

# Unidade II:

## Camada de Rede

Prof. Max do Val Machado



PUC Minas

Instituto de Ciências Exatas e Informática  
Departamento de Ciência da Computação  
Disciplina Arquitetura TCP/IP I

- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

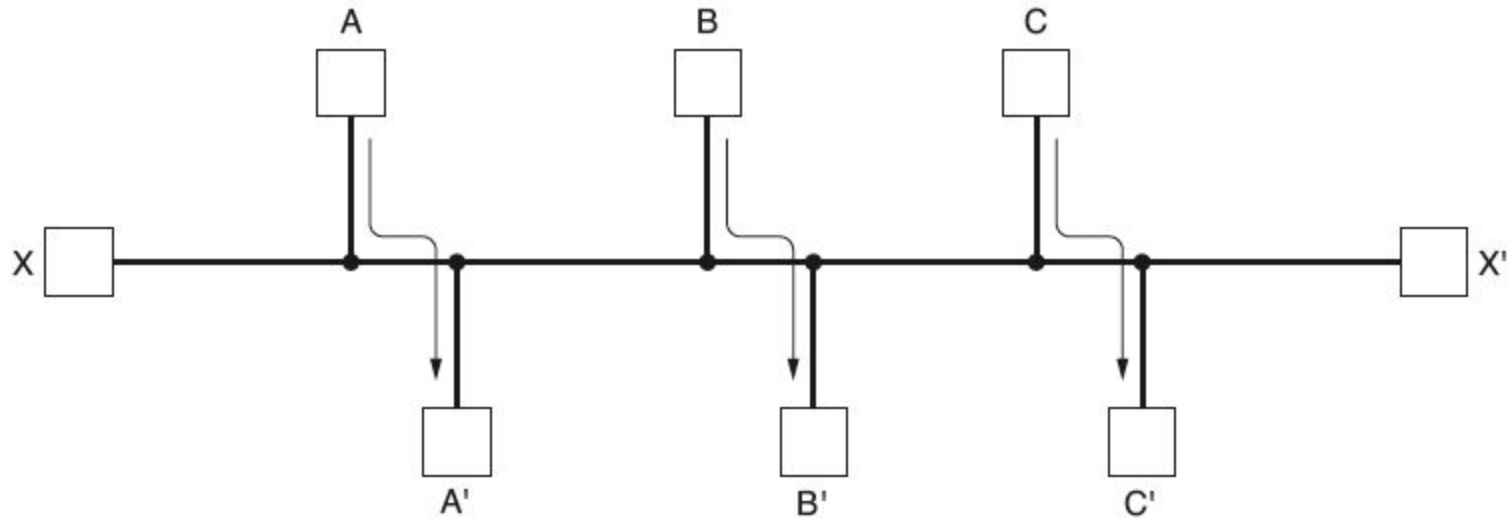
- **Introdução**
- Algoritmos de Roteamento
- Roteamento na Internet

# Algoritmo de Roteamento

- É a parte do software da camada de rede responsável pela decisão sobre a interface de saída a ser usada na transmissão do pacote de entrada
- Tem como propriedades desejadas:
  - Exatidão
  - Simplicidade
  - Robustez (pensando em escalabilidade)
  - Estabilidade (convergir para uma rota viável)
  - Equidade
  - Eficiência

# Equidade vs Eficiência

- Supondo um tráfego intenso de cada  $i$  para  $i'$ , a eficiência global máxima acontece desativando completamente o fluxo de  $x$  para  $x'$ , perdendo equidade. Nesse caso, o ideal é um meio-termo



# Roteamento em Redes de Datagrama vs Circuito Virtual

- Em redes de datagrama, a decisão será tomada para cada vez que o pacote de dados recebido
- Em redes de circuito virtual, a decisão será tomada somente quando um novo circuito virtual estiver sendo estabelecido

# Algoritmos de Roteamento Estáticos e Dinâmicos

- Estáticos ou não-adaptativos
  - A escolha da rota é previamente calculada off-line, sendo transferida para os roteadores quando a rede é inicializada
  - Desconsidera medidas e estimativas atuais do tráfego, topologia e carga da rede
- Dinâmicos ou Adaptativos
  - Mudam suas decisões de roteamento para refletir mudanças na topologia ou tráfego

- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- Algoritmo do caminho mais curto
- *Flooding*
- *Gossiping*
- Roteamento por vetor de distância
- Roteamento de estado de enlace
- Roteamento Hierárquico
- Roteamento em redes infraestruturadas
- Roteamento em redes *ad hoc*
- Roteamento em redes *mesh*

- Introdução
- **Algoritmos de Roteamento**
- Roteamento na Internet



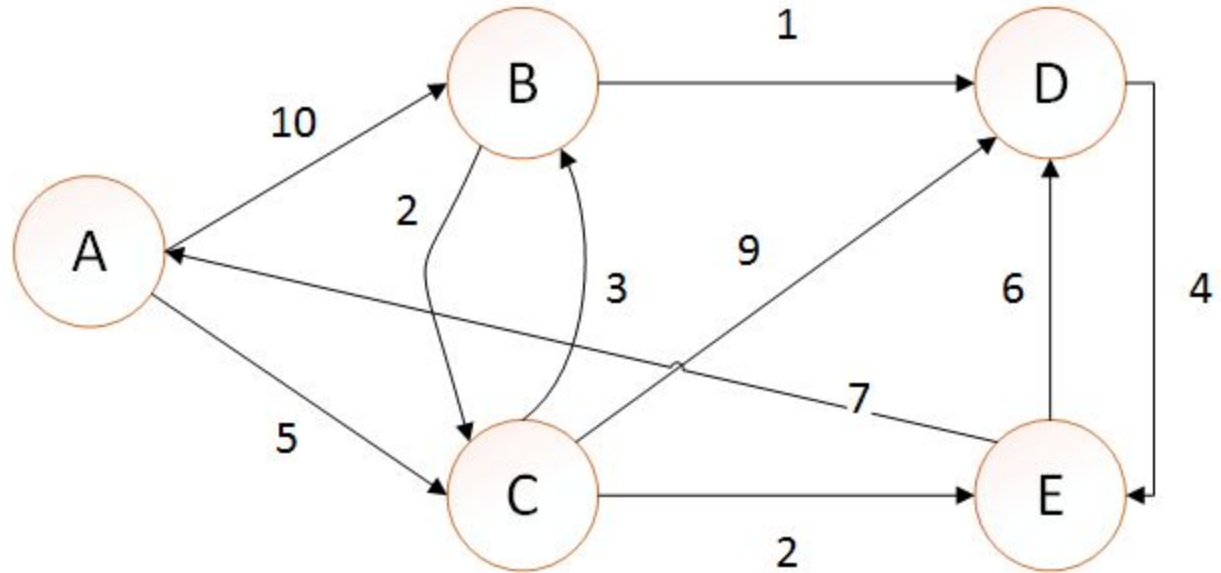
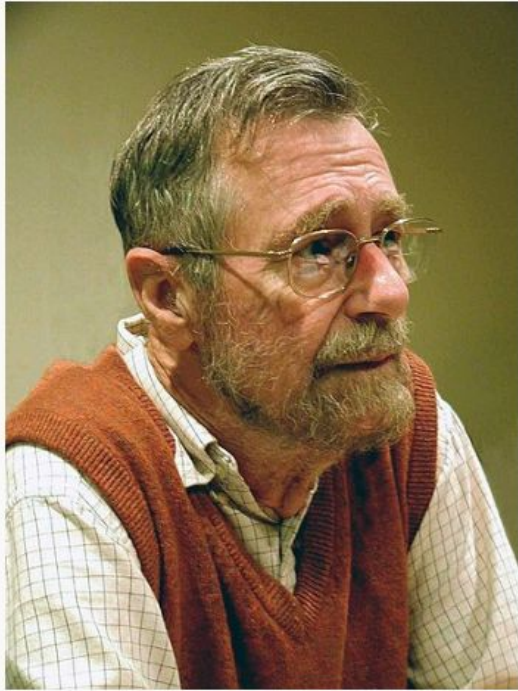
- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- **Algoritmo do caminho mais curto**
- *Flooding*
- *Gossiping*
- Roteamento por vetor de distância
- Roteamento de estado de enlace
- Roteamento Hierárquico
- Roteamento em redes infraestruturadas
- Roteamento em redes *ad hoc*
- Roteamento em redes *mesh*

- Introdução
- **Algoritmos de Roteamento**
- Roteamento na Internet

# Algoritmo do Caminho Mais Curto (Dijkstra)

Edsger Dijkstra



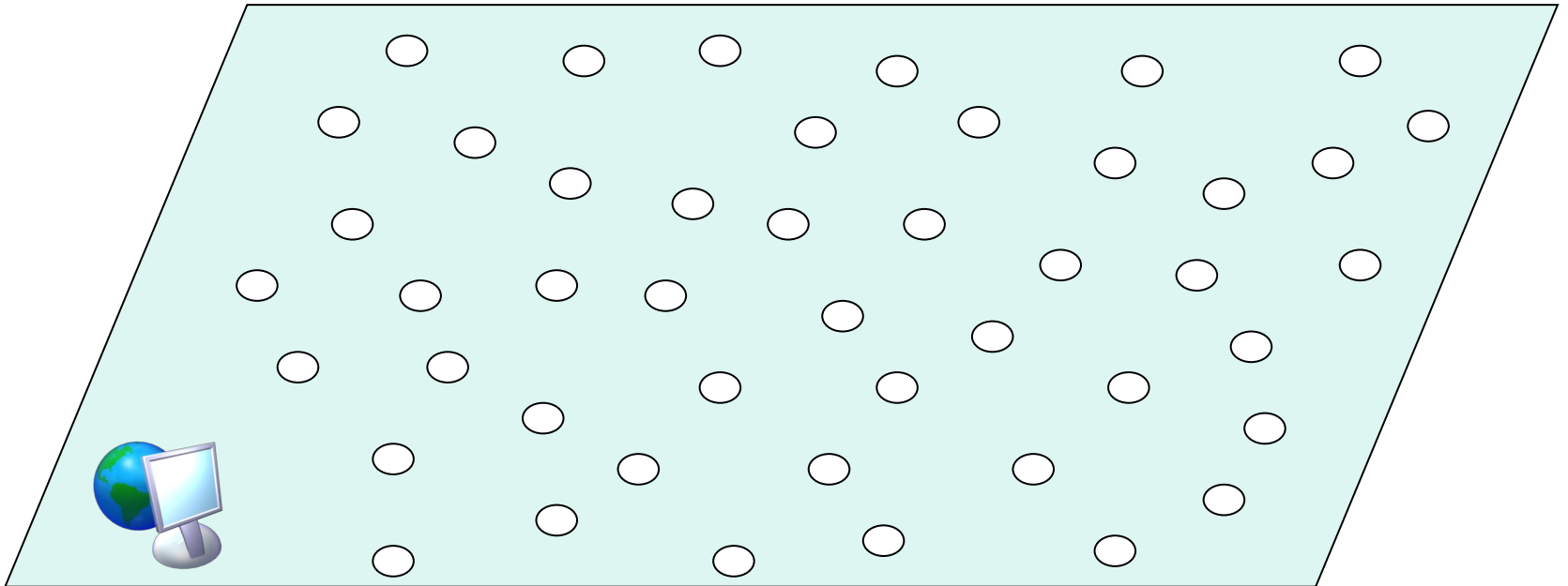
- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- Algoritmo do caminho mais curto
- ***Flooding***
- *Gossiping*
- Roteamento por vetor de distância
- Roteamento de estado de enlace
- Roteamento Hierárquico
- Roteamento em redes infraestruturadas
- Roteamento em redes *ad hoc*
- Roteamento em redes *mesh*

- Introdução
- **Algoritmos de Roteamento**
- Roteamento na Internet

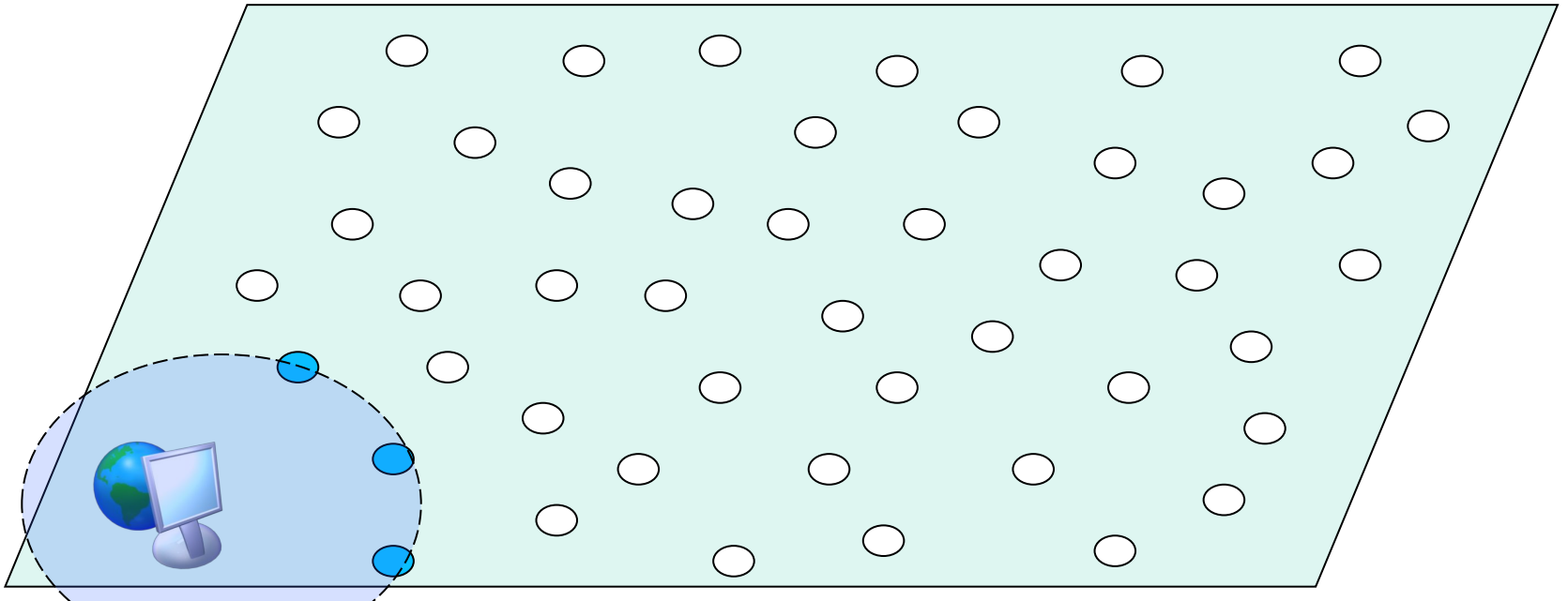
# Flooding (Inundação)

- Quando um nó recebe um pacote pela primeira vez, ele propaga



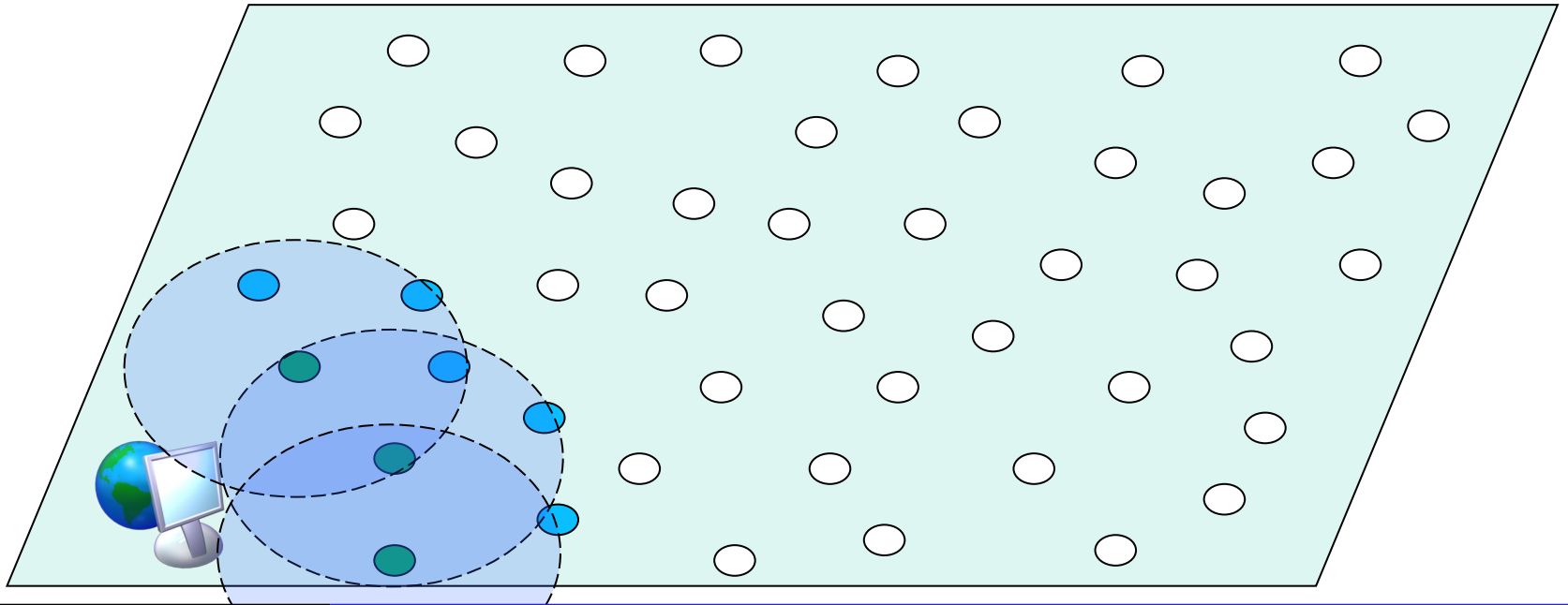
# Flooding (Inundação)

- Quando um nó recebe um pacote pela primeira vez, ele propaga



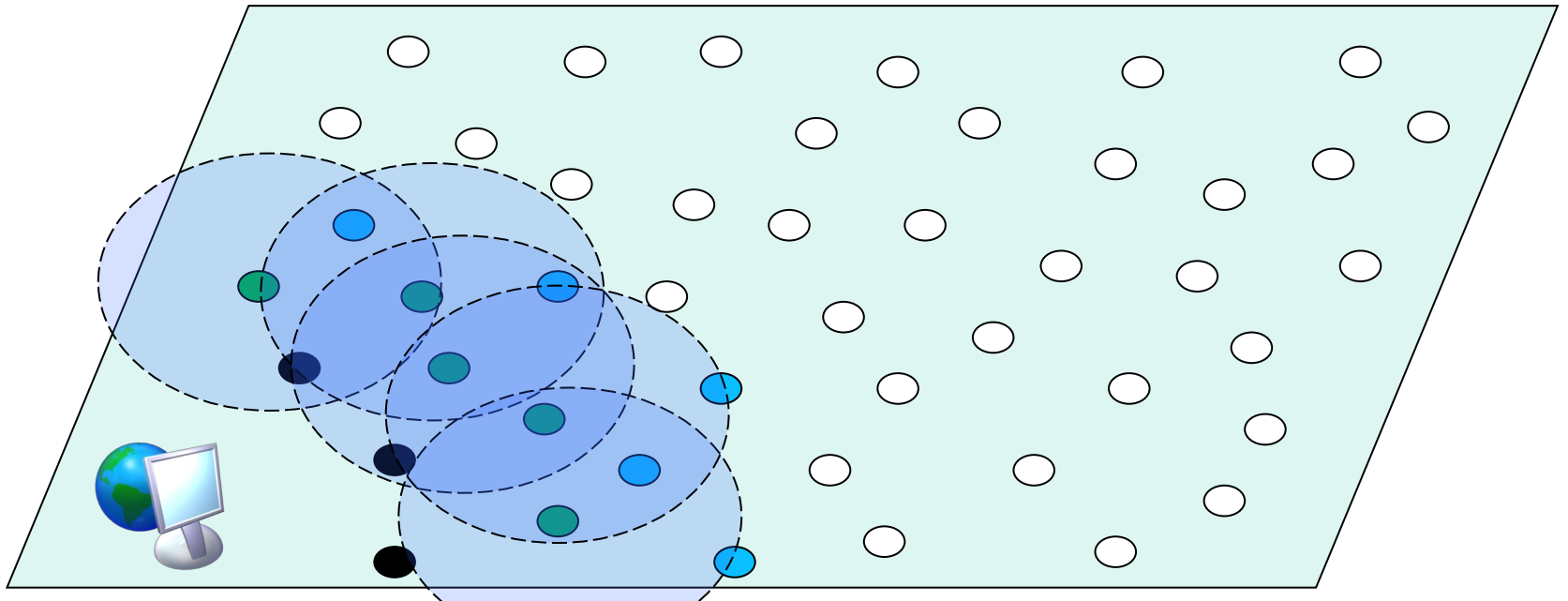
# Flooding (Inundação)

- Quando um nó recebe um pacote pela primeira vez, ele propaga



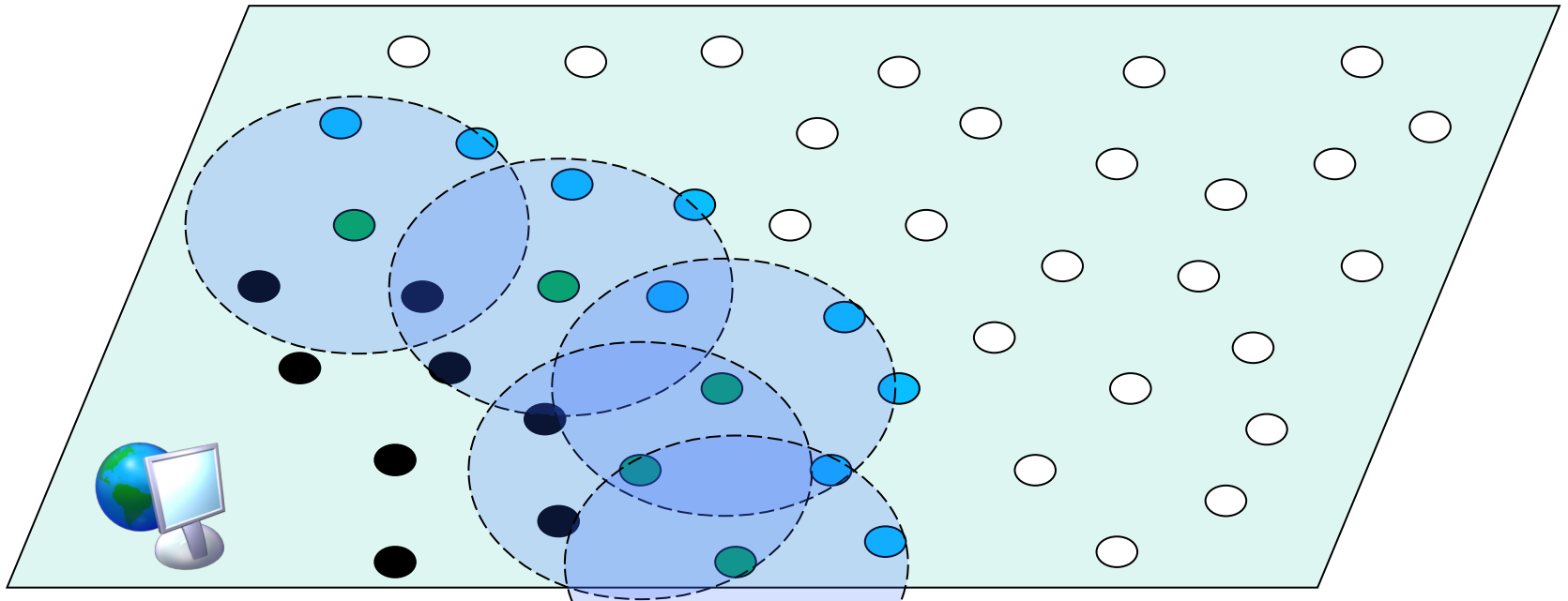
# Flooding (Inundação)

- Quando um nó recebe um pacote pela primeira vez, ele propaga



# Flooding (Inundação)

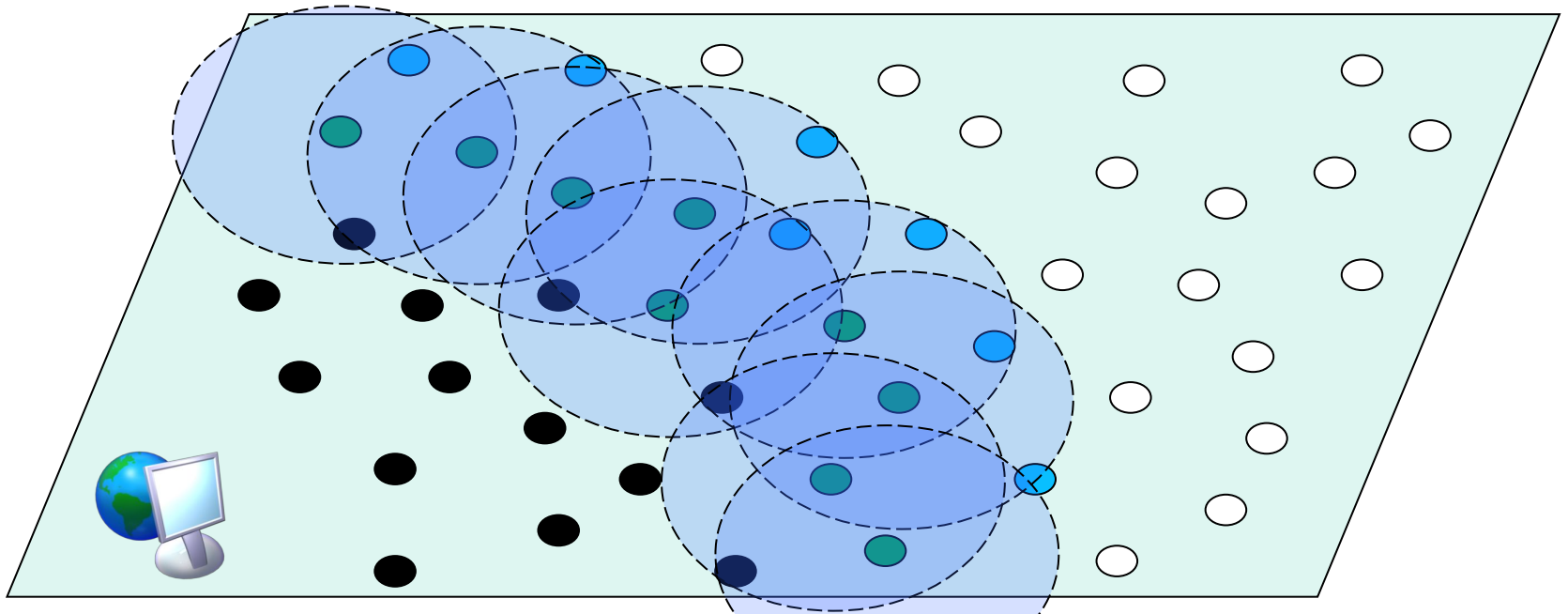
- Quando um nó recebe um pacote pela primeira vez, ele propaga





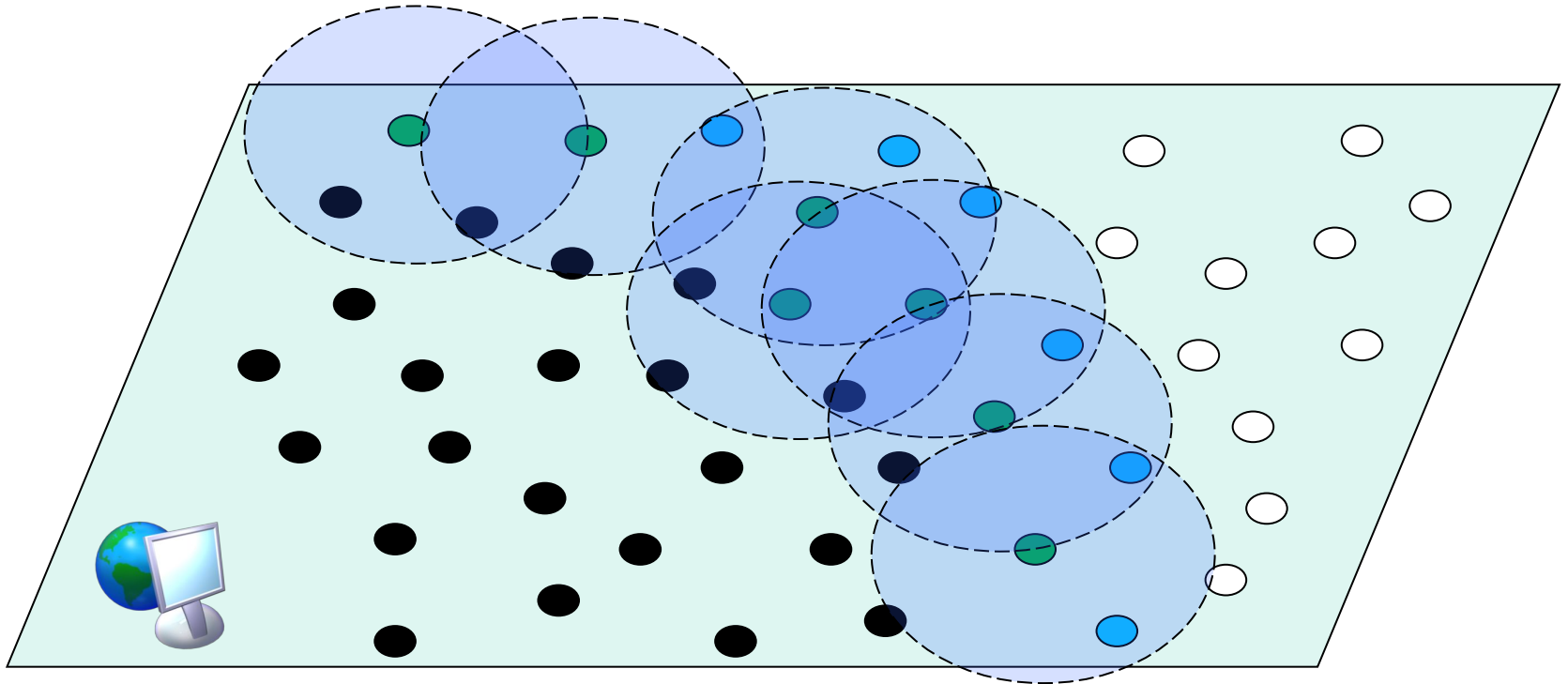
# Flooding (Inundação)

- Quando um nó recebe um pacote pela primeira vez, ele propaga



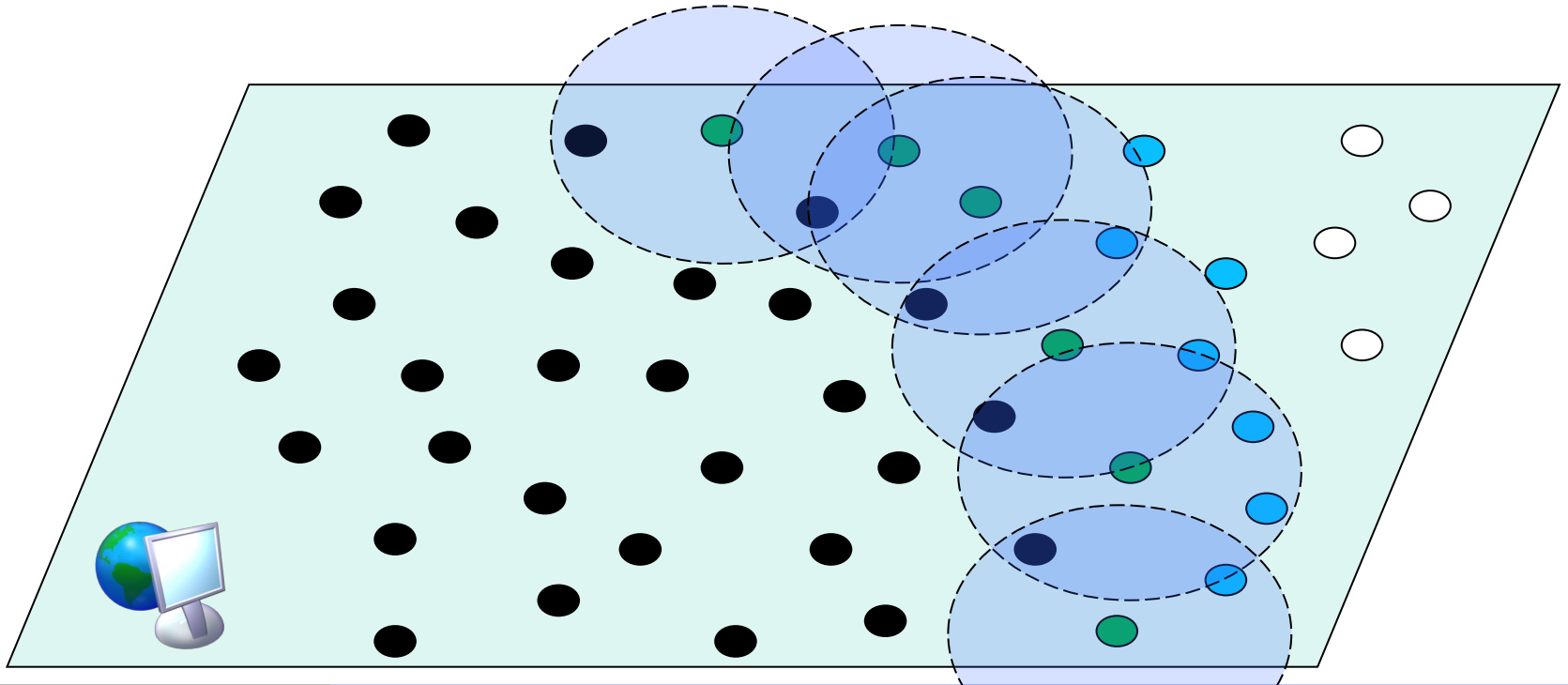
# Flooding (Inundação)

- Quando um nó recebe um pacote pela primeira vez, ele propaga



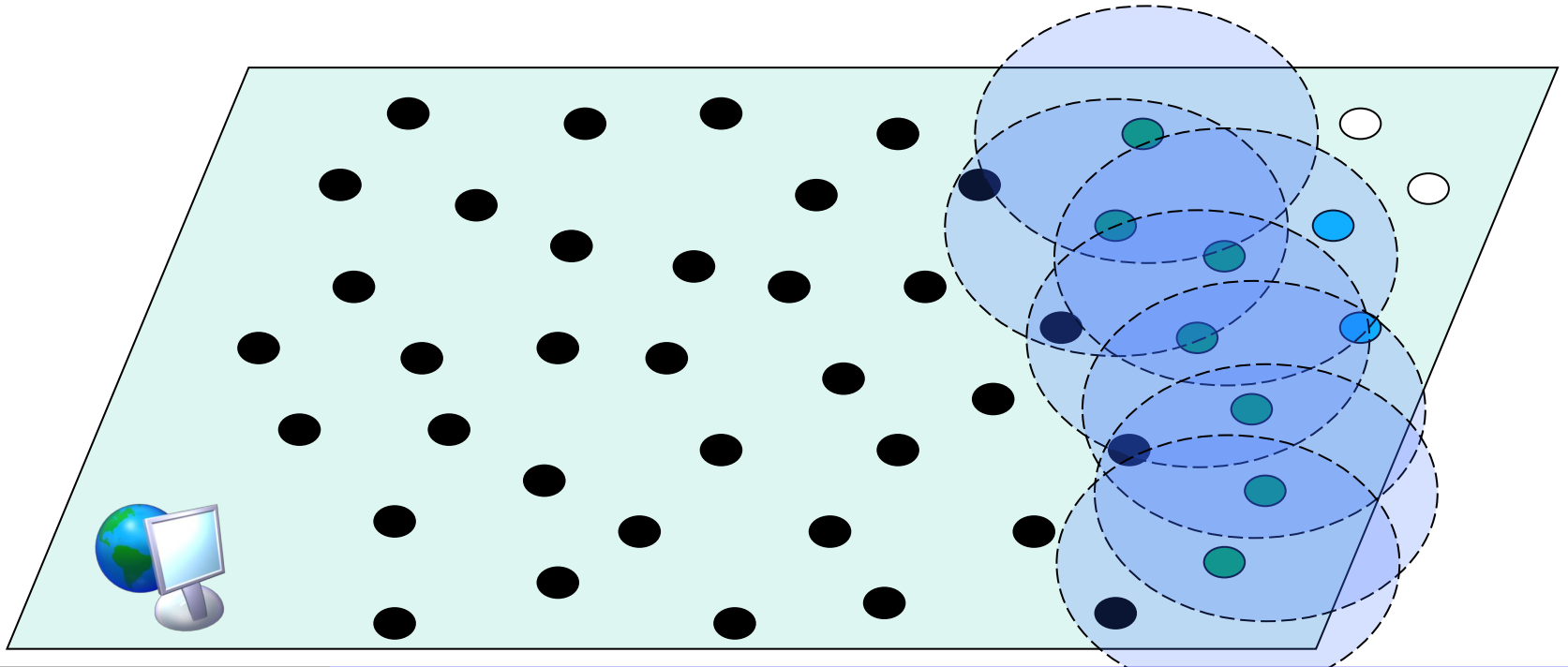
# Flooding (Inundação)

- Quando um nó recebe um pacote pela primeira vez, ele propaga



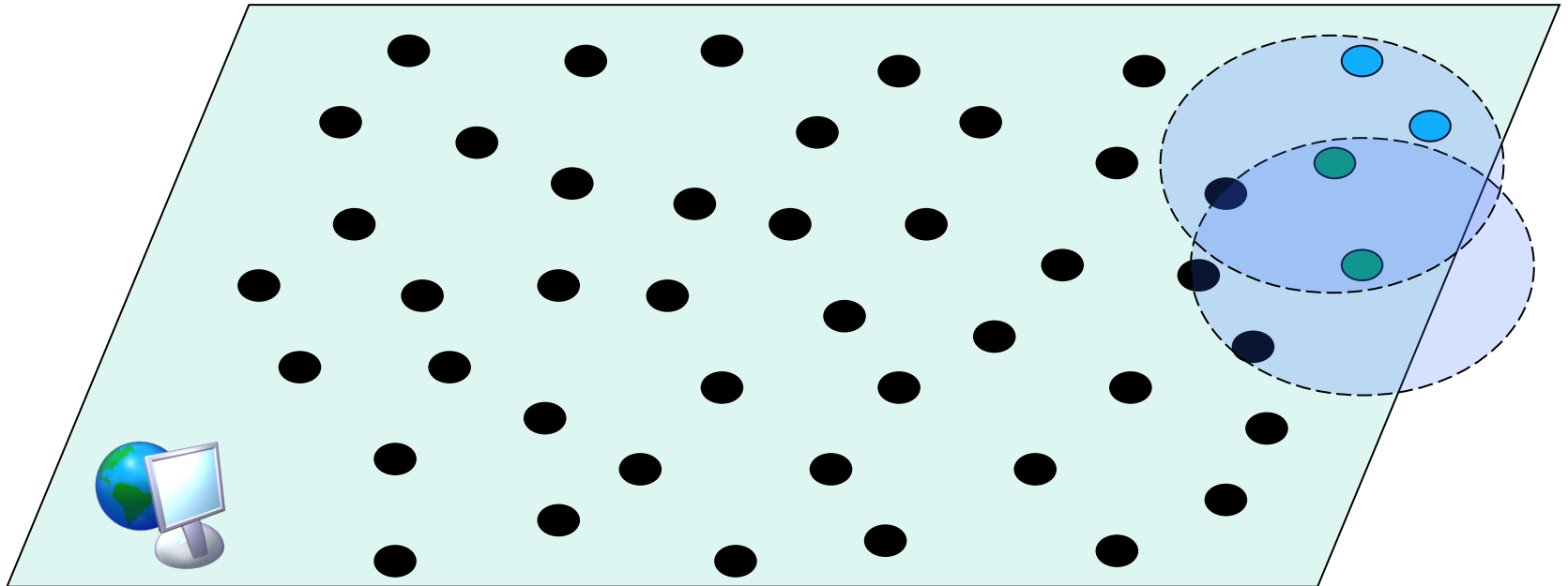
# Flooding (Inundação)

- Quando um nó recebe um pacote pela primeira vez, ele propaga



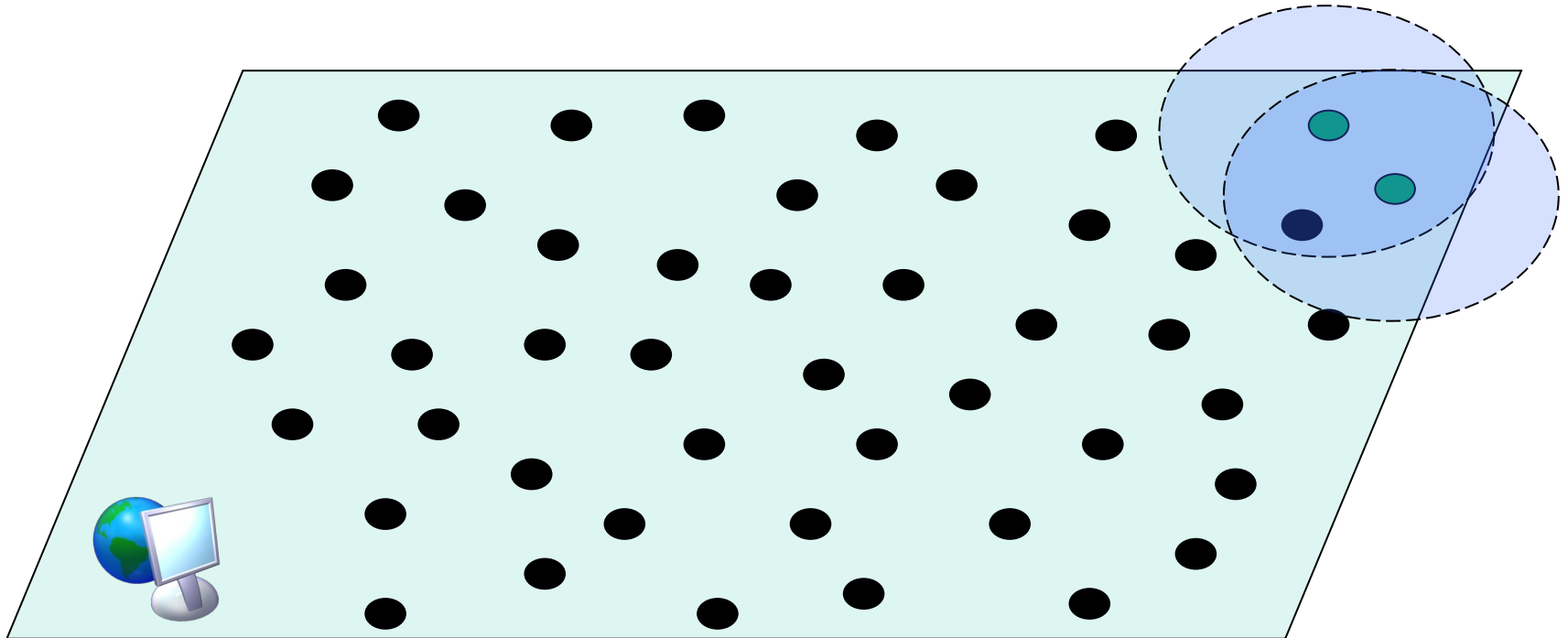
# Flooding (Inundação)

- Quando um nó recebe um pacote pela primeira vez, ele propaga



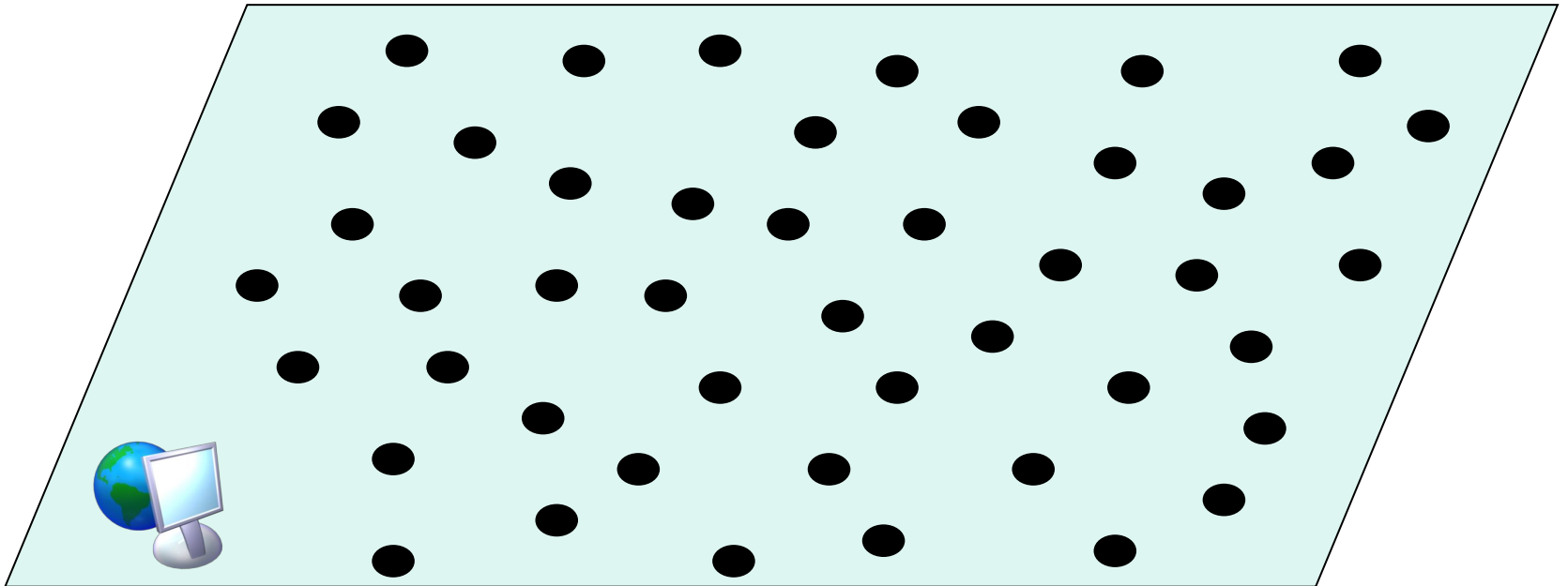
# Flooding (Inundação)

- Quando um nó recebe um pacote pela primeira vez, ele propaga



# *Flooding* (Inundação)

- Quando um nó recebe um pacote pela primeira vez, ele propaga



- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

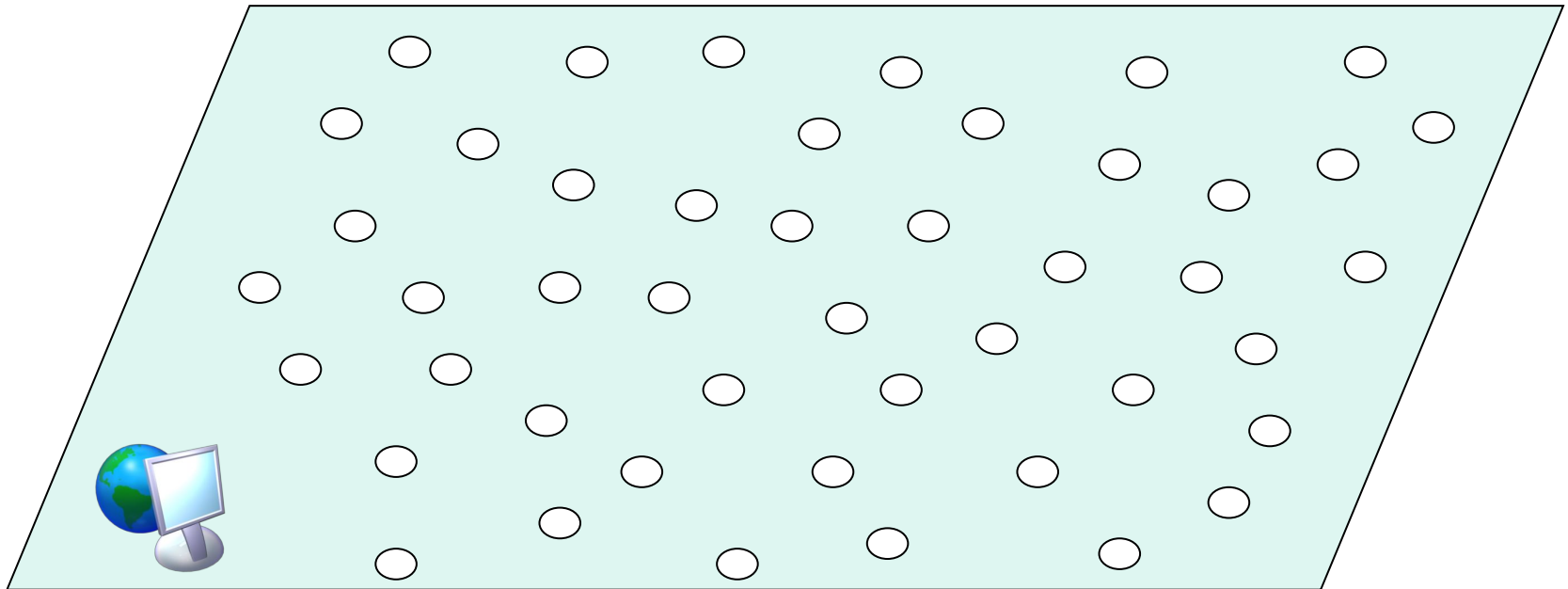
- Algoritmo do caminho mais curto
- *Flooding*
- ***Gossiping***
- Roteamento por vetor de distância
- Roteamento de estado de enlace
- Roteamento Hierárquico
- Roteamento em redes infraestruturadas
- Roteamento em redes *ad hoc*
- Roteamento em redes *mesh*

- Introdução
- **Algoritmos de Roteamento**
- Roteamento na Internet



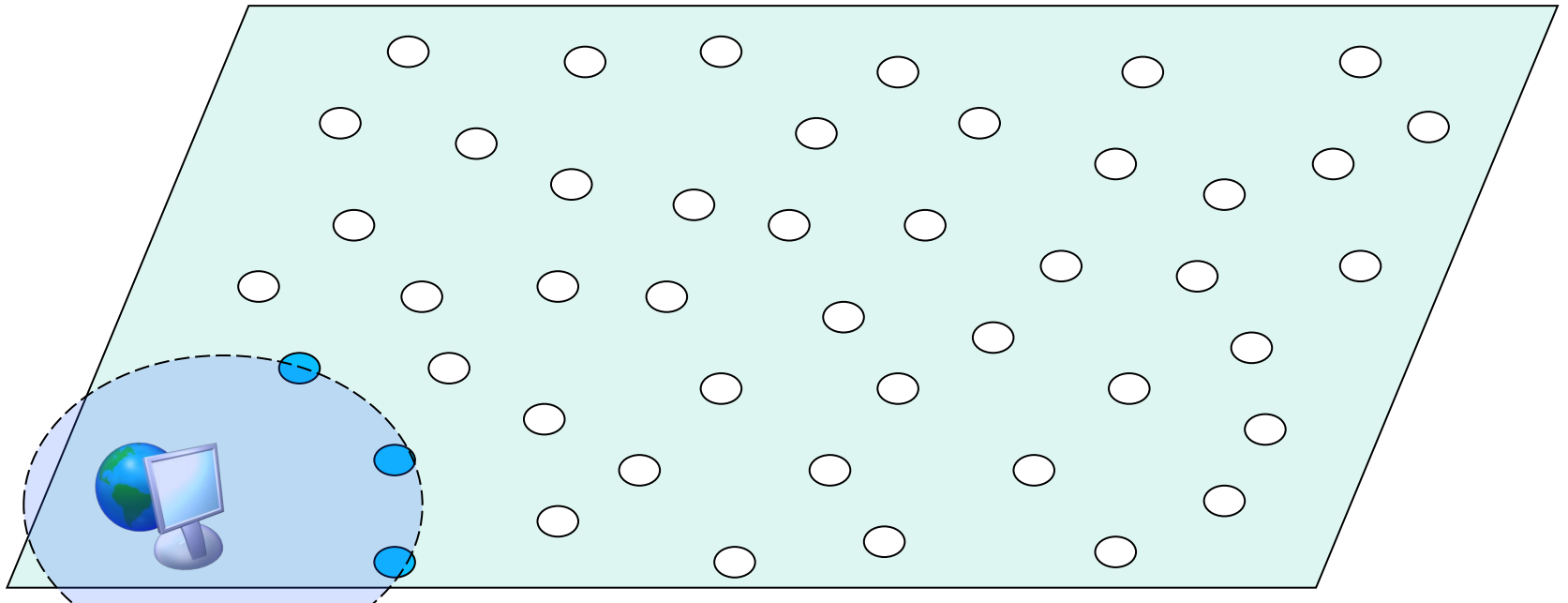
# *Gossiping* (Fofoca ou *Flooding* Probabilístico)

- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)



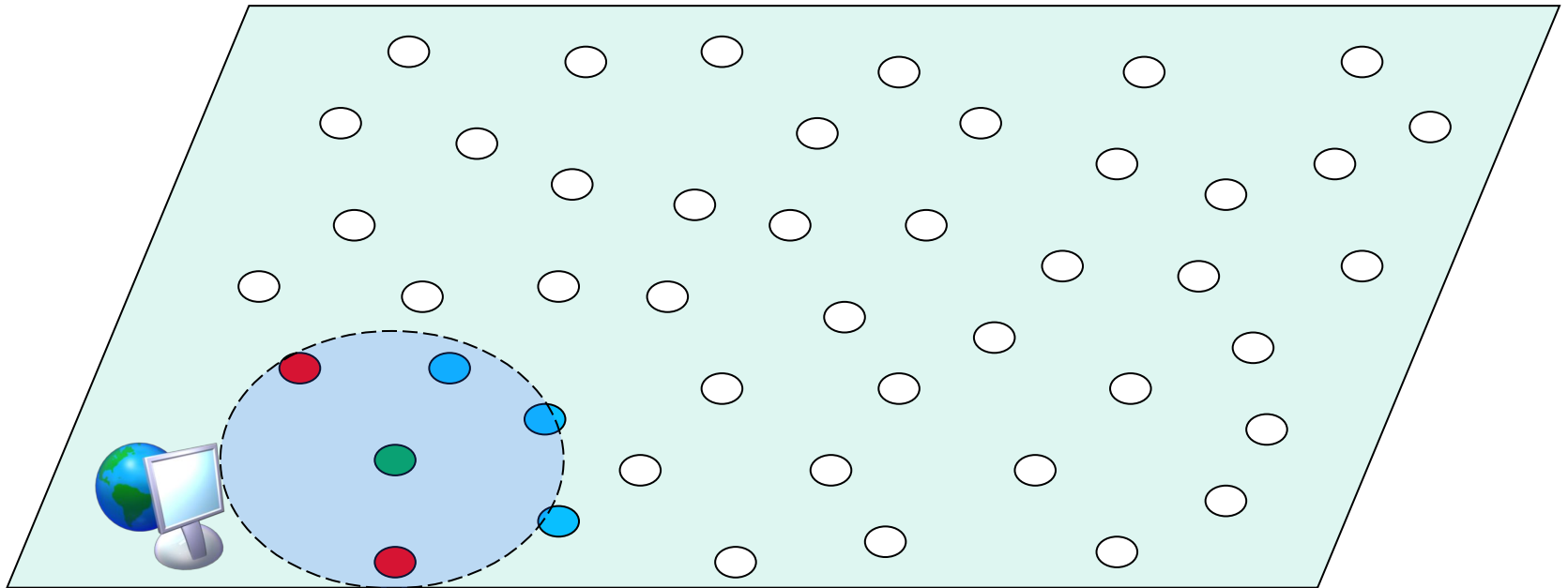
# *Gossiping* (Fofoca ou *Flooding* Probabilístico)

- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)



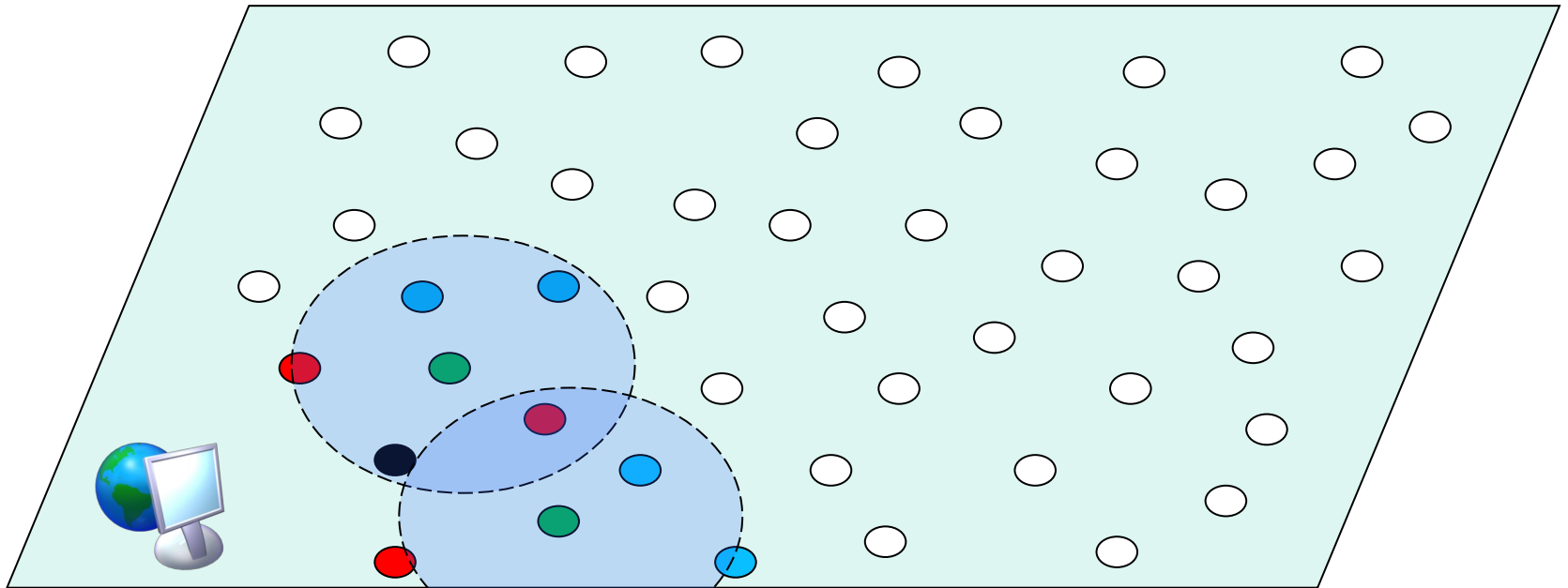
# Gossiping (Fofoca ou Flooding Probabilístico)

- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)



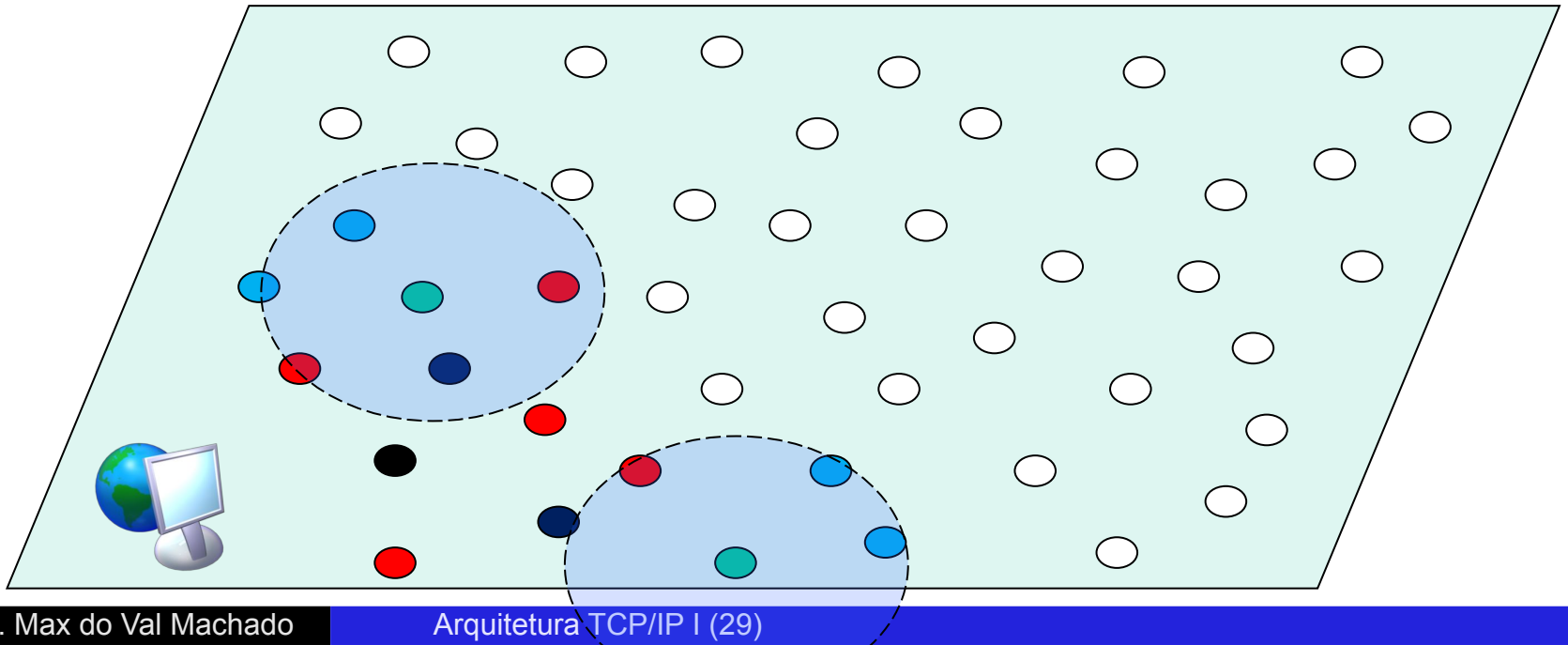
# Gossiping (Fofoca ou Flooding Probabilístico)

- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)



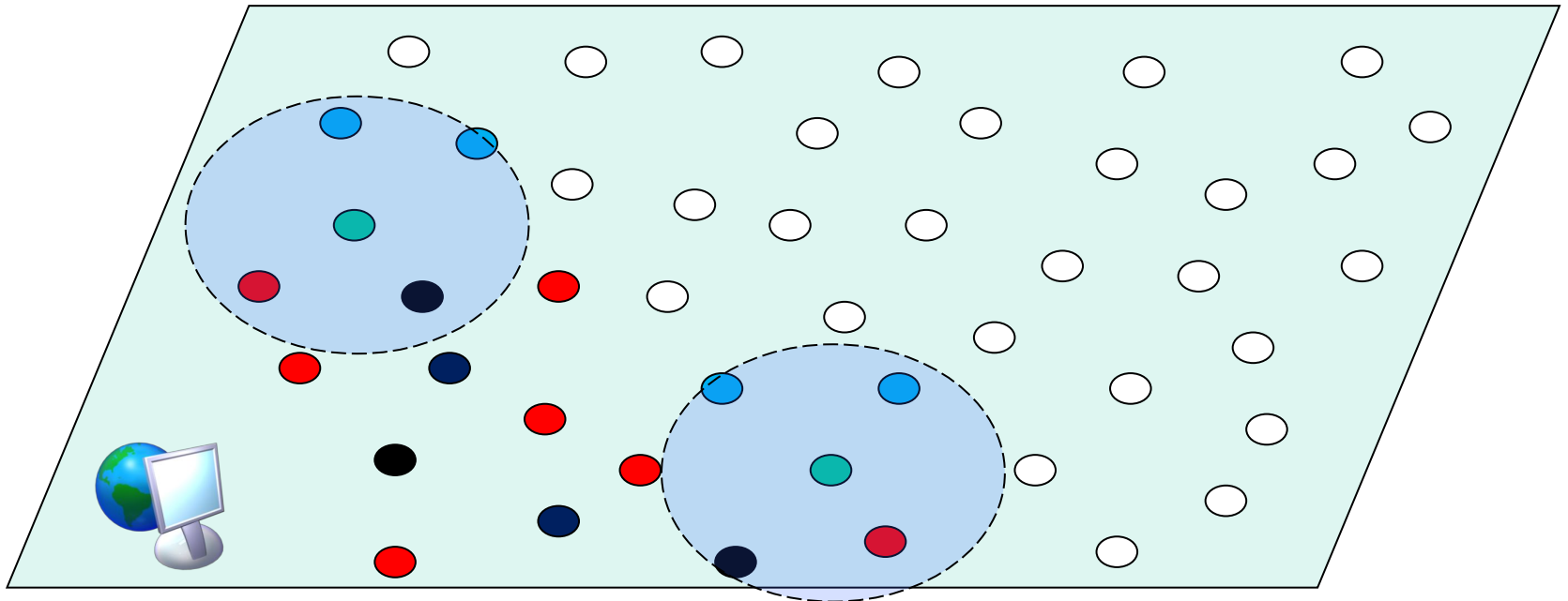
# Gossiping (Fofoca ou Flooding Probabilístico)

- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)



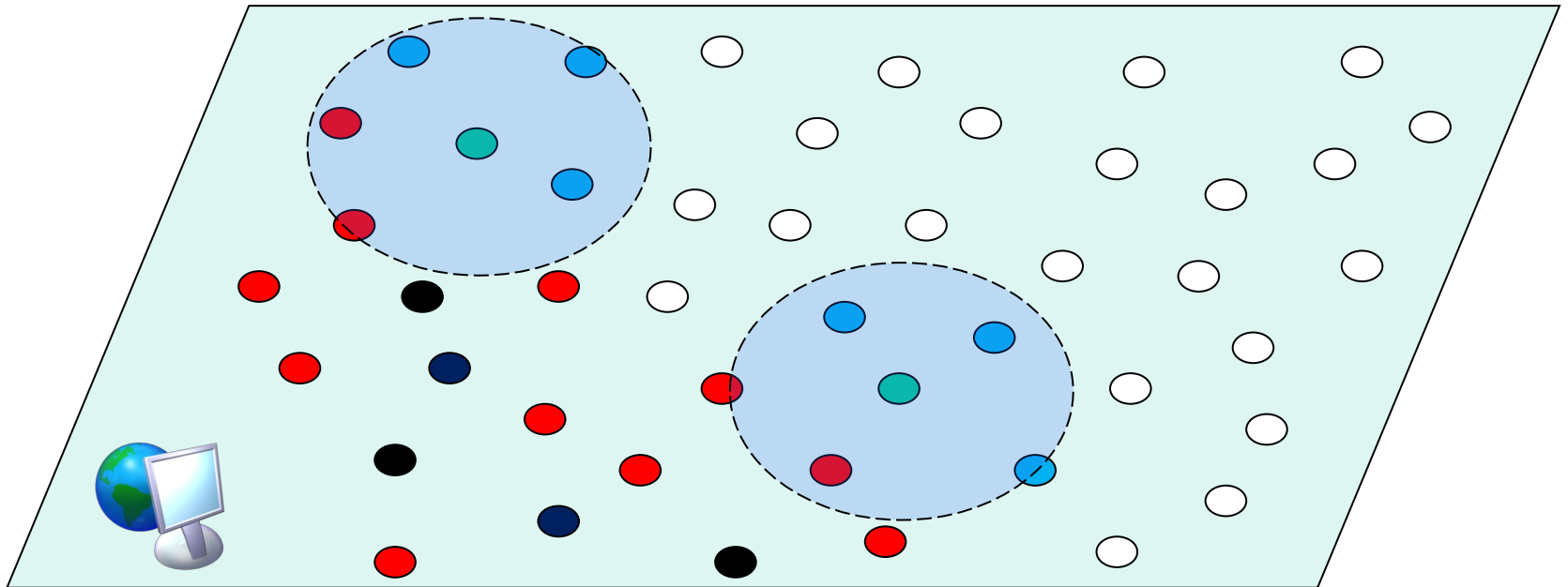
# Gossiping (Fofoca ou Flooding Probabilístico)

- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)



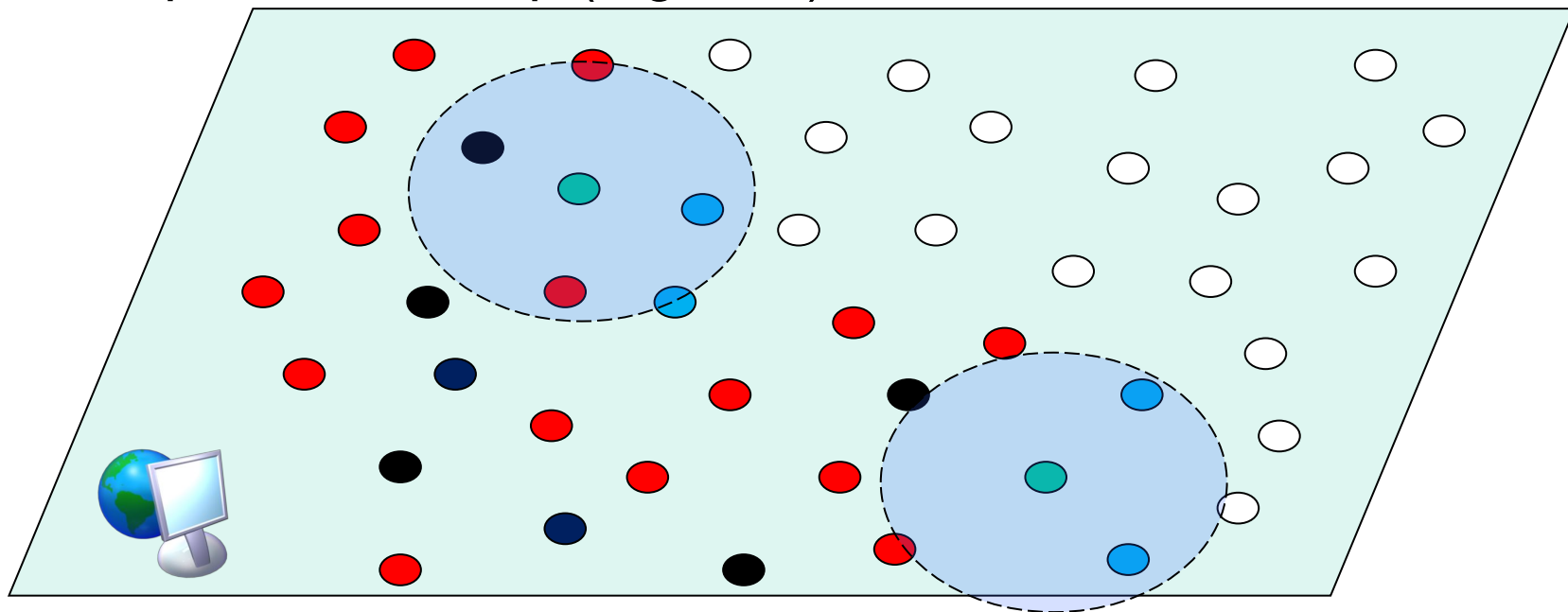
# Gossiping (Fofoca ou Flooding Probabilístico)

- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)



# Gossiping (Fofoca ou Flooding Probabilístico)

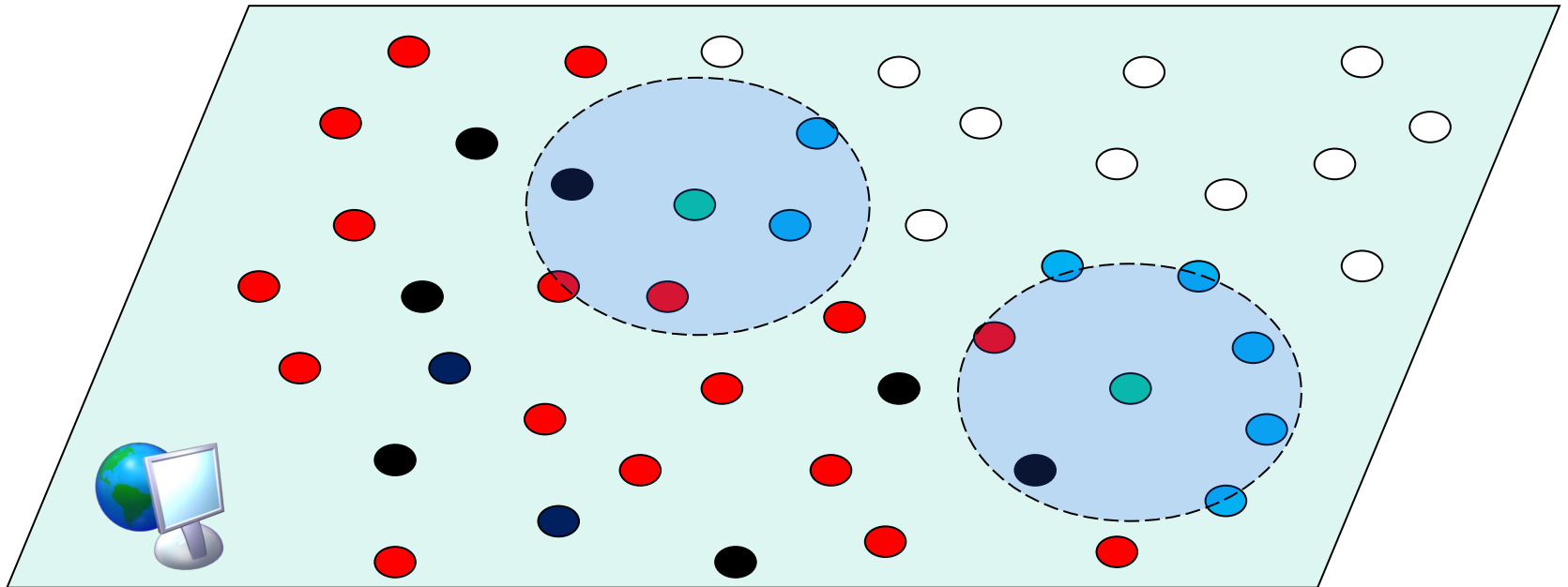
- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)





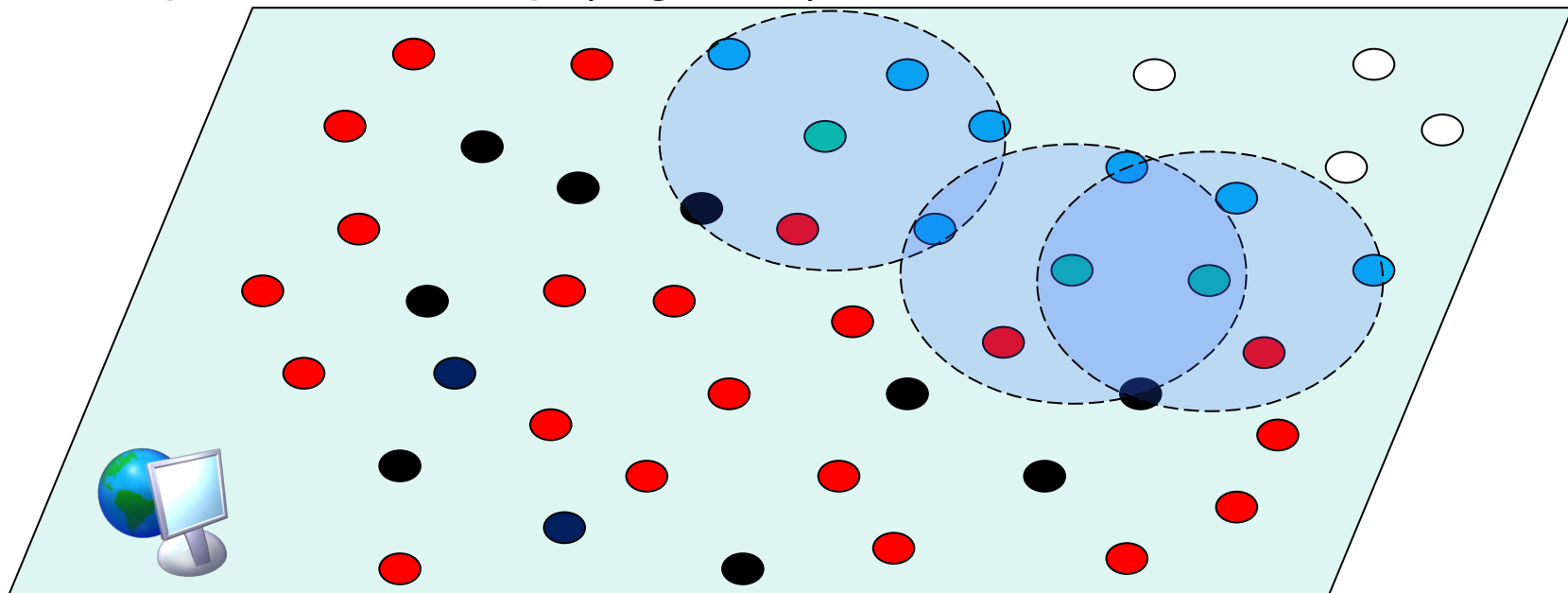
# Gossiping (Fofoca ou Flooding Probabilístico)

- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)



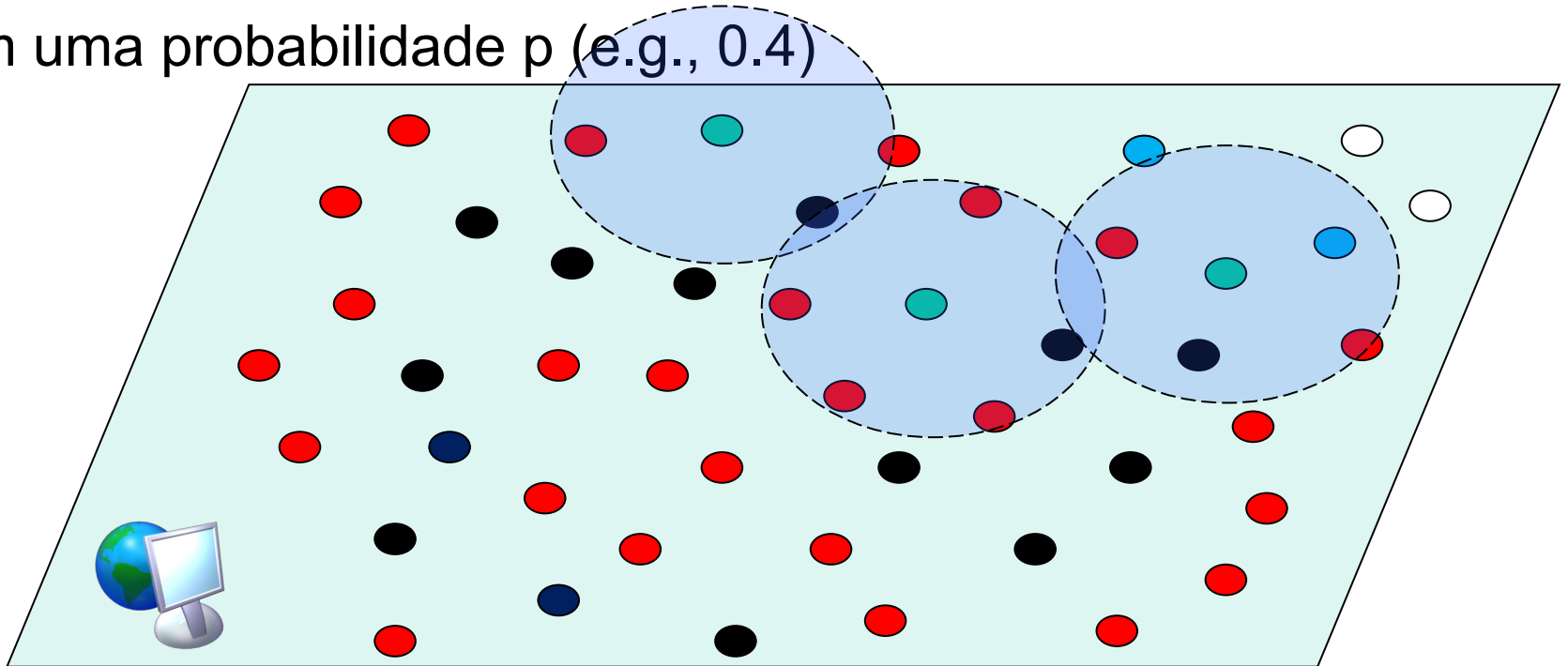
# Gossiping (Fofoca ou Flooding Probabilístico)

- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)



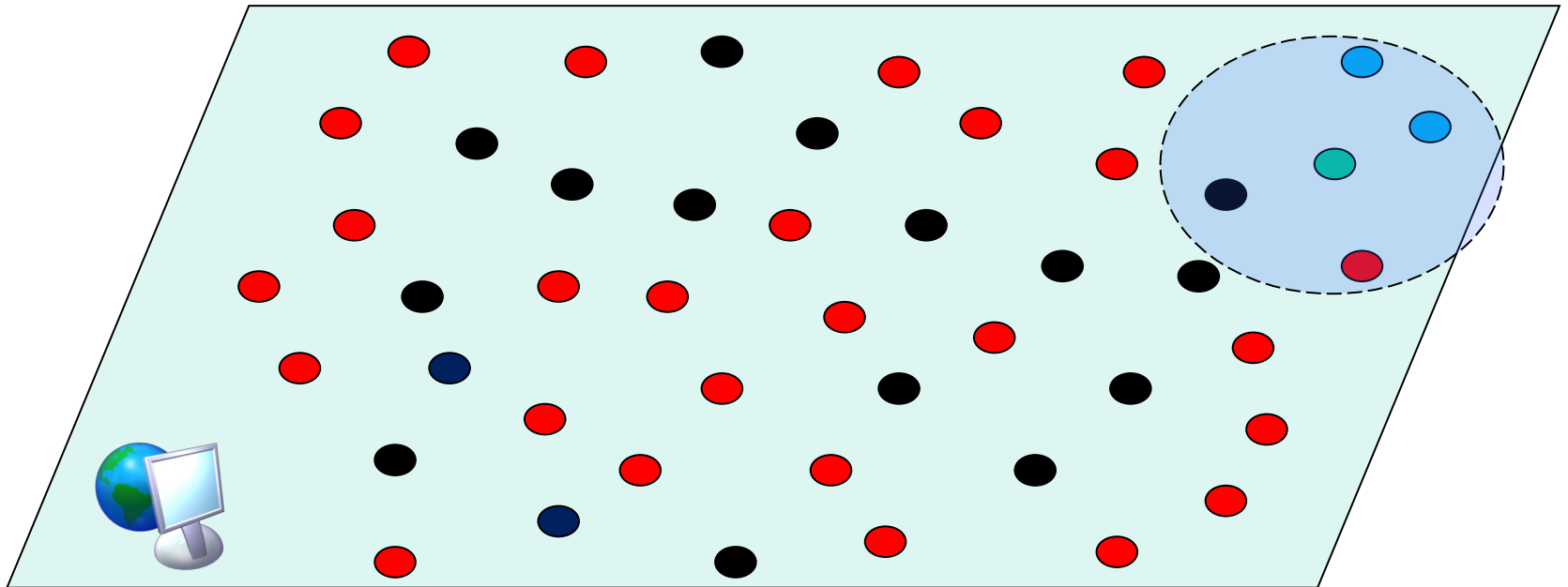
# Gossiping (Fofoca ou Flooding Probabilístico)

- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)



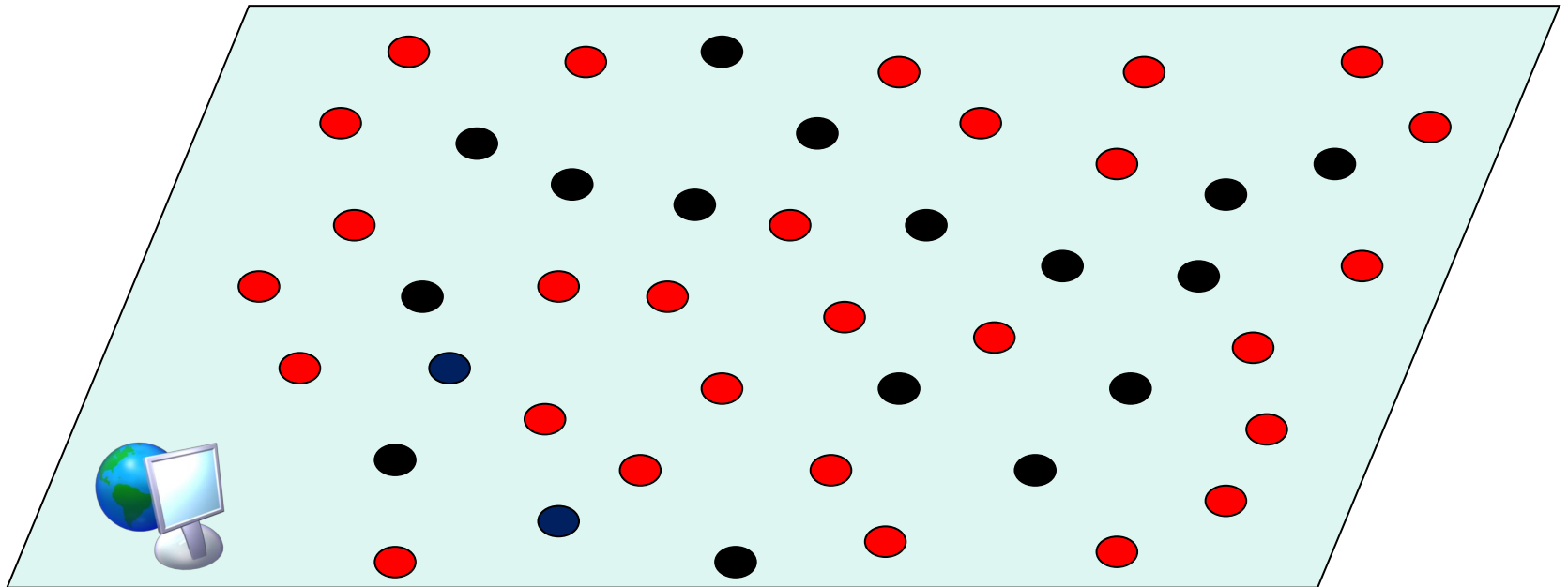
# Gossiping (Fofoca ou Flooding Probabilístico)

- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)



# Gossiping (Fofoca ou Flooding Probabilístico)

- Quando um nó recebe um pacote pela primeira vez, ele propaga com uma probabilidade  $p$  (e.g., 0.4)



- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- Algoritmo do caminho mais curto
- *Flooding*
- *Gossiping*
- **Roteamento por vetor de distância**
- Roteamento de estado de enlace
- Roteamento Hierárquico
- Roteamento em redes infraestruturadas
- Roteamento em redes *ad hoc*
- Roteamento em redes *mesh*

- Introdução
- **Algoritmos de Roteamento**
- Roteamento na Internet

# Roteamento por Vetor de Distância

- Conhecido também como roteamento distribuído de Bellman-Ford ou Algoritmo de Ford-Fulkerson
- Utilizado na ARPANET até 1979 (*Routing Information Protocol*, RIP)

# Roteamento por Vetor de Distância

- Cada roteador mantém uma tabela indexada por cada roteador da rede contendo:
  - A linha de saída preferencial a ser utilizada para esse destino
  - Estimativa do tempo ou da distância até o destino
- Cada roteador conhece a distância (e.g., número de saltos, comprimento da fila, atraso) até cada um dos seus vizinhos



# Roteamento por Vetor de Distância

- Periodicamente, cada roteador envia a cada vizinho uma lista de suas distâncias estimadas até cada destino
- Periodicamente, todos os roteadores atualizam suas tabelas utilizando as informações recebidas dos vizinhos
- A antiga tabela de roteamento não é utilizada no cálculo

# Exemplo de Atualização da Tabela do Nó J

## Listas recebidas

To	A	I	H	K
A	0	24	20	21
B	12	36	31	28
C	25	18	19	36
D	40	27	8	24
E	14	7	30	22
F	23	20	19	40
G	18	31	6	31
H	17	20	0	19
I	21	0	14	22
J	9	11	7	10
K	24	22	22	0
L	29	33	9	9
	JA delay is 8	JI delay is 10	JH delay is 12	JK delay is 6

Vetores recebidos de  
quatro vizinhos de J

Novo atraso estimado  
a partir de J

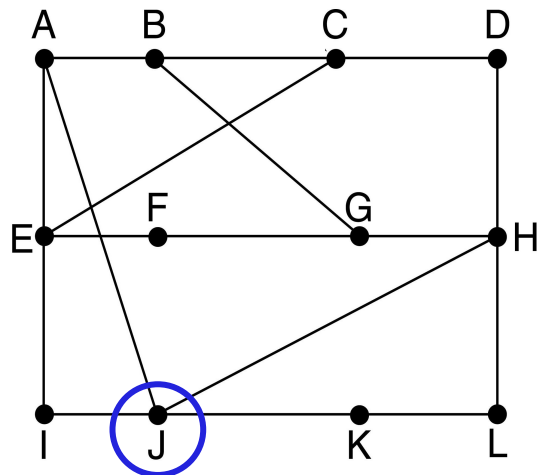
$\min(8+0, 10+24, 12+20, 6+21)$   
 $\min(8+12, 10+36, 12+31, 6+28)$   
 $\min(8+25, 10+18, 12+19, 6+36)$   
 $\min(8+40, 10+27, 12+8, 6+24)$

...

Line	
8	A
20	A
28	I
20	H
17	I
30	I
18	H
12	H
10	I
0	-
6	K
15	K

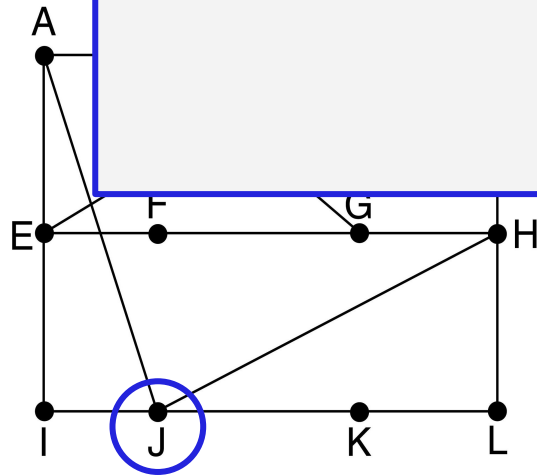
Nova tabela de J

Sub-rede



# Exemplo de Atualização da Tabela do Nó J

**Exercício:** Dada a tabela abaixo, justifique porque a distância de J para L é 15



H	17	20	0	19
I	21	0	14	22
J	9	11	7	10
K	24	22	22	0
L	29	33	9	9
	JA delay	JI delay	JH delay	JK delay
	is	is	is	is
	8	10	12	6

Vetores recebidos de  
quatro vizinhos de J

Nova tabela de J	
Line	
0	A
1	A
2	I
3	H
4	I
5	I
6	H
7	H
8	I
9	-
10	K
11	K

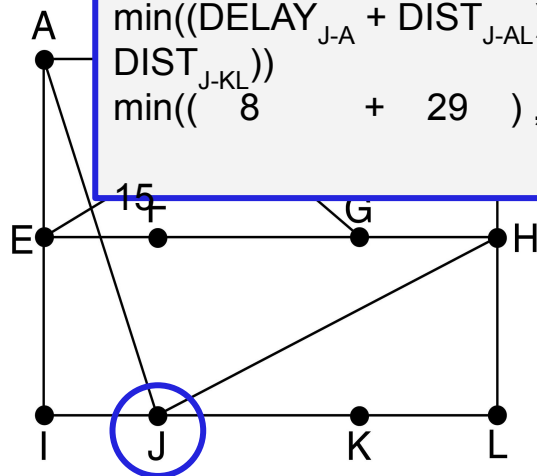
*Nova tabela de J*

# Exemplo de Atualização da Tabela do Nó J

**Exercício:** Dada a tabela abaixo, justifique porque a distância de J para L é 15

**Resposta:**

$\min((\text{DELAY}_{J-A} + \text{DIST}_{J-AL}), (\text{DELAY}_{J-I} + \text{DIST}_{J-IL}), (\text{DELAY}_{J-H} + \text{DIST}_{J-HL}), (\text{DELAY}_{J-K} + \text{DIST}_{J-KL}))$   
 $\min((8 + 29), (10 + 33), (12 + 9), (6 + 9))$



H	17	20	0	19
I	21	0	14	22
J	9	11	7	10
K	24	22	22	0
L	29	33	9	9
	JA delay is 8	JI delay is 10	JH delay is 12	JK delay is 6

Vetores recebidos de  
quatro vizinhos de J

Novo atraso estimado  
para chegar de J

	Line
	A
	A
	I
	H
	I
	I
18	H
12	H
10	I
0	-
6	K
15	K

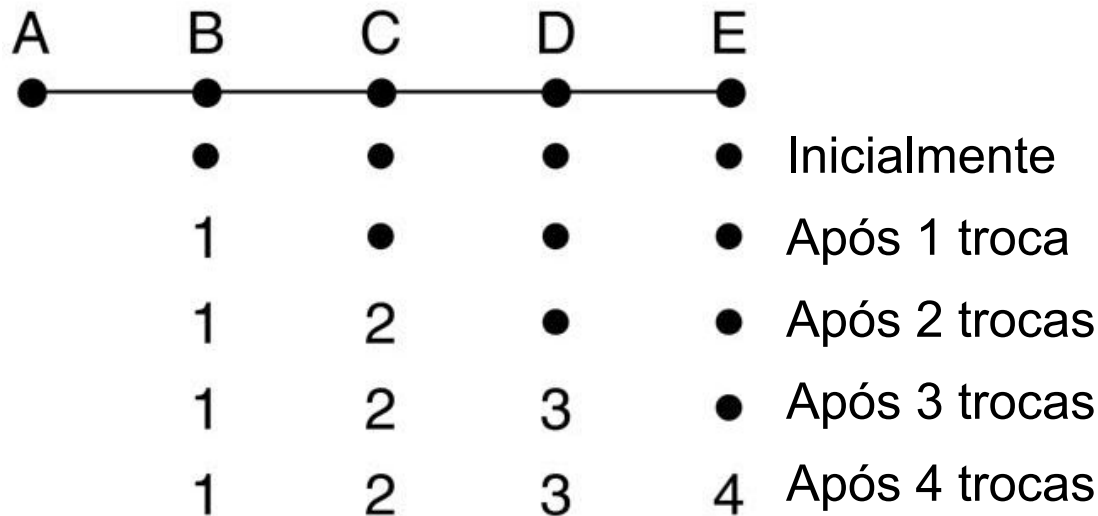
*Nova tabela de J*

# Roteamento por Vetor de Distância

- Reage com rapidez a boas notícias, mas devagar a más notícias
- Boas notícias:
  - Um roteador A possui uma rota grande até o destino X
  - Se um vizinho B relatar uma pequena distância até X, o roteador A passará a enviar o tráfego de X para B
  - Em uma troca de vetores, a boa notícia é sempre processada

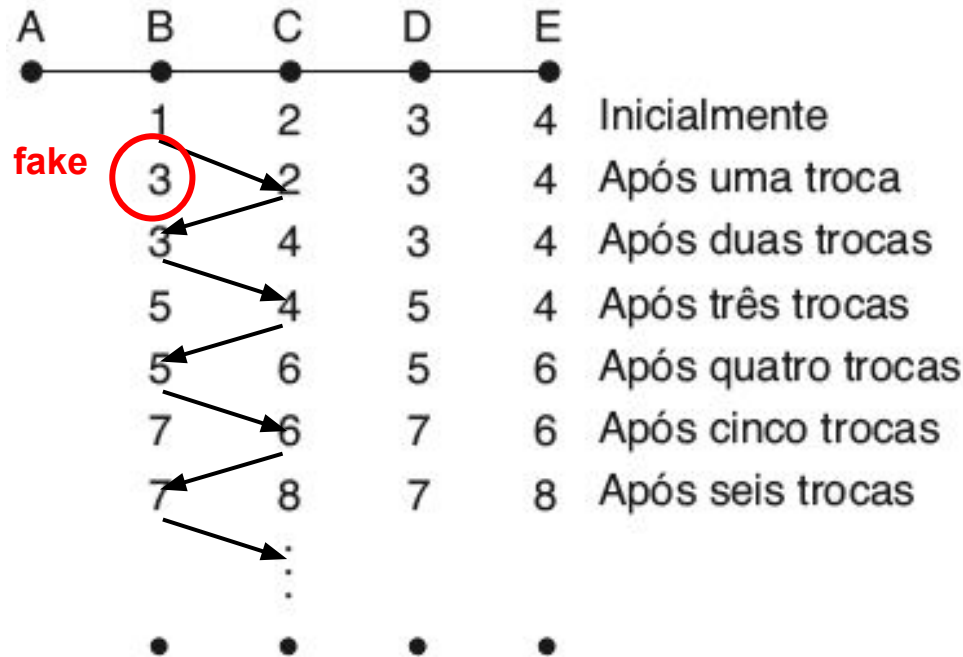
# Exemplo de Boas Notícias em um Grafo de Linha

- Métrica de retardo: número de hops
- Inicialmente, A está inativo e todos os roteadores sabem disso



# Exemplo de Más Notícias em um Grafo de Linha

- Inicialmente, A está ativo e a primeira linha mostra sua distância para os demais
- De repente, A é desativado e na primeira troca, B atualiza a distância de A por C



*O problema  
da contagem  
até o infinito*

# Problema da Contagem até o Infinito

- Gradualmente, todos os roteadores seguem seu caminho até infinito
- O número de trocas necessárias depende do valor numérico utilizado para infinito
- Quando X informa a Y que tem um caminho em algum lugar, Y não tem como saber se ele próprio está no caminho



- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- Algoritmo do caminho mais curto
- *Flooding*
- *Gossiping*
- Roteamento por vetor de distância
- **Roteamento de estado de enlace**
- Roteamento Hierárquico
- Roteamento em redes infraestruturadas
- Roteamento em redes *ad hoc*
- Roteamento em redes *mesh*

- Introdução
- **Algoritmos de Roteamento**
- Roteamento na Internet

# Roteamento de Estado de Enlace

- Motivação: Resolver o problema de contagem até infinito do Roteamento por Vetor de Distâncias
- Substituiu o Roteamento por Vetor de Distâncias na ARPANET a partir de 1979
- Utilizado pelo OSPF

# Algoritmo Atualizar Informações sobre Vizinhos

- 1) Descobrir meus vizinhos e aprender seus endereços de rede
- 2) Medir o custo (e.g., atraso) até cada um dos meus vizinhos
- 3) Criar um pacote que informe tudo o que acabei de aprender
- 4) Enviar esse pacote a todos os outros roteadores
- 5) Calcular o caminho mais curto até cada um dos outros roteadores (Dijkstra)

# Algoritmo Atualizar Informações sobre Vizinhos

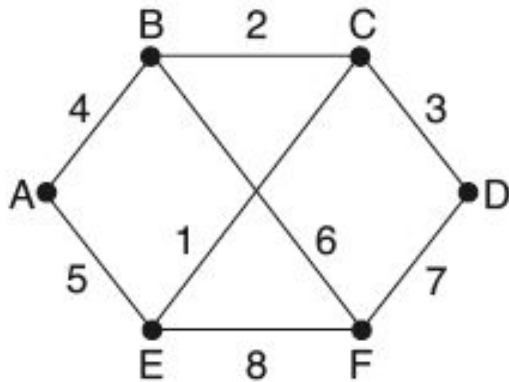
- 1) Descobrir meus vizinhos e aprender seus endereços de rede
- 2) Medir o custo (e.g., atraso) até cada um dos meus vizinhos

Para realizar essas duas etapas, os nós trocam pacotes de HELLO

O atraso pode ser calculado, por exemplo, de forma inversamente proporcional à largura de banda do enlace. Assim, a Ethernet de 1 Gbps tem custo 1 e a de 100 Mbps, custo 10

# Algoritmo Atualizar Informações sobre Vizinhos

3) Criar um pacote que informe tudo o que acabei de aprender



*Uma rede*

A	B	C	D	E	F
Seq.	Seq.	Seq.	Seq.	Seq.	Seq.
TTL	TTL	TTL	TTL	TTL	TTL
B   4	A   4	B   2	C   3	A   5	B   6
E   5	C   2	D   3	F   7	C   1	D   7
	F   6	E   1		F   8	E   8

*Pacotes de estado de enlace correspondentes a essa rede*

# Algoritmo Atualizar Informações sobre Vizinhos

4) Enviar esse pacote a todos os outros roteadores

Tarefa difícil e custosa em que cada nó efetua uma difusão

# Algoritmo Atualizar Informações sobre Vizinhos

5) Calcular o caminho mais curto até cada um dos outros roteadores (Dijkstra)

- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- Algoritmo do caminho mais curto
- *Flooding*
- *Gossiping*
- Roteamento por vetor de distância
- Roteamento de estado de enlace
- **Roteamento Hierárquico**
- Roteamento em redes infraestruturadas
- Roteamento em redes *ad hoc*
- Roteamento em redes *mesh*

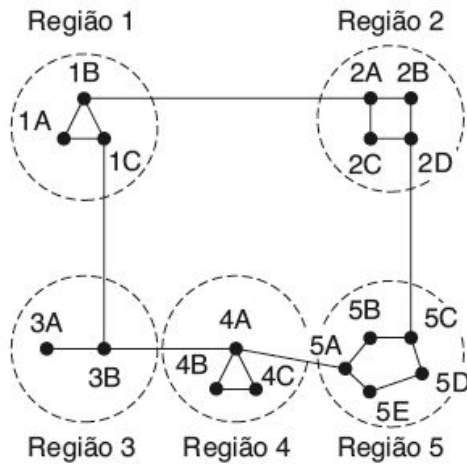
- Introdução
- **Algoritmos de Roteamento**
- Roteamento na Internet



# Roteamento Hierárquico

- Motivação: As tabelas de roteamento crescem proporcionalmente às redes, implicando em aumento de: memória para armazenar as tabelas; CPU para processá-las; e largura de banda para transmiti-las
- As redes são organizadas por regiões e o líder de uma região não conhece a estrutura interna de outras regiões

# Exemplo do Roteamento Hierárquico



*Uma rede*

*Tabela  
para 1A*

1A	–	–
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

*Tabela Hierárquica  
para 1A*

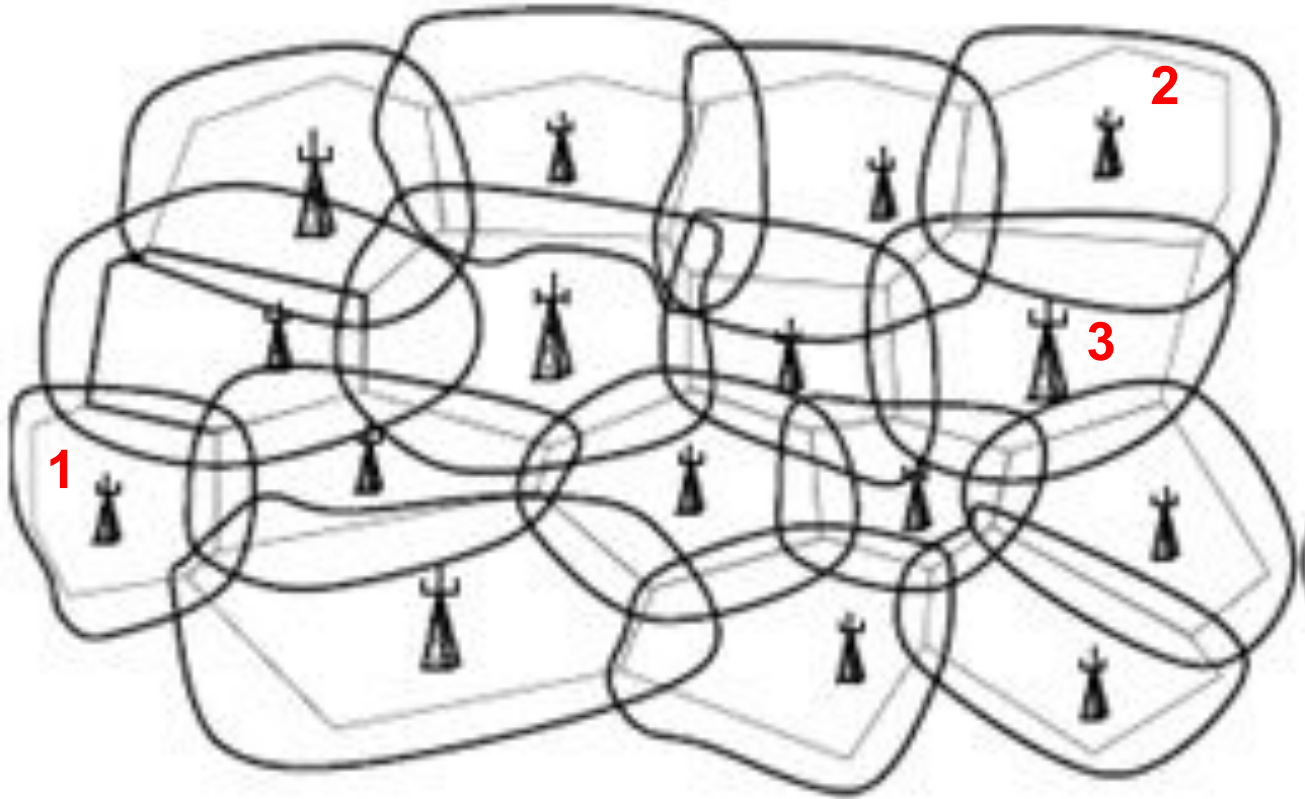
1A	–	–
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- Algoritmo do caminho mais curto
- *Flooding*
- *Gossiping*
- Roteamento por vetor de distância
- Roteamento de estado de enlace
- Roteamento Hierárquico
- **Roteamento em redes infraestruturadas**
- Roteamento em redes *ad hoc*
- Roteamento em redes *mesh*

- Introdução
- **Algoritmos de Roteamento**
- Roteamento na Internet

# Roteamento em Redes Infraestruturadas

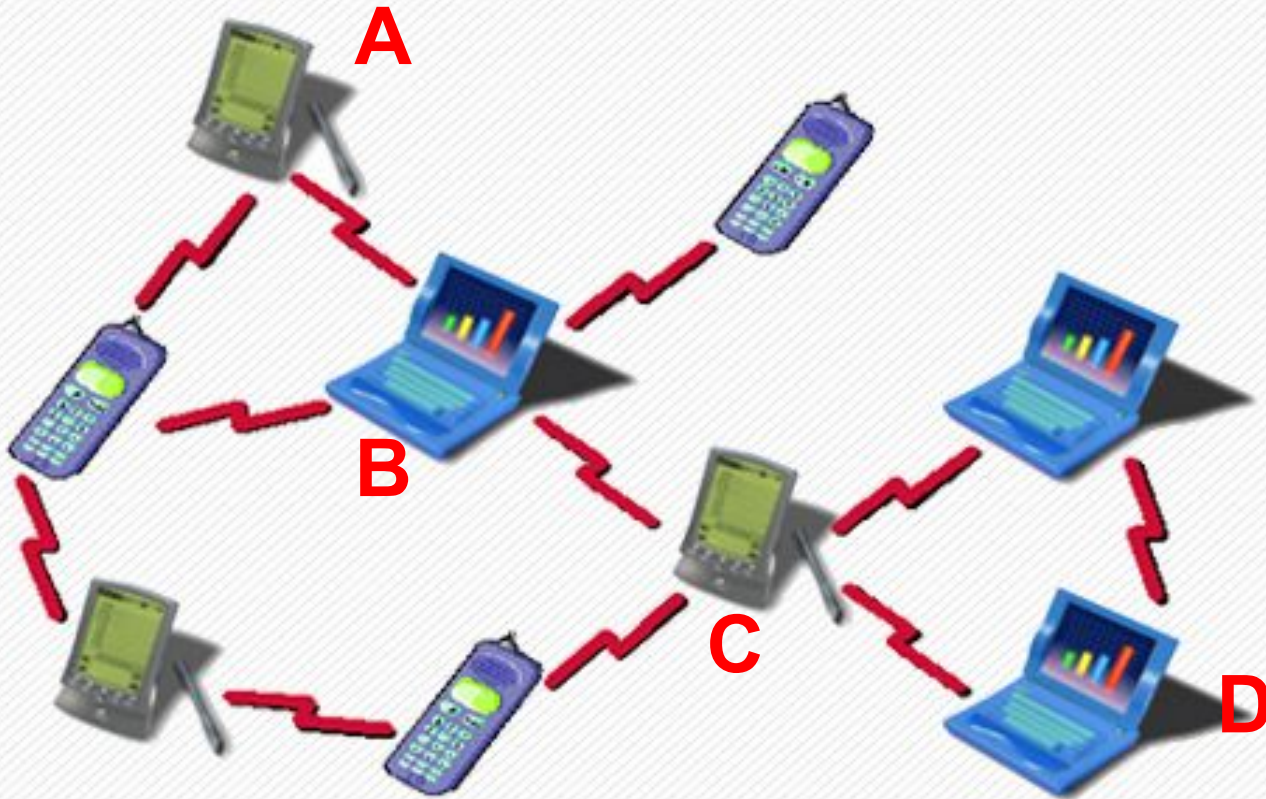


- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- Algoritmo do caminho mais curto
- *Flooding*
- *Gossiping*
- Roteamento por vetor de distância
- Roteamento de estado de enlace
- Roteamento Hierárquico
- Roteamento em redes infraestruturadas
- **Roteamento em redes *ad hoc***
- Roteamento em redes *mesh*

- Introdução
- **Algoritmos de Roteamento**
- Roteamento na Internet

# Roteamento em Redes *Ad Hoc*

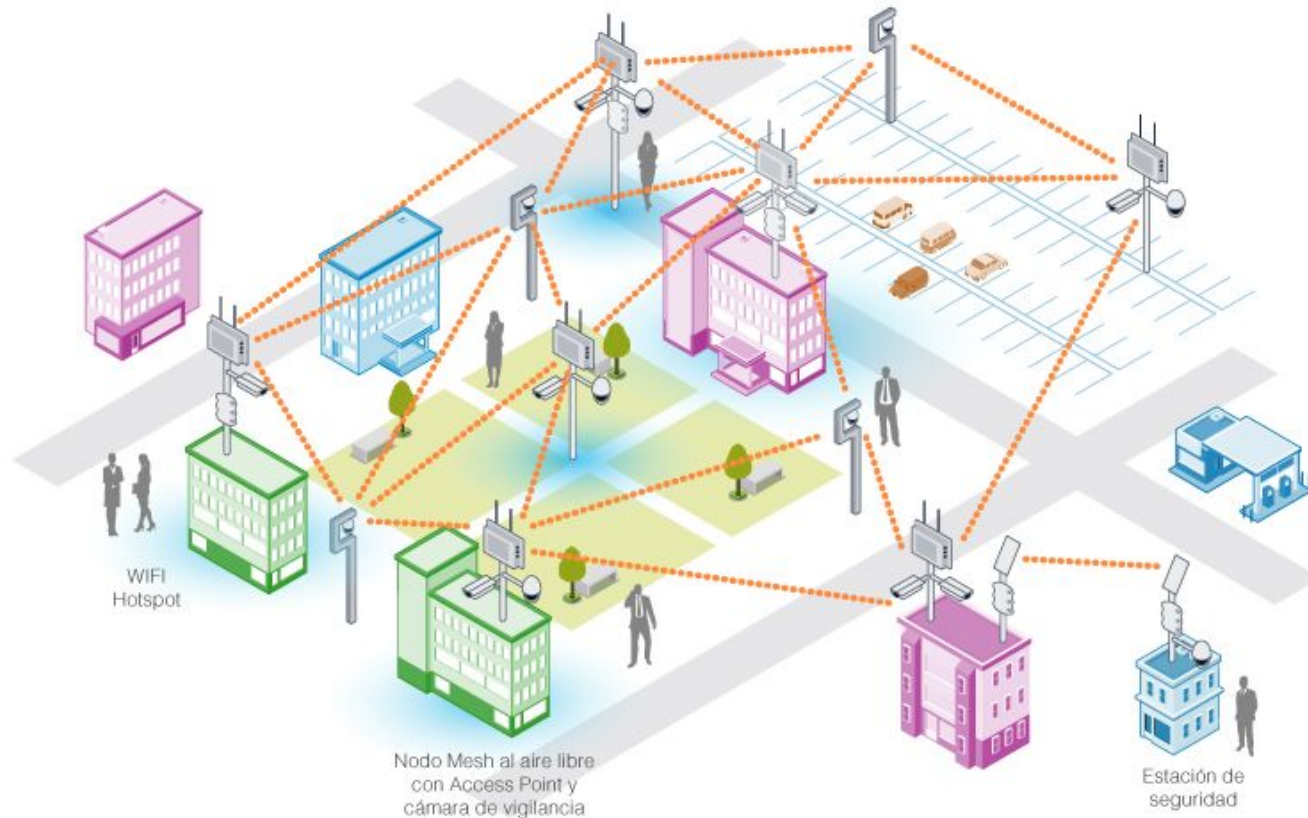


- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- Algoritmo do caminho mais curto
- *Flooding*
- *Gossiping*
- Roteamento por vetor de distância
- Roteamento de estado de enlace
- Roteamento Hierárquico
- Roteamento em redes infraestruturadas
- Roteamento em redes *ad hoc*
- **Roteamento em redes *mesh***

- Introdução
- **Algoritmos de Roteamento**
- Roteamento na Internet

# Roteamento em Redes *Mesh*





## Exercício (10)

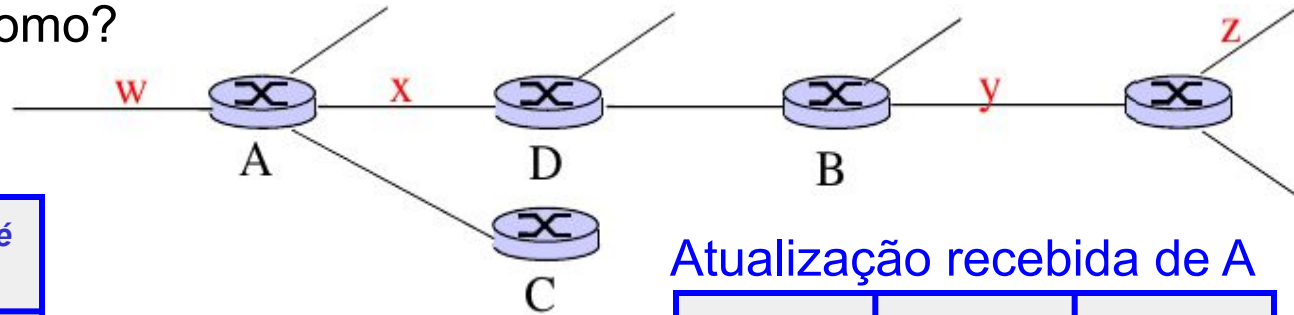
- Descreva o funcionamento básico dos algoritmos de *Flooding* e *Gossiping*

## Exercício (11)

- No SA, considere a tabela do nó D e ele recebeu uma atualização de A. A tabela em D mudará? Se sim, como?

Tabela de D

Rede de Destino	Próximo roteador	Saltos até destino
w	A	2
y	B	2
z	B	7
x	--	1
....	....	....



Atualização recebida de A

Rede de Destino	Próximo roteador	Saltos até destino
w	--	1
z	C	10
x	--	1
....	....	....

## Exercício (12)

- Descreva os roteamentos por vetor distância e de estado de enlace

## Exercício (14)

- Faça um paralelo entre o roteamento em redes infraestruturadas e *ad hoc*

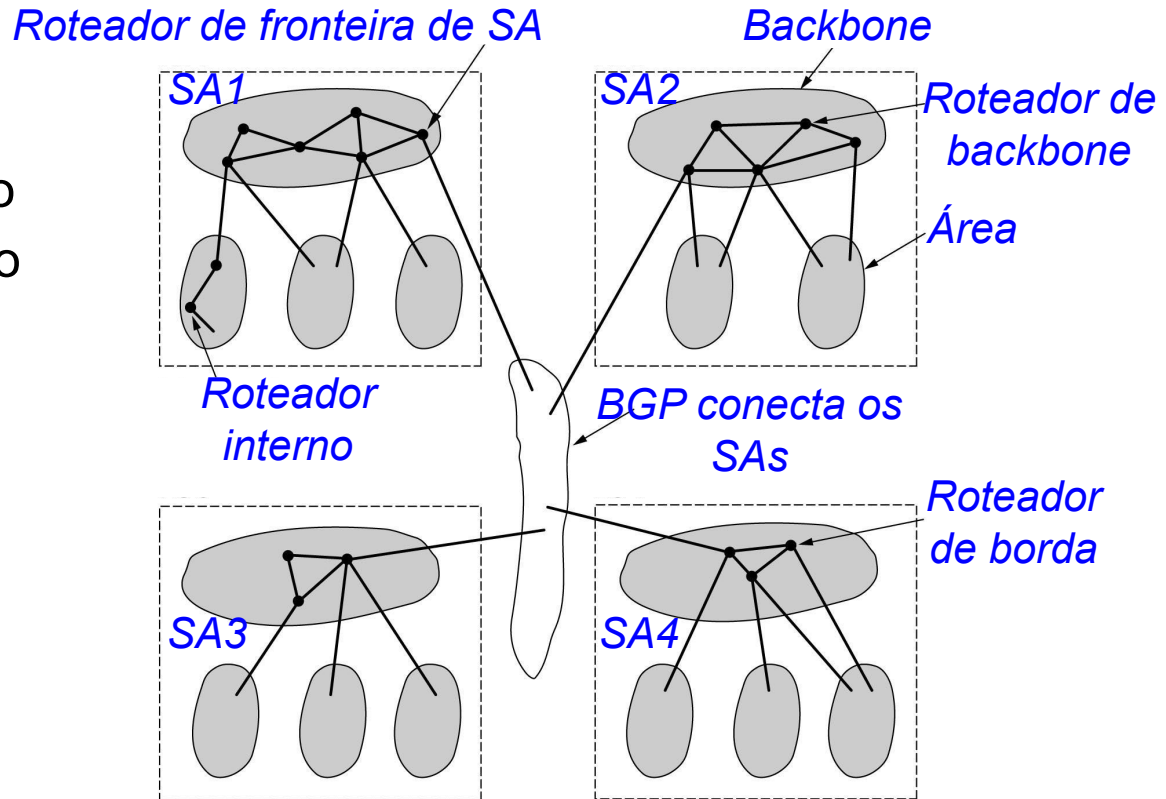
- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- **Introdução**
- *Open Shortest Path First* (OSPF)
- *Border Gateway Protocol* (BGP)

- Introdução
- Algoritmos de Roteamento
- **Roteamento na Internet**

# Internet, um Conjunto de Sistemas Autônomos (SAs)

- A Internet é formada por um grande número de SAs e cada um deles pode usar seu próprio algoritmo de roteamento interno



# Protocolos de Gateway Interior vs. Exterior

- **Protocolos de gateway interior**: usados para comunicação entre roteadores dentro de um mesmo SA, **roteamento intradomínio**. Existem algumas opções de protocolos, contudo, o OSPF é o mais popular
- **Protocolos de gateway exterior**: usados para comunicação entre roteadores de SAs diferentes, **roteamento interdomínio**. Nesse caso, todas as redes usam o mesmo protocolo que, atualmente, é o BGP

- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- Introdução
- ***Open Shortest Path First (OSPF)***
- *Border Gateway Protocol (BGP)*

- Introdução
- Algoritmos de Roteamento
- **Roteamento na Internet**



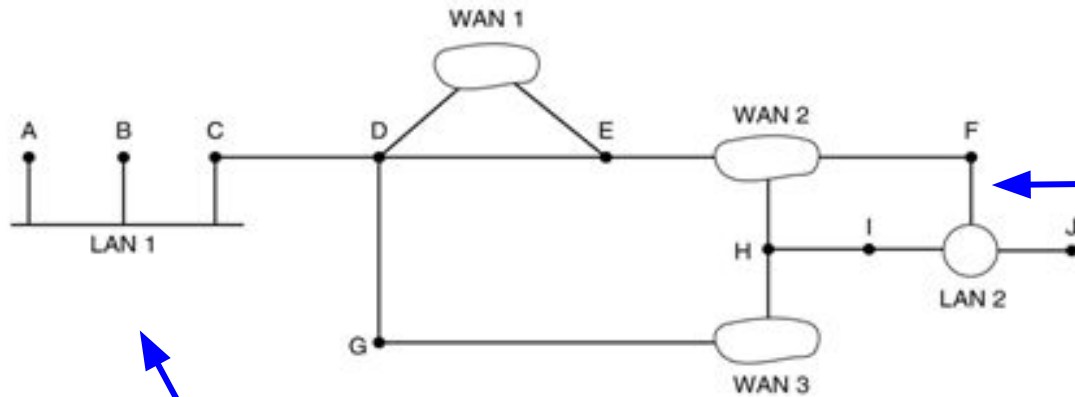
# *Interior Gateway Routing Protocol*

- O protocolo de *gateway* interior da Internet original era um protocolo de vetor de distância, o *Routing Information Protocol* (RIP)
- Os problemas do RIP eram a contagem até infinito e convergência lenta
- Em 1990, o protocolo de estado de enlace *Open Shortest Path First* (OSPF) se tornou o protocolo de *gateway* interior padrão

# Características do OSPF

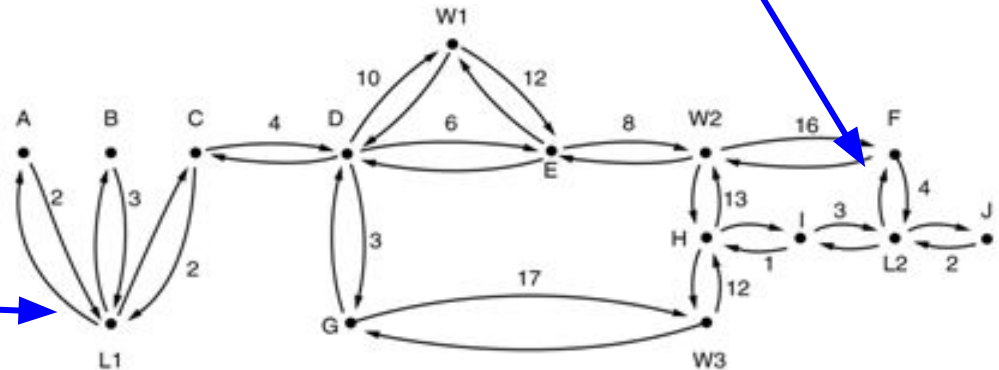
- Amplamente divulgado na literatura especializada (“*open*”)
- Admite várias unidades de medida de distância (retardo, distância física, etc...)
- Adapta-se de forma rápida e automática a alterações na topologia
- Admite roteamento baseado no tipo de serviço
- Faz balanceamento de carga entre caminhos múltiplos de igual custo
- Possui suporte para hierarquia dentro de um único domínio de roteamento
- Segurança: as trocas entre os roteadores podem ser autenticadas
- Possui suporte para tunelamento

# Representação da Rede com Grafo Orientado

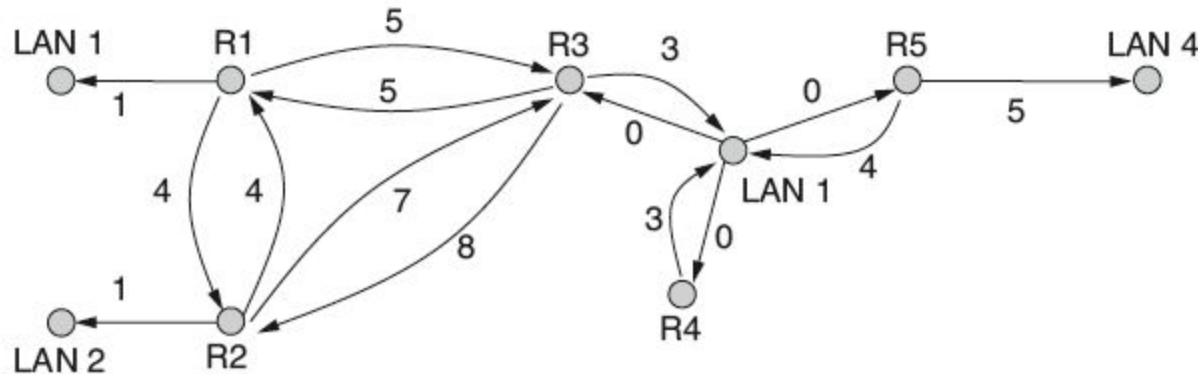
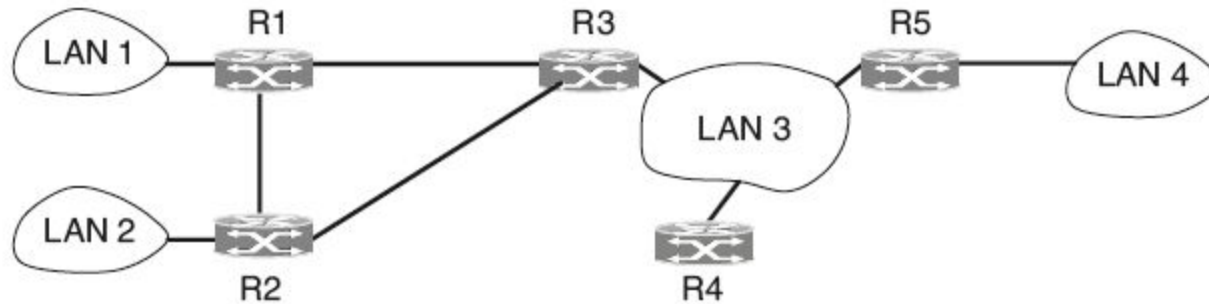


Conexão ponto-a-ponto vira par de arestas bidirecionais

Rede difusão vira vértice conectado com cada *host*



# Sistema Autônomo e seu Grafo Representativo

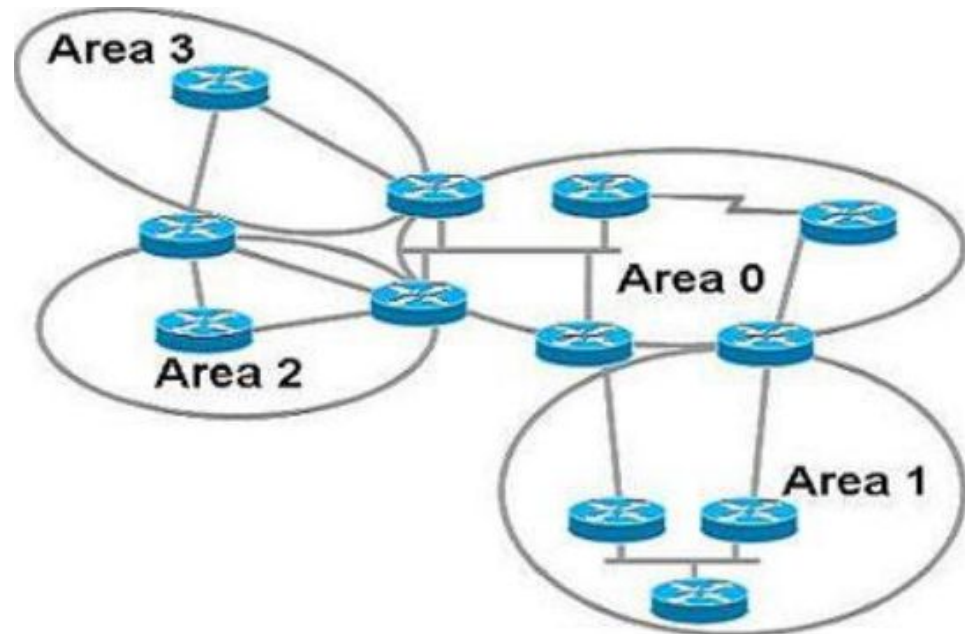


# Funcionamento Básico

- Aplica o algoritmo estado de enlace para atualizar as tabelas de vizinhos
  - Cada *host* envia suas informações para todos os outros do SA
  - Atualizações acontecem quando houver uma mudança no estado de enlace ou periodicamente (30 segundos)
- Aplica Dijkstra para encontrar o caminho mais curto
- ***Equal Cost MultiPath (ECMP)***: Quando dois roteadores tiverem múltiplos caminhos mais curtos, todos são lembrados, proporcionando um balanceamento de carga

# Hierarquização

- Grandes SAs podem ser divididos em áreas numeradas onde cada área representa uma rede ou mais redes
- Roteadores da mesma área executam o algoritmo de roteamento de estado de enlace



# Hierarquização

- Áreas não se sobrepõem nem precisam ser completas
- Eventualmente, um roteador não pertence a uma área
- Fora de uma área, a topologia e os detalhes da sub-rede não são visíveis
- Dentro de cada área, um ou mais roteadores de borda são responsáveis pelo roteamento de pacotes para fora da área
- Um roteador que se conecta a duas áreas armazena o banco de dados de ambas e executa o algoritmo de caminho mais curto para cada uma delas separadamente

- Responsável por rotear o tráfego entre as outras áreas do sistema autônomo
- Denominado área 0 do sistema autônomo
- Tem conexão com todas as áreas
- Sempre contém todos os roteadores de borda de área que estão dentro do SA



# Roteamento Interárea Dentro do SA

- Pacote é roteado até um roteador de borda de área (roteamento intra-área)
- Em seguida, o pacote é roteado por meio do *backbone* até o roteador de borda de área da área de destino
- Por fim, o pacote é roteado até seu destino final

# Cinco Tipos de Mensagens

Tipo de mensagem	Descrição
Hello	Usada para descobrir quem são os vizinhos
Link state update	Oferece os custos do transmissor aos seus vizinhos
Link state ack	Confirma a atualização do estado de enlace
Database description	Anuncia quais atualizações o transmissor tem
Link state request	Solicita informações do parceiro

- Introdução
- Questões de projeto da camada de rede
- A camada de rede da Internet
- *Internet Protocol* v4 (IPv4)
- Protocolos de controle da Internet
- **Roteamento**
- *Internet Protocol* v6 (IPv6)

- Introdução
- *Open Shortest Path First* (OSPF)
- ***Border Gateway Protocol* (BGP)**

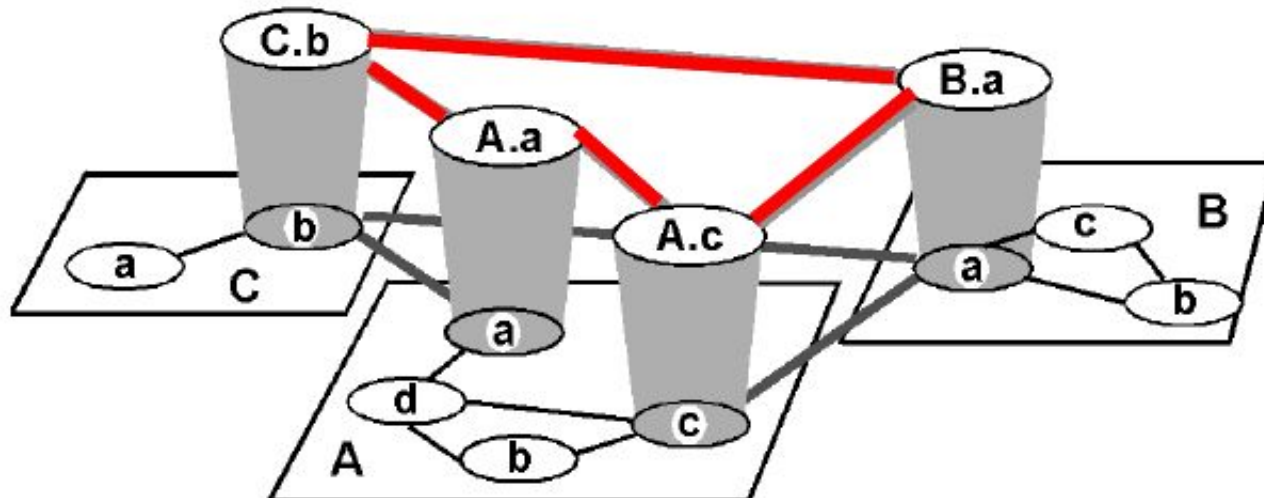
- Introdução
- Algoritmos de Roteamento
- **Roteamento na Internet**

# Objetivos de Protocolos de *Gateway* Interior vs. Exterior

- Protocolos de *gateway* interior precisam movimentar pacotes da origem até o destino da forma mais eficiente possível
- Protocolos de *gateway* exterior devem se preocupar com política, por exemplo:
  - Nenhum tráfego deve passar por certos SAs
  - Nunca colocar o Iraque em uma rota que comece no Pentágono
  - Não usar os Estados Unidos para ir da Colúmbia Britânica até Ontário
  - Só passar pela Albânia se não houver outra alternativa para chegar ao destino
  - O tráfego que começar ou terminar na IBM não deve transitar pela Microsoft

# Border Gateway Protocol (BGP)

- A Internet consiste de SAs e nas linhas que os conectam
- Dois SAs são considerados conectados se existe uma linha entre roteadores de borda de cada um deles

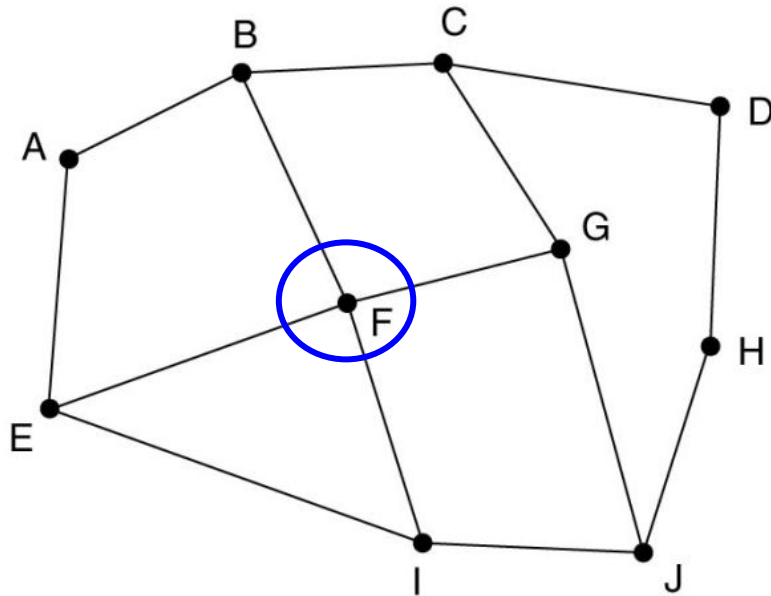


# Funcionamento Básico

- Pares de roteadores BGP se comunicam entre si estabelecendo conexões TCP
- Utiliza o Roteamento por Vetor de Distâncias
  - Ao invés de manter apenas o custo para cada destino, cada roteador BGP tem controle de qual caminho está sendo usado
  - Em vez de fornecer periodicamente a cada vizinho apenas o custo estimado para cada destino possível, o roteador BGP informa a seus vizinhos o caminho exato que está usando

# Exemplo da Tabela de Roteamento de F

- F descarta caminhos que passam por ele ou que violam alguma restrição política. Após os descartes, o roteador F adota as rotas mais curtas



*Informações sobre D que F  
recebe de seus vizinhos:*

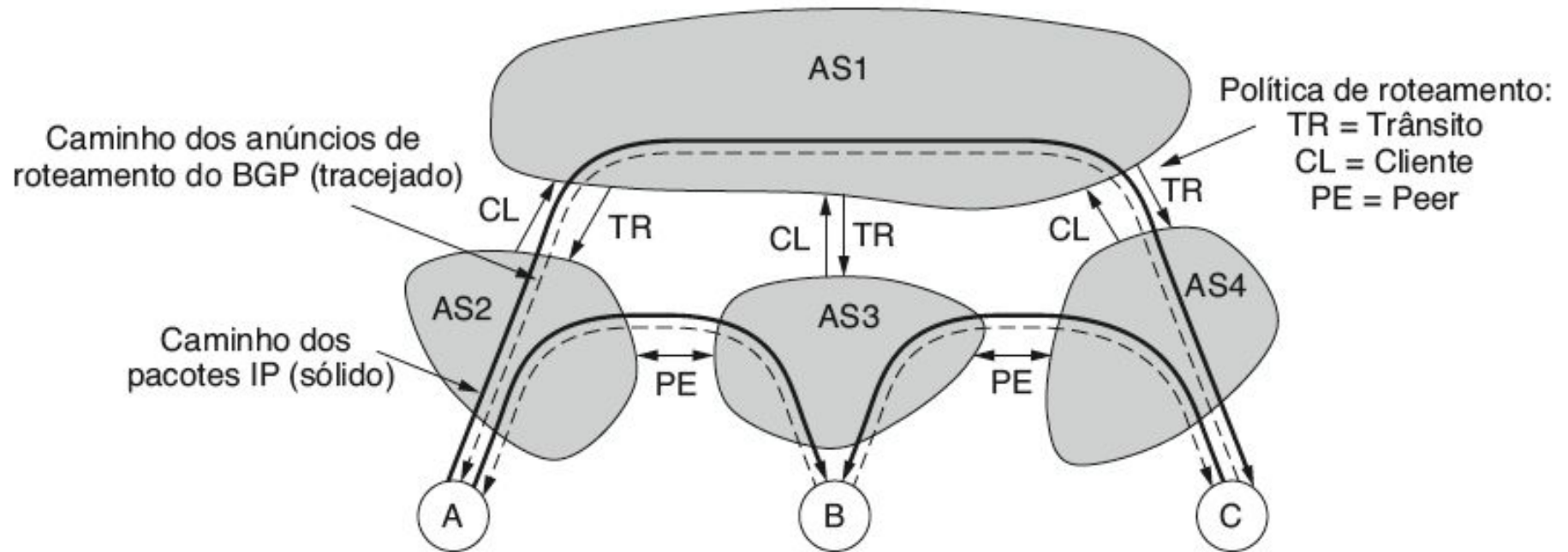
*De B: "Eu utilizo BCD"*

*De G: "Eu utilizo GCD"*

*De I: "Eu utilizo IFGCD" (descartado)*

*De E: "Eu utilizo EFGCD" (descartado)*

# Trânsito e *Peering*





# Classificação das Redes

- **Redes stub**: possuem apenas uma conexão com o grafo BGP, logo, não podem ser usadas para tráfego porque não há “outro” lado
- **Redes multiconectadas**: podem ser usadas para tráfego, a menos que se recusem
- **Redes de trânsito**: tem como objetivo tratar pacotes de terceiros

## Exercício (15)

- Como o BGP resolve o problema da contagem até infinito que atinge outros algoritmos de Roteamento por Vetor de Distâncias?