

的网格。这种拓扑结构不仅较网格具有更强的容错能力而且其直径也比较小,因为对角之间的通信只需要两跳。

图8-16e中的立方体(cube)是一种规则的三维拓扑结构。我们展示的是 $2 \times 2 \times 2$ 立方体,更一般的情形则是 $k \times k \times k$ 立方体。在图8-16f中,是一种用两个三维立方体加上对应边连接所组成四维立方体。我们可以仿照图8-16f的结构并且连接对应的节点以组成四个立方体组块来制作五维立方体。为了实现六维,可以复制四个立方体的块并把对应节点互连起来,以此类推。以这种形式组成的 n 维立方体称为超立方体(hypercube)。许多并行计算机采用这种拓扑结构,因为其直径随着维数的增加线性增长。换句话说,直径是节点数的自然对数,例如,一个10维的超立方体有1024个节点,但是其直径仅为10,有着出色的延迟特性。注意,与之相反的是,1024的节点如果按照 32×32 网格布局则其直径为62,较超立方体相差了六倍多。对于超立方体而言,获得较小直径的代价是扇出数量(fanout)以及由此而来的连接数量(及成本)的大量增加。

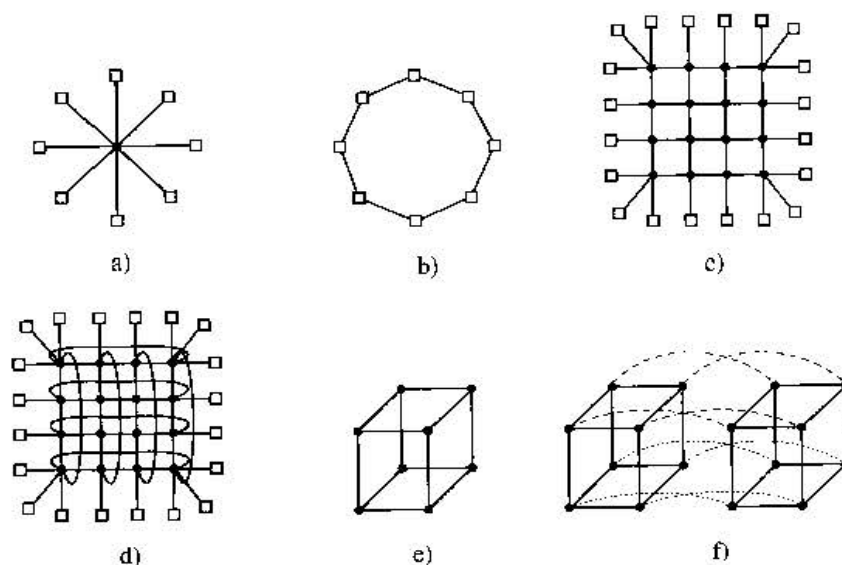


图8-16 各种互连拓扑结构: a) 单交换机; b) 环; c) 网格; d) 双凸面; e) 立方体; f) 四维超立方体

在多计算机中可采用两种交换机制。在第一种机制里,每个消息首先被分解(由用户软件或网络接口进行)成为有最大长度限制的块,称为包(packet)。该交换机制称为存储转发包交换(store-and-forward packet switching),由源节点的网络接口卡注入到第一个交换机的包组成,如图8-17a所示。比特串一次进来一位,当整个包到达一个输入缓冲区时,它被复制到沿着其路径通向下一个交换机的队列当中,如图8-17b所示。当包到达目标节点所连接的交换机时,如图8-17c所示,该包被复制进入目标节点的网络接口卡,并最终到达其RAM。

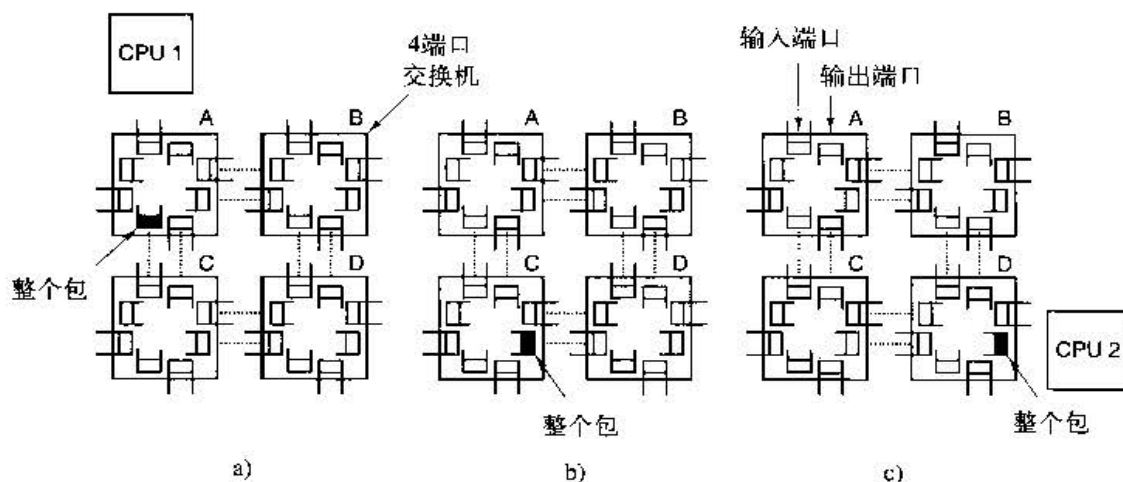


图8-17 存储转发包交换