

的解决方案。特别是,每个对等方只需要知道两个对等方,即它的直接后继和直接前任。但该解决方案也引入了一个新问题。尽管每个对等方仅知道两个邻居对等方,但为了找到负责的键(在最差的情况下),DHT 中的所有 N 个结点将必须绕环转发该报文;平均发送 $N/2$ 条报文。

所以,在设计 DHT 时,在每个对等方必须跟踪的邻居数量与 DHT 为解析一个查询而需要发送的报文数量之间存在着折中。一方面,如果每个对等方联系所有其他对等方(网状覆盖网络),则每个查询仅发送一个报文,但每个对等方必须关联 N 个对等方。在另一方面,使用环形 DHT,每个对等方仅知道两个对等方,但对每个查询平均要发送 $N/2$ 条报文。幸运的是,我们能够完善 DHT 设计,使每个对等方的邻居数量和每次查询的报文数量都保持在可接受的范围内。细化的方案之一是以该环形覆盖网络为基础,但增加“捷径”,使每个对等方不仅联系它的直接后继和直接前任,而且联系分布在环上的数量相对少的捷径对等方。这种具有某些捷径的环形 DHT 的例子显示在图 2-27b 中。使用捷径来加速查询报文的路由选择。具体来说,当某对等方接收到一条查询一个键的报文时,它向最接近该键的邻居(后继邻居或捷径邻居之一)转发该报文。所以,在图 2-27b 中,当对等方 4 接收到请求键 11 的报文,它确定(在它的邻居中)对该键最邻近的对等方是它的捷径邻居 10,并且直接向对等方 10 转发该报文。显然,捷径能够大大减少用于处理查询的报文数量。

下一个自然的问题是:“一个对等方应当有多少条捷径,哪些对等方应当成为这些捷径邻居?”该问题已经受到研究界的高度重视 [Balakrishnan 2003; Androutsellis - Theotokis 2004]。重要的是,研究表明 DHT 能被设计成每个对等方的邻居数量以及每个请求的报文数量均为 $O(\log N)$,其中 N 是对等方的数量。这种设计给出了使用网状和环形覆盖网络拓扑的两种极端解决方案之间的一种满意折中方案。

2. 对等方扰动

在 P2P 系统中,对等方能够不加警示地到来和离去。因此,当设计一个 DHT 时,我们也必须关注存在这种对等方扰动时维护 DHT 的情况。为了更好地理解处理这个问题所使用的方法,我们再次考虑图 2-27a 中的环形 DHT。为处理对等方扰动,我们此时将要求每个对等方联系其第一个和第二个后继(即知道它们的 IP 地址);例如,对等方 4 此时联系对等方 5 和对等方 8。我们也要求每个对等方周期性地证实它的两个后继是存活的(例如,通过周期性地向它们发送 ping 报文并寻求响应)。现在我们考虑当某对等方突然离开时 DHT 如何维护 DHT。例如,假定图 2-27a 中的对等方 5 突然离开。在此情况下,因为对等方 5 不再响应 ping 报文,在离开的对等方之前的两个对等方(4 和 3)得知 5 已经离开。对等方 4 和 3 因此需要更新它们后继的状态信息。我们考虑对等方 4 更新其状态的方法:

- 对等方 4 用它的第二个后继(对等方 8)来代替它的第一个后继(对等方 5)。
- 然后对等方 4 向新的第一个后继询问它的直接后继(对等方 10)的标识符和 IP 地址。然后对等方 4 将对等方 10 标记为它的第二个后继。

在课后习题中,请你确定对等方 3 是如何更新它的覆盖路由选择信息的。

在简要地讨论了一个对等方离开时必须要做的事后,我们现在考虑当一个对等方要加入 DHT 时发生的事情。假如一个标识符为 13 的对等方要加入该 DHT,在加入时,它仅知