170.rdns.pingpipe.com(173.208.32.170)0.125ms 0.058ms 0.055ms

[0246] 2 v3508.er01.sea.ubiquity.io(23.105.64.1)6.931ms 7.117ms 7.069ms

[0247] 3 38.88.0.25(38.88.0.25)0.976ms 1.161ms 1.111ms

[0248] 4 154.24.19.33(154.24.19.33)1.560ms 1.510ms 1.689ms

[0249] 5 154.24.42.225(154.24.42.225)1.653ms 1.580ms 1.753ms

[0250] 6 te0-1-0-7.ccr22.sea01.atlas.cogentco.com(154.54.41.145)1.567ms 1.325ms 1.459ms

[0251] 7 be2084.ccr21.sea02.atlas.cogentco.com(154.54.0.254)1.740ms 1.946ms 2.126ms

[0252] 8 38.104.126.78(38.104.126.78)1.177ms 1.134ms 1.076ms

[0253] 9＊＊＊

[0254] 10＊＊＊

[0255] 11 ae15-0.cys01-96cbe-1a.ntwk.msn.net(191.234.84.11)55.000ms 55.482ms 55.377ms

[0256] 12 191.234.85.3(191.234.85.3)55.327ms 55.269ms 55.436ms

[0257] 两次跟踪中的倒数第二跳包含嵌入的rDNS。这些跳中的每一个都在其rDNS中具有cys01(CYS是Cheyenne的IATA机场代码)。最终目标的等待时间通常在倒数第二跳的1毫秒内,这强烈地暗示倒数第二个IP地址和目标IP地址位于同一个数据中心。考虑到倒数第二跳和目标之间的等待时间以及倒数第二跳的rDNS,地理定位服务器将这个目标IP地址置于与倒数第二跳相同的位置,即,怀俄明州的Cheyenne。

[0258] 将IP别名和MPLS标签用于地理定位

[0259] 网络互连设备(例如,路由器、防火墙和交换机)通常具有许多不同的网络接口;这些接口中的每一个可以具有唯一的IP地址,并且可以物理地连接到某个其它设备(通常通过铜缆或光缆)。例如,单个计算机设备在单个地理定位中,分配给其(潜在地多个)接口的所有IP地址也是如此。查找属于单件装备的所有IP地址的过程被称为去别名。

[0260] 在一些实现中,除了基于MPLS标签的技术之外,上述地理定位服务器可以使用一个或多个去别名技术(例如,墨卡托(Mercator)技术)来查找别名IP地址。这些技术使得地理定位服务器能够确定IP别名集合,并且然后推断每个集合的公共地理定位。这样的集合通常表示互联网上单个路由器上的接口。因此,地理定位服务器可以更准确地确定整个集合的地理定位,并且可以校正许多观察到的跟踪路由路径的地理定位,并且向一个或多个用户通知或提醒关于与所讨论的设备相邻的跳的潜在地理定位。

[0261] 墨卡托技术可以包括向选定的IP地址的随机端口发送任意分组,并且观察何时从不是特定跟踪路由数据收集器的目标的IP地址返回端口不可达消息。当从与跟踪路由数据收集器的目标不同的IP地址接收到适当的消息时,有可能推断出IP地址具有一个或多个IP别名。全球的跟踪路由数据收集器集合可以以这种方式以在全球跟踪路由中观察到的所有IP地址为目标,从而收集可信的IP别名对。此外,从这种方法新发现的IP地址也可以添加到IP地址的列表以迭代的方式进行探测,从而允许发现更多的IP别名。可以经由可转接闭合过程将一致观察到的各对收集到公共集合中。

[0262] 替代的实现涉及观察发送到地理定位服务器的全球跟踪路由数据中的MPLS标签的公共序列。对于给定的序列,标签将不改变,但是当不同的跟踪遍历不同的路由器接口时,遇到的IP地址可能不同。然后,地理定位服务器可以使用MPLS标签将不同的IP地址与同一设备相关联。

[0263] 为了理解MPLS标签可以如何用于将不同的IP地址与同一设备相关联,考虑如图6A所示的MPLS隧道600。MPLS隧道600包括耦合在也称为边缘LSR的入口标签边缘路由器(LER)602和出口LER 606之间的若干标签交换路由器(LSR)604a-604c(统称为LSR 604)。当入口LER 602接收分组时,它确定分组的转发等价类(FEC)及其标签交换路径(LSP)、为该分组创建MPLS报头、并且在将分组在MPLS隧道600中传送到第一LSR 604a之前将适当的标签插入到MPLS报头中。MPLS报头中的标签指定入口和出口之间的MPLS隧道600中的节点(在这个示例中为LSR 604)。

[0264] 在接收到分组后,第一LSR 604a检查分组的MPLS报头中的标签以确定分组的目的地。但是,与其它路由器不同,第一LSR 604a不一定具有任何IP路由信息。相反,它简单地检查分组的MPLS报头中的标签以确定分组在MPLS隧道600内的下一个目的地(这里是第二LSR 604b)。第一LSR604a更新MPLS报头,然后将分组传送到LSR 604b,以此类推,直到分组到达具有完整的IP路由表并适当地路由分组的出口LER 606。

[0265] 由于LSR 604使用特定的MPLS标签信息而不是完整的IP路由表执行,因此它们可以相对快地路由流量。但是由于它们依赖于MPLS标签而不是IP路由信息,因此LSR 604不一定将流量路由到MPLS隧道600之外的目的地。这意味着如果第一LSR 604a从入口LER 602接收TTL=1的分组,那么它将把分组转发到第二LSR 604b,而不是向入口LER 602返回“时间超时”消息。第二LSR 604b将把分组转发到第三LSR 604c,第三LSR 604c又将分组转发到出口LER606,其向入口LER 602返回“时间超时”消息。如果入口LER 602分别向第二LSR 604b和第三LSR 604c传送具有为2和3的TTL的分组,那么会发生同样的事情:它们被转发到出口LER 606,出口LER606向入口LER 602返回“时间超时”消息。出口LER 606还响应于接收到TTL=4的分组而(正确地)返回“时间超时”消息。因此,虽然报告的到达中间跳的计时实际上相对于LER 606,但是这个实际的终点在TTL达到4之前是看不见的。鉴于互联网上MPLS隧

道的普遍存在，因此寻找MPLS隧道并且然后忽略MPLS隧道中的中间跳等待时间对于在地理定位中使用等待时间提供了显著的优点。

[0266] 图6B图示了用于在跟踪路由数据中使用MPLS标签来估计特定路由器的地理定位的过程600。给定通过已知MPLS域的跟踪，地理定位服务器确定由入口LER用于选择LSP的FEC(步骤602)。对于跟踪中的每个MPLS跳，地理定位服务器将MPLS跳的IP地址映射到跟踪中的该点处的MPLS标签序列(步骤604)。地理定位服务器在选定的时间帧内在通过MPLS域的所有跟踪路径序列上重复该IP地址映射(步骤606)。地理定位服务器聚合在每个唯一传入MPLS标签序列上看到的所有IP地址(步骤608)，然后使用等待时间、rDNS信息或两者估计聚合IP地址的地理定位(步骤610)。

[0267] 作为示例，考虑列出遇到的IP地址和MPLS标签的以下三个跟踪序列。在所有三种情况下，MPLS标签序列是相同的。下面加下划线的IP地址属于同一路由器，并且由标签序列(1314,1496,1793)来识别。下面斜体的IP地址属于不同的公共路由器，并且由标签序列(1314,1496,1793,1807,1609)来识别。地理定位服务器可以将共享公共标签序列的通过同一MPLS域跨多个跟踪序列看到的MPLS IP地址识别为属于同一路由器。由于MPLS IP地址属于同一路由器，因此它们具有相同的地理定位。

[0268]

跟踪片段#1		跟踪片段#2		跟踪片段#3	
IP 地址	MPLS 标签	IP 地址	MPLS 标签	IP 地址	MPLS 标签
4.69.143.238	1314	4.69.143.238	1314	4.69.143.238	1314
4.69.161.114	1496	4.69.161.114	1496	4.69.161.114	1496
4.69.137.58	1793	4.69.137.50	1793	4.69.137.54	1793

[0269]

4.69.134.154	1807	4.69.134.154	1807	4.69.134.154	1807
4.69.134.129	1609	*4.69.134.137*	1609	*4.69.134.141*	1609

[0270] 使用上述技术，地理定位服务器收集其中有强有力的证据表明它们属于单件装备并且因此处于单个地理定位的IP地址集合。然后，地理定位服务器尝试使用等待时间和/或DNS信息将每个集合地理定位到公共位置或一组位置。

[0271] 对于通过墨卡托子过程找到的IP别名集合，IP地址本身通常是直接可能的。地理定位服务器从多个位置ping给定集合的成员，以寻找最靠近的收集器，并基于等待时间测量结果计算一组可信的地理定位。

[0272] 在一些实现中，地理定位服务器从最靠近的收集器使用三角测量技术。地理定位服务器反向解析每个集合中的所有IP，从而查找机场代码、城市名称或其它地理缩写。使来自DNS信息的一组位置与来自等待时间测量结果的一组位置相交可以提供一组较小的潜在地理定位。如果该集合为空，那么DNS信息(其是人类输入的并且容易出错)被地理定位服务器忽略。每个IP别名集合的最终结果是一组可信的地理定位，其与所有观察到的等待时间测量结果一致，并且在可能的情况下，与从DNS标签导出的地理“提示”一致。

[0273] 对于经由MPLS标签找到的IP别名集合，IP地址本身不一定是直接可能的，并且可

能仅在经由跟踪路由转接到其它目的地时才可观察到。而且，在这种跟踪路由测量结果中观察到的等待时间信息可能不用于MPLS隧道的中间跳，因为计时可能是相对于隧道的末端。因此，通常，对于从跟踪路由导出的任何等待时间测量结果，地理定位服务器可以丢弃来自MPLS隧道中的中间跳的等待时间，其可以由地理定位服务器经由如上所述的MPLS标签的使用来识别。在直接探测或间接获得准确等待时间信息不可行的情况下，地理定位服务器可以恢复到DNS标签来地理定位属于MPLS隧道的IP别名。

[0274] 图7A和图7B图示了由地理定位服务器和过程利用的IP去别名技术的示例。下面提供了由地理定位服务器用于识别公共路由器上的IP地址并且然后改进整个集合的地理定位估计的墨卡托技术的示例。以下12个IP地址被识别为一组相关的IP别名：

[0275] 68.86.83.46,68.86.83.42,68.86.83.38,68.86.83.34,68.86.82.94,50.242.148.85,68.86.82.82,68.86.85.218,68.86.82.86,68.86.82.90,23.30.206.41,23.30.206.153

[0276] 地理定位服务器通过使用图7A所示的过程700在图中查找连接的组件来识别该集合。即，如果对IP地址A的墨卡托探测返回IP地址B，那么IP地址A和B可以彼此相关联。这种关联可以用图论的术语描绘为两个节点，A和B各有一个，在它们之间具有有向边。如果A是B的别名并且B是C的别名，那么A、B和C是彼此的别名。

[0277] 换句话说，过程700包括从传感器向可以表示路由器接口的选定IP地址的随机端口发送任意分组(例如，传输控制协议或用户数据报协议分组)。传感器测量往返等待时间并观察端口不可达消息(步骤704)。如果传感器接收到端口不可达消息，那么传感器或地理定位服务器确定端口不可达消息是否从与任意分组的目的地不同的IP地址返回(步骤706)。如果是，那么地理定位服务器确定目的地IP地址具有一个或多个IP别名，包括返回端口不可达消息的IP地址(步骤708)。然后，地理定位服务器确定所有IP别名处于相同的地理定位(步骤710)，并使用上述技术来估计其地理定位。

[0278] 以上给出的12个IP地址及其学到的关联可以由图7B中所示的连接图来表示，其指示它们都是相关的。图7B示出这些12个IP地址中的7个形成强连接的子图，意味着每个IP地址在两个方向上“可见”。即，探测A返回组件的至少一个成员，而探测组件的一个或多个成员也返回A。(虽然强连接的组件提供非常强的别名证据，但是对于我们的目的而言我们只需要连接的组件)。

[0279] 基于上述连接性，地理定位服务器可以推断哪些IP属于同一个设备并且因此应该位于相同的地理定位。等待时间测量结果指示所有这些IP地址都在加利福尼亚州圣何塞附近。所有这些IP地址的DNS信息都指向圣何塞的Great Oaks社区。例如，图7中的68.8.86.83.46节点705解析为17pe02.11greatoaks.ca.ibone.comcast.net。基于由地理定位服务器执行的自动分析，所有12个IP地址都位于美国加利福尼亚州圣何塞的Great Oaks。

[0280] 在撰写本文时，三个商业IP地理定位提供商对这12个IP地址给出以下地理定位估计：

[0281] 提供商1:美国，没有指定城市

[0282] 提供商2:若干美国城市，包括弗吉尼亚州的阿什本(Ashburn,Virginia)、乔治亚州的玛丽埃塔(Marietta,Georgia)、科罗拉多州的丹佛(Denver,Colorado)和加利福尼亚

州的洛杉矶(Los Angeles,California)。

[0283] 提供商3:若干美国城市,包括华盛顿州的西雅图(Seatle,Washington)、科罗拉多州的丹佛(Denver,Colorado)、德克萨斯州的达拉斯(Dallas,Texas)和乔治亚州的玛丽埃塔(Marietta,Georgia)。

[0284] 这是相当典型的商业上可用的IP地理定位数据。很难找到将所有别名置于相同位置的商业地理定位提供商,并且看见置于不同国家的别名并不罕见(特别是对于具有国际连接性的路由器)。

[0285] 转接树

[0286] 在一些实现中,地理定位服务器可以识别在国家级别不正确的地理定位,用于随后使用等待时间和DNS解析进行校正和改进。为了实现这一点,地理定位服务器执行从来自多个(例如,超过600个)对等会话的BGP路由数据导出的分析。地理定位服务器从BGP数据收集器接收包含关于到互联网上的每个路由网络前缀的自治系统(AS)路径的信息的BGP数据。每个AS路径包含表示交换路由并维持某种商业关系的相邻自治系统的AS-AS边缘。

[0287] 图8A图示了由地理定位服务器生成的从其起源转接出到互联网核心的单个网络前缀。在一些实现中,地理定位服务器采用机器学习分类器来将这些AS边缘分类为几个不同类别(标记)中的一个:转接、对等、集群、交换等。可以预计本地或区域转接提供商在有限的地理范围内运营,并且因此转接地理定位到其运营国家的网络前缀。生成的BGP边缘标记之后被地理定位服务器利用,以确定每个网络前缀如何从其起源转接出到互联网的核心,例如,图8A中示出的转接树。

[0288] 图8B图示了用于基于在相同转接边缘上承载的其它前缀的假定地理定位来确定所讨论的前缀的可能地理定位的过程800。在步骤802中,地理定位服务器基于生成的边缘标记来计算所讨论的前缀的转接树。接下来,地理定位服务器检查转接树的边缘上承载的一些或全部前缀的地理定位(步骤804)。为了观测到前缀的每个边缘进行转接,地理定位服务器计算跨边缘的所有前缀的地理分布,并且基于跨该边缘观察所讨论的前缀的BGP对等方的数量对地理分布进行加权(步骤806)。地理定位服务器将这些加权的地理分布组合成为所讨论的前缀建议的国家级别地理定位(在可能的情况下)(步骤808)。在步骤810中,地理定位服务器将从转接树推断出的地理定位与由被检查的前缀的一个或多个第三方报告的(一个或多个)地理定位进行比较。(例如,如上所述,地理定位服务器可以自动获得报告的地理定位)。如果推断出的地理定位与报告的(一个或多个)地理定位不匹配,那么地理定位服务器例如使用上述技术来利用等待时间测量结果验证前缀的地理定位(步骤812)。

[0289] 关于由一个主要商业提供商地理定位到俄勒冈州波特兰(Portland,Oregon)的前缀118.150.0.0/20,图示了图8A中的转接树的示例实用工具。检查这个前缀的转接树显示转接边缘很大程度上携带该提供商地理定位到中国台湾地区的前缀,并且大多数BGP对等方观察到这些边缘,尤其是那些最靠近前缀的起源的边缘。由地理定位服务器执行的评分指示该前缀最有可能的地理定位实际上是中国台湾地区,美国则处于遥远的第二处。等待时间测量结果确认这个前缀位于中国台湾地区的可能性以及其位于美国的完全不可信性。

[0290] 路由时间相关的事件

[0291] 在一些实现中,地理定位服务可以通过使用历史BGP中断和不稳定性数据来识别不正确的地理定位用于随后的校正和改进。为了实现这一点,地理定位服务器对从来自多

个(例如,超过600个)对等会话的BGP路由数据导出的互联网上的路由前缀中的一些或全部执行中断和不稳定性计算,并将“事件”识别为在大致同一时间展示某些行为的前缀集合。随着时间的推移,这些事件的相关性通常揭示了联网基础设施和这些前缀的路由路径的共同点。

[0292] 这些计算出的事件通过表示特定地理定位中的大型网络中断和修复。例如,地理定位服务器观察到其中同时撤回121个前缀-这些前缀中的96%是通过商业服务地理定位到印度的事件。同一天的十五分钟之后,地理定位服务器几乎看到所有这些前缀都返回。由于这对事件共同包含99个前缀,因此它们有可能捕获相同物理基础设施的故障和修复。它们中几乎全部都地理定位到印度的这一事实增加了对这一解释的支持。

[0293] 通过分析不位于印度的少数前缀,地理定位服务器能够生成潜在错误地理定位的候选集合。前缀恰好在这些时间被撤回和恢复而不参与相同网络中断的机会非常低,但不是零,并且因此它们的地理定位需要验证。

[0294] 例如,注册到美国威斯康星州拉辛(Racine,Wisconsin)的Modine Manufacturing公司的198.40.150.0/24是一个这样的前缀。在2011年,所有可用的商业服务都将这个前缀置于印度而且大部分仍然这样做。地理定位服务器观察到198.40.150.0/24由AS21758(Modine Manufacturing公司)通告并且经由AS 18101(Reliance Communications Mumbai)专门转接。AS 18101具有专门印度的覆盖范围,许多前缀在孟买(Mumbai)和钦奈(Chennai)。如果发现它们也在Modine所基于的威斯康星州拉辛地区提供互联网服务,那将是非常令人惊讶的。来自ARIN的注册机构在威斯康星州拉辛列出这个前缀,这无疑是这些常见错误地理定位的来源。

[0295] 等待时间数据表明这个前缀在印度钦奈。这并不令人惊讶,因为Modine在2008年12月通过新闻发布通告了在印度钦奈的新制造工厂。卫星局的前缀在总部的实际地址处注册并不罕见。地理定位服务器能够在从时间相关的BGP路由中断和恢复的集群中发现潜在的错误地理定位并且然后经由等待时间数据验证新的地理定位之后将这个前缀置于钦奈。虽然等待时间数据本身会足以识别这种级别的错误地理定位,但是BGP路由事件可以帮助在较小的地理范围内改进和校正错误地理定位,特别对于等待时间单独是不确定的邻近城市而言。

[0296] 图9图示了在短期的区域互联网不稳定期间的网络前缀的集合。时间沿着x轴表示,并且每个感兴趣的前缀沿着y轴的每一“行”显示。值得注意的是,模式清晰地显示出来,其允许显示公共不稳定模式的前缀进行算法上的聚类。地理定位服务器在这些集群内寻找地理不一致性,并且使用上述技术来验证任何异常值或使任何异常值无效。

[0297] 到任播实例的距离

[0298] 在一些实现中,地理定位服务可以使用查询全球任播网络的观察到的IP地址的地理分布来识别不正确的地理定位用于随后校正和改进。该系统可以包括广泛分布的全球任播网络,用于作为商业DNS服务的一部分提供权威的DNS响应,从而向全球数千万的递归提供答案。通过观察查询任播网络的每个实例的递归的地理分布,可以发现地理异常值并通过这里描述的技术进一步调查以便进行可能的校正。此外,类似rDNS,任播实例的地理定位是对该实例的查询者的位置的弱标识符。即,精心设计的任播网络往往将查询保持在本地。

[0299] 参数路径拟合

[0300] 在一些实现中，地理定位服务可以使用在跟踪路由边缘的下游观察到的IP地址的地理分布来识别不正确的地理定位用于随后的校正和改进。每个跟踪路由包含沿着从跟踪路由收集器到目的地IP地址的路径观察到的连续IP地址序列。沿着这样的路径的每对连续的IP地址可以被视为图论术语中的有向边。对于每个这样的边缘或连续边缘序列，地理定位服务器计算在遍历给定路径片段的所有跟踪路由的边缘下游观察到的IP地址集合，从而建立等待时间和跳在数据中典型地看起来像什么的模型。地理定位服务器然后在与每个这样的有向路径片段相关联的每组下游IP地址中寻找地理异常值。这些异常值可以通过这里描述的技术进行进一步调查以便进行可能的校正。

[0301] 非可地理定位的IP地址

[0302] 在一些实现中，地理定位服务器识别不能以任何合理程度的特异性(例如，除“地球”或非常大的区域(诸如大洲)之外)进行地理定位的IP地址。对于这种特异性缺乏存在若干原因。IP地址可能属于移动设备，并且因此可以自由漫游。IP地址可能位于卫星链路的另一端。卫星通常处于地球同步轨道(从而意味着超过500毫秒RTT的等待时间)并且具有覆盖大地理区域的宽波束。等待时间可以用于识别卫星连接，但IP的实际位置可能在卫星波束内的任何地方。

[0303] 此外，IP地址可以是任播，即，经由BGP路由从多个位置通告的。在这种情况下，这些IP地址没有单一的地理定位，因为它们同时在许多不同的位置。任播路由技术通常由内容提供商用于减少从流行内容的复制存储库到最终用户的等待时间。如前所述，由地理定位服务器通过等待时间测量结果识别为地理不一致的IP地址被标记为任播。

[0304] 地理定位提供商通常将它们包括在其数据库中的每个IP地址置于单个地理定位中。这种策略对于任播网络可能产生误导。例如，考虑Google在8.8.8.8上流行的公共DNS递归解析器。使用在前面的示例中使用的相同商业提供商，地理定位服务器输出以下地理定位：

[0305] 提供商1:美国加利福尼亚州山景城

[0306] 提供商2:美国加利福尼亚州山景城

[0307] 提供商3:美国加利福尼亚州山景城

[0308] 地理定位服务器确定Google DNS 8.8.8.8在110个跟踪路由数据收集器的5ms内，并且在26,704个不同的收集器对之间是地理不一致的，从而意味着高度的任播(即，许多实际的Google位置)。但是，地理定位服务器测量结果指示8.8.8.8的这些许多Google实例中没有一个实际上在山景城中，这一结论得到Google自己公开发布的信息的支持。地理定位服务器不仅可以使用等待时间测量结果发现任播前缀，而且它还可以识别每个任播实例的可能位置。

[0309] 结论

[0310] 虽然各种发明性实施例已经在本文中进行了描述和图示，但本领域普通技术人员将容易想到用于执行本文所述的功能和/或获得本文所述的结果和/或优点中的一个或多个的各种其它装置和/或结构，并且这些变化和/或修改当中每一个都被认为在本文所述的发明性实施例的范围内。更一般而言，本领域技术人员将容易认识到的是，本文所述的所有参数、维度、材料和配置都意在是示例性的，并且实际的参数、维度、材料和配置将取决于本发明性教导所用于的具体应用(一个或多个)。本领域技术人员将认识到，或者能够仅仅使

用常规实验来确定，本文所描述的具体发明性实施例的许多等价物。因此，应当理解的是，前述实施例仅仅是作为示例给出的，并且，在所附权利要求及其等价物的范围内，发明性实施例可以以不同于具体描述并要求保护的方式的方式来实践。本公开的发明性实施例涉及本文所述的每个单独的特征、系统、制品、材料、套件、和/或方法。此外，如果这种特征、系统、制品、材料、套件、和/或方法不相互不一致，那么两个或更多个这种特征、系统、制品、材料、套件、和/或方法的任意组合包括在本公开的发明性范围内。

[0311] 本发明的上述实施例可以以多种方式中的任何一种来实现。例如，一些实施例可以使用硬件、软件或其组合来实现。当实施例的任何方面至少部分地用软件实现时，软件代码可以在无论在单个计算机中提供或者在多个计算机之间分布的任何合适的处理器或处理器集合上执行。

[0312] 在这方面，本发明的各个方面可以至少部分地体现为利用一个或多个程序编码的计算机可读存储介质(或多个计算机可读存储介质)(例如，计算机存储器、一个或多个软盘、紧凑盘、光盘、磁带、闪速存储器、现场可编程门阵列或其它半导体器件中的电路构造、或者其它有形计算机存储介质或非瞬态介质)，当所述一个或多个程序在一个或多个计算机或其它处理器上执行时，执行实现以上讨论的本技术的各个实施例的方法。计算机可读介质(一个或多个)可以是便携式的，使得存储在其上的程序(一个或多个)可以被加载到一个或多个不同的计算机或其它处理器上，以实现如上讨论的本技术的各个方面。

[0313] 本文使用的术语“程序”或“软件”在一般意义上指任何类型的计算机代码或处理器可执行指令集，该计算机代码或处理器可执行指令集可以用来对计算机或其它处理器编程以实现如以上讨论的本技术的各个方面。此外，应当认识到的是，根据本实施例的一个方面，当被执行时执行本技术的方法的一个或多个计算机程序无需驻存在单个计算机或处理器上，而是可以以模块方式分布在多个不同的计算机或处理器当中以实现本技术的各个方面。

[0314] 计算机可执行指令可以以许多形式，诸如由一个或多个计算机或其它设备执行的程序模块。一般而言，程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等。通常，在各个实施例中可以如所期望地组合或分布程序模块的功能。

[0315] 此外，本文描述的技术可以体现为已经提供了至少一个示例的方法。作为该方法的一部分执行的动作可以以任何合适的方式进行排序。因此，实施例可以被构造为其中动作是以不同于图示的次序执行的，其可以包括同时执行一些动作，尽管在说明性实施例中示出为顺序的动作。

[0316] 如本文所定义和使用的，所有的定义都应该被理解为优先于字典的定义控制通过引用被结合的文档中的定义和/或所定义术语的普通含义。

[0317] 如本文在说明书和在权利要求中所使用的，除非明确地指示为相反，否则不定冠词“一”和“一个”应当被理解为意指“至少一个”。

[0318] 如本文在说明书和在权利要求中所使用的，短语“和/或”应当被理解为意在如此连接的元素的“任一个或两者”，即，在一些情况下结合存在并且在另一些情况下分离存在的元素。利用“和/或”列出的多个元素应该以同样的方式进行解释，即，如此结合的元素的“一个或多个”。除由“和/或”子句具体识别的元素之外，其它元素也可以可选地存在，无论

是与具体识别的那些元素有关还是无关。因此，作为非限制性的示例，当结合诸如“包括”的开放式语言使用时，对“A和/或B”的引用可以，在一个实施例中，仅仅指A(可选地包括除B之外的元素)；在另一个实施例中，仅仅指B(可选地包括除A之外的元素)；在还有的另一个实施例中，指A和B两者(可选地包括其它元素)；等等。

[0319] 如本文在说明书和权利要求中所使用的，“或”应当被理解为具有与如上定义的“和/或”相同的含义。例如，当分隔列表中的项时，“或”或“和/或”应当被解释为包括性的，即，包括多个元素或元素列表当中至少一个，但也包括一个以上，以及可选地，包括附加的未列出的项目。只有明确地指示相反的术语，诸如“仅…之一”或者“确切地…之一”，或者当在权利要求中使用时的“由…组成”，将指包括多个元素或元素列表当中确切一个元素。一般而言，如本文所使用的，当前面有排他性术语(诸如“任意一个”、“…之一”、“仅…之一”或者“确切地…之一”)时，如本文中使用的术语“或”将只被解释为指示排他性的替代(即“一个或另一个但不是两者”)。当在权利要求书中使用时，“基本上由…组成”将具有其如在专利法领域中使用的普通含义。

[0320] 如本文在说明书和权利要求中所使用的，短语“至少一个”在引用一个或多个元素的列表时应当理解为意味着从元素列表中的元素中的任何一个或多个元素中选择的至少一个元素，但是未必包括在元素列表内具体列举的每一个元素中的至少一个元素并且未排除元素列表中的元素的任意组合。这一定义也允许可以可选地存在除了在短语“至少一个”所引用的元素列表内具体识别的元素之外的、无论是与具体识别的那些元素有关还是无关的元素。因此，作为非限制性示例，“A和B中的至少一个”(或等价地，“A或B中的至少一个”，或等价地，“A和/或B中的至少一个”)可以，在一个实施例中，指至少一个，可选地包括多于一个A，其中不存在B(并且可选地包括除B之外的元素)；在另一个实施例中，指至少一个，可选地包括多于一个B，其中不存在A(并且可选地包括除A之外的元素)；在还有的另一个实施例中，指至少一个，可选地包括多于一个A，和至少一个，可选地包括多于一个B(并且可选地包括其它元素)；等等。

[0321] 在权利要求中，以及在上述说明书中，所有过渡短语(诸如“包含”、“包括”、“携带”、“具有”、“含有”、“涉及”、“持有”等)被理解为是开放式的，即，意味着包括但不限于。只有过渡短语“由…组成”和“基本上由…组成”才将分别是封闭或半封闭过渡短语，如在美国专利局专利审查程序手册2111.03节中所阐述的。

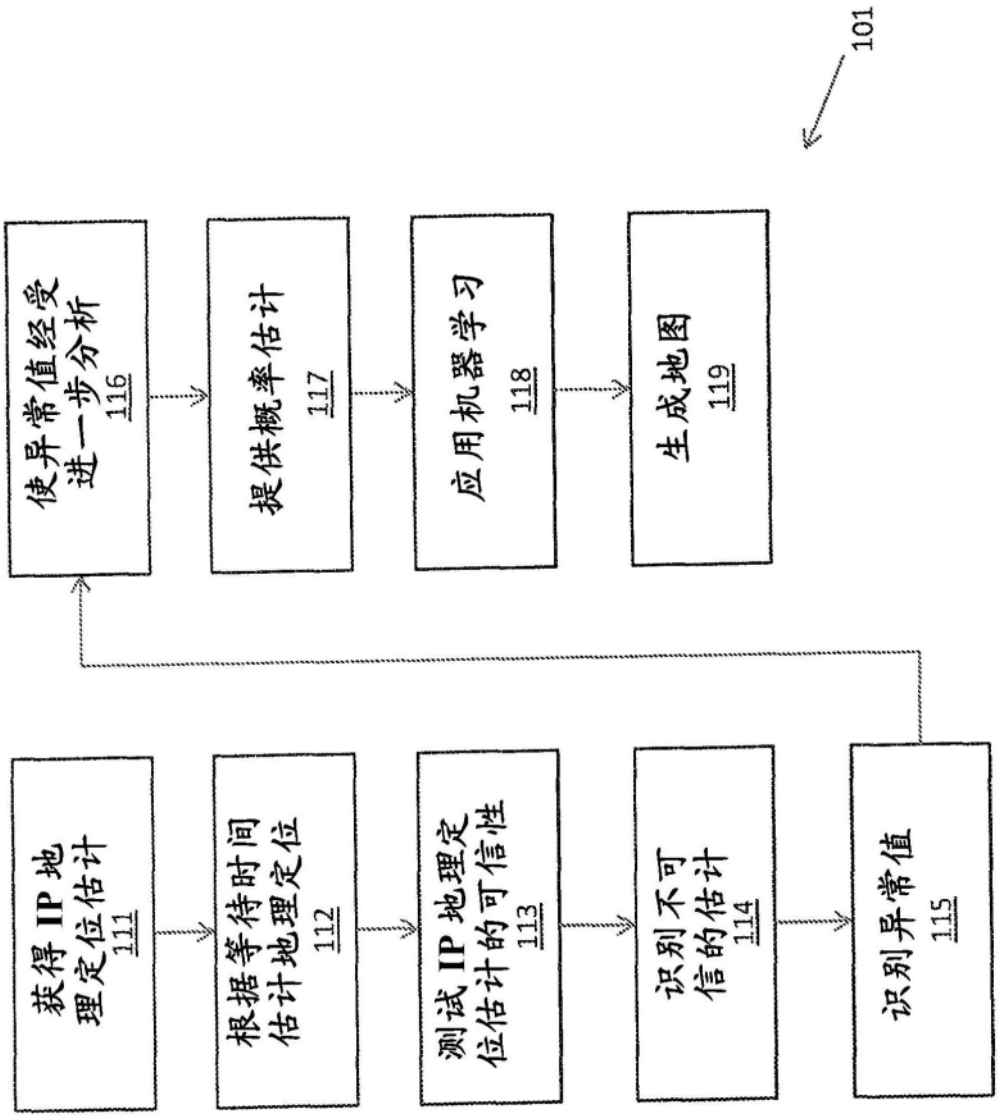

101
获得IP地理定位估计 111
根据等待时间估计地理定位 112
测试IP地理定位估计的可信性 113
识别不可信的估计 114
识别异常值 115
使异常值经受进一步分析 116
提供概率估计 117
应用机器学习 118
生成地图 119

图1A

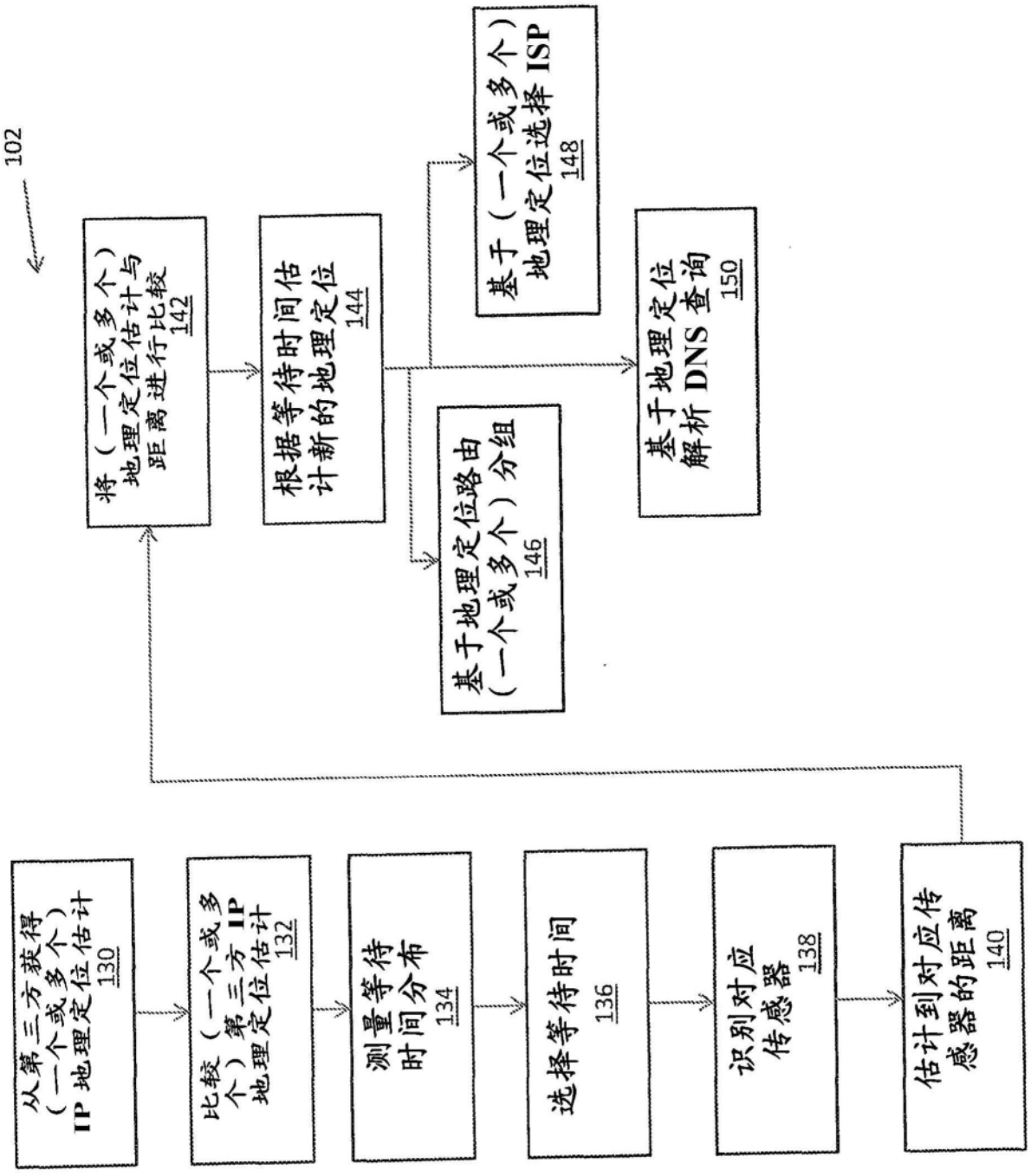

102
从第三方获得（一个或多个）IP 地理定位估计 130
比较（一个或多个）第三方 IP 地理定位估计 132
测量等待时间分布 134
选择等待时间 136
识别对应传感器 138
估计到对应传感器的距离 140
将（一个或多个）地理定位估计与距离进行比较 142
根据等待时间估计新的地理定位 144
基于地理定位路由（一个或多个）分组 146
基于（一个或多个）地理定位选择 ISP 148
基于地理定位解析 DNS 查询 150

图1B

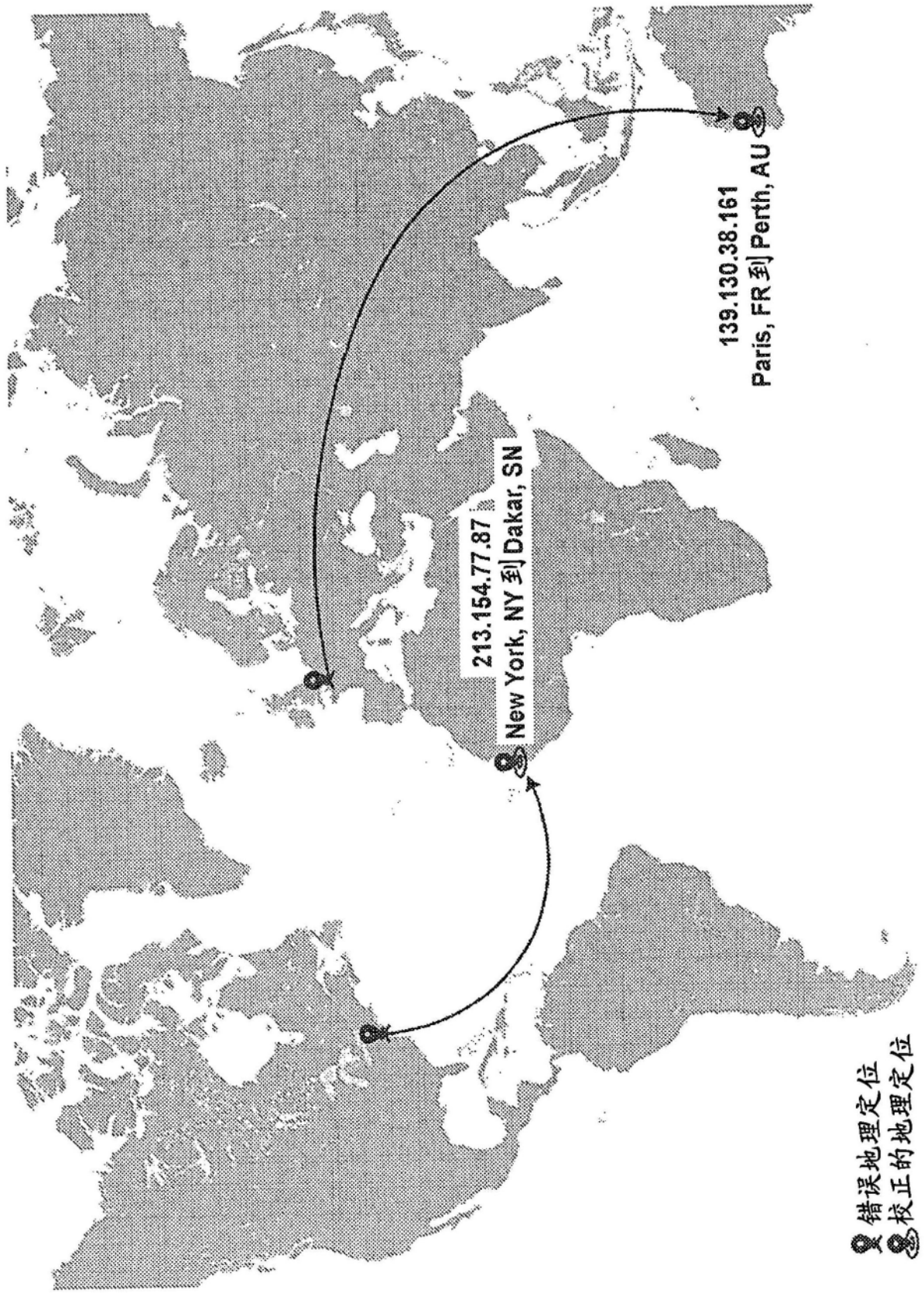

139.130.38.161
Paris, FR到Perth, AU
213.154.77.87
New York, NY到Dakar, SN
错误地理定位
校正的地理定位

图1C

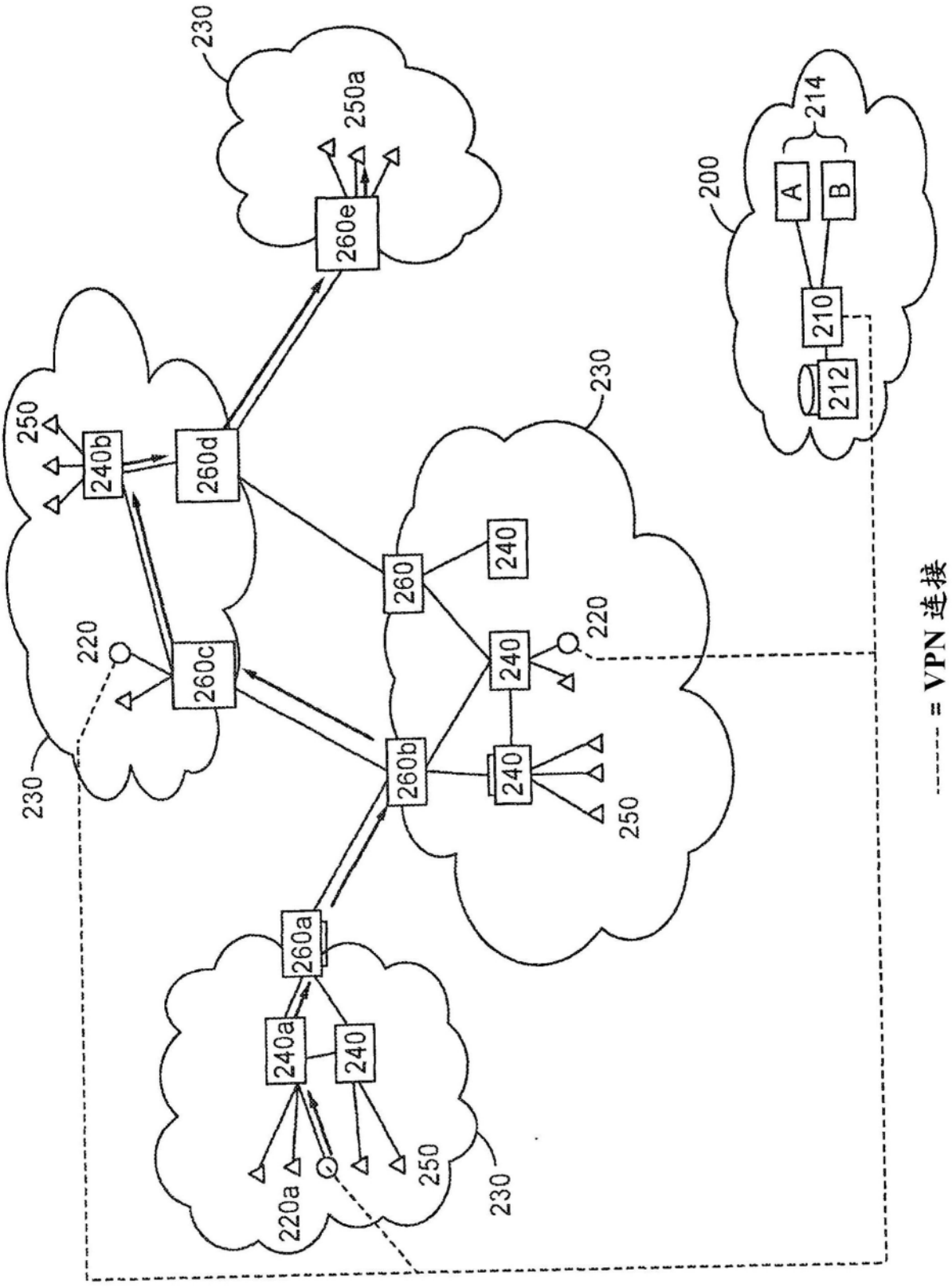

230
250a
260e
200
214
A
B
210
212
250
240b
260d
230
260
240
220
260c
240
220
230
260b
240
250
260a
240a
240
250
220a
230
----- = VPN 连接

图2A

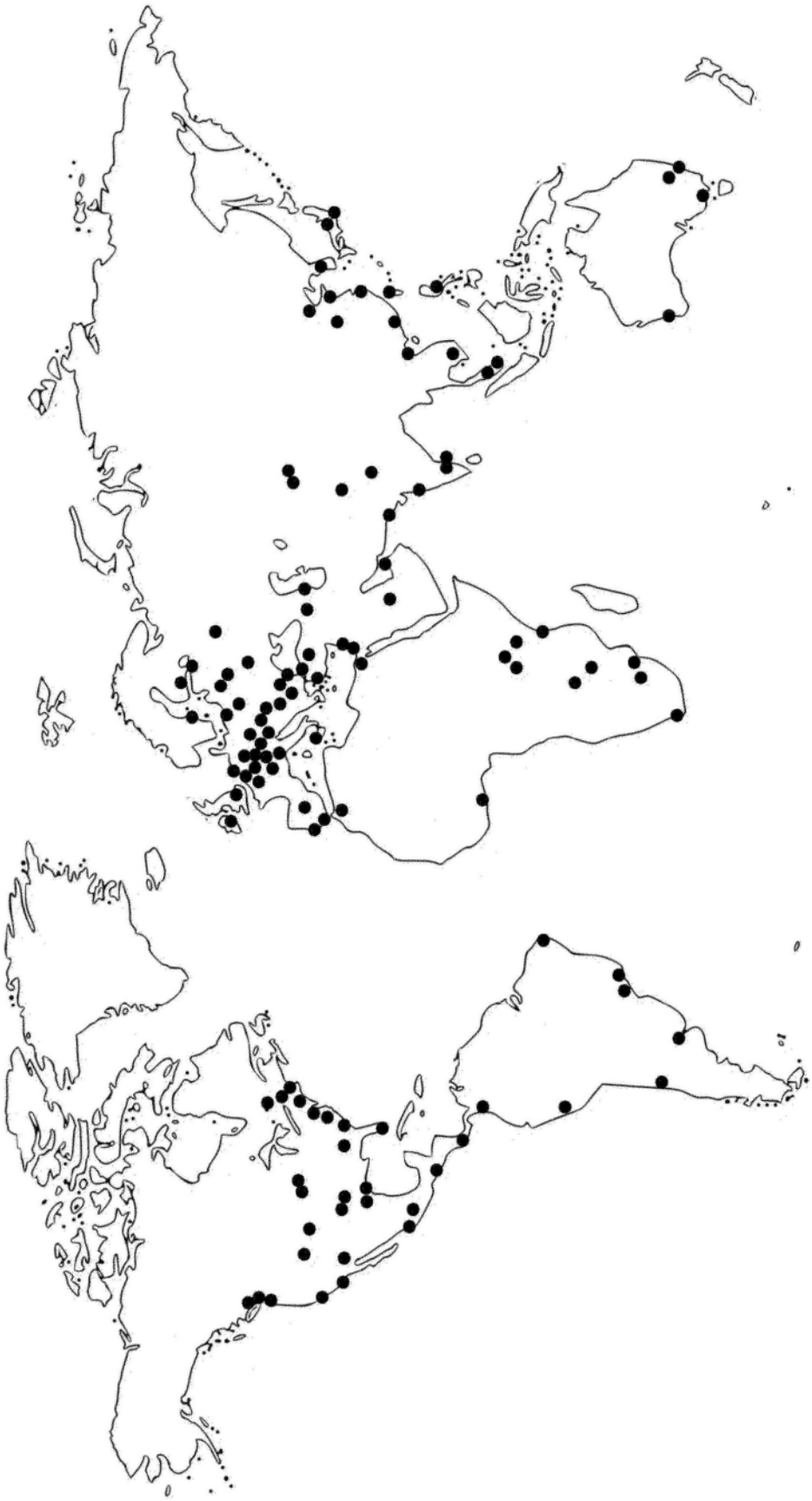


图2B

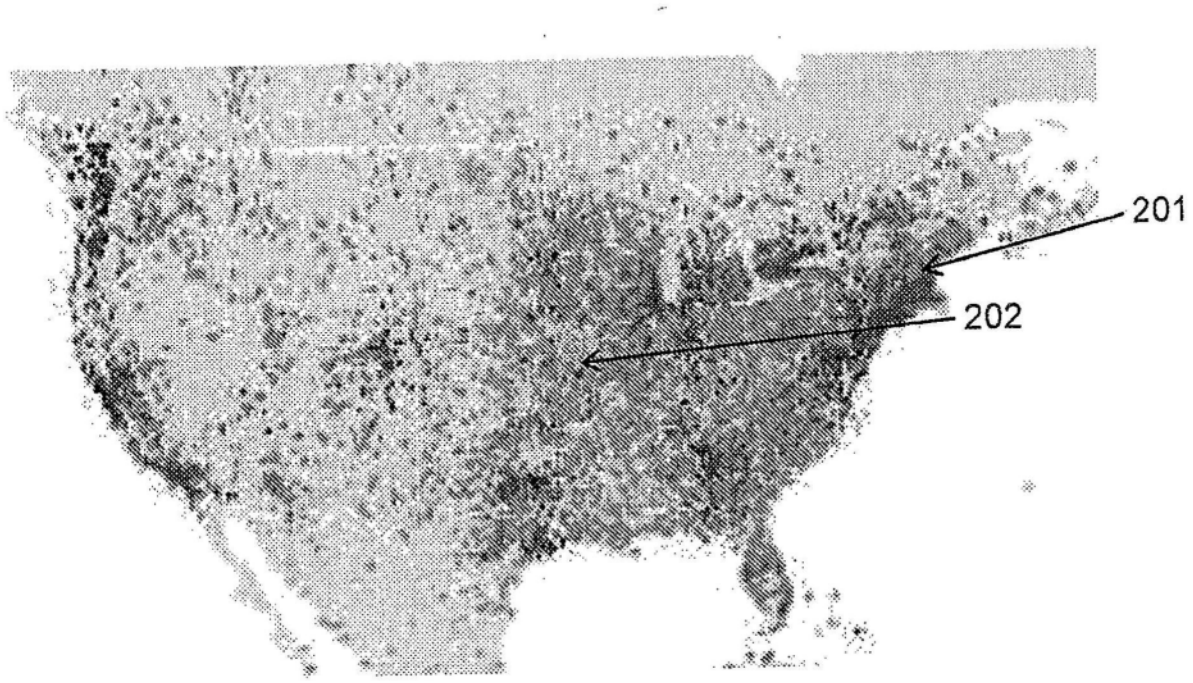


图2C

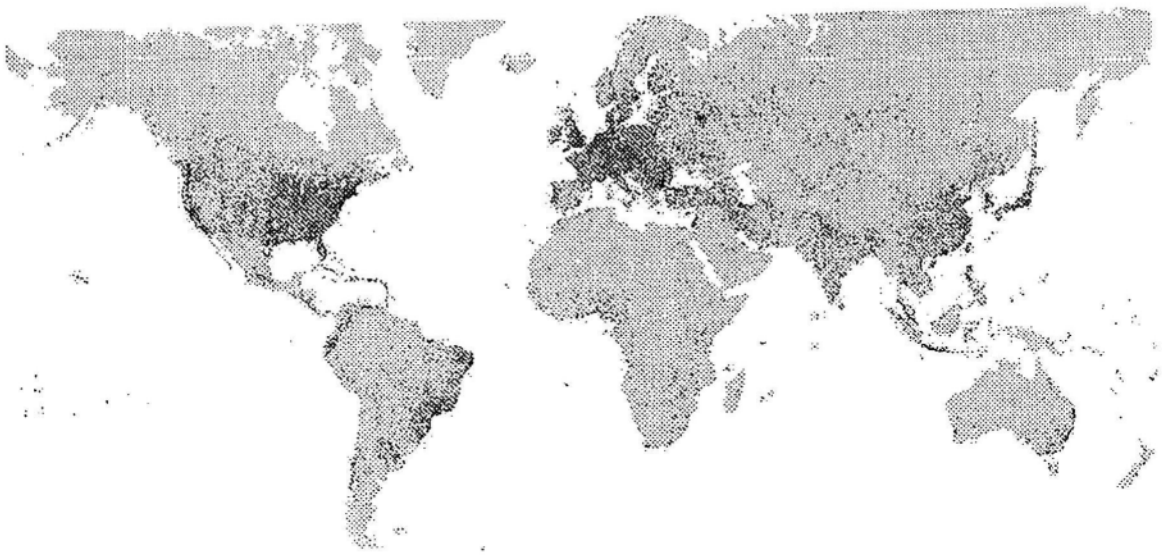


图2D

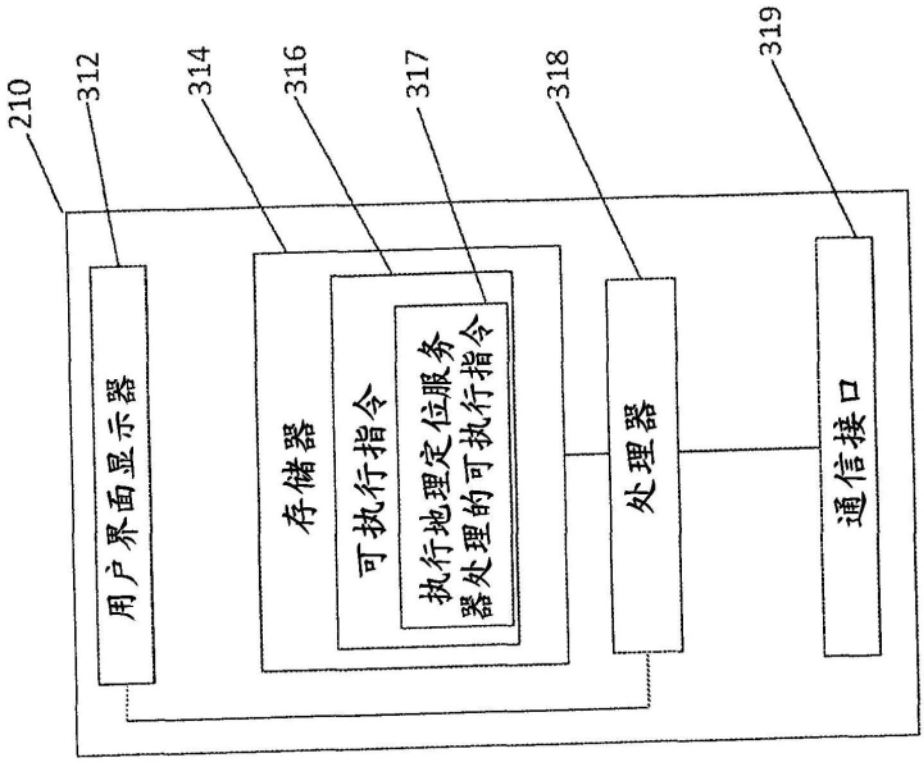

210
312
314
316
317
318
319
用户界面显示器
存储器
可执行指令
执行地理定位服务器处理的可执行指令
处理器
通信接口

图3A

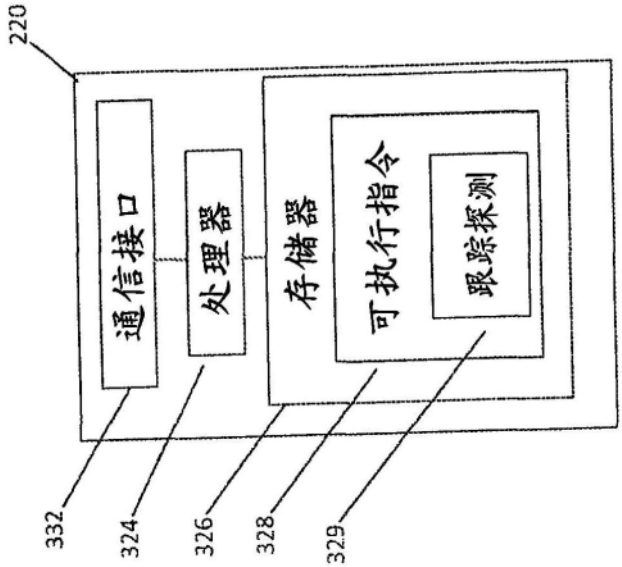

220
通信接口
处理器
存储器
可执行指令
跟踪探测
332
324
326
328
329

图3B

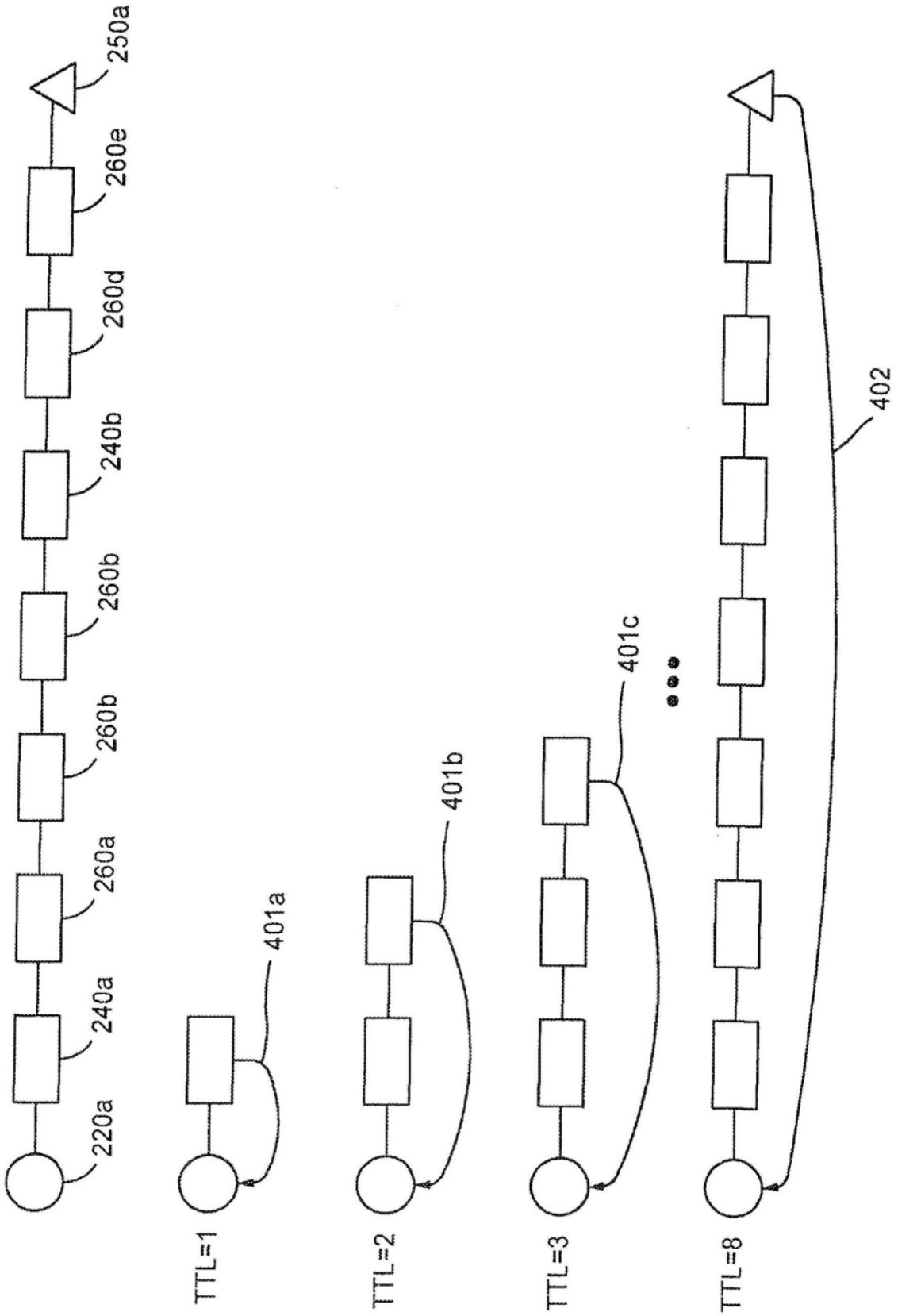

250a
260e
260d
240b
260b
260b
260a
240a
220a
401a
401b
401c
402
TTL=1
TTL=2
TTL=3
TTL=8

图4

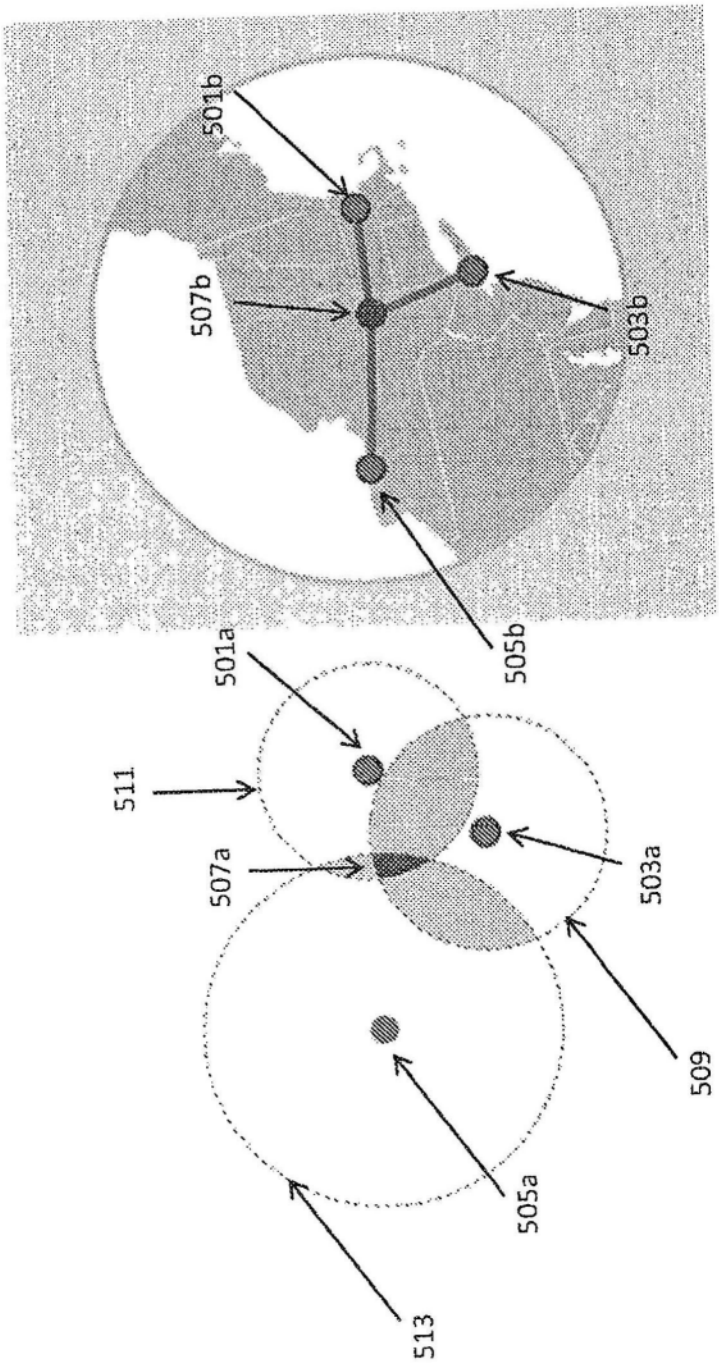


图5

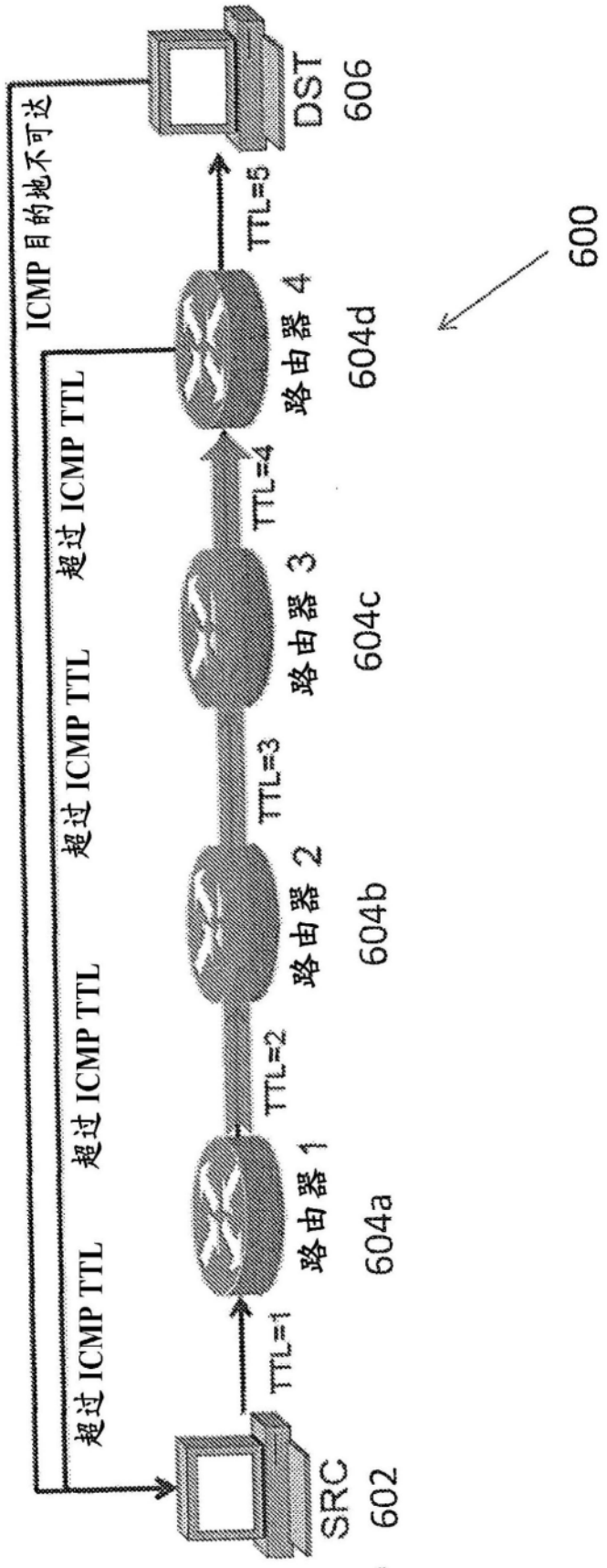

ICMP 目的地不可达
超过 ICMP TTL
超过 ICMP TTL
超过 ICMP TTL
SRC
602
TTL=1
路由器 1
604a
TTL=2
路由器 2
604b
TTL=3
路由器 3
604c
TTL=4
路由器 4
604d
TTL=5
DST
606
600

图6A

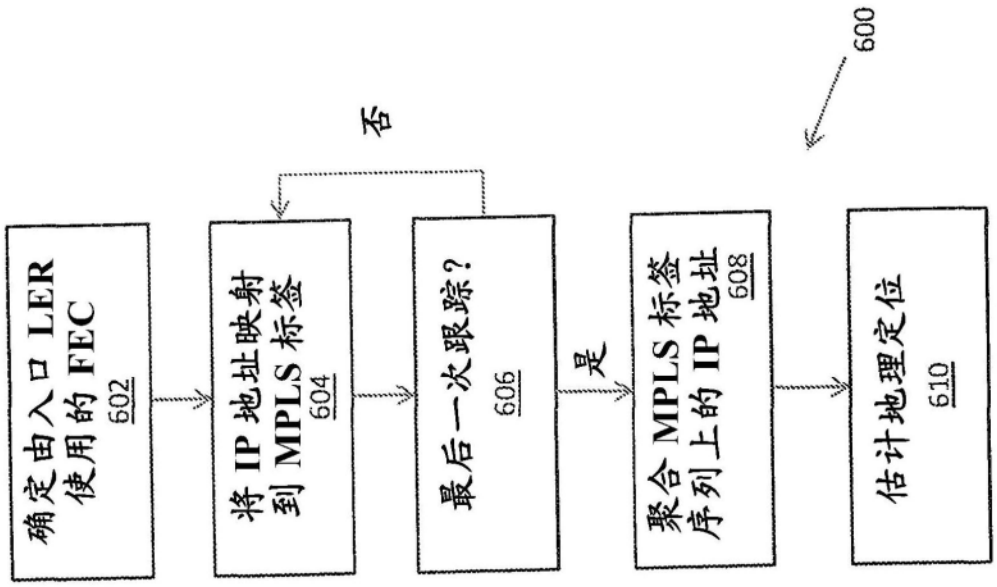

600
确定由入口 LER 使用的 FEC
602
将 IP 地址映射到 MPLS 标签
604
最后一次跟踪？
606
否
是
聚合 MPLS 标签序列上的 IP 地址
608
估计地理定位
610

图6B

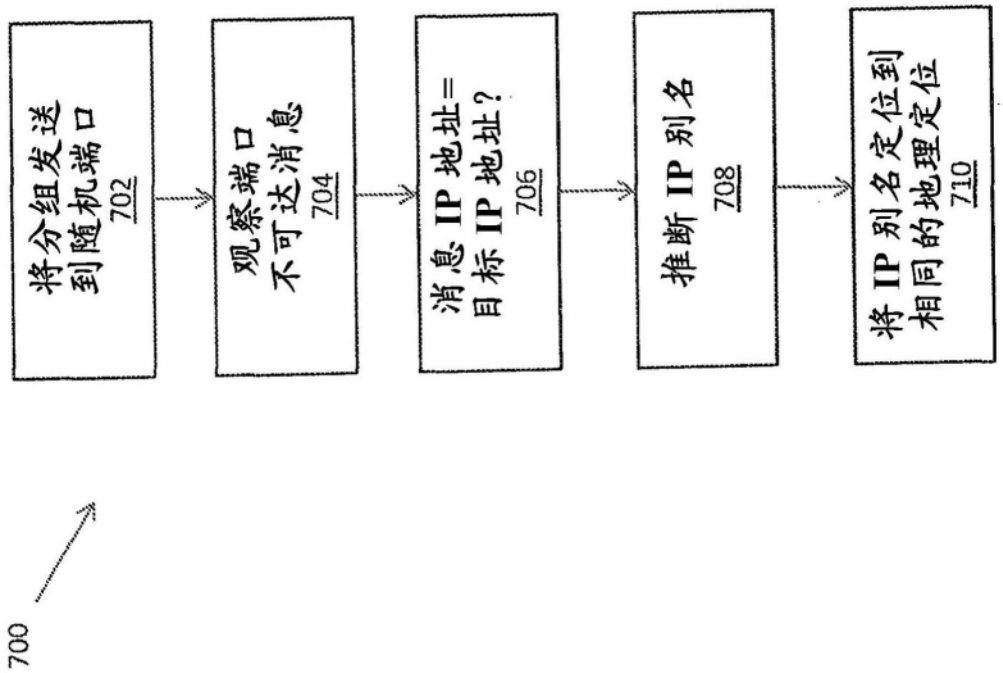

700
将分组发送到随机端口
702
观察端口不可达消息
704
消息 IP 地址=目标 IP 地址？
706
推断 IP 别名
708
将 IP 别名定位到相同的地理定位
710

图7A

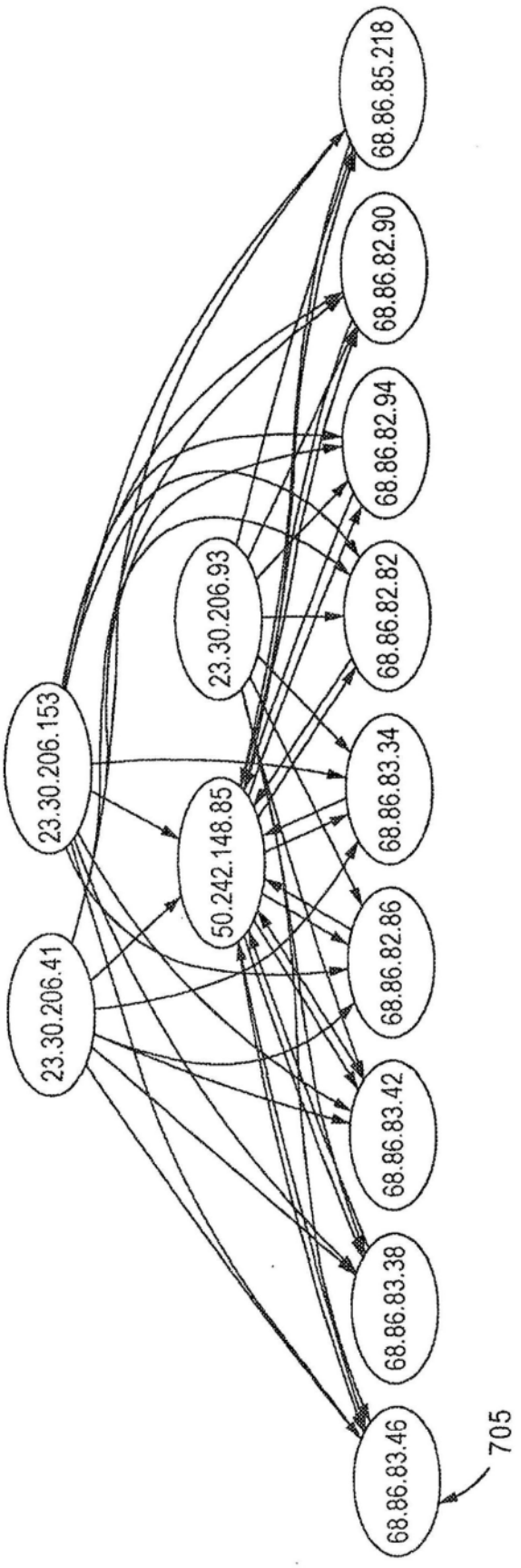

68.86.85.218
68.86.82.90
68.86.82.94
68.86.82.82
68.86.83.34
68.86.82.86
68.86.83.42
68.86.83.38
68.86.83.46
705
23.30.206.153
23.30.206.41
50.242.148.85
23.30.206.93

图7B

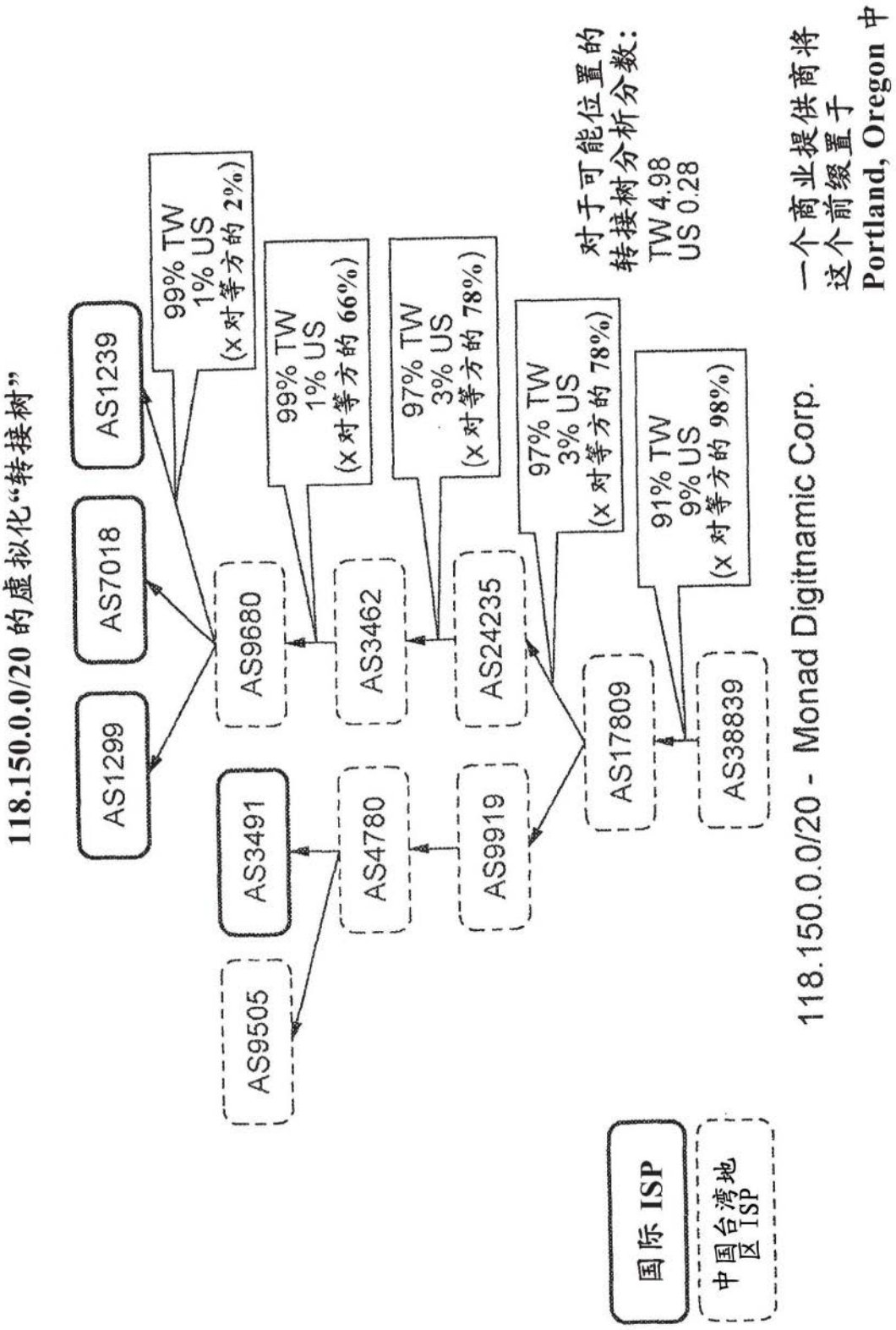

118.150.0.0/20 的虚拟化“转接树”
AS1299
AS7018
AS1239
AS3491
AS9505
AS9680
AS4780
AS3462
AS9919
AS24235
AS17809
AS38839
99% TW
1% US
(x 对等方的 2%)
99% TW
1% US
(x 对等方的 66%)
97% TW
3% US
(x 对等方的 78%)
97% TW
3% US
(x 对等方的 78%)
91% TW
9% US
(x 对等方的 98%)
对于可能位置的
转接树分析分数:
TW 4.98
US 0.28
118.150.0.0/20 - Monad Digitnamic Corp.
一个商业提供商将
这个前缀置于
Portland, Oregon 中
国际 ISP
中国台湾地区 ISP

图8A

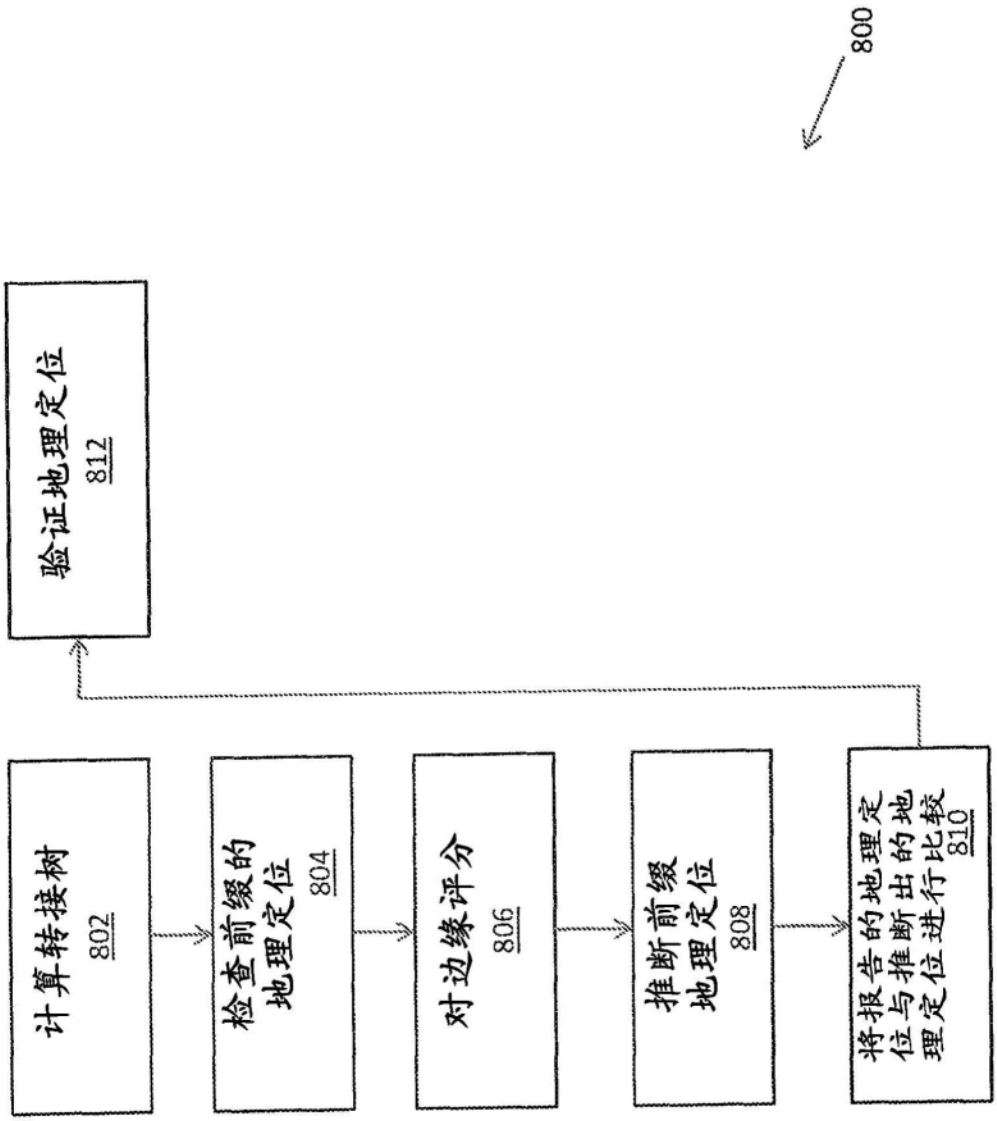

800
计算转接树
802
检查前缀的
地理定位
804
对边缘评分
806
推断前缀
地理定位
808
将报告的地理定
位与推断出的地
理定位进行比较
810
验证地理定位
812

图8B

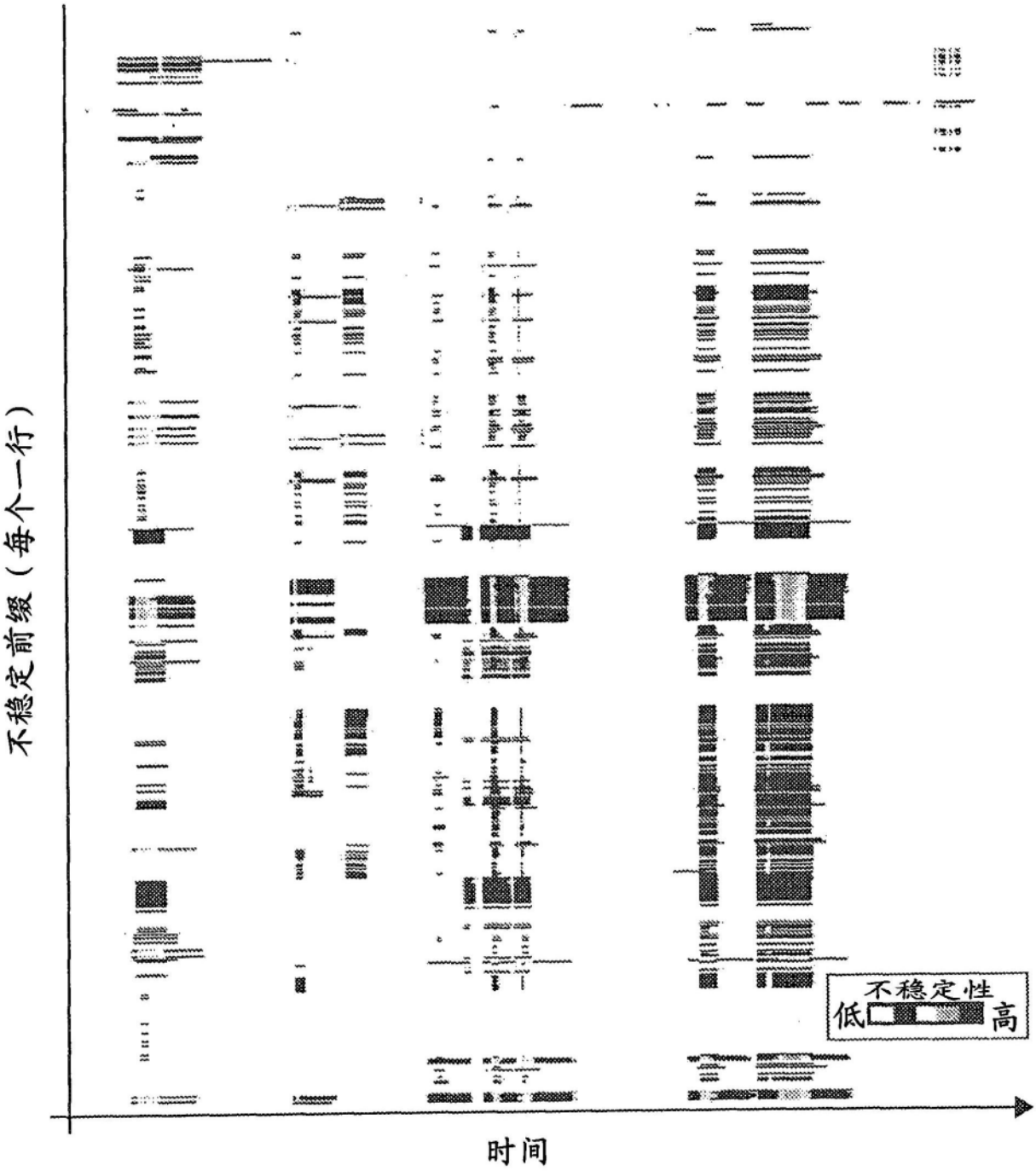


图9