

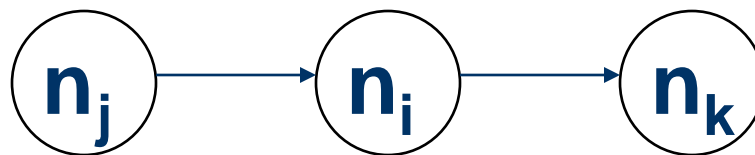
# **Sistemas Distribuídos – Fundamentos de Algoritmos Distribuídos**

Prof. DSc Marcelo  
Lisboa Rocha



# Introdução

- O modelo fundamental é definido sobre um Grafo (G)
- $G=(N, E/D)$ , onde  $N$  é o conjunto de vértices representando processadores, processos, entre outros, e  $E$  é o conjunto de arestas não-orientadas e  $D$  é o conjunto de arestas orientadas.
- Para cada nó  $n_i \in N$ , temos dois conjuntos:
  - (In):  $In_i \subseteq N$ : conjunto de vizinhos  $n_j$  de  $n_i$  tais que  $n_j \rightarrow n_i \in D$
  - (Out):  $Out_i \subseteq N$ : conjunto de vizinhos  $n_k$  de  $n_i$  tais que  $n_i \rightarrow n_k \in D$



# Introdução

- **Modelos de Temporização de G**
  - G é Síncrono
  - G é Assíncrono
- **Observações**
  - Não existe sistema real síncrono, mas sincronizável.
  - No caso assíncrono, as computações são todas reativas no sentido em que um nó realiza computação em duas situações:
    - Ao iniciar “espontaneamente”.
    - Ao receber mensagem de algum vizinho.

# Exemplo de Algoritmo Assíncrono

## Algoritmo

Para cada  $n_i \in N$  (tem  $|In_i|$  vizinhos)

Início espontâneo do algoritmo (um ou mais dos  $n_i$ )

- Compute;
- Se necessário Envie Msg para os membros de algum subconjunto de  $Out_i$ ;

Repetir

Receba Msg de algum  $n_i$

- Compute;
- Se necessário Envie Msg para os membros de algum subconjunto de  $Out_i$ ;

Até  $n_i$  saiba que (ocorreu terminação global) ou (não receberá mais mensagens).

Fim Para

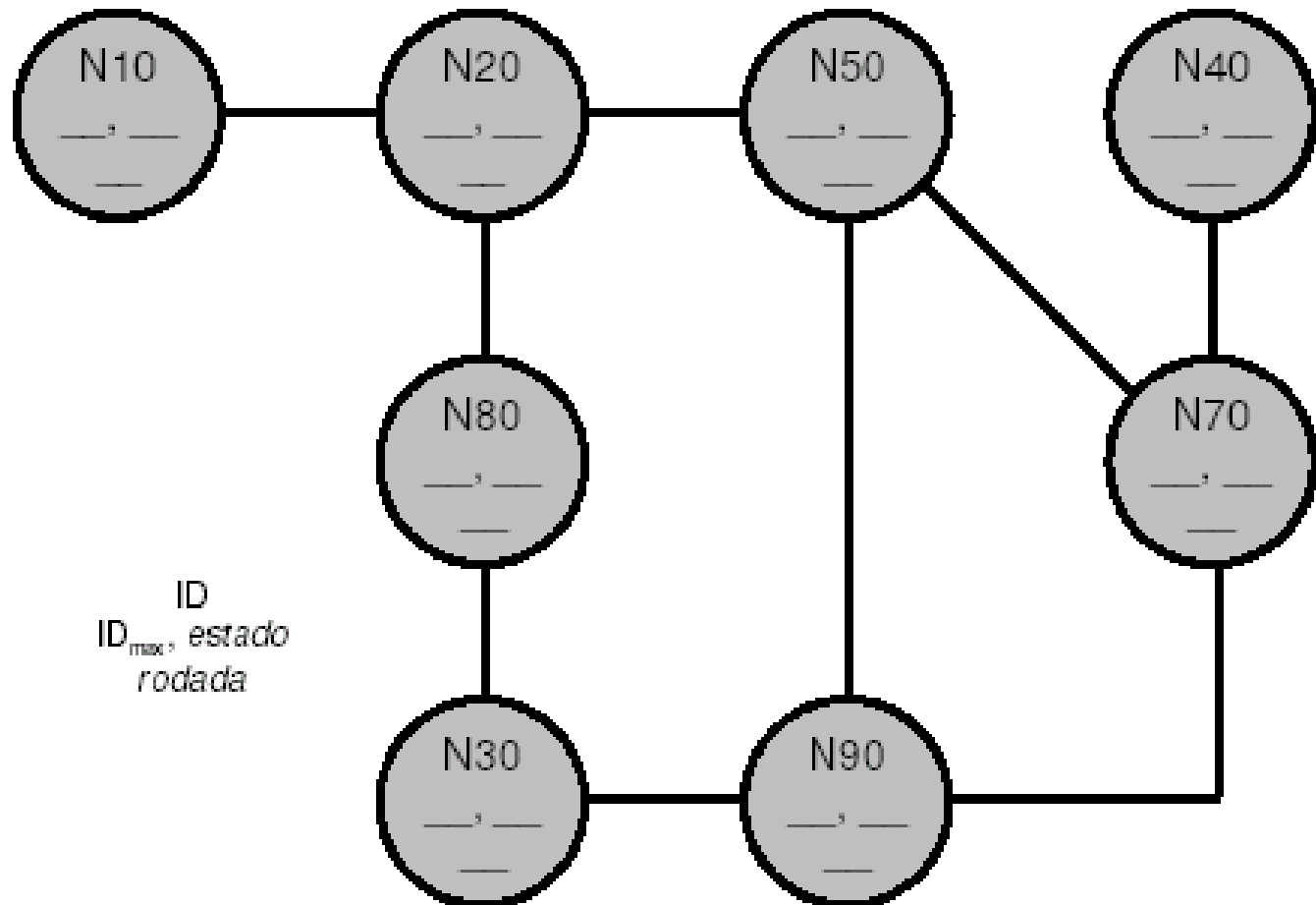
# Eleição de Líder em Redes Arbitrárias

- Cada nodo é univocamente identificado por um ID.
- A rede é representada por um grafo  $G$  conectado não-direcionado.
- Objetivo: um nodo apenas se declarar líder, mudando seu status para líder.
- Em alguns casos, que demais elementos reconheçam que **não** se tornaram líderes, mudando seu status para não-líder.
- É possível que o número de nodos,  $N$ , e o diâmetro,  $diam$ , seja conhecido de todos os processos.

# Eleição de Líder: Algoritmo Simples de Inundação

- Premissa: processos conhecem o diâmetro *diam*
- Algoritmo inunda a rede com o ID máximo, de forma que cada nodo propague a seus vizinhos o valor mais alto registrado até o momento
- Descrição:
  - cada processo armazena o máximo ID visto
  - a cada rodada, cada processo envia o atual valor máximo para todos seus vizinhos
  - após *diam* rodadas, se o ID máximo registrado por um processo é seu próprio ID, então ele é o líder, caso contrário, ele é um não-líder
- E o conhecimento da identidade do líder em outros processos?

# Eleição de Líder: Algoritmo Simples de Inundação



# Eleição de Líder: Algoritmo Simples de Inundação

- **Análise de Complexidade:**
  - Complexidade de tempo:  $\text{diam}$  rodadas
  - Complexidade de comunicação:  $\text{diam} \times \text{número\_arestas}$
- **Otimização 1:**
  - Se o processo  $p_i$  recebe valor maior de processo  $p_j$ ,  $p_j$  não precisa enviar mensagem para  $p_i$
- **Otimização 2:**
  - Processos enviam o valor máximo de ID apenas quando o mesmo é descoberto (acrescentar uma variável “new\_info” a cada nó)
  - No pior caso dá o mesmo resultado.