Escola de Engenharia Mauá

ECM511 – Pesquisa Operacional e ~Métodos de Otimização

Prof. Joyce M Zampirolli
joyce.zampirolli@maua.br

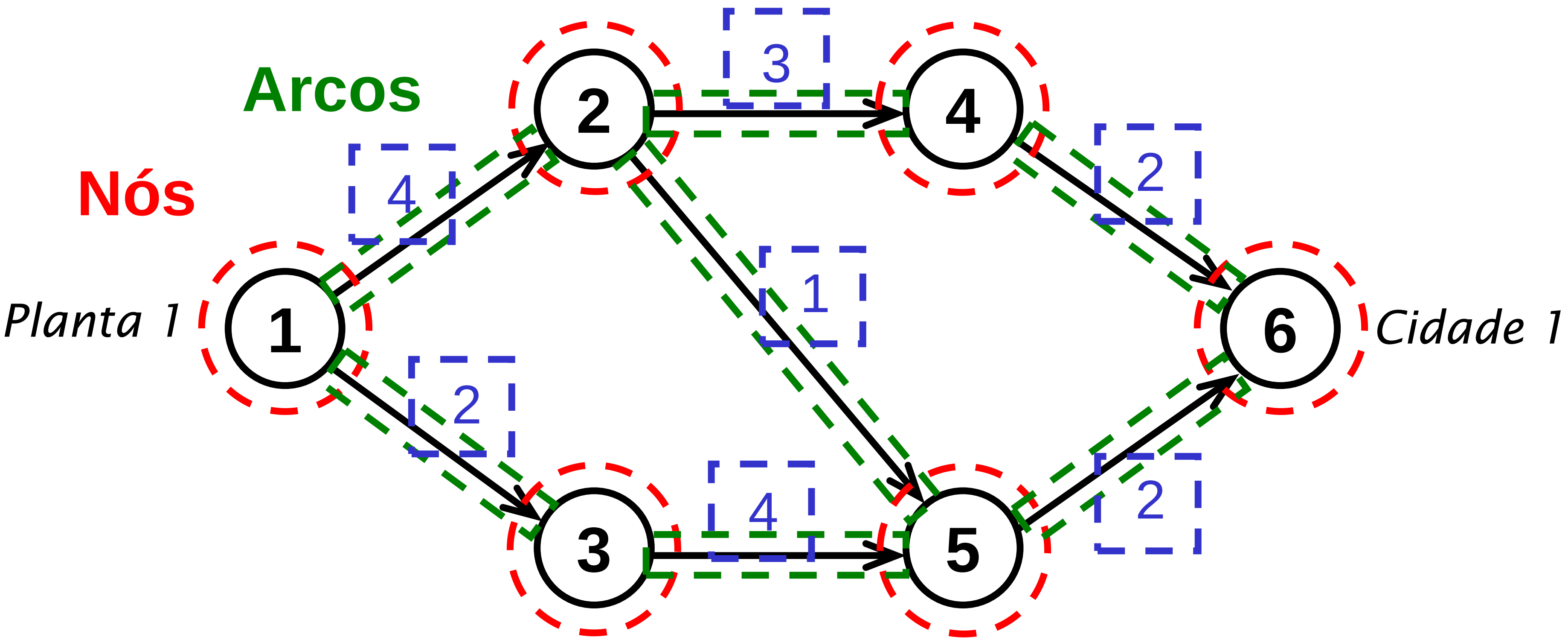
Otimização em Redes: Problema do Caminho Mais Curto e do Fluxo Máximo

Maio/2019

Caminho mais curto

Exemplo: Junior Wells

Como discutido no estudo do PCV, um **grafo** é formado por um conjunto de **nós** e **arcos**. Se existirem **pesos** associados aos arcos, o grafo se torna uma **rede**.



Caminho mais curto

Exemplo: Junior Wells

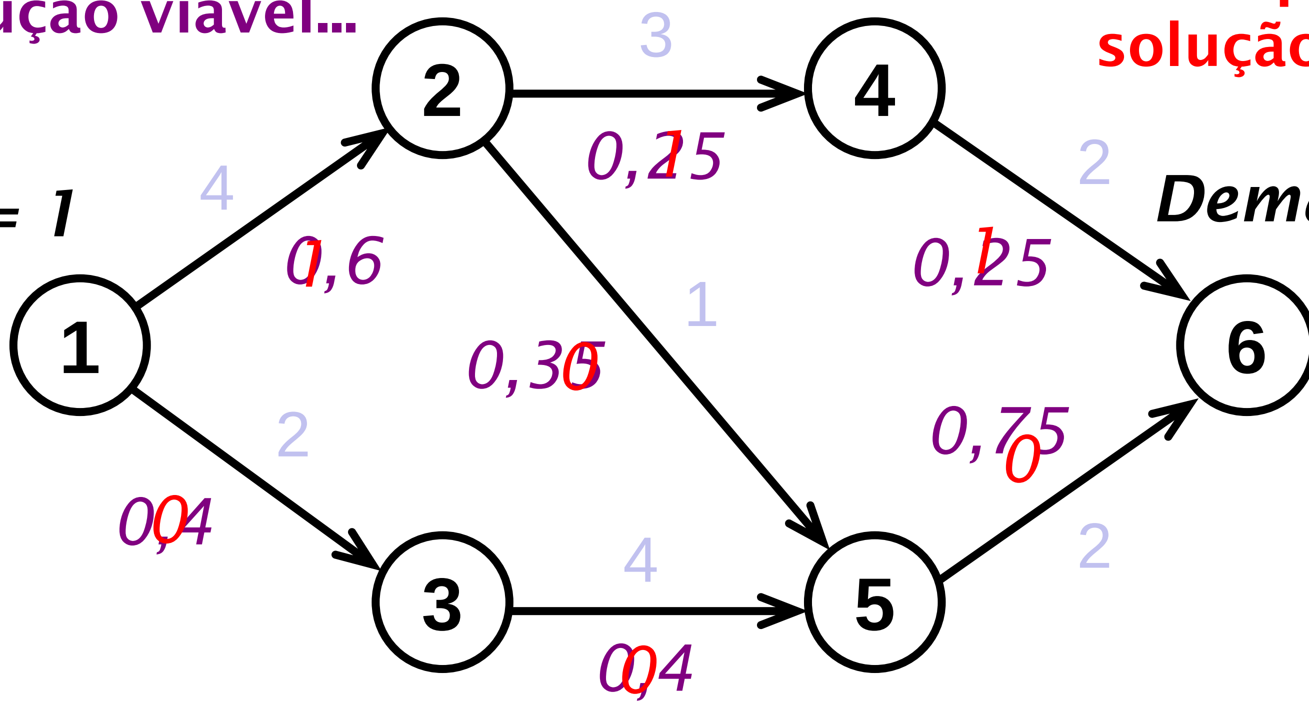
O problema do transporte de **uma unidade** de produto entre os nós **1** e **6** pode ser usado para determinar o **caminho mais curto** entre estes nós (por quê?)

Uma solução viável...

Uma "possível" solução ótima.

Oferta = 1

Planta 1

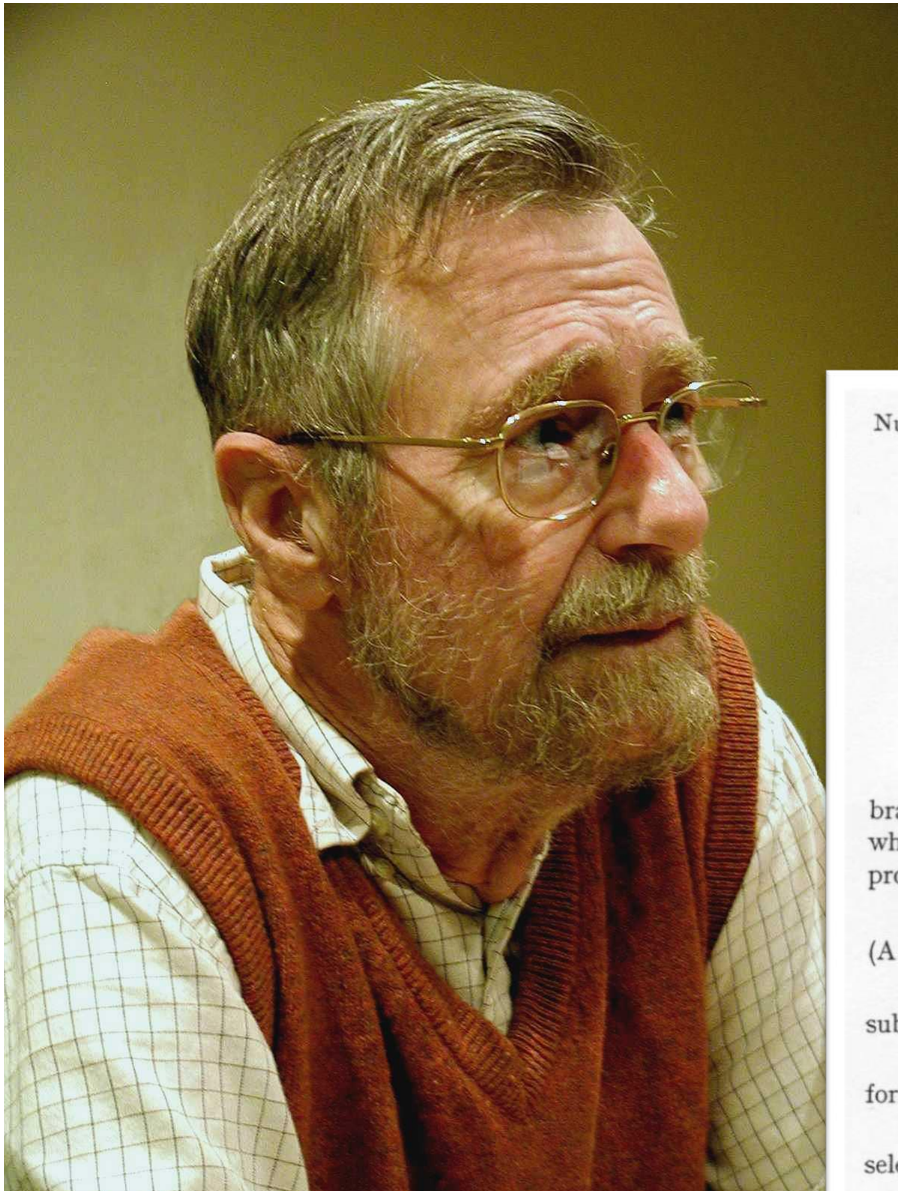


Demanda = 1

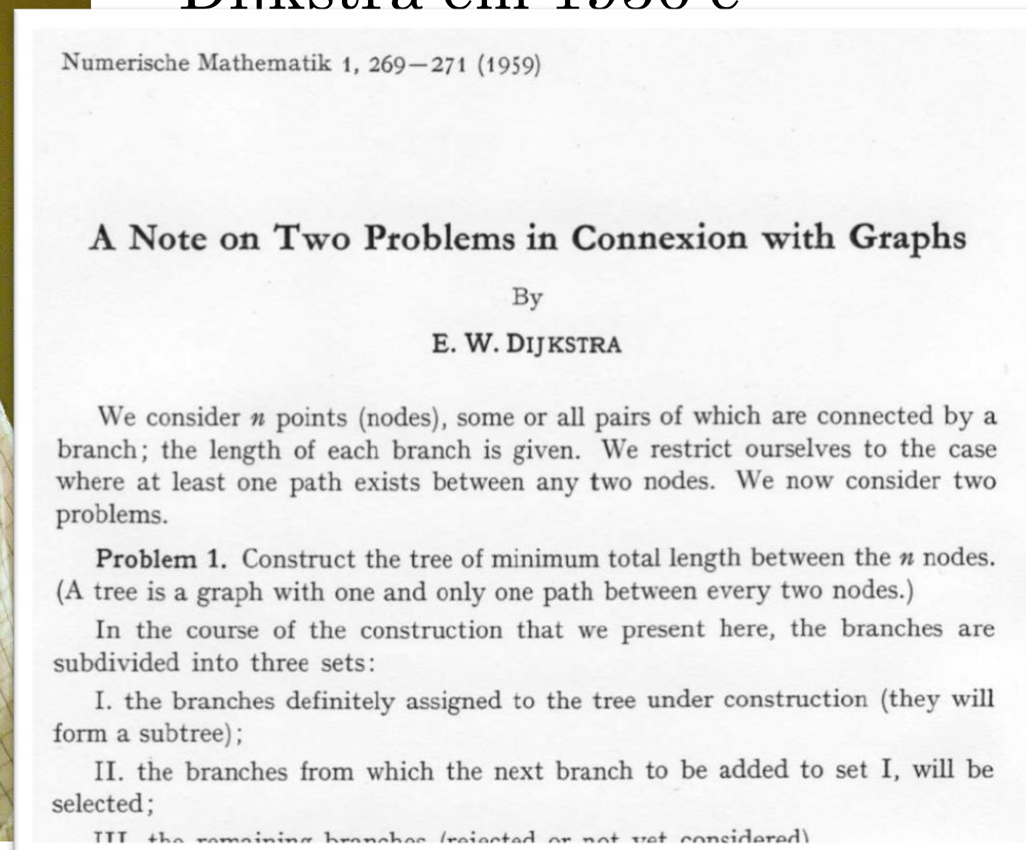
Cidade 1

Distância = 4×1 + 3×1 + 2×1

Algoritmo de Dijkstra



Wikipédia: o algoritmo de Dijkstra, concebido pelo cientista da computação holandês Edsger Wybe Dijkstra em 1956 e



Numerische Mathematik 1, 269—271 (1959)

A Note on Two Problems in Connexion with Graphs

By

E. W. DIJKSTRA

We consider n points (nodes), some or all pairs of which are connected by a branch; the length of each branch is given. We restrict ourselves to the case where at least one path exists between any two nodes. We now consider two problems.

Problem 1. Construct the tree of minimum total length between the n nodes. (A tree is a graph with one and only one path between every two nodes.)

In the course of the construction that we present here, the branches are subdivided into three sets:

I. the branches definitely assigned to the tree under construction (they will form a subtree);

II. the branches from which the next branch to be added to set I, will be selected;

III. the remaining branches (rejected or not yet considered).

Algoritmo de Dijkstra

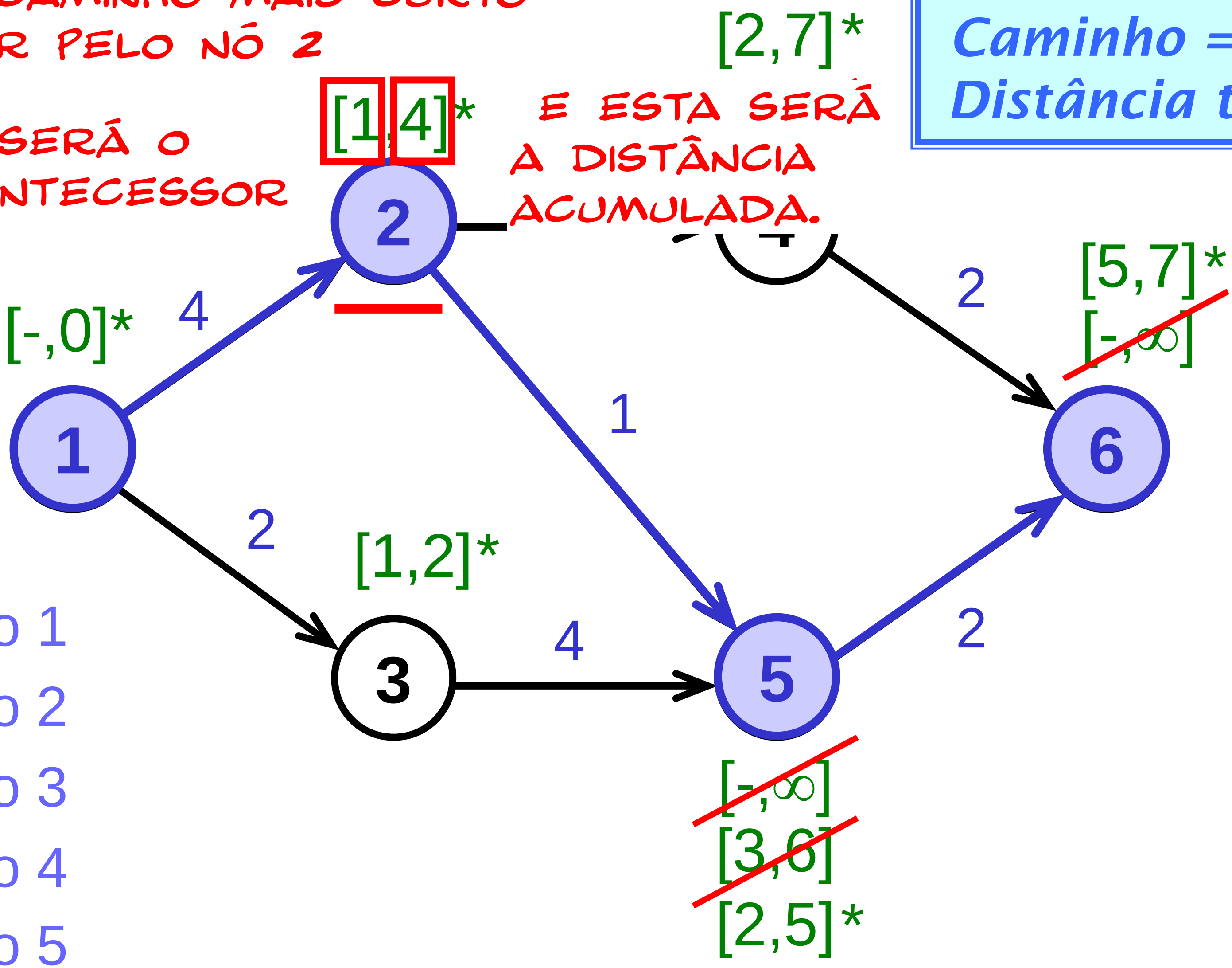
Caminho mais curto

SE O CAMINHO MAIS CURTO PASSAR PELO NÓ 2

ESTE SERÁ O SEU ANTECESSOR

E ESTA SERÁ A DISTÂNCIA ACUMULADA.

Caminho = 1-2-5-6
Distância total = 7



Passo 1

Passo 2

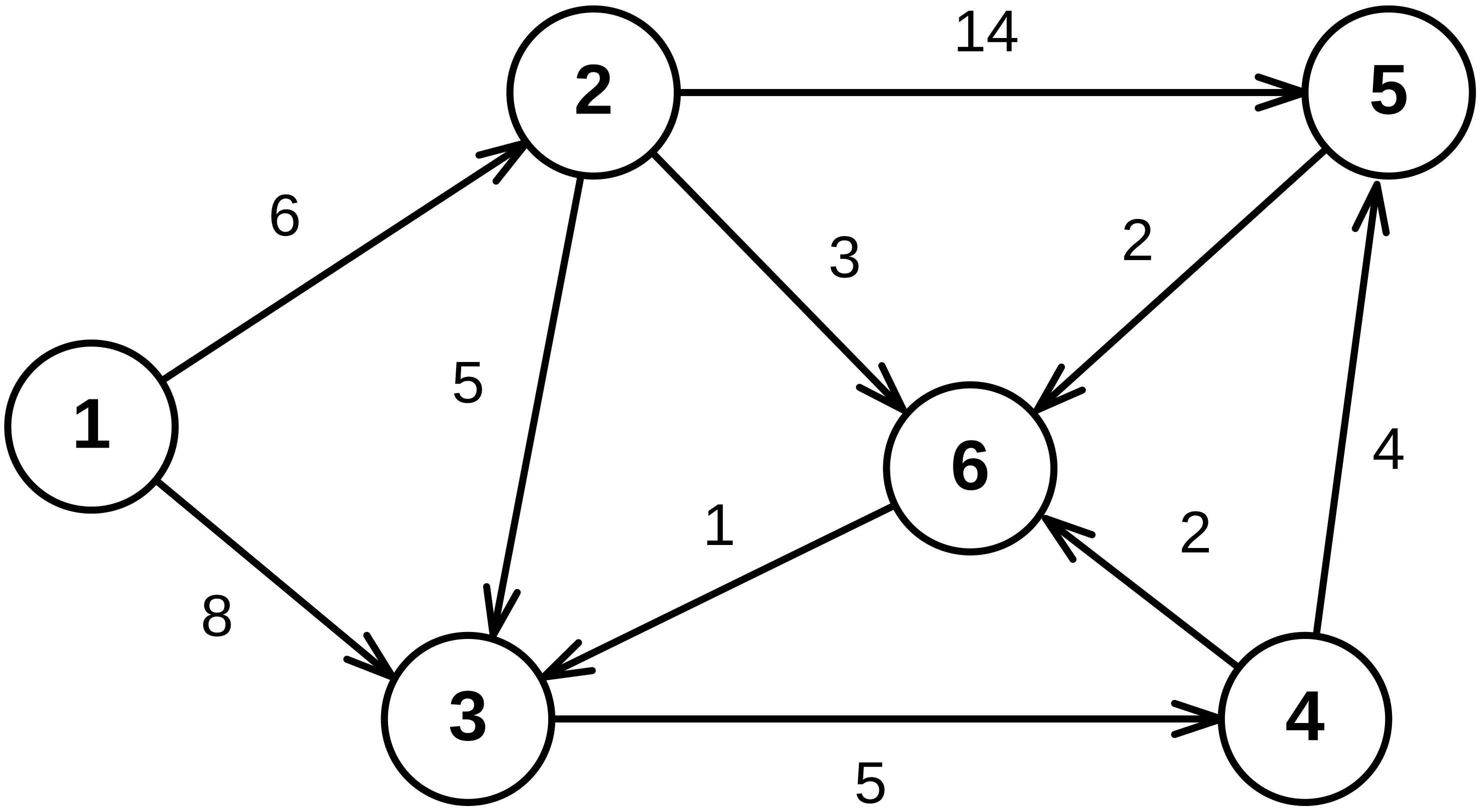
Passo 3

Passo 4

Passo 5

Dijkstra - exercício proposto

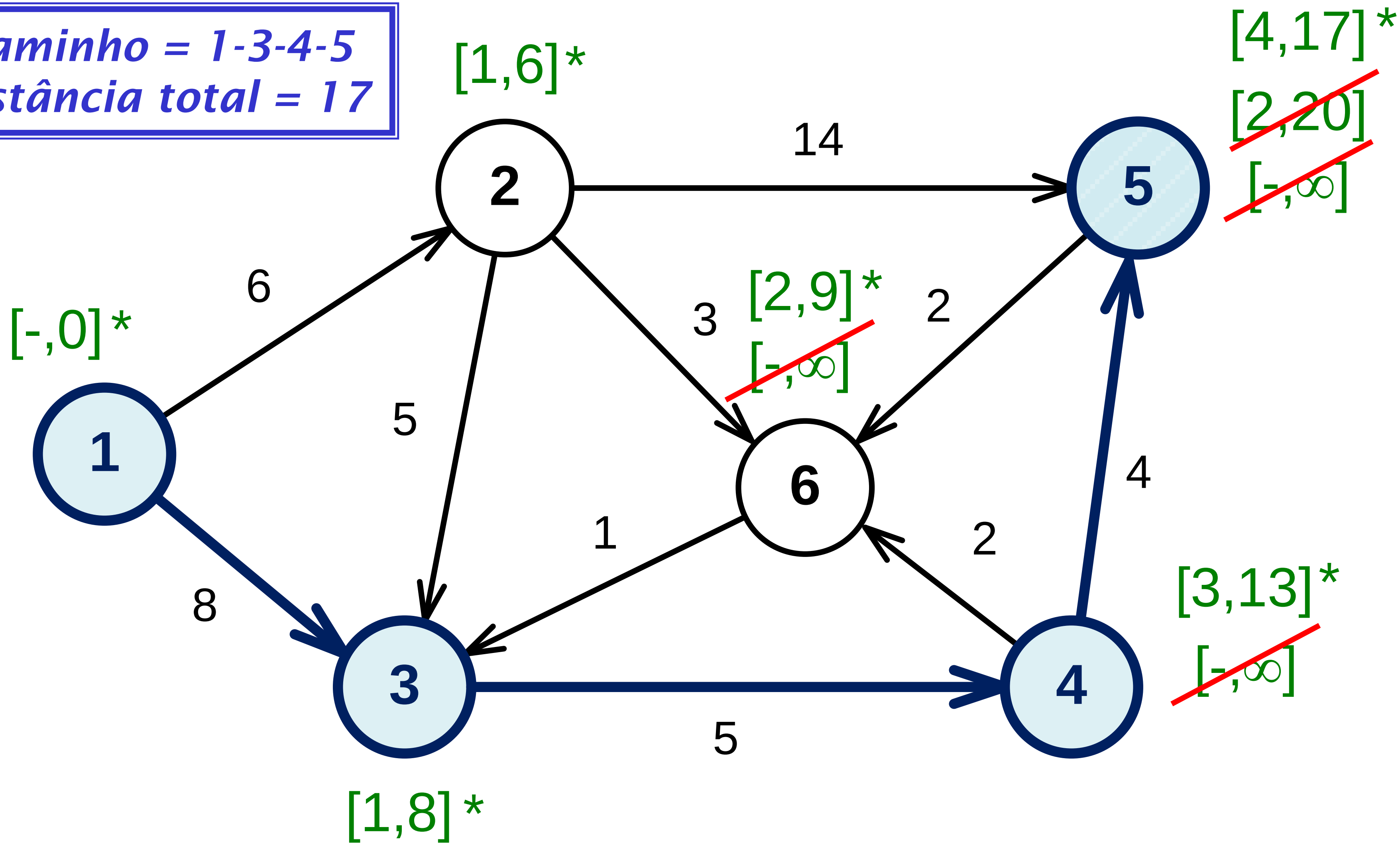
caminho mais curto entre os nós 1 e 5



Dijkstra - exercício proposto

solução

Caminho = 1-3-4-5
Distância total = 17



Algoritmo de Ford-Fulkerson

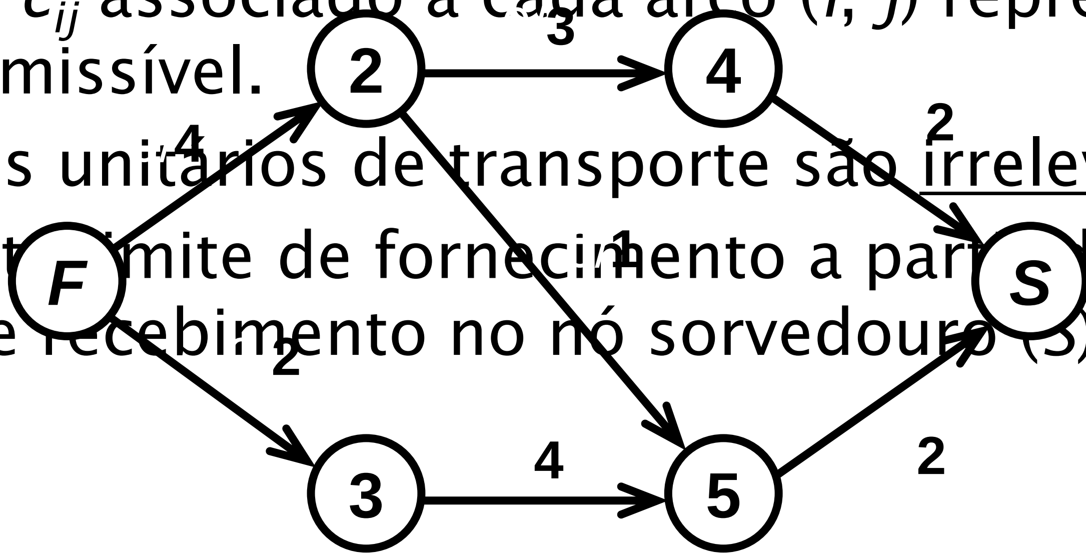
Problema do fluxo máximo

1) Objetivo: determinar o fluxo máximo entre *F* e *S*.

2) O "peso" c_{ij} associado a cada arco (i, j) representa o fluxo máximo admissível.

3) Os custos unitários de transporte são <u>irrelevantes</u>.

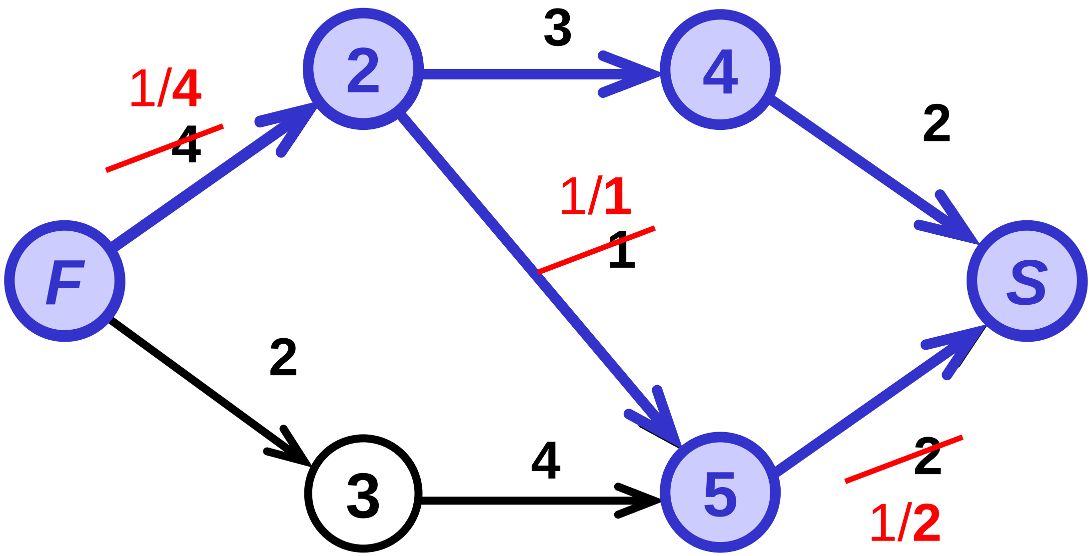
4) Não existe limite de fornecimento a partir do nó fonte (*F*) ou limite de recebimento no nó sorvedouro (*S*).



Algoritmo de Ford-Fulkerson

Rotina A: Identificação de um caminho de aumento de fluxo.

Rotina B: Atualização dos fluxos.



Importante: As atualizações podem ocorrer em fluxos diretos ou reversos!

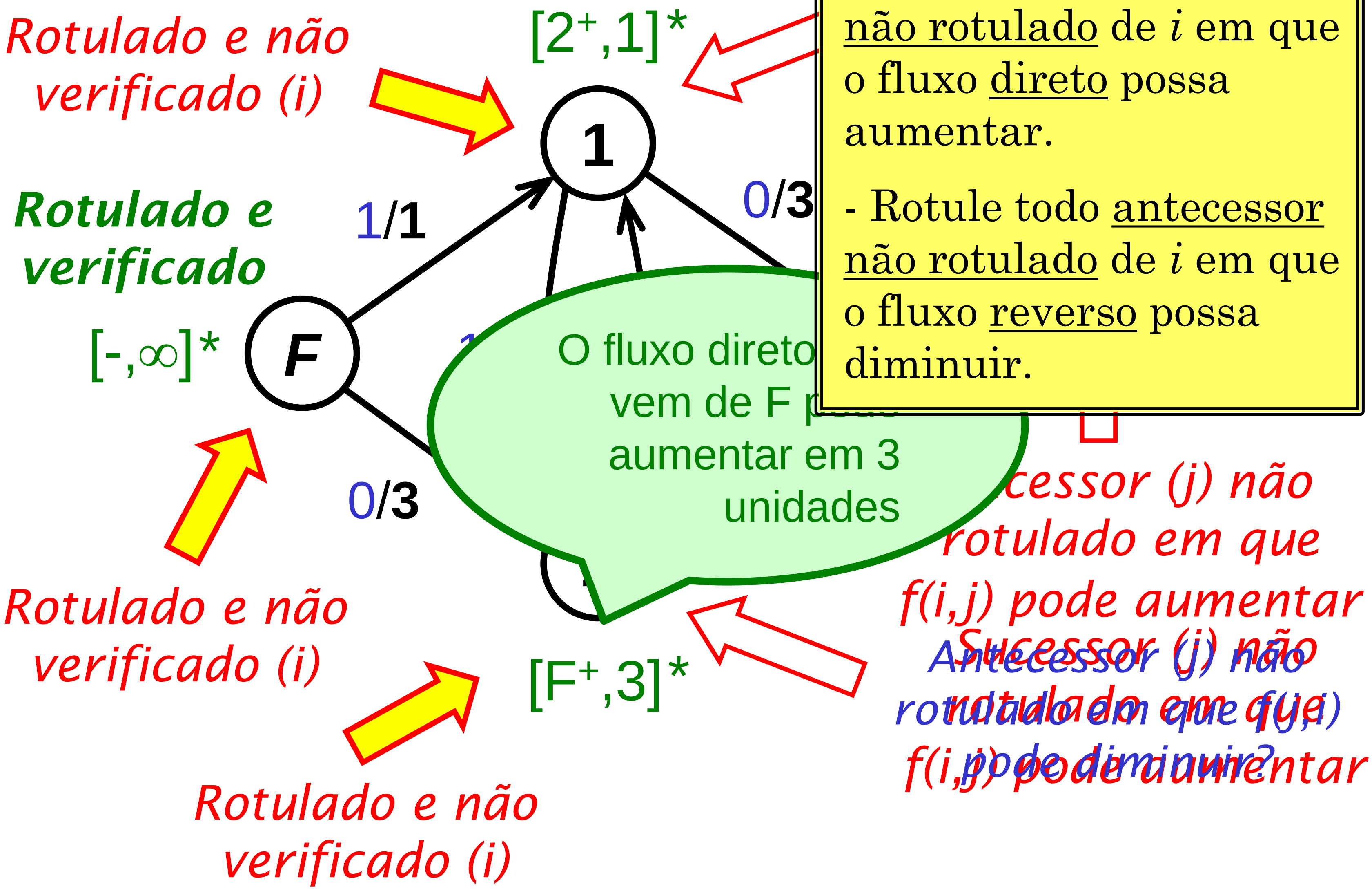
Algoritmo de Ford-Fulkerson

Fluxo máximo: rotina A

Passo genérico:

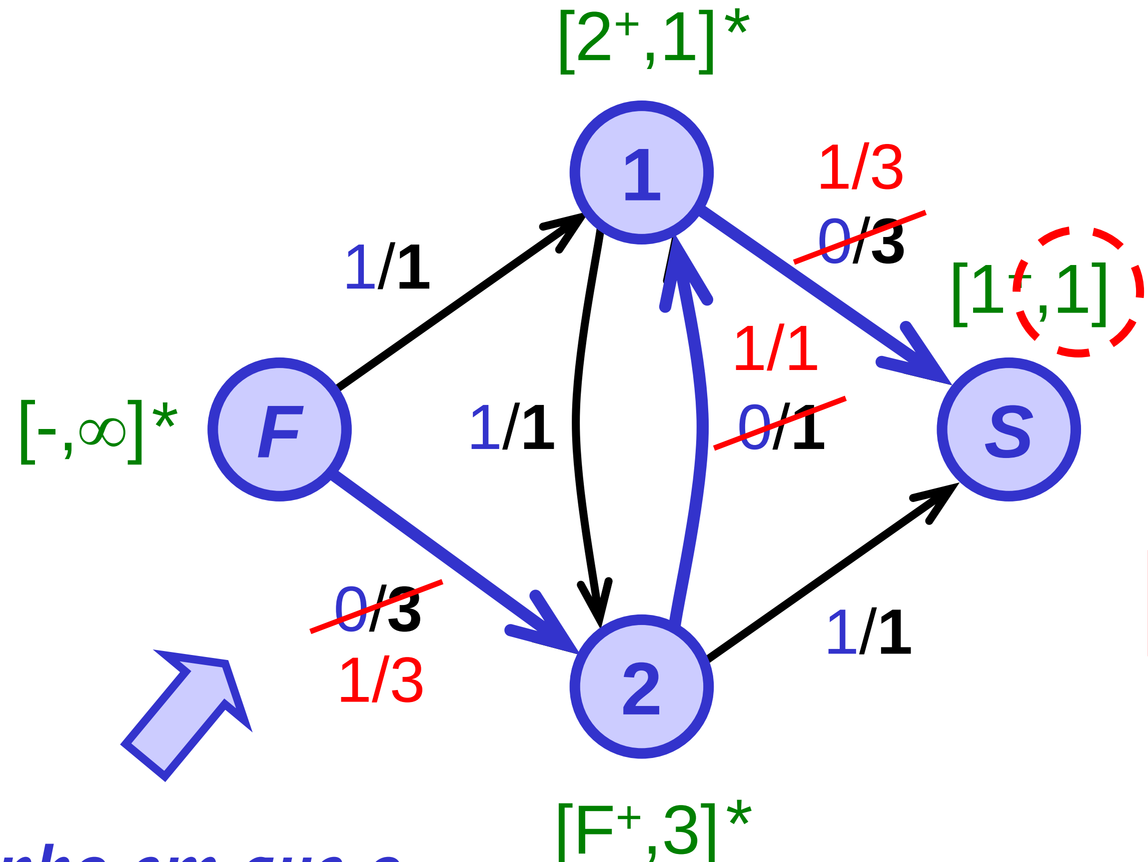
- Rotule todo sucessor não rotulado de i em que o fluxo direto possa aumentar.

- Rotule todo antecessor não rotulado de i em que o fluxo reverso possa diminuir.



Algoritmo de Ford-Fulkerson

Fluxo máximo: rotina B



Atualizando os fluxos...

Δfluxo = 1

Caminho em que o fluxo pode aumentar

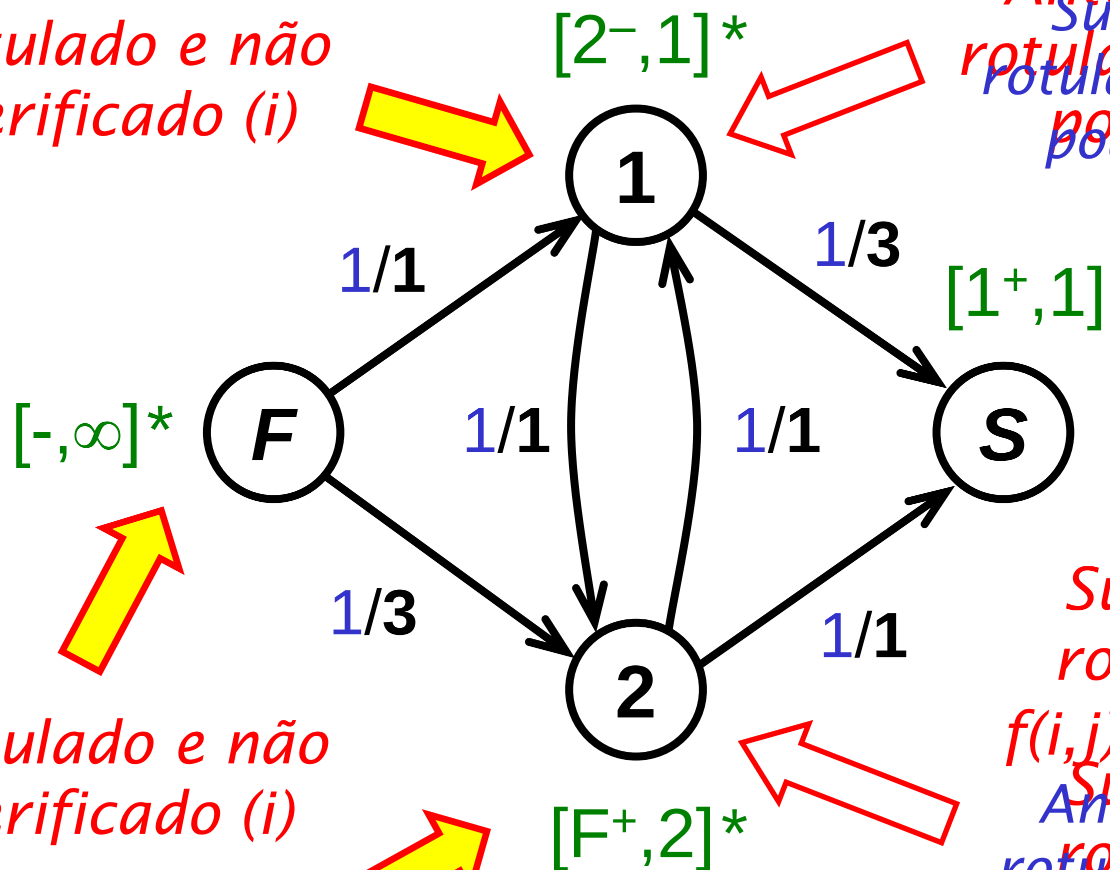
Algoritmo de Ford-Fulkerson

Fluxo máximo: rotina A

Rotulado e não verificado (i)

*Antecessor (j) não rotulado em que f(j,i) pode **diminuir**!*

Sucessor (j) não rotulado em que f(i,j) pode aumentar?



Rotina B!

Rotulado e não verificado (i)

Sucessor (j) não rotulado em que f(i,j) pode aumentar

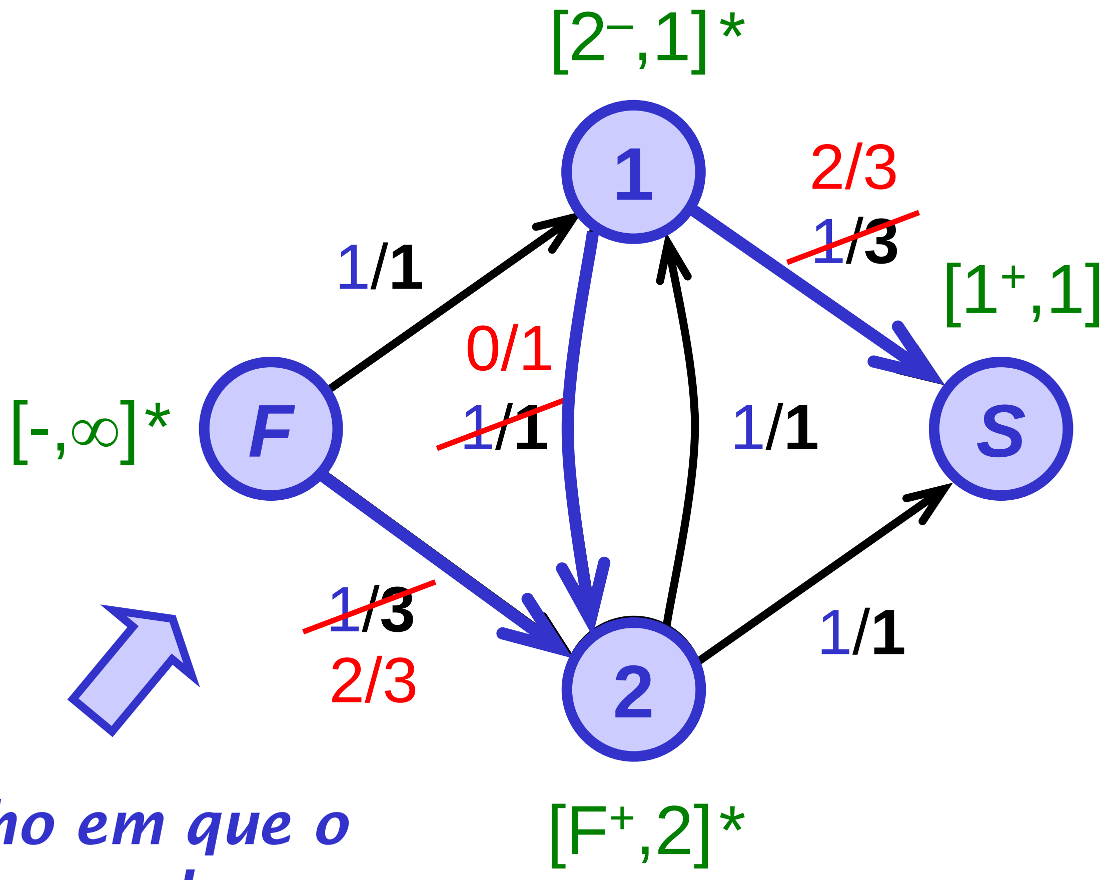
Sucessor (j) não rotulado em que f(i,j) pode aumentar

Antecessor (j) não rotulado em que f(j,i) pode diminuir?

Rotulado e não verificado (i)

Algoritmo de Ford-Fulkerson

Fluxo máximo: rotina B



Atualizando os fluxos...

Δfluxo = 1

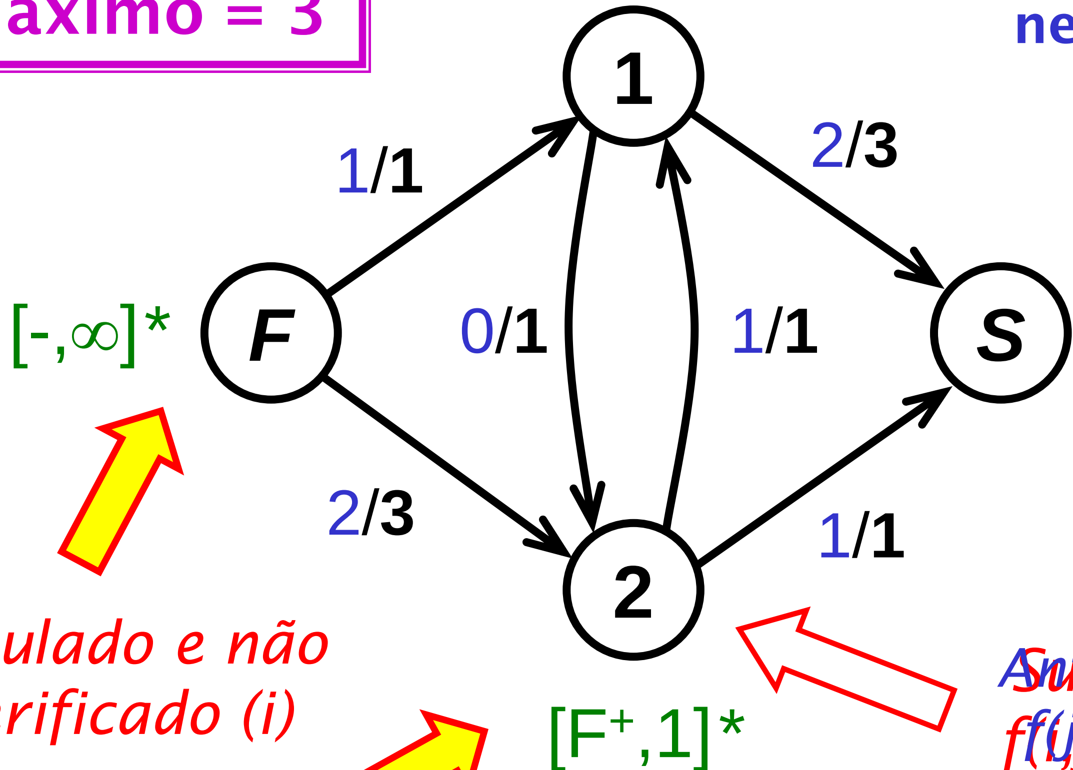
Caminho em que o fluxo pode aumentar

Algoritmo de Ford-Fulkerson

Fluxo máximo: rotina A

Solução ótima:
fluxo máximo = 3

Não há como rotular nenhum outro nó!



Rotulado e não verificado (i)

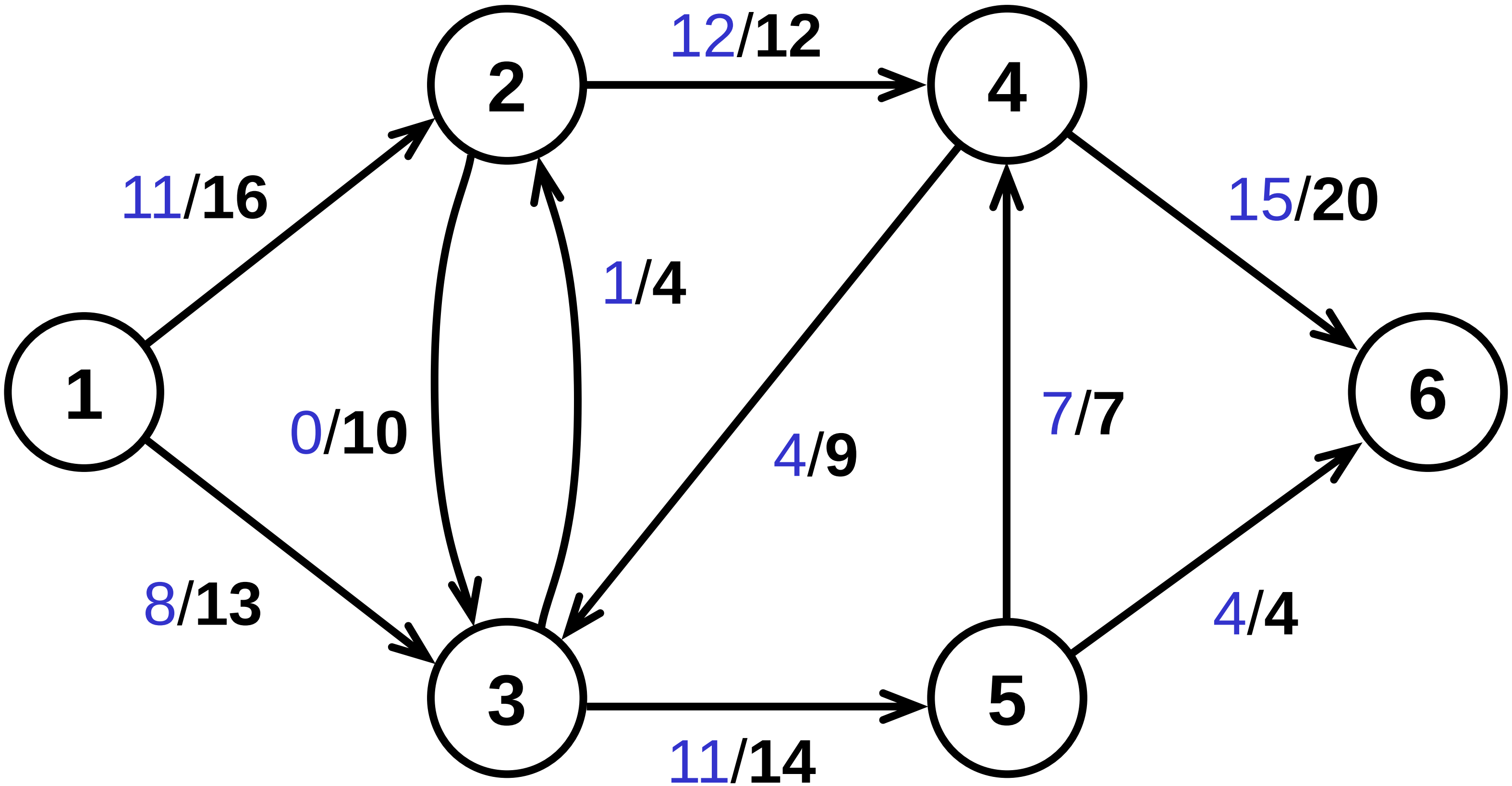
Antecessor (j) em que f(j,i) pode diminuir?

Sucessor (j) em que f(i,j) pode aumentar

Rotulado e não verificado (i)

Exercício proposto

Aplique, apenas 1 vez, as rotinas A e B do algoritmo de Ford-Fulkerson à rede mostrada na pág. 130. No passo genérico da rotina A, se houver mais de um nó disponível, escolha o nó de menor numeração:



Exercício proposto

Rotina B

Δfluxo = 4