

## DHT (Tabelas de Hash Distribuídas)

- Será considerado o sistema Chord sem considerar proximidade de redes para explicação do funcionamento da DHT.
  - Usa um espaço de endereçamento de  $m$  bits para chaves (usualmente 128 ou 160);
  - Uma entidade com chave  $k$  cai sob a jurisdição do nó que tenha o menor identificador  $id \geq k$ , nó denominado *sucessor* de  $k \rightarrow succ(k)$
  - Cada nó  $p$  monitora  $succ(p+1)$  e  $pred(p)$
  - Cada nó mantém uma **tabela de derivação** de no máximo  $m$  entradas, então  $FT_p[i] = succ(p + 2^{i-1})$

# DHT – Tabelas de hash distribuídas

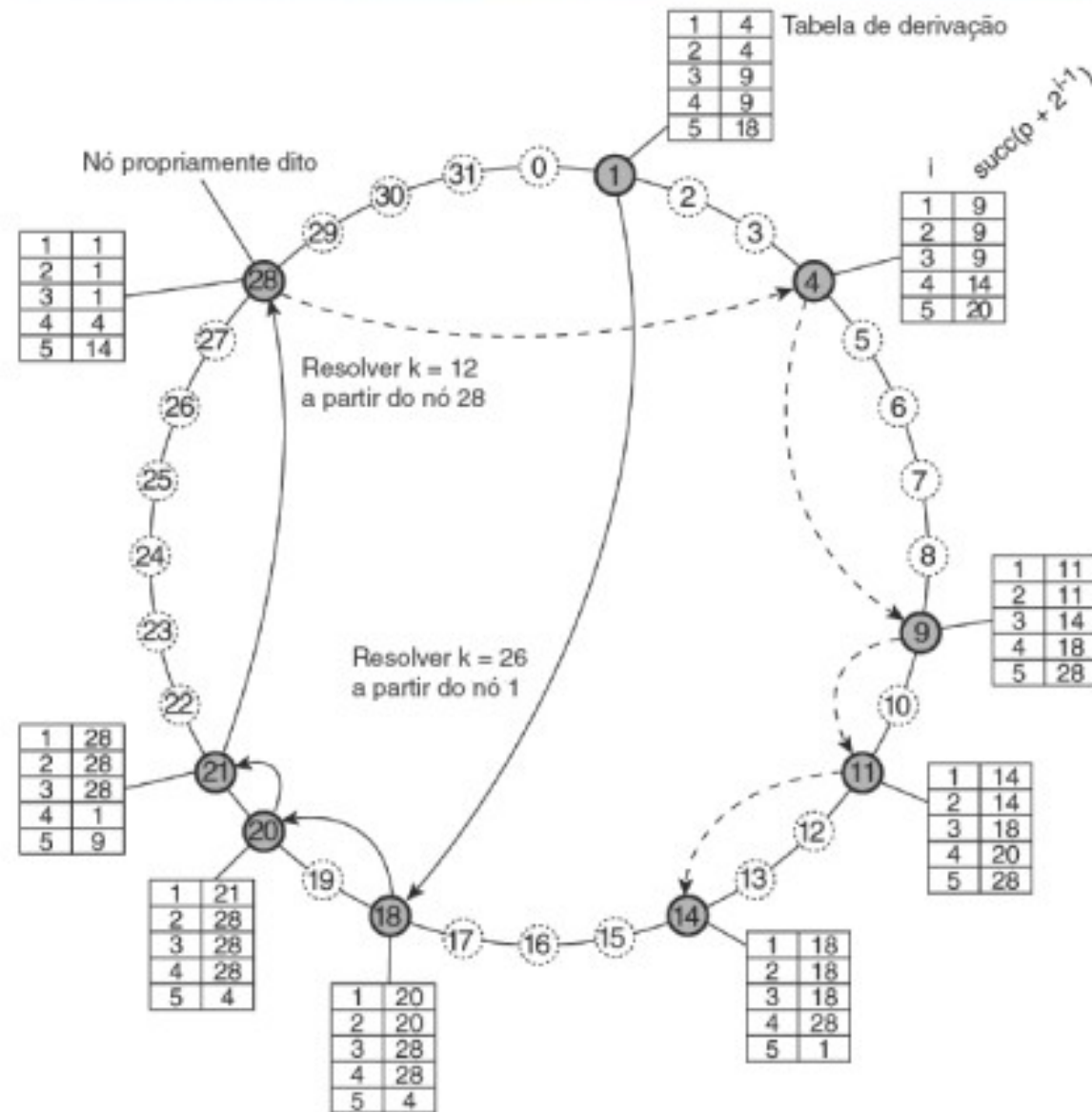


Figura 5.4 Resolução da chave 26 a partir do nó 1 e da chave 12 a partir do nó 28 em um sistema Chord.



# DHT – Inclusão e Remoção de nós

- Para um nó entrar na rede:
  - $p$  contata nó arbitrário e procura  $\text{succ}(p+1)$ , adicionando-se ao anel.
- Para um nó sair da rede:
  - $p$  contata seu predecessor informando a saída.
- Correção de falhas:
  - Verificação periódica:  
 $\text{pred}(\text{succ}(q+1)) == q?$  → Caso negativo, ajusta  $\text{FT}[1]$  para o resultado.
  - $\text{pred}(q) == \text{falha?}$  → ajusta  $\text{pred}(q)$  para 'desconhecido'.
    - Corrigido automaticamente quando um outro nó  $q$  verifica  $\text{pred}(\text{succ}(q+1)) == q$  e o resultado é 'desconhecido' → informa a  $q+1$  que possivelmente é seu predecessor!