

居（除了从其接收该分组的那个邻居）转发之。显然，如果图是相连的，这种方案将最终将广播分组的副本交付给该图中的所有结点。虽然这种方案简单而优雅，但它具有致命缺点（在你继续阅读前，看看你能否猜出这个致命缺点）：如果该图具有圈，则每个广播分组的一个或多个分组副本将无休止地循环。例如，在图 4-44 中，R2 将洪泛到 R3，R3 将洪泛到 R4，R4 将洪泛到 R2，R2 将（再次）洪泛到 R3，等等。这种简单情况导致两个广播分组无休止地循环，一个朝顺时针方向，一个朝逆时针方向。但还可能存在着一种更为不幸的致命缺陷：当一个结点与两个以上其他结点连接时，它将生成并转发广播分组的多个副本，这些副本中的每个又产生多个它们自身的副本（在有两个以上邻居的其他结点处），等等。这种**广播风暴**（broadcast storm）导致无休止的广播分组的复制，将最终导致在该网络中生成大量的广播分组，使得网络变得毫无用处。（参见本章课后作业中分析这种广播风暴增长速率的问题。）

2. 受控洪泛

避免广播风暴的关键是每个结点明智地选择何时洪泛分组，何时不洪泛分组（例如，如果它已经接收并洪泛了某分组的较早副本）。在实践中，这能够以几种方式来实现。

在**序号控制洪泛**（sequence-number-controlled flooding）中，源结点将其地址（或其他唯一的标识符）以及**广播序号**（broadcast sequence number）放入广播分组，再向它的所有邻居发送该分组。每个结点维护它已经收到的、复制的和转发的源地址和每个广播分组的序号列表。当结点接收到一个广播分组时，它首先检查该分组是否在列表中。如果在，丢弃该分组；如果不在，复制该分组并向该结点的所有邻居转发（除了接收到该分组从它的那个结点）。在第 2 章中讨论的 Gnutella 协议在它的覆盖网络中对广播查询使用了序号控制洪泛。（在 Gnutella 中，报文复制和转发在应用层执行，而不是在网络层执行。）

受控洪泛的第二种方法被称为**反向路径转发**（Reverse Path Forwarding, RPF）[Dalal 1978]，有时也称为**反向路径广播**（RPB）。RPF 的基本思想简单且优雅。当一台路由器接收到具有给定源地址的广播分组时，仅当该分组到达的链路正好是位于它自己的返回其源的最短单播路径上，它才向其所有出链路（除了它接收分组的那个）传输报文；否则，路由器只是丢弃该分组而不向任何它的出链路转发分组。因为路由器知道它在这样一条链路上将接收或者已经接收了该分组的一个副本（该链路位于自身返回发送方最短路径上），故这样一个分组能够被丢弃。（你也许应当搞清，事实上，如果发生了这样的情况，环路和广播风暴将不会出现。）注意到 RPF 不使用单播路由选择以实际将分组交付给目的地，它也不要求路由器知道从它自己到源的完整最短路径。RPF 仅需要知道在它到发送方的单播最短路径上的下一个邻居；它仅使用这个邻居的身份以决定是否洪泛一个接收到的广播分组。

图 4-44 举例说明了 RPF。假定用粗线画的链路表示了从接收方到源（A）的最低费用路径。结点 A 最初广播一个源为 A 的分组到结点 C 和 B。结点 B 将向结点 C 和 D 转发它从结点 A 接收到的源为 A 的分组（因为 A 位于到 A 的最低费用路径

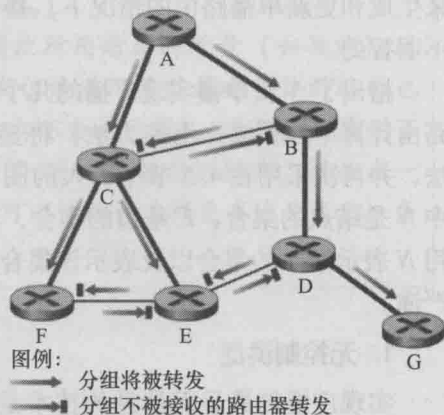


图 4-44 反向路径转发