IGRP 简介

目录

简介

本技术文档简要介绍了内部网关路由选择协议（IGRP）。 本文档有两个目的。第一个目的是向那些对使用、评估、实施IGRP技术的读者提供该技术概要。第二个目的是更广泛地探讨IGRP中所蕴含的一些有趣的想法。请参考“配置 IGRP”、“Cisco IGRP 实施”和“IGRP 命令”以了解有关如何配置 IGRP 的信息。

IGRP 的目标

IGRP是一种协议，可以让多个网关协调它们的路由。其目标是：

- 即使是在非常大而复杂的网络里也能生成稳定的路由。不会出现路由环路，哪怕是瞬时的路由环路也不会出现。
- 快速对网络拓扑的更改做出响应。
- 低开销。也就是说，IGRP本身不会使用比实际需要更多的带宽。
- 当几条路由的状况大概相同时，在这几条平行的路由之间平分流量。
- 考虑不同路径上的出错率和流量水平。

当前IGRP的实施方案可以处理TCP/IP的路由选择。然而，基本设计是希望能够处理不同类型的协议。

没有一种工具能够解决所有的路由选择问题。从传统来说，路由选择问题可以分为几类。IGRP等协议称为“内部网关协议”（IGP）。 它们可以用在一组单一的网络中，用一种管理系统或者一些能够密切配合的管理系统来管理。这样的网络组通过“外部网关协议”（EGP）连接。 IGP专门用来追踪有关网络拓扑的大量细节。IGP设计优先考虑的是生成最优路由以及对变化做出快速响应。EGP可以防止一个网络系统出错或者有意被其它系统误传。EGP设计的优先考虑是稳定性和管理性控制。对EGP来说，生成一个合理的路由就已足够，而不是最优路由。

IGRP同Xerox公司的路由信息协议、Berkeley的RIP以及Dave Mill的Hello协议等较早的协议有一些相似之处。它同这些协议的主要区别在于它适用于更大、更复杂的网络。第4节将进一步同RIP进行对比，因为RIP是较早的协议中使用最为广泛的一种类型。

同这些较早协议相类似，IGRP是一种距离向量协议。在这种协议内，网关只同毗邻的网关交换路由信息。这些路由信息包括有关网络其余部分的信息总结。用数学方式表示就是，将所有的网关聚集在一起来解一个最优化问题，为分布式算法分配最优流量。每个网关只需要解决部分问题，而且只会接收整体数据的一部分。

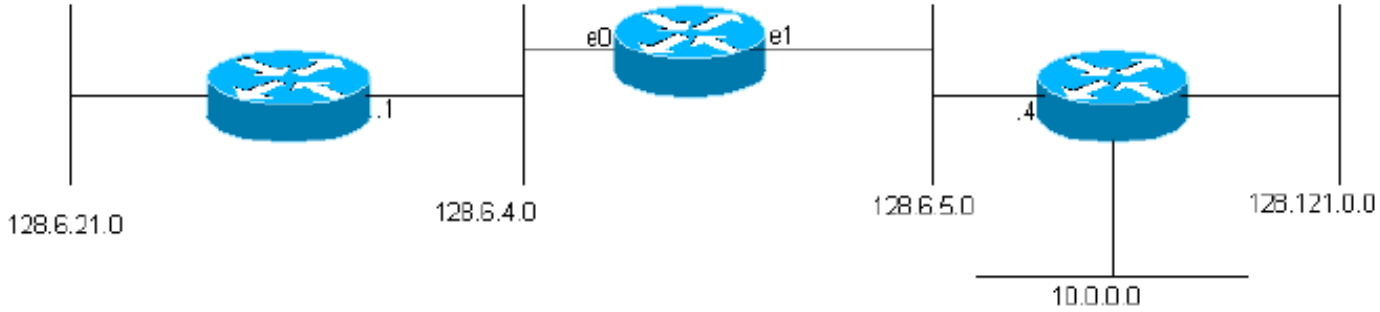
另一种主要协议是称为SPF（最短路径优先）的算法。OSPF 使用的就是这个概念。要了解有关 OSPF 的详细信息，请参阅 OSPF 设计指南。OSPF 这类协议基于泛洪技术，每个网关会随时了解有关其他各个网关上各接口最新状态的信息。每个网关从自己的角度出发，通过用于整个网络的数据独立地解决优化问题。每种方法都有其自己的优点。在一些情况下，SPF可以非常快地对变动做出响应。为了防止出现路由环路，IGRP必须在进行某种更改之后几分种内忽略新数据。因为SPF可以直接从每个网关获得信息，因而能够避免这种路由环路。这样，它就可以马上传输新的信息。但是，无论是在内部数据结构还是网关之间的消息方面，SPF 要处理的数据都远远多于 IGRP。

路由问题

IGRP一般用于连接不同网络的网关。我们假设网络使用基于数据包的技术。实际上，网关担当分组交换机的角色。当连接到一个网络的系统想要将数据包发送到另外网络中的系统时，它就将数据包发送到网关上。如果目的地址是同该网关相连接的网络中的一个，则网关就会将数据包转发到目的地。如果目的地更远一些，则网关将把数据包转发到同目的地较近的其它网关上。这些网关使用路由表来决定如何处理数据包。以下是路由表的一个简单例子。下面是一个简单的路由表的例子（在例子中所使用的地址都是Rutgers大学的地址。注意，基本的路由选择问题同其它协议也相似，但是这里的描述假定IGRP正被用于路由IP）。

图 1

```
network      gateway     interface
  -------      -------     ---------
  128.6.4      none        ethernet 0
  128.6.5      none        ethernet 1
  128.6.21     128.6.4.1   ethernet 0
  128.121      128.6.5.4   ethernet 1
  10           128.6.5.4   ethernet 1
```



（正如我们所看到的，实际的IGRP路由中有关于每个网关的附加信息）。 此网关连接到两个Ethernet，分别称为0和1。它们被指定了IP网络号（实际子网号）128.6.4和128.6.5。因此，为这些特定网络寻址的数据包可以直接发送到目的地，只需使用适当的以太网接口。有两个邻近的网关，128.6.4.1和128.6.5.4。用于128.6.4和128.6.5以外的网络的数据包将转发到这些网关中的一个或另一个。路由表中说明哪个网络应该使用哪个网关。例如，发往网络10上主机的数据包应转发到网关128.6.5.4。我们希望此网关更接近网络10，即到网络10的最佳路径通过此网关。IGRP的主要目的是使得网关可以建立并维护象这样的路由表。

IGRP 小结

正如前面所提到的，IGRP是一种协议，使得网关可以通过同其它网关交换信息而构建自己的路由表。网关最开始查找直接同它相连的所有网络的相关条目。通过与毗邻的网关交换路由选择更新，它可以获得有关其它网络的信息。在最简单的情况下，网关将找到一条能够到达每个网络的最佳路径。下一个网关将标记出数据包应该发送的一条路径、应该使用的网络接口以及度量信息等。度量信息是一组数据，说明当前路径情况是否良好。这就使得该网关可以比较来自不同网关的路径，并决定最终使用哪一条路径。常常有这样的情况：需要在两条或多条路径之间分离流量。IGRP在两条或多条路径一样好的情况下会这样做。用户还可以在路径几乎一样好的情况下配置使用分离路径。在这种情况下，该路径将能够传输更多的流量，而且其度量信息比单一路径要好。其意图是流量可以在一条9600 bps的线路和一条19200 bps的线路间分担，而且19200线路可以获得约为9600 bps线路两倍的流量。

IGRP所使用的度量值包括：

- 拓扑延迟时间
- 路径带宽最窄部分的带宽值
- 路径的信道占用情况
- 路径的可靠性

拓扑延迟时间是指沿着该路径到达目的地所需要花费的时间量，假设是在一个无负载的网络上。当然，当网络有负载的时候会产生额外的延迟。然而，载荷可以考虑在信道占用数中，而不是通过测量实际延迟来计算。路径带宽就是路径中以速率最慢的链路的每秒比特率计算的带宽值。信道占有情况说明当前有多少带宽在使用。它将用负载来计算，并且将随着负载而改变。可靠性说明的是当前的出错率，指无损地到达目的地的那部分数据包。它是完好无损到达目的地的数据包比率，这个数值是测量出来的。

尽管没有被用作度量值的一部分，但是度量信息中还含有两项信息：跳次及MTU。跳数和最大传输单位 (MTU)。跳次就是数据包到达目的地时所需要经过的网关数。MTU是可以不经过分拆就可以沿着整条路径发送的最大数据包尺寸（也就是说，它是路径所涉及到的所有网络中的最小MTU值）。（换言之，它是该路径中涉及的所有网络的最小 MTU。）

根据度量信息，可以为路径单独计算一个“合成度量值”。合成度量可以将不同度量成分的效果合成到代表该路径“好坏程度”的一个数值中。实际决定最佳路径时所采用的正是合成度量值。

每个网关定期将其完整的路由表（根据分割水平规则，需要做一定的审查和修改）广播到所有邻近网关。当某个网关从其它网关获得这种广播时，它就同已有的列表进行对比。所有新的目的地和路径都被添加到网关的路由表上。广播中的路径可以同现有的路径进行对比。如果新路径更好，它就会替换已有路径。广播中的信息还可以用来更新同现有路径相关的信道占有情况和其它信息。这个通用流程类似于所有距离向量协议所使用的流程。在数学文献中这被称为“Bellman-Ford”算法。有关基本程序的详细开发，请参阅RFC 1058 ，该程序描述了RIP（一种较旧的距离矢量协议）。

IGRP对标准的Bellman-Ford算法在三个重要方面做了修改。首先，不采用简单的度量值，而是使用度量向量来度量路径。其次，不是用最小的度量值来挑选某条路径，而是将流量在不同路径之间进行分配，这些路径的度量值都落在某个特定范围内。第三，在拓扑被改动的情况下，可以采用一些特性来提供稳定性。

根据合成的度量值来选择最优路径：

```
[(K1 / Be) + (K2 * Dc)] r
```

K1, K2 = 常数 Be = 未加载路径带宽×（1 - 信道占用率） Dc = 拓扑延迟 r = 可靠性

具有最小合成度量的路径将是最佳路径。有多条路径可以到达同一个目的地，网关可以在多条路径上路由数据包。这可以根据每条数据路径的合成度量来完成。例如，如果一条路径的合成度量值为1，而另外一条的合成度量值为3，则合成度量值为1的数据路径上将能够发送三倍以上的数据包。

采用度量信息向量有两方面的优势。首先，他能够支持基于同样数据组的多种类型服务。第二个方面的优势是能够提高精确度。在使用一个度量向量时，通常可以将其视为延迟来处理。路径中每个链路都加到总度量中。如某一个链路的带宽较低，这通常意味着大延迟。但是，带宽限制并不能真正按照延迟的累积方式累计。将带宽视为一个单独的要素才是正确的处理方式。类似地，负载可以用一个单独的信道占用数值来加以处理。

IGRP可以为相互连接的计算机网络提供一个系统，稳定地处理包括环路在内的通用图形拓扑。系统包括完全的路径度量信息，也就是说，它知道同每一个网关相连的所有其它网络的路径参数。流量可以在并行路径上进行分配，多个路径参数可以同时在整个网络上计算。

与 RIP 的比较

这一部分将把IGRP同RIP对比。这种对比是有用的，因为RIP广泛地被用于和IGRP类似的目的。然而，这样做并不完全公平，因为RIP同IGRP的目的并不完全相同。RIP 的本意并非是为了实现与 IGRP 相同的所有目标，RIP主要用于技术比较单一的小型网络。对于这样的应用，通常用RIP就足够了。

IGRP和RIP之间最根本的差别在于它们的度量结构不同。不幸的是，这种度量结构并不能只经过简单的变化就用于RIP。它需要IGRP中所用到的一些新算法和数据结构。

RIP采用简单的“跳次”度量来描述网络。与IGRP不同，RIP中每条路径都以延迟、带宽等来描述，而RIP中则以1到15的数字来描述。通常，此数字用于表示路径到达目的地之前经过的网关数量。这意味着在速率很慢的串行线路和以太网之间并无区别。在RIP的一些实际实施过程中，系统管理员可以具体指定某个给定的跳数应该被计算一次以上。较慢的网络可以用一个较大的跳数来表示。但是，由于最大值是 15，此数值的计算次数不能过多。例如，如果用 1 表示以太网，用 3 表示 56Kb 线路，则一条路径最多只能有 5 条 56Kb 线路，否则就会超过最大值 15。为了代表各种可用的网络速率，并能在大型网络中应用，Cisco的研究结果建议需要24位的度量值。如果最大度量值太小，系统管理员就会面临很不舒服的选择：他不是无法区别快速和慢速路由，就是无法将整个网络放入限制数值范围内。要么无法区分快速和慢速路由，要么就不能将整个网络都纳入限制范围之内。实际上，国家范围内的网络现在常常大到RIP根本无法处理的程度，即使每个跳数只计算一次也仍然如

此。RIP显然不能用于这样的网络。

最容易想到的做法就是修改RIP，使其允许更大的度量值。可惜的是，这并非有效。同所有的距离向量协议一样，RIP也存在“无穷大计数”的问题。 RFC 1058中对此进行了更详细的描述 。一旦拓扑结构改变，就会产生假路由。同这些假路由相关的度量值就会慢慢增加直至达到15，达到这个点时该路由就会被删除。15是一个非常小的最大值，假定使用触发更新，这个流程也会快速收敛。如果修改RIP以允许24位度量，则环路的持续时间将足够长，度量的计数最多可达2**24。这是不可容忍的。IGRP具有一些特性，可以专门防止产生假路由。下文第5.2节将讨论这些内容。如果不引入此类功能或更改为SPF等协议，则处理复杂网络是不现实的。

IGRP不是简单地增加容许度量的范围。它重新改造了度量的结构，用其来描述延迟、带宽、可靠性和载荷状态。可以将这些考虑反映在RIP等单个度量中。例如，采用单个度量标准，几条连续的快速链路将看起来如同一条较慢的链路。交互式流量可能就是这种情况，延迟是主要的考虑因素。然而，对于大批量数据传输，主要的考虑因素是带宽，在这种情况下将度量简单地加在一起并不是一种好方法。IGRP可以分别处理延迟和带宽，将延迟累计，但是仍然考虑最小的带宽值。我们很难看出如何将可靠性和负载的影响效果融入一个单一成分的度量标准。

我认为，IGRP的一个最大优势就是容易配置。它可以直接表示具有实际意义的数值。这意味着可以根据接口类型、线速等自动对其进行设置。使用单一要素的度量，更有可能必须“修改”度量来纳入多项不同因素的影响。

其它方面的创新更多地在于算法和数据结构方面，而不是路由选择协议。例如，IGRP规定了支持不同路由之间分离流量的算法和数据结构。当然也可以设计一种让RIP也这样做的实施方案。然而，既然在重新实施路由选择，就没有理由再坚持采用RIP。

迄今为止，我已经说明了“通用IGRP”这种能够支持任何网络协议路由选择的技术。然而，在这部分，特定TCP / IP的实施方案也值得再提一提。也就是将与RIP进行比较的实施方案。

RIP更新消息只包括路由表的情况。也就是说，只含有一些目的地和度量标准，而其它信息则很少。IGRP的IP实施方案有一些额外的结构。首先，更新消息由一个“自治系统号码”来识别。 这个术语来自Arpanet的传统，在这里有特定的含义。然而，对于大多数网络来说，这意味着您可以在同一网络上运行几种不同的路由系统。这对于多个组织的网络需要合并的情况尤为有用。每个组织都维护自身的路由。因为每个更新都加上了标签，因此可以将网关配置为只注意最适合的更新。将特定网关配置为接收来自多个自治系统的更新。它们以可控方式在系统之间传送信息。请注意，这不是路由安全问题的完整解决方案。任何网关都可以配置为侦听来自任何自治系统的更新。但是，在实施路由策略时（在这种情况下，网络管理员之间的信任程度尚可），这仍然不失为一种非常有用的工具。

有关 IGRP 更新消息的第二项结构上的功能会影响 IGRP 处理默认路由的方式。大多数路由协议都有默认路由的概念。路由更新要列出世界上每个网络往往是不切实际的。通常，一组网关需要详细列出组织内部网络的路由信息。发往组织外部目的地的所有流量均可发送到几个边界网关之一。这些边界网关可能有着更完整的信息。通往最佳边界网关的路由就是“默认路由”。 我们使用它来到达内部路由更新中未明确列出的任何目的地，从这个意义上来说，它是默认路由。RIP 和一些其他路由协议如同对待真实网络一样传播有关默认路由的信息。IGRP 却采用不同的方法。IGRP 允许将真实网络标记为默认路由的候选，而不是用一个虚假条目作为默认路由。它通过将有关这些网络的信息放到更新消息的特殊外部部分中来实现这一点。但也可将其视为打开一个与这些网络相关联的位。IGRP 定期扫描所有候选的默认路由，并选择度量最小的路由作为实际默认路由。

这种默认路由方法有可能比大多数 RIP 实施所采用的方法更加灵活一点。RIP 网关通常可以设置为生成具有某个指定度量的默认路由。其目的是让此操作在边界网关上完成。

详细说明

此部分提供对 IGRP 的详细说明。

整体说明

当某个网关第一次打开时，其路由表被初始化。这可能由一个操作人员从控制台终端完成，也可以通过读取配置文件中的信息来实现。此时系统可以提供同网关相连的各个网络的描述，包括链路之间的拓扑延迟（也就是说，某个数位穿越该链路所需要的时间）和链路的带宽。

图 2

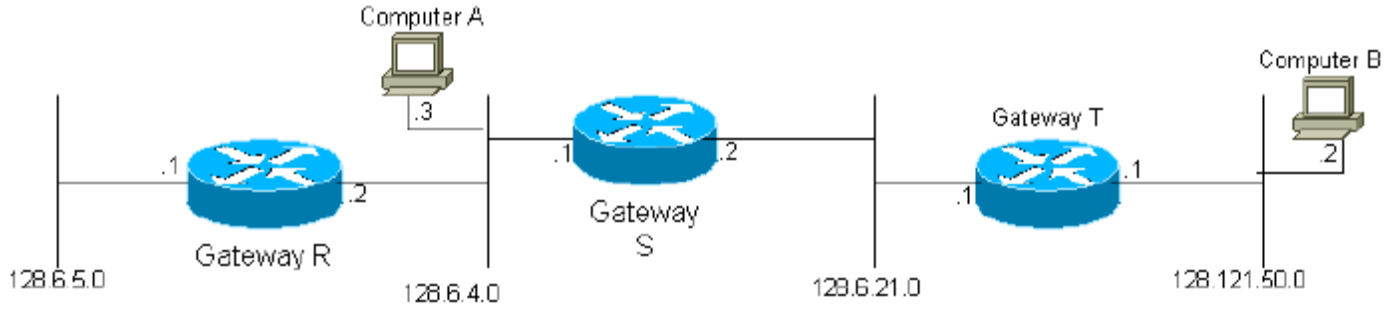


图1. 一个简单的网络范例 例如，在图1中，网关S可能被告知可以通过相应接口连接到网络2和网络3。因此，最初网关2只知道它可以到达网络2和3中的任何目的计算机。所有网关都被编程为定期向其相邻网关发送它们已初始化的信息以及从其他网关收集的信息。网关S通过网关R和网关T接收网关R和网关T的更新信息，获知网关S可以通过网关R到达网络1的计算机，网络4的计算机可以通过网关T到达。完全连接。

图 3

```
________ Network 1
       |
       gw A --nw2-- gw C
       |            / |
       |          /   |
       nw3    nw4     nw5
       |      /        |
       |    /          |
       gw B            gw D
      _|_______________|____ Network 6
```

图2. 替代路径范例 每个网关计算合成度量值以确定到达目的计算机的理想数据传输路径。例如，在上图中，对于网络6中的目标，网关A(gw A)将通过网关B和C计算两条路径的度量函数。请注意，路径仅由下一跳定义。从A到网络6实际上有3条可用的路由。

- 直接到 B
- 先到C接着再到B
- 先到C然后再到D

但是，网关A无需在涉及C的两条路由中进行选择。A中的路由表有一个条目代表通往C的路径。其度量代表从C到最终目的地的最佳方式。如果A发送一个数据包到C，则由C来决定是否使用B或者D。

等式 1

用来计算每条路径的度量函数如下：

```
[(K1 / Be) + (K2 * Dc)] r
```
在这里Be = 有效带宽：无负载带宽×（1 - 信道占用率）

等式 2

从原理上来说，总体延迟Dc应该可以用下公式来确定：

```
Dc = Ds + Dcir + Dt
```
在这里：

然而，在实践中，每种类型的网络技术都使用标准的延迟数据。例如，以太网就有标准的延迟数据，对于每种特定位速率的串行线路也有标准延迟数据。

下面例子说明在图2情况下网关A中的路由表。（注意，出于简单的目的，度量值向量的每个部分都没有显示）。

路由表示例：

网络	接口	下一跳网关	量度
1	NW 1	无	直连
2	NW 2	无	直连
3	NW 3	无	直连
4	NW 2	C	1270
	NW 3	B	1180
5	NW 2	C	1270
	NW 3	B	2130
6	NW 2	C	2040
	NW 3	B	1180

图3. 路由表范例 Bellman - Ford算法描述了通过相邻网关之间的信息交换来构建路由表的基本流程。该算法被用于RIP（RFC 1058）等较早协议中。 为了处理更复杂的网络，IGRP在基本的Bellman - Ford算法上添加了三种特性。

1. 不是简单地采用度量值，而是用度量向量来区分不同的路径。可以根据公式1计算出一个单一的合成度量值。使用矢量可让网关通过使用公式1中的几个不同系数来适应不同类型的服务。它还使网络特征的表示比单个度量更精确。
2. 不是用最小度量值来挑选出一条路径，而是将流量分流到几条路径中，这些路径的度量值都在某一特定范围内。这就可以同时使用多条路由，提供比单一路由更大的有效带宽。网络管理人员可以指定一个差异值V。所有具有最小合成度量值的路径都留下。此外，度量值小于V×M的所有路径也留下。流量在多条路径之间以合成度量值相反的比例进行分配。
3. 差异值概念可能会带来一些问题。很难提出一种策略可以在使用大于1的差异值的同时还能够不导致数据包环路传输。在Cisco 8.2版中并没有采用差异值特性。（我并不确认到底在哪个版本中删除了这种特性）。 其后果是只能将差异值设置为1。
4. 推出了一些特性，在拓扑被更换的情况下可以提供稳定性。这些特性可以防止出现路由环路和“无穷计数”的情况。主要的稳定特性有“抑制（holddowns）”、“触发式更新（triggered updates）”、“分离线（split horizon）”以及“中毒（poisoning）”等。 这将在下面进一步讨论。

流量分离（点2）可能会带来一些比较微小的危险性。差异值V可用来让网关同时使用不同速率的路

径。例如，出于冗余性考虑，可能有一条9600 BPS的线路同一条19200 BPS的线路平行运行（然而，如果好几条路径都相同，则负载将在它们之间分担）。如果差量 V 为 1，则只会使用最佳路径。因此，如果 19200 BPS 线路的可靠性尚可，则不会使用 9600 BPS 线路。（但是，如果多条路径相同，则会在其之间分摊负载。）通过增加差异值，我们允许流量在最佳路由和其它接近最佳水平的路由之间分配。采用一个足够大的差异值，流量可以在两条线路之间分配。危险性在于，如果采用足够大的差异值，则允许的流量不仅可能速度较慢，而且实际上可能“在错误方向上”传输。 这样，就需要有其它规则来防止流量向“上游”传输：不能在远程合成度量值（在下一个网段计算的合成度量值）大于该网关上所计算出的合成度量值的路径上传输流量。如果路径的远程复合度量（在下一跳计算的复合度量）大于在网关处计算的复合度量，则不沿该路径发送流量。一般情况下，不鼓励系统管理员将差异值设置为大于1的数，除非是在需要使用平行路径的情况下。在这种情况下，应该认真设置差异值以提供“正确”结果。

IGRP适用于处理多个“服务类型”和多种协议。服务类型在数据包中有说明，根据服务类型需要修改评估路径的方法。例如，TCP / IP协议允许数据包指定高带宽、低延迟或者高可靠性的相对重要性。通常交互式的应用将强调低延迟，而大批量数据传输应用则强调高带宽。这些需求决定了适合公式1使用的K1和K2的相对值。1.数据包中要支持的每种规范组合都称为“服务类型”。 对于每种类型的服务，必须选择一组参数K1和K2。每种类型的服务都对应一个路由表。这样做的原因是路径需要根据公式1所定义的合成度量值来进行选择和排序。1.每种服务类型都不同。来自所有这些路由表中的信息合并起来，就生成了由网关交换的路由更新消息，如图 7 所示。

稳定性功能

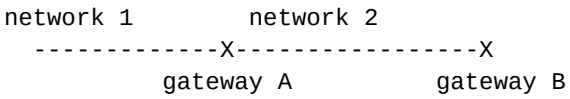
这一部分说明抑制、触发式更新、分离线和中毒等特性。这些特性的目的是防止网关不会挑选错误的路由。如RFC 1058中所述 ，当路由因网关或网络故障而变得不可用时，可能会发生这种情况。从原理上来说，毗邻的网关可以探测出故障。接下来它们将发送路由更新信息，说明老的路由已经不可用。然而，有可能更新根本没有到达网络的一些部分，或者在达到某些特定网关时被延迟。而某个网关仍然认为旧路由没有问题，并继续扩散这样的信息，从而重新将出现故障的路由纳入系统内。最后，这个信息将通过网络传播，并且返回重新将其纳入的网关。其结果是出现环路路由。

事实上，采取相应措施时考虑了冗余性。从理论上来说，抑制和触发式更新就足以首先防止错误路由发生。然而，在实践中，由于会产生不同类型的通信故障，这样仅这两种措施还远不够。分离线和路由中毒就可以在任何情况下防范路由环路。

通常，新的路由表被定期发送到邻近的网关（默认是每90秒种一次，不过，系统管理员可以调整这个周期）。触发更新是一种新路由表，此类路由表会在某些变化发生后立即发送出去。最重要的变动是删除某条路由。等候超时（可能是邻近网关或者线路出了故障）或者接收到路径中下一个网关发出的关于路径不能再使用的信息等情况都可能导致路由被取消。当网关G探测到某条路由不再可用，它就可以立即出发一个更新信息。这条更新信息将显示该路由不可用。下面我们考虑当这条更新信息到达邻近网关时将发生什么状况。如果相邻路由往回指向G，则邻居必须删除路由。这会导致邻居触发更新，依次类推。因此，一次故障会触发一波更新消息。这些消息将在网络的某一部分传播，且在这部分网络中，路由经过出现故障的网关或网络。

如果我们能够保证这些更新消息波可以立即到达每个相应的网关，则触发式更新就已经足够。但这里存在两个问题。首先，包括更新消息的数据包可能被网络中某些链路丢弃或者损坏。其次，触发式更新并不会同步发生。有可能某个网关尚未获得触发式更新就发送一条正常更新消息，从而导致失效的路由被重新插入已经获得触发式更新的相邻网关中。抑制专门用来解决这样的问题。抑制的规则是，当某条路由被删除，则在一定时期间内不会接收同一目的地址的任何新路由。这就让触发式更新有一段时间可以到达其它所有网关，从而可以确保我们所获得的所有新路由都不是某些网关中被插入的旧路由。抑制时间段必须足够长才能让触发的更新能够穿过网络。此外，它还应该包括一些定期广播周期来处理丢弃的数据包。下面我们考虑在触发式更新中有一条被丢弃或者损坏的情况下会导致什么状况。发送该更新信息的网关将在下一个定期更新时间发送另外一条更新信息。这将在丢失了最初更新波的相邻网关上触发新的更新消息波。

触发式更新和抑制等特性相结合，应该足以消除过期路由，并防止它们被重新插入。然而，还需要采取一些其它的预防措施。这些措施主要是针对那些性能损坏严重或者已经被分成不同区域的网络。IGRP所采用的其它预防措施称为分离线和路由中毒。分离线主要基于这样一个观察，即将一条路由沿着源方向发回原处是没有意义的。考虑下述情况：



网关A将告知B它有到网络1的路由。当B向A发送更新时，从来没有任何理由提及网络1。由于A离1更近，因此没有理由考虑通过B。水平分割规则规定应为每个邻居（实际上是每个相邻网络）生成单独的更新消息。 用于某个给定邻居的更新应该省略指向该邻居的路由。此规则可防止相邻网关之间形成环路。例如，假设 A 与网络 1 连接的接口发生故障。如果没有水平分割规则，B会告诉A它可以到达1。由于它不再有实际路由，A可能会接收该路由。在这种情况下，A和B都有到1的路由。但A将指向B，B将指向A。当然，触发更新和抑制会阻止这种情况发生。但是由于没有理由将信息发还给产生这条信息的地方，因而分离线仍然有价值。除了能够防止出现环路，分离线还能够减少更新消息的大小。

分离线应该能够防止相邻网关之间的环路。路由中毒则可能打破较大的环路。规则是，当某条更新消息显示某个已有路由的度量值已经增加妻足够多的时候，就有环路产生。该路由应该被删除并进入抑制状态。目前的规则是，如果复合度量增加超过1.1倍，则会删除路由。仅仅增加复合度量来触发删除路由并不安全，因为由于信道占用率或可靠性的变化，可能会发生较小的度量更改。因此1.1的系数仅仅是试探性的。实际值并不重要。我们希望这条规则只在打破很大环路的时候才会用到，因为较小的环路通过触发式更新和抑制就已经能够预防。

禁用抑制功能

在8.2版之后，Cisco的代码就能提供是否禁用抑制功能的选项。抑制的缺点在于当一条老路由出现故障时将会延迟采用新路由的时间。在默认参数下，在某个变动产生之后路由器需要花费数分钟才能够获得新路由。然而，出于上面所解释的原因，简单地消除抑制功能并不安全。其结果是计数到无穷大，如 RFC 1058 中所述。我们推测（但不能证明），如果使用加强版的路由毒化，则不再需要通过抑制来阻止计数到无穷大。这样禁用抑制就应该启用这种功能更强的路由中毒。注意，分离线和触发式更新将仍然有效。

路由中毒的更强大模式则是基于跳次。如果某条路径的跳次增加，就将该路由删除。这显然有可能删除仍然有效的路由。如果网络的某个部分发生了某种改变，以至于当前路径需要增加一个网关的路径，则跳次将增加。在这种情况下，路由仍然有效。然而，并没有一种完全安全的方法可以将这种情况同路由环路（无穷计数）区分开。 这样，最安全的方法是只要跳次增加就将该路由删除。如果该路由仍然合法，则在下一次更新的时候重新安装，从而将导致一次触发式更新，在系统的其它部分重新安装该路由。

总之，距离向量算法可以轻松地获取新路由。问题是它将彻底将旧路由从系统中清除。这样，在删除可疑路由方面更激进一些的规则应该比较安全。

更新过程详细信息

图4 - 8所描述的流程主要针对某种网络协议的处理，如TCP/IP，DECnet或者ISO/OSI协议等。不过，只给出针对TCP/IP的协议细节。某个网关可以处理服从多个协议的数据。因为每个协议都有不同的寻址结构和数据包格式，对于不同的协议来说，用来实现图4 - 8过程的计算机代码一般也不相同。如图4的详细说明所述，图4中描述的流程变化最大。图5至图8中描述的流程将具有相同的一般结构。协议之间的差别在于路由更新数据包的格式，这个数据包必须能够符合特定协议的规范。

注意，目的地在不同协议中可能有不同的定义。这里所描述的方法可以用于到某个主机、网络或者更复杂的分层地址机制的路由。采用哪一类型的路由将取决于协议的地址编排结构。当前的TCP/IP实施仅支持到IP网络的路由。这样“目的地”确实是指IP网络或者子网号。子网的信息只保存给相连的网络。

图4 - 7显示了网关所使用路由流程不同部分的伪代码。在程序开始的时候，需要输入说明每个端口可用的协议和参数。

网关将仅处理列表中的特定协议。任何来自采用列表之外协议的系统的通信都将被忽略。数据输入信息包括：

- 同网关相连的网络
- 每个网络的无负载带宽
- 每个网络的拓扑延迟
- 每个网络的可靠性
- 每个网络的信道占用率
- 每个网络的MTU

然后，根据公式1计算每个数据路径的度量函数。请注意，前三项是相当永久的。它是一个基本的网络技术函数，同载荷无关。它们可以通过配置文件设置，也可以通过操作人员直接输入来配置。注意，IGRP并不使用测量到的延迟值。理论和实践都表明，协议要采用测量的延迟值来维持稳定路由非常困难。有两个可度量的参数：可靠性和信道占用率。可靠性和信道占用率。可靠性是基于网络接口硬件或者固件所报告的出错率来确定的。

除了这些输入，路由算法还需要一些路由参数的值。这包括计时器值、差异值以及抑制功能是否启用等。这通常由一个配置文件或者操作 人员的输入来规定。（在Cisco 8.2版中，差异值被固定设置为1）。

输入初始信息后，网关中的操作将由事件（数据包到达某一个网络接口或计时器到期）触发。图4 - 7中 所描述的流程可以用如下方式触发：

- 当数据包到达时，将根据图4进行处理。这会导致数据包从另一个接口发送、丢弃或接受以供进一步处理。
- 当网关接收了数据包以求进一步处理时，它可以用一种取决于所用协议的方法进行分析，这在本规范中没有进行说明。如果数据包是一个路由更新，则可以根据图5进行处理。
- 图6说明了计时器所触发的事件。计时器可以设置为每秒钟产生一个中断。当中断发生时，就执行图6中所显示的流程。
- 图7显示了一个路由更新的子路由。图5和图6显示了到这条子路由的呼叫。
- 此外，图8显示了度量值计算的详细过程，请参见图5和图7。

有4个关键时间常数控制路由的传播和到期。这些时间常数可以由系统管理员进行设置。不过，它们都有默认的数值。这些时间常数是：

- 广播时间 - 所有网关按照此频率在所有连接的接口上广播更新。默认值是每90秒钟广播一次。
- 无效时间 - 如果没有在此时间内收到给定路径的更新，则认为该路径已超时。它应该是广播时间的几倍，目的是考虑到包含有更新消息的数据包有可能被网络丢弃。默认值是广播时间的3倍。
- 保持时间 - 当某个目的地已无法访问（或度量的增加足以导致毒化）时，该目的地会进入“抑制”状态。 在这种状态下，在保持时间所规定的时间段内，不再接收传往同一目的地的新路径。保持时间说明这种状态将持续多久。这可能是广播时间的几倍。默认值是广播时间的3倍再加上10秒（如5.2.1中所描述，可能会禁用抑制功能）。（如禁用抑制部分所述，可以禁用抑制功能。）

- 刷新时间 - 如果在此时间内未收到给定目的地的更新，则从路由表中删除该路径的条目。注意有效时间和清除时间之间的区别：在有效时间过后，某条路径超时并且会被删除。超过无效时间后，路径将超时并被删除。如果没有保留到某个目的地的路径，则目的地现在就无法到达。然而该目的地的数据库字段仍然保留着。这个数据段必须保留以强制实现抑制。在清除时间之后，该数据库条目将从列表中删除。这个时间可以比有效时间与抑制时间之和长一些。默认值是广播时间的7倍。

这些数字预示着下面这些主要的数据结构。网关所支持的每种协议都单独保留一套数据结构。在每种协议中，为其所支持的每种类型服务也分别保留一套数据结构。

对于系统所知道的每个目的地，都有（可能为空）一个能到达该目的地的路径清单，一个抑制到期时间以及一个最后更新时刻。最后更新时刻说明到这个目的地的每条路径都已经包括在来自其它网关的更新信息中。注意，每条路径也都保留了更新时间。当到达某个目的地的最后路径被删除之后，目的地就进入抑制状态，除非抑制被禁用（见5.2.1节）。 抑制到期时间说明抑制到期的时间。这个时间不是零将意味着目的地正处在抑制状态下。为了节约计算时间，为每个目的地保留一个“最佳度量值”也是一个很好的思路。这个值就可以简单地取为到目的地的所有路径中最小的那个合成度量值。

到达目的地的每条路径中都有路径中下一个跳数的地址、将使用的接口、描述路径特征的一个度量向量（包括拓扑延迟、带宽、可靠性以及信道占用率等）。其他信息也与每条路径关联，包括跳数、MTU、信息源、远程复合度量以及根据公式1根据这些数字计算的复合度量。还有上次更新时间。信息来源是指该路径的最新更新消息的来源。在实践中，这同下一个网段的地址相同。最后更新时刻就是这条路径的最近更新到达的时间。它可以用于确定路径是否超时。

注意，一条IGRP更新消息包括三个部分：内部、系统（意思是“这个自治系统”而非内部）以及外部。内部、系统（表示“此自治系统”而非内部）和外部。内部部分用于到子网的路由。并不包括所有的子网信息，而只包括一个网络的子网。而是只包括一个网络的子网。这个子网就是同更新正在发送的目的地址相关的网络。通常，这些更新在每个接口上广播，因而 这就是正在发送广播的那个网络（其它情况下用于对一条IGRP请求和点到点IGRP的响应）。（对 IGRP 请求的响应和点对点 IGRP 会出现其他情况。）主网络（不是子网）被置于更新消息的系统部分 ，除非它们被特别标记为外部。

如果某个网络是从其它网关获取，并且该信息到达的时候位于更新消息的外部部分，则该网络被标记为外部网络。Cisco的实施还允许系统管理员将某些网络标记为外部网络。外部路由也被称为“候选默认”。 它们是流往或者流经那些被考虑为合适的默认网关的路由，当没有清晰的路由到某个目的地时就可以采用这些路由。例如，在Rutgers，我们配置网关连接Rutgers和我们的地区网络，因而它连接到NSFnet骨干网的路由就标记为外部。Cisco的方案将具有最小度量值的外部路由选为默认路由。

下面各节将清晰地说明图4 - 8的一些特定部分。

数据包路由

图4说明了输入数据包的总体处理过程。这可以简单地用来说明术语。很显然，这并不是对IP网关功能的完整的描述。

这个流程使用支持协议清单和网关初始化时所进入的接口的相关信息。详细的数据包处理过程取决于数据包所使用的协议。这可以在步骤A中决 定。一旦已知协议类型，则就会根据该协议采取图4中的相应措施。数据包的详细内容将在协议规 范中描述。协议规范包括确定数据包目的地的流程、将目的地与网关自身地址进行比较以确认网关本身是否就是目的地的流程、确定数据包是 否为广播的流程，以及确定目的地是否为某个特定网络一部分的流程等。图4的步骤B和步骤C中使用了这些步骤。步骤D中的测试需要搜索路由表中列出的目标。如果路由表中有关于该目的地的项目，并且该目的地至少同一条可用路由相关，则该测试就算完成。注意，在这里和 下一步骤中所使用的目的地

和路径数据都根据支持的服务类型分别进行保存。这样，这个步骤一开始就是确定数据包所规定的服务类型，并且 选择用于这一步和下一步的相应数据结构组。

如果某条路径的远程合成度量值低于其合成度量，这条路径就可以用于步骤D和步骤E。如果某条路径的合成度量值大于其合成度量值，如果用度量值来衡量的话，该路径的下一个跳数就会离目的地“更远”。在这里将其称为“上游路径”。 通常情况下，人们希望使用度量值可以防止上游路径被选中。很容易看到，上游路径绝不可能是最好的路径。然而，如果允许较大的差异值，则非最佳路径也可能被使用。其中一些这样的路径就有可能是上游路径。

步骤E计算将要使用的路径。那些远程合成度量值不超过其合成度量值的路径将不在考虑之内。如果有一条以上的路径可接受，则这些路径将按照循环交替加权方式使用。路径的使用频率与其复合度量成反比。

接收路由更新

图5说明从相邻网关接收路由更新的过程。这些更新包括一个列表，其中每一个条目都给出某个目的地的信息。在单个路由更新上同某个目的地相关的条目可能有多条，以针对不同的服务类型。如图5所述，每个条目都单独处理。如果条目在更新的外部部分，则如果作为此过程的结果添加了目标，则将为其设置外部标志。

对于网关所支持的每种服务类型，图5所描述的整个流程都会重复一次，方法是采用同那种类型相关的目的地/路径信息组。图5中最外层的环路中显示了这一点。每种服务类型必须处理一次整个路由更新。对于每种类型的服务都必须将整个路由更新完全处理一次（注意，当前的IGRP版本中并不支持多种服务类型。

在步骤A，在路径上完成了基本可接收性测试。这应该包括目的地的合理性检测。不可能的（“Martian”）网络号应该被拒绝。(有关详细信息，请参阅RFC 1009 和RFC 1122 。) 如果所指向的目的地正处于抑制状态，也就是说抑制到期时间非零而且晚于当前时间，则这样的更新也会被拒绝。

在步骤B中将搜索路由表，以查找是否有描述某条已知路径的条目。路由表中的路径由其关联的目的地、作为该路径一部分列出的下一跳、要用于该路径的输出接口和信息源（发送更新的地址 - 实际上通常与下一跳相同）来定义。 来自更新数据包中的条目将对某条路径进行描述，该路径的目的地列在该条目中、输出接口正是更新进入的接口，且下一个跳数和信息源正是发送更新的网关的地址（“源”S）。

图7中所描述更新流程中的步骤H和T按照时间表计划运行。这个流程将在图5中所描述的整个流程运行结束之后再运行。也就是说，图7中描述的更新过程只会发生一次，即使在图5中描述的处理过程中多次触发也是如此。此外，如果网络变化迅速，必须采取预防措施，防止更新过于频繁地发出。

如果更新数据包中所描述的当前条目所描述的目的地已经存在于路由表，则进入步骤K。这个步骤将根据更新数据包数据计算出来的新合成度量值同该目的地的最佳合成度量值进行对比。注意，最佳合成度量值在这个步骤并不重新计算。

步骤L用于那些比已有最佳合成度量值差的路径。这既包括那些比已有路径差的新路径，也包括那些合成度量值增加的已有路径。步骤 L 测试新路径是否可接受。注意，这项测试不仅查看新路径是否足够好而值得保留，还查看该路由是否中毒。延迟值必须不能是指示着一个无法到达的目的地的特定值（对于当前的IP部署来说，所有的延迟值都写入一个24位的字段），而且合成度量值（按照图8中指定的方法计算）也必须可接受，该路由才能够被接受。要确定合成度量值是否可接受，可以将其与通往目的地的所有其它路径的合成度量值进行对比。假定M是这些度量值中的最小值。则如果它小于V×M就可以接受，其中，V是网关初始化时所设置的差异值。如果V = 1（在Cisco 8.2版中，总是这个值），则比已 有度量值差的所有度量值都不会被接受。在这里有一个唯一的例外情况

：如果路径已经存在，而且是到目的地的唯一路径，则如果度量值增加不超过10%（或者在禁用抑制的情况下，如果跳次没有增加）该路径将保留。如果该路径已经存在且是通往目的地的唯一路径，则当度量的增加未超过 10%（或在禁用抑制的情况下跳数未增加）时将保留该路径。

当针对某条路径的新信息表明合成度量值降低的情况下将会执行步骤V。到目的地D的所有合成度量值都被进行了比较。在这个比较过程中，使用了针对P的新合成度量值，而不是出现在路由表中的那个度量值。然后计算最低合成度量值M。接下来对通往D的所有路径再次进行检验。如果任何一条路径的合成度量值>M×V，则该路径就被删除。V是差异值，在网关初始化的时候输入。（在Cisco 8.2版中，差异值被固定设置为1）。

定期处理

图6中所描述的流程每秒钟触发一次。它可以校验路由表中的不同计时器是否已经超时。这些计时器已经在前面描述过。

在步骤U，图7中所描述的流程被激活。

步骤R和S都是必需的，因为存储在路由表中的合成度量值取决于测量出的信道占用率，这个度量随着时间会不停地变动。采用通过接口的流量测量值的移动平均值，定期计算信道占用率。如果新计算的值不同于已有的值，则涉及该接口的所有合成度量值都必需进行调整。路由表中所 显示的每条路径都需要校验。下一个跳数使用接口“I”的每条路径都会重新计算其合成度量值。在作为路径度量值的一部分而存储在路由表中 的原有信道占用率以及该接口最新计算出的信道占用率中，选取较大的一个作为信道占用率，根据公式1计算来计算新的合成度量值。

生成更新消息

图7说明了网关如何生成发送到其它网关的更新消息。网关为同该网关相连的每个网络接口都分别生成一条消息。该消息接下来就发送到可以通过接口（步骤J）到达的所有其它网关上。 通常，将该消息作为广播业务进行发送就可以完成发送。不过，如果网络技术或者协议不允许广播， 则可能需要将消息单独发送到每个网关上。

通常，消息是通过在步骤G中为路由表中的每个目标添加一个条目来构建的。请注意，必须使用与每种服务类型关联的目标/路径数据。在最坏的情况下，针对每种服务类型、每个目的地的更新都需要添加一个新的条目。然而，在步骤G中将新条目添加到更新消息中时，需要扫 描已经添加的条目。如果新条目在更新消息中已经存在，则就不再添加。当目的地和下一跳数的网关相同的时候，新条目就会复制已有的条目。

为简单起见，伪代码将省略一件事 - IGRP 更新消息分为三个部分：这样，在目的地上实际有三个环。第一个只包括该网络中正在发送更新的子网。第二个包括没有被标记为外部的所有主网络（也就是，非子网的网络）。第三个包括被标记为外部的所有主网络 。

步骤E实施分离线检测。在一般情况下，这种检测对于一些路由无效，这些路由所通过的接口正是目前发送更新消息的接口。但是，如果更新是发送到特定目的地（例如，响应来自其他网关的 IGRP 请求，或作为“点对点 IGRP”的一部分），则只有当最佳路径最初来自该目的地（其“信息源”与目的地相同）且其输出接口与传入请求的接口相同时，水平分割才会失败。

计算度量值信息

图8说明了如何从网关接收到的更新消息处理度量值信息，以及被网关发送的更新消息如何生成。注意，条目是基于到目的地的某条特殊路径的 。如果到该目的地的路径超过一条以上，则就会选择合成度量值最小的路径。如果有一条以上的路径有最小合成度量值，则可以使用任意解绑

（tie - breaking）规则。（对于大多数协议来说，这将根据下一个跳数网关的地址来确定）。

图 4 - 处理传入数据包

```
Data packet arrives using interface I

A   Determine protocol used by packet

    If protocol is not supported
      then discard packet

B   If destination address matches any of gateway's addresses
    or the broadcast address
      then process packet in protocol-specific way

C   If destination is on a directly-connected network
      then send packet direct to the destination, using
      the encapsulation appropriate to the protocol and link type

D   If there are no paths to the destination in the routing
    table, or all paths are upstream
      then send protocol-specific error message and discard the packet

E   Choose the next path to use. If there are more than
    one, alternate round-robin with frequency proportional
    to inverse of composite metric.

    Get next hop from path chosen in previous step.

    Send packet to next hop, using encapsulation appropriate
    to protocol and data link type.
```

图 5 - 处理传入路由更新

```
Routing update arrives from source S

    For each type of service supported by gateway
      Use routing data associated with this type of service

      For each destination D shown in update

A       If D is unacceptable or in holddown
          then ignore this entry and continue loop with next destination D

B       Compute metrics for path P to D via S (see Fig 8)

        If destination D is not already in the routing table
        then Begin

          Add path P to the routing table, setting last
          update times for P and D to current time.

H         Trigger an update

          Set composite metric for D and P to new composite
          metric computed in step B.

          End

        Else begin (dest. D is already in routing table)
```

```
K         Compare the new composite metric for P with best
          existing metric for D.

           New > old:

L            If D is shown as unreachable in the update,
             or holddowns are enabled and
              the new composite metric >
                (the existing metric for D) * V
                [use 1.1 instead of V if V = 1,
                as it is as of Cisco release 8.2]
O            or holddowns are disabled and
             P has a new hop count > old hop count
             then Begin

              Remove P from routing table if present

              If P was the last route to D
                then Unless holddowns are disabled
                    Set holddown time for D to
                     current time + holddown time
T                  and Trigger an update

              End

             else Begin

              Compute new best composite metric for D

              Put the new metric information into the
              entry for P in the routing table

              Add path P to the routing table if it
              was not present.

              Set last update times for P and D to
              current time.

              End

           New <= OLD:

V            Set composite metric for D and P to new
             composite metric computed in step B.

             If any other paths to D are now outside the
             variance, remove them.

             Put the new metric information into the
             entry for P in the routing table

             Set last update times for P and D to
             current time.

          End

        End of for

      End of for
```

图 6 - 定期处理

```
Process is activated by regular clock, e.g. once per second
```

```
    For each path P in the routing table (except directly
    connected interfaces)

      If current time < P'S LAST UPDATE TIME + INVALID TIME
        THEN CONTINUE WITH THE NEXT PATH P

      Remove P from routing table

      If P was the last route to D
        then Set metric for D to inaccessible
          Unless holddowns are disabled,
            Start holddown timer for D and
          Trigger an update

        else Recompute the best metric for D

      End of for

    For each destination D in the routing table

      If D's metric is inaccessible
        then Begin

        Clear all paths to D

        If current time >= D's last update time + flush time
          then Remove entry for D

        End

      End of for

    For each network interface I attached to the gateway

R     Recompute channel occupancy and error rate

S     If channel occupancy or error rate has changed,
        then recompute metrics

      End of for

    At intervals of broadcast time

U     Trigger update
```

图 7 - 生成更新

```
Process is caused by "trigger update"

    For each network interface I attached to the gateway

      Create empty update message

      For each type of service S supported

        Use path/destination data for S

        For each destination D

E         If any paths to D have a next hop reached through I
            then continue with the next destination
```

```
          If any paths to D with minimal composite metric are
          already in the update message
            then continue with the next destination

G         Create an entry for D in the update message, using
          metric information from a path with minimal
          composite metric (see Fig. 8)

          End of for

        End of for

J     If there are any entries in the update message
       then send it out interface I

      End of for
```

图 8 - 度量计算的详细信息

这一节描述了通过一个到达的路由更新消息来计算出度量值和跳次的流程。这个函数的输入值是路由更新数据包中的某个特定目的地。输出值是一个度量值向量，可以用来计算合成度量以及跳次。如果这条路径添加到路由表上，则整个度量值向量就被输入到列表中。下述定义中所使用的接口参数都是在网关初始化设置时为那些路由更新到达的接口所设置的参数，不过信道占用率和可靠性都是基于该接口上所测流量的移动平均值。

- 延迟 = 数据包延迟 + 接口拓扑延迟
- 带宽 = 最大值（数据包带宽，接口带宽）
- 可靠性 = 最小值（数据包可靠性，接口可靠性）
- 信道占用率 = 最大值（数据包信道占用率，接口信道占用率）（最大值被用于带宽，因为带宽度量值以反向形式存储。从概念上来说，我们需要的是最小带宽。 ）注意，必须保存数据包的初始信道占用率 ，因为在接口信道占用率发生变化的时候，需要重新计算有效的信道占用率。

下面并非度量值向量的一部分，但是也出现在路由表中，作为该路径的特征值：

- 跳次 = 数据包中的跳次
- MTU = 最小值（数据包中的MTU，接口MTU）
- 远程合成度量––采用数据包中的度量值根据公式1计算。也就是说，度量值的成分是数据包中所包括的那些，而非上面所显示的更新后的值。很显然，这必须在上面所显示的调整完成之前就计算完。
- 合成度量值––采用这一节所计算出的度量值根据公式1计算而出。

这一部分说明计算将要发送的路由更新的度量值和跳次的流程：

这个函数用来确定需要置入外发更新数据包的度量值信息和跳次。它基于到目的地的某条特定路径，如果有可用路径的话。如果没有路径或者路径都是上游路径，则目的地就称为“不可访问”。

```
If destination is inaccessible, this is indicated by using a specific
        value in the delay field. This value is chosen to be larger
        than the largest valid delay. For the IP implementation this is
        all ones in a 24-bit field.

   If destination is directly reachable through one of the interfaces, use
        the delay, bandwidth, reliability, and channel occupancy of the
        interface. Set hop count to 0.

   Otherwise, use the vector of metrics associated with the path in the
        routing table. Add one to the hop count from the path in the
        routing table.
```

IP实施方案细节

这一节将简要说明Cisco IGRP所使用的数据包格式。IGRP可以采用代IP协议9（IGP）的IP数据表来发送。 数据包最前端是一个报头。它紧跟着IP报头。

```
unsigned version: 4;  /* protocol version number */
    unsigned opcode: 4;   /* opcode */
    uchar edition;        /* edition number */
    ushort asystem;       /* autonomous system number */
    ushort ninterior;     /* number of subnets in local net */
    ushort nsystem;       /* number of networks in AS */
    ushort nexterior;     /* number of networks outside AS */
    ushort checksum;      /* checksum of IGRP header and data */
```

对于更新消息来说，路由信息紧跟着报头。

当前的版本号为1。

1. 更新

这能够表示信息类型。两种信息类型的格式将在下面给出。

版本是一个串行号码，当路由表出现变动时就会增加（在上面的伪代码要求触发一次路由更新的情况下会完成这种操作）。（这种变化发生在上述伪代码触发路由更新的情况下。）版本号使得网 关可以避免处理包含已有信息的更新（当前还没有采用这种方式。（这一点当前未实现。也就是说，版本号仍然正确生成，但是在输入中被忽略。因为对于那些要被 丢弃的数据包无法采用版本号。有必要搞清楚与该版本相关的所有数据包都已经得到处理）。

*A系统就是自治系统号码。*在思科的方案中，网关可以加入一个以上的自治系统。每个这样的系统都运行其自己的IGRP协议。从概念上来说，对于每个自治系统都有完全独立的路由表。通过IGRP来自于一个自治系统的路由只能在针对那个AS的更新发送。这个字段使得网关在处理这些消息时，可以选择采用哪一组路由表。如果网关接收到一条针对没有配置的AS的IGRP消息，就会将其忽略。事实上，思科的方案允许信息从一个AS“泄露”到另外一个。不过，我将它视为一种管理性工具，而不是协议的一部分。

*Ninterrior、nsystem以及nexterior都表示更新消息中三个部分中每个条目的号码。*这些部分在上面已经描述过。在这些部分之间没有其它划分。第一个ninterior条目被当做内部，下面的nsystem条目则视为系统，而最后的nexterior Checksum则为IP校验和，采用与UDP校验和相同的算法来计算。

校验和是 IP 校验和，使用与 UDP 校验和相同的校验和算法进行计算。校验和在IGRP报头以及其后所有路由信息上计算。在计算校验和的过程中，校验和字段设置为0。校验和并不包括IP报头，那里也不像UDP和TCP中那样有虚报头。

请求

IGRP请求要求接收方发送自己的路由表。请求消息只有一个报头。只用到了版本、操作码以及系统字段。所有其它字段都是零。接受者需要将一个标准IGRP更新消息发送给请求者。

更新

IGRP更新消息包括报头，其后紧跟着路由条目。包括的许多路由更新条目可以装入一个1500字节的数据报（包括IP报头）中。 采用当前的结构声明，最多允许104个条目。如果需要更多的条目

，则可以发送多条更新消息。由于更新消息可以简单地逐个条目地进行处理，因而用一个分段的消息取代几个独立的消息并没有优势。

下面是路由条目的结构：

```
uchar number[3];       /* 3 significant octets of IP address */
    uchar delay[3];        /* delay, in tens of microseconds */
    uchar bandwidth[3];    /* bandwidth, in units of 1 Kbit/sec */
    uchar mtu[2];          /* MTU, in octets */
    uchar reliability;     /* percent packets successfully tx/rx */
    uchar load;            /* percent of channel occupied */
    uchar hopcount;        /* hop count */
```

uchar [2]和 uchar [3]字段是16位和24位二进制整数，按照标准的IP网络顺序。

号码定义描述的目的地。它是一个 IP 地址。为了节约空间，除非在内部部分，否则只给出IP地址的前3个字节。在内部部分，给出的是最后的3个字节。对于系统和外部路由，由于不可能有子网，因而低位的字节总设置为0。内部路由总是一个已知网络的子网，因而只提供该网络号码的第一个字节。

延迟的单位为 10 微秒。其范围为10微秒到168秒，看上去应该足够。全1的延迟说明该网络无法到达。

带宽是以比特/秒为单位的反向带宽，以1.0e10为倍数。范围从1200 BPS线路到10 Gbps。（也就是说，如果带宽为N Kbps，则使用的数值就是10000000/N）。

MTU单位为字节。

可靠度为255的一小部分，即255是100%。

负载也按照255的分子数给出。

跳次就是一个地数。

由于带宽和延迟所使用的单位稍有点奇怪，因而我们下面给出几个例子。这些都是一些常用介质所使用的默认值。

	Delay	Bandwidth
Satellite	200,000 (2 sec)	20 (500 Mbit)
Ethernet	100 (1 ms)	1,000
1.544 Mbit	2000 (20 ms)	6,476
64 Kbit	2000	156,250
56 Kbit	2000	178,571
10 Kbit	2000	1,000,000
1 Kbit	2000	10,000,000

度量计算

下面的例子是Cisco 8.0（3）中的合成度量值的实际计算方法说明。

```
metric = [K1*bandwidth + (K2*bandwidth)/(256 - load) + K3*delay] *
                        [K5/(reliability + K4)]

    If K5 == 0, the reliability term is not included.
```

```
The default version of IGRP has K1 == K3 == 1, K2 == K4 == K5 == 0
```

相关信息

- IP 路由 支持页
- IGRP 支持页
- 技术支持 - Cisco Systems