PROCURA EM ESPAÇOS DE ESTADOS

(PARTE 2)

Luís Morgado

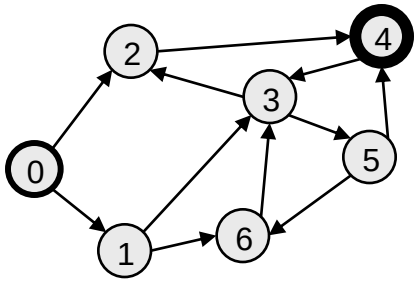
2015

MÉTODOS DE PROCURA

Procura em profundidade (*Depth-First Search*)

function DEPTH-FIRST-SEARCH(*problem*) **returns** a solution, or failure
GENERAL-SEARCH(*problem*,ENQUEUE-AT-FRONT)

[Russel & Norvig, 2003]



Grafo do
Espaço de Estados

Fronteira de exploração [ ]

[0]

0 [ ]

[**1, 2**]

1 [2]

[**3, 6**, 2]

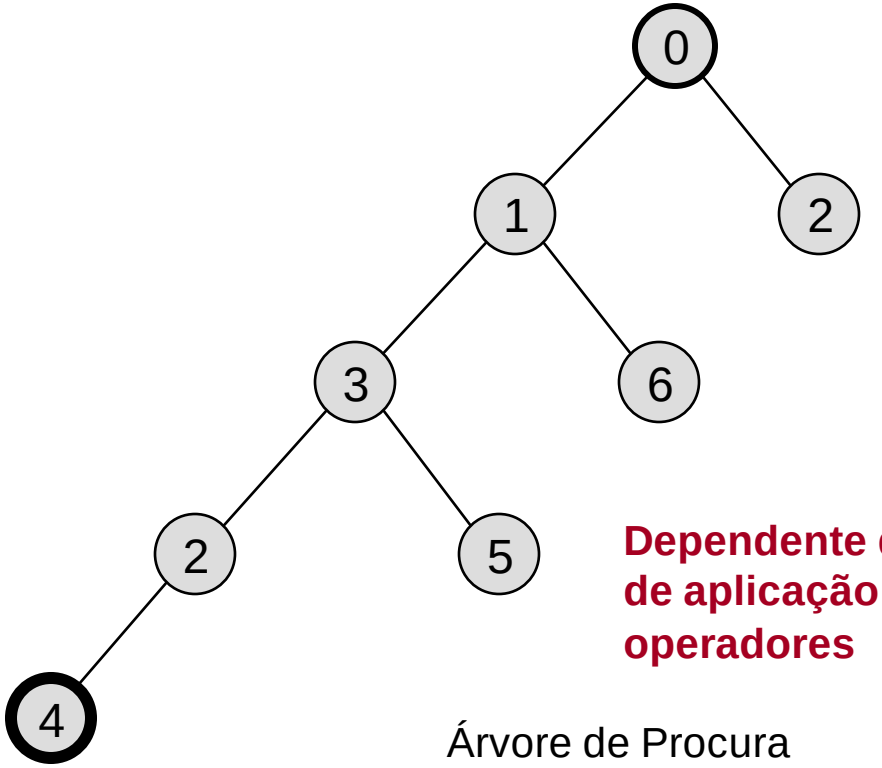
3 [6,2]

[**2, 5**, 6, 2]

2 [5,6,2]

[**4**, 5, 6, 2]

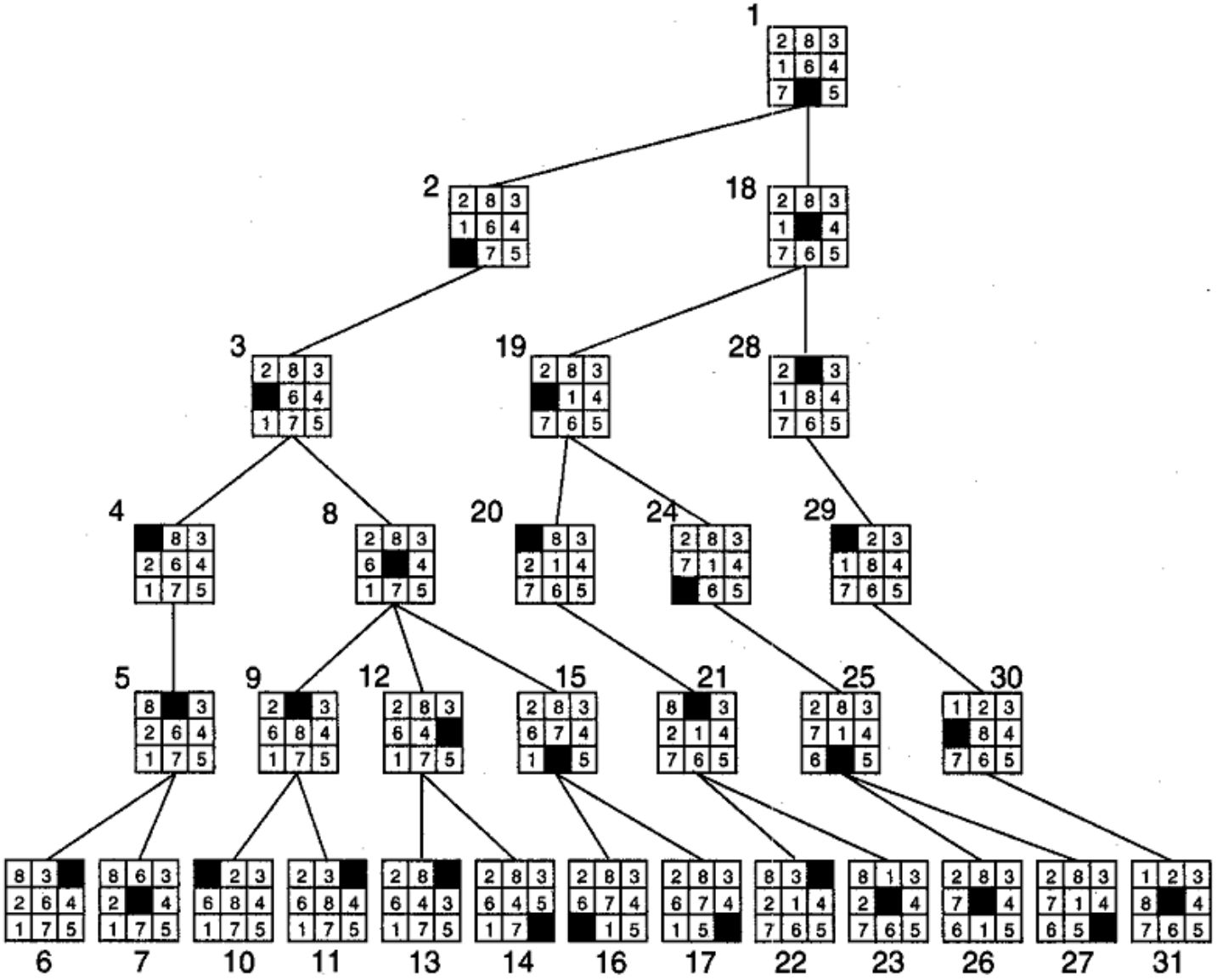
4 [5,6,2]



Árvore de Procura

Dependente da ordem de aplicação dos operadores

PROCURA EM PROFUNDIDADE



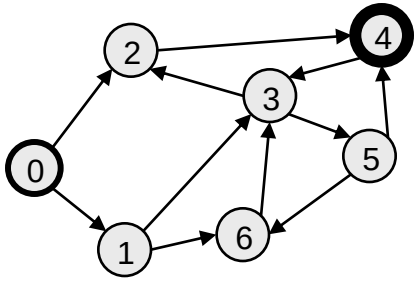
[Luger, 2009]

MÉTODOS DE PROCURA

Procura em largura (*Breadth-First Search*)

```
function BREADTH-FIRST-SEARCH(problem) returns a solution or failure
  return GENERAL-SEARCH(problem, ENQUEUE-AT-END)
```

[Russel & Norvig, 2003]



Grafo do
Espaço de Estados

Fronteira de exploração [ ]

[0]

0 [ ]

[**1, 2**]

1 [2]

[2, **3, 6**]

2 [3,6]

[3, 6, **4**]

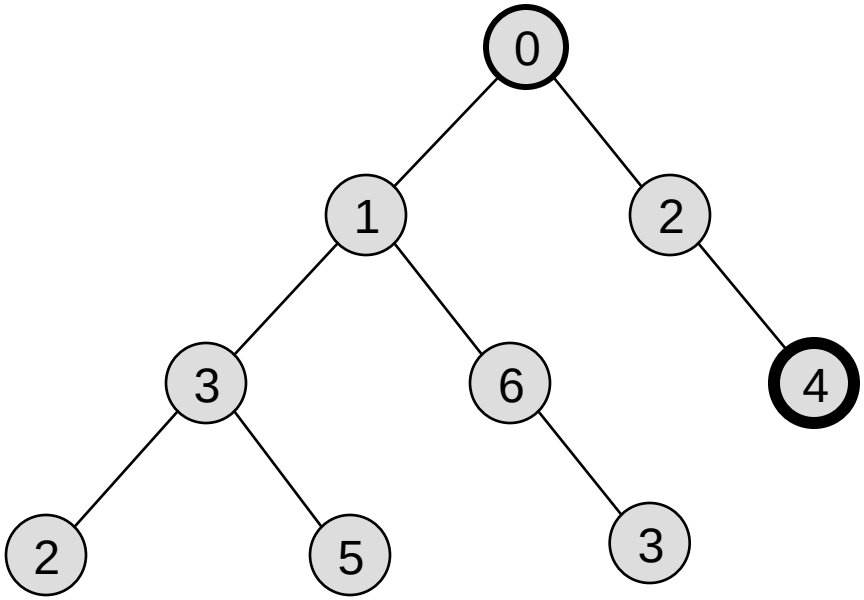
3 [6,4]

[6,4,**2,5**]

6 [4,2,5]

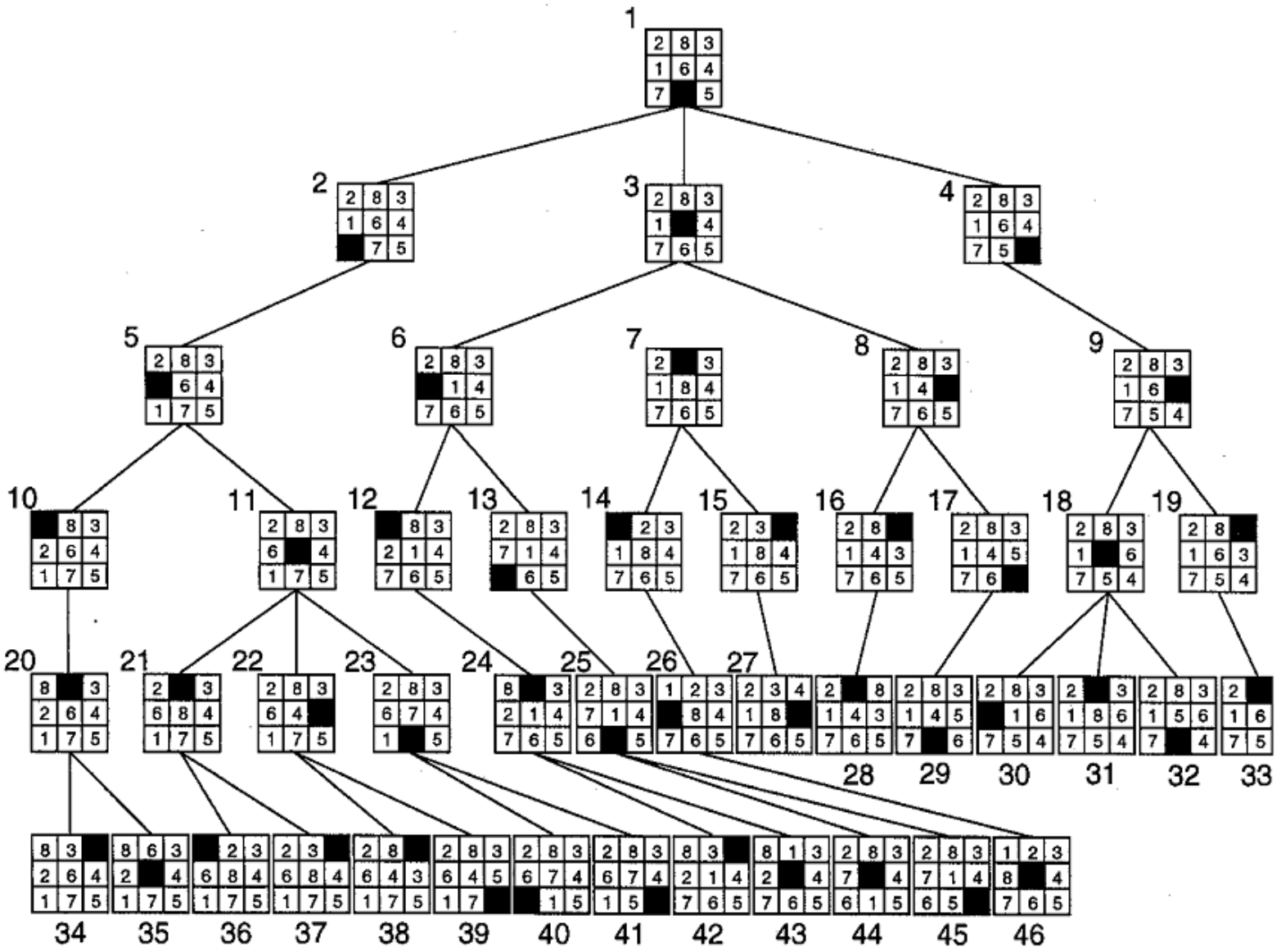
[4,2,5,**3**]

4 [2,5,3]



Árvore de Procura

PROCURA EM LARGURA



[Luger, 2009]

CRITÉRIOS DE EXPLORAÇÃO DO GRAFO DE ESPAÇO DE ESTADOS

PROCURA EM PROFUNDIDADE

- Explorar primeiro nós com **maior profundidade**

PROCURA EM LARGURA

- Explorar primeiro nós com **menor profundidade**

MÉTODOS DE PROCURA

QUAL O MELHOR MÉTODO DE PROCURA ?

- Aspectos a considerar num método de procura
  - **COMPLETO**
    - O método de procura garante que, caso exista solução, esta será encontrada
  - **ÓPTIMO**
    - O método de procura garante que, existindo várias soluções, a solução encontrada é a melhor
  - **COMPLEXIDADE**
    - **TEMPO** (complexidade temporal)
      - **Tempo necessário** para encontrar uma solução
    - **ESPAÇO** (complexidade espacial)
      - **Memória necessária** para encontrar uma solução

MÉTODOS DE PROCURA

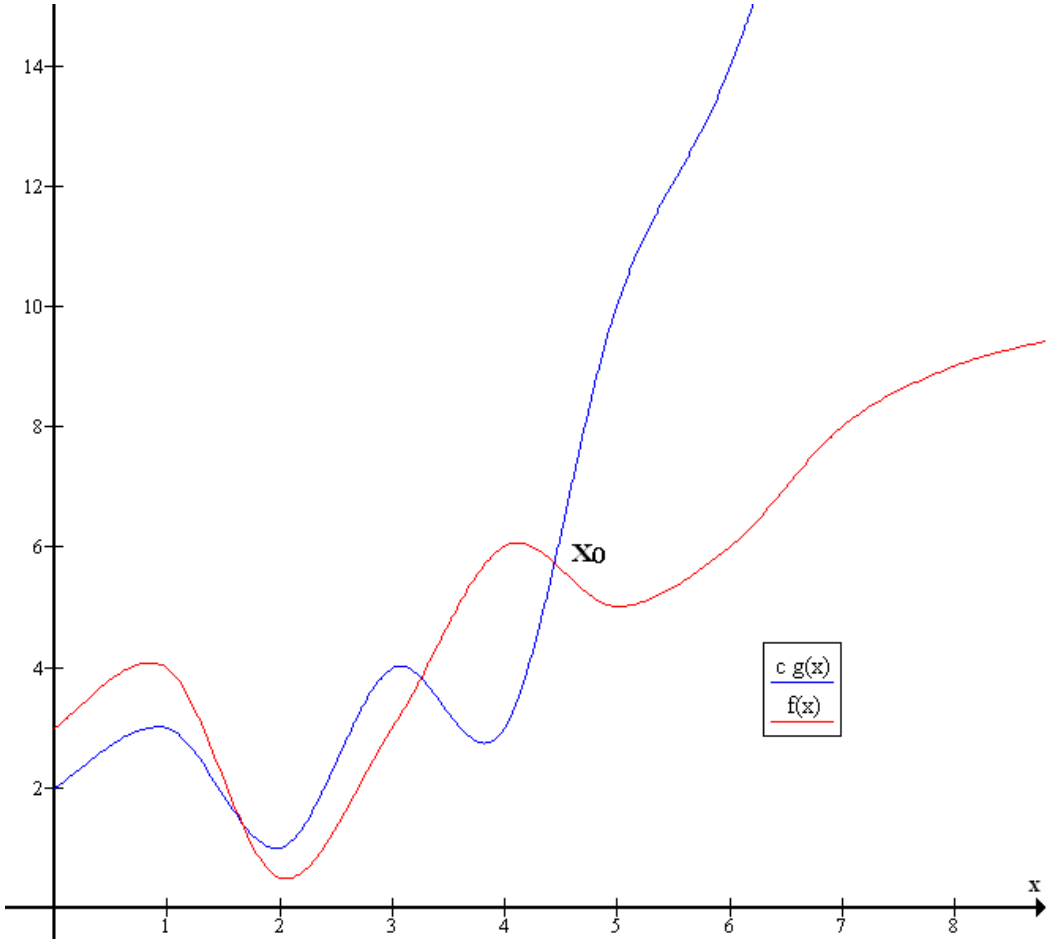
- Parâmetros de caracterização de um método de procura:
  - **FACTOR DE RAMIFICAÇÃO** - ***b***
    - Número máximo de sucessores para um qualquer estado
  - **PROFUNDIDADE DA PROCURA** - ***d***
    - Profundidade do nó objectivo menos profundo na árvore de procura
    - Dimensão do percurso entre o estado inicial e o estado objectivo

COMPORTAMENTO LIMITE DE UMA FUNÇÃO

Notação $f = O(g)$

$f(x)$ é de ordem $O(g(x))$ se existirem duas constantes positivas x_0 e c tal que:

$(x > x_0) : f(x) \leq cg(x)$



COMPLEXIDADE COMPUTACIONAL

PROCURA EM LARGURA

Factor de ramificação *(branching factor)*: ***b***

Número de nós a expandir para encontrar uma solução de dimensão ***d***

$1 + b + b^2 + b^3 + \ldots + b^d$ ⟶ Complexidade espacial: $O(b^d)$

Complexidade temporal: $O(b^d)$

UTILIZAÇÃO EXTENSIVA DE MEMÓRIA

PROCURA EM PROFUNDIDADE

Número de nós a expandir para explorar até uma profundidade ***m***

Complexidade espacial: $O(bm)$

Complexidade temporal: $O(b^m)$

PODE NÃO ENCONTRAR SOLUÇÃO

COMPLEXIDADE COMPUTACIONAL

Método de Procura	Tempo	Espaço	Óptimo	Completo
Profundidade	$O(b^m)$	$O(bm)$	Não	Não
Largura	$O(b^d)$	$O(b^d)$	Sim	Sim

b – factor de ramificação
d – dimensão da solução
m – profundidade da árvore de procura
C^* – Custo da solução óptima
ε – Custo mínimo de uma transição de estado ($\varepsilon > 0$)

MÉTODOS DE PROCURA

PROCURA EM PROFUNDIDADE LIMITADA
(*Depth-Limited Search*)

- Limitar procura a uma profundidade máxima

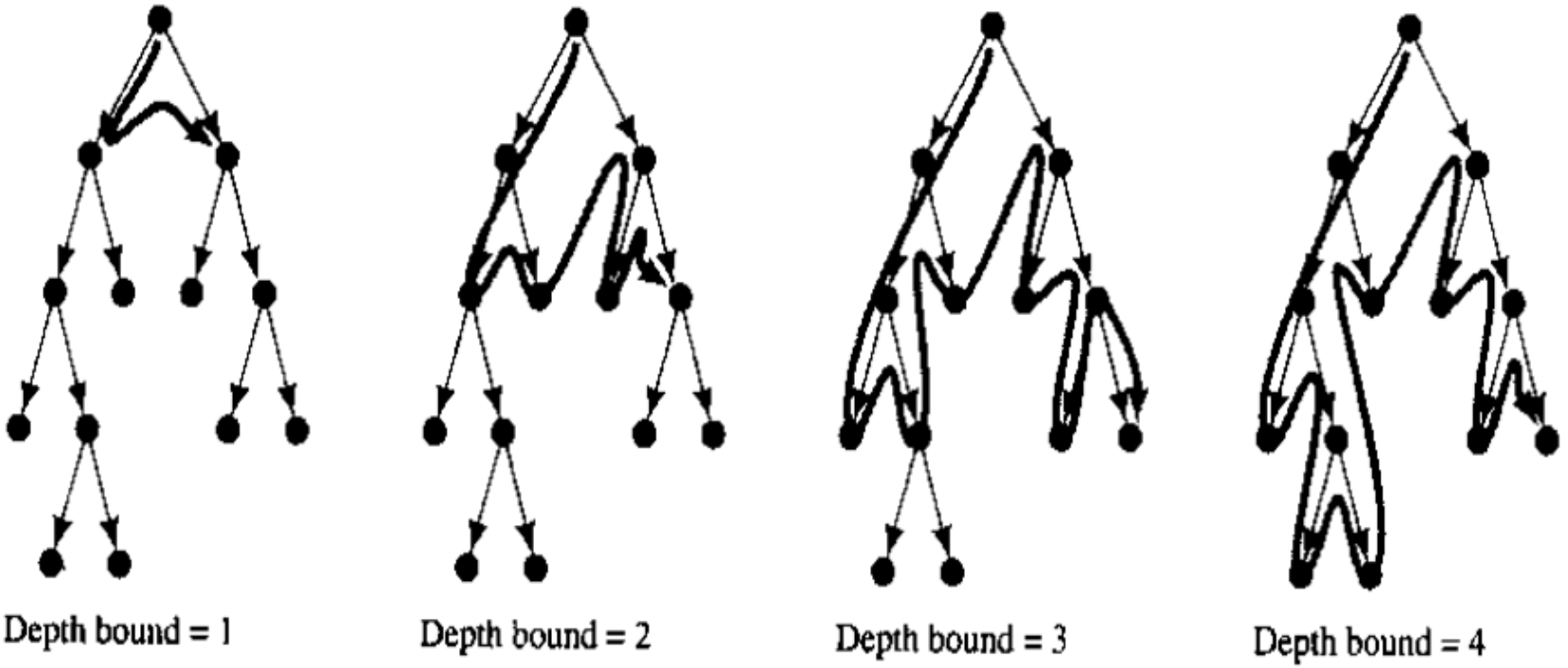
PROCURA EM PROFUNDIDADE ITERATIVA
(*Iterative Deepening Search*)

```
function ITERATIVE-DEEPENING-SEARCH(problem) returns a solution, or failure
    inputs: problem, a problem

    for depth ← 0 to ∞ do
        result ← DEPTH-LIMITED-SEARCH(problem, depth)
        if result ≠ cutoff then return result
```

[Russel & Norvig, 2003]

PROCURA EM PROFUNDIDADE ITERATIVA



[Nilsson, 1998]

Número de nós a expandir para encontrar uma solução de dimensão ***d***

$(d+1) + (d)b + (d-1)b^2 + \ldots + 2b^{d-1} + 1b^d$

Complexidade espacial: $O(bd)$

Complexidade temporal: $O(b^d)$

PROCURA BIDIRECCIONAL

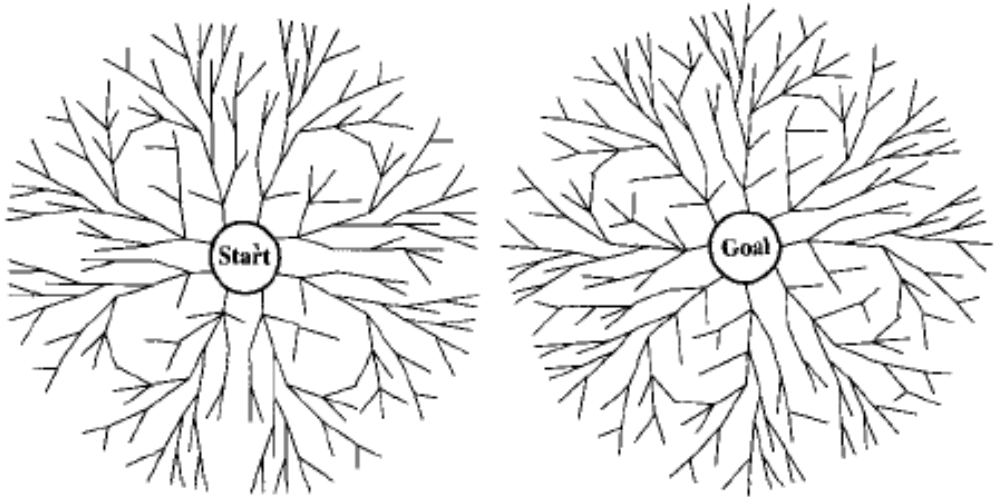


Figure 3.17 A schematic view of a bidirectional breadth-first search that is about to succeed, when a branch from the start node meets a branch from the goal node.

[Russel & Norvig, 2003]

BIBLIOGRAFIA

[Russel & Norvig, 2009]

S. Russell, P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd Edition, Prentice Hall, 2009

[Nilsson, 1998]

N. Nilsson , *Artificial Intelligence: A New Synthesis*, Morgan Kaufmann 1998

[Luger, 2009]

G. Luger , *Artificial Intelligence: Structures and Strategies for Complex Problem Solving* , Addison-Wesley, 2009

[Jaeger & Hamprecht, 2010]

M. Jaeger, F. Hamprecht, *Automatic Process Control for Laser Welding*, Heidelberg Collaboratory for Image Processing (HCI) , 2000