

Algoritma Branch & Bound

Bahan Kuliah IF2211 Strategi Algoritma

Program Studi Informatika – STEI ITB
2018

Overview

- Pembentukan pohon ruang status (state space tree) dinamis untuk mencari solusi persoalan
 - BFS
 - DFS
 - DLS
 - IDS
 - Backtracking

Overview

- BFS: solusi dgn minimum step, exponential space
- DFS: lebih efisien (1 solusi) , lintasannya dapat terlalu panjang (pohon ruang status tidak berhingga kedalamannya)
- DLS: variasi DFS, solusi bisa tidak ketemu (depth-limited)
- IDS: sekuens DLS (depth ++)
- Backtracking: basis DFS, expand simpul jika arahnya benar, fungsi pembatas

Algoritma Branch & Bound (B&B)

- Digunakan untuk persoalan optimisasi → meminimalkan atau memaksimalkan suatu fungsi objektif, yang tidak melanggar batasan (*constraints*) persoalan
- B&B: BFS + least cost search
 - BFS murni: Simpul berikutnya yang akan diekspansi berdasarkan urutan pembangkitannya (FIFO)
- B&B:
 - Setiap simpul diberi sebuah nilai cost:
 $\hat{c}(i)$ = nilai taksiran lintasan termurah ke simpul status tujuan yang melalui simpul status i .
 - Simpul berikutnya yang akan di-expand **tidak lagi** berdasarkan urutan pembangkitannya, tetapi simpul yang memiliki *cost* yang paling kecil (least cost search) – pada kasus minimasi.

B&B vs Backtracking

- Persamaan:
 - Pencarian solusi dengan pembentukan pohon ruang status
 - ‘Membunuh’ simpul yang tidak ‘mengarah’ ke solusi
- Perbedaan:
 - ‘nature’ persoalan yang bisa diselesaikan:
 - Backtracking: Tak ada batasan (optimisasi), umumnya untuk persoalan non-optimisasi
 - B&B:
 - Persoalan optimisasi
 - Untuk setiap simpul pada pohon ruang-status, diperlukan suatu cara penentuan batas (bound) nilai terbaik fungsi objektif pada setiap solusi yang mungkin, dengan menambahkan komponen pada solusi sementara yang direpresentasikan simpul
 - Nilai dari solusi terbaik sejauh ini
 - Pembangkitan simpul: ...

B&B vs Backtracking (2)

- Perbedaan:
 - Pembangkitan simpul:
 - Backtracking: umumnya DFS
 - B&B : beberapa 'aturan' tertentu → paling umum 'best-first rule'

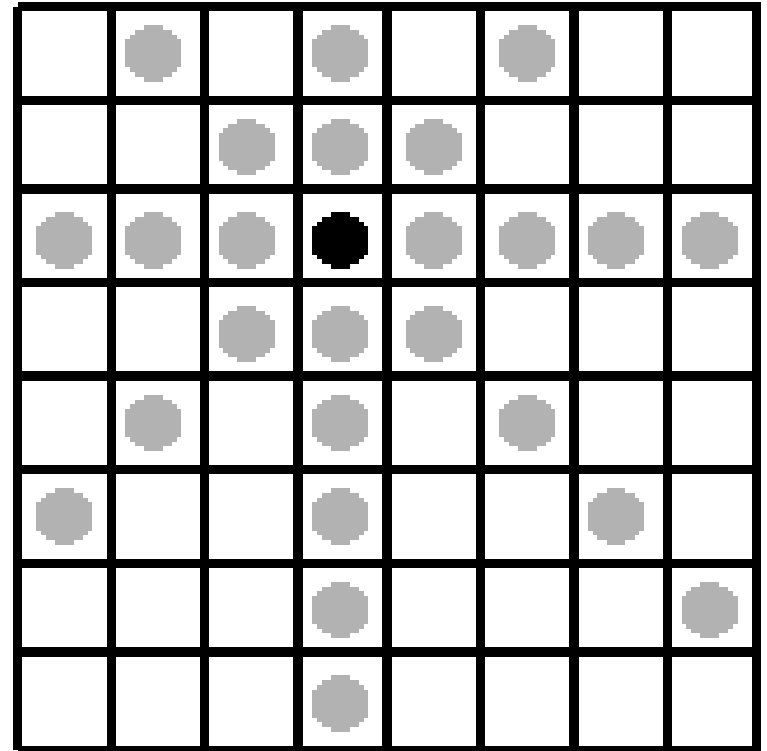
“Fungsi Pembatas”

- Algoritma B&B juga menerapkan “pemangkasan” pada jalur yang dianggap tidak lagi mengarah pada solusi.
- Kriteria pemangkasan secara umum:
 - Nilai simpul tidak lebih baik dari nilai terbaik sejauh ini
 - Simpul tidak merepresentasikan solusi yang ‘feasible’ karena ada batasan yang dilanggar
 - Solusi yang feasible pada simpul tersebut hanya terdiri atas satu titik → tidak ada pilihan lain; bandingkan nilai fungsi obyektif dengan solusi terbaik saat ini, yang terbaik yang diambil

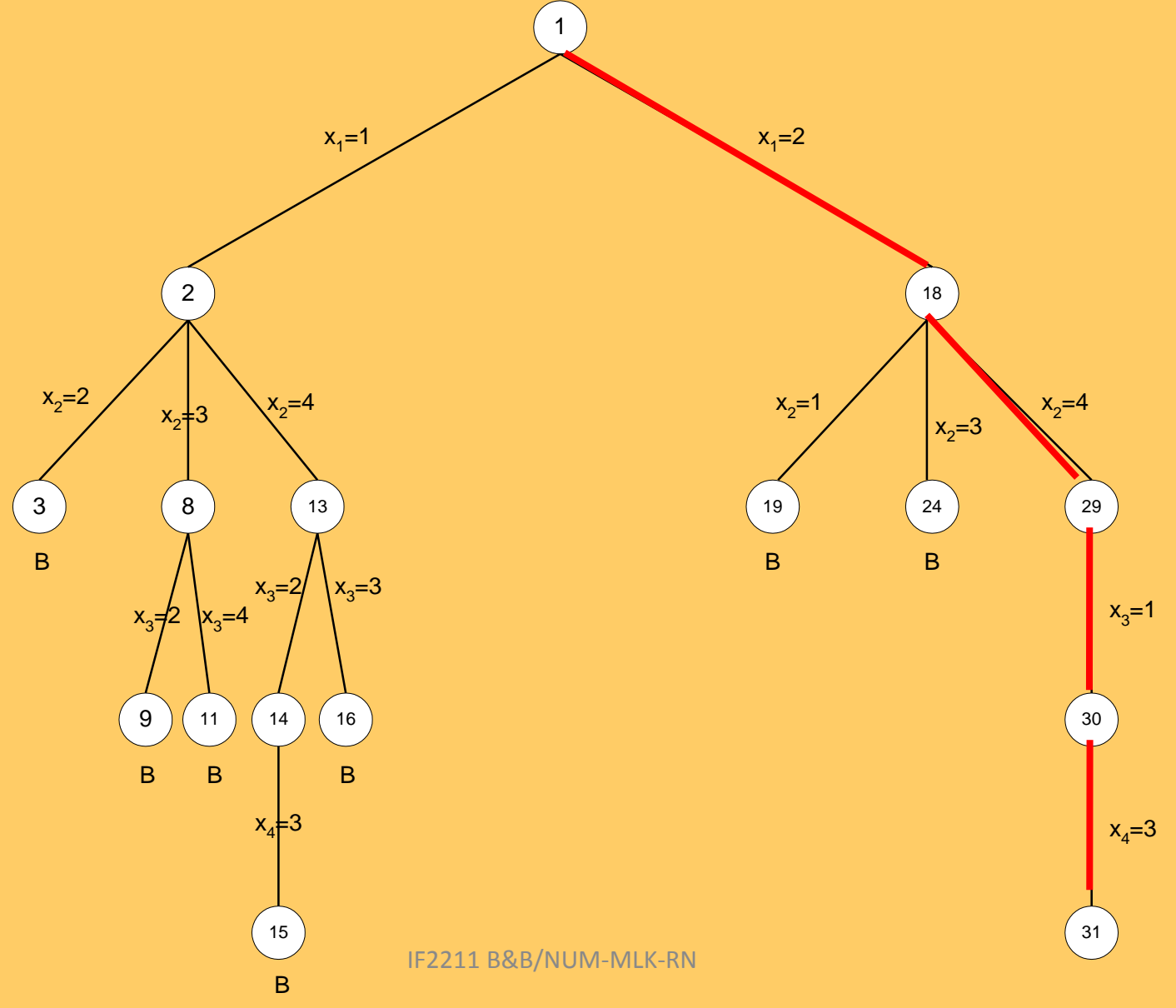
Persoalan N-Ratu

(The N-Queens Problem)

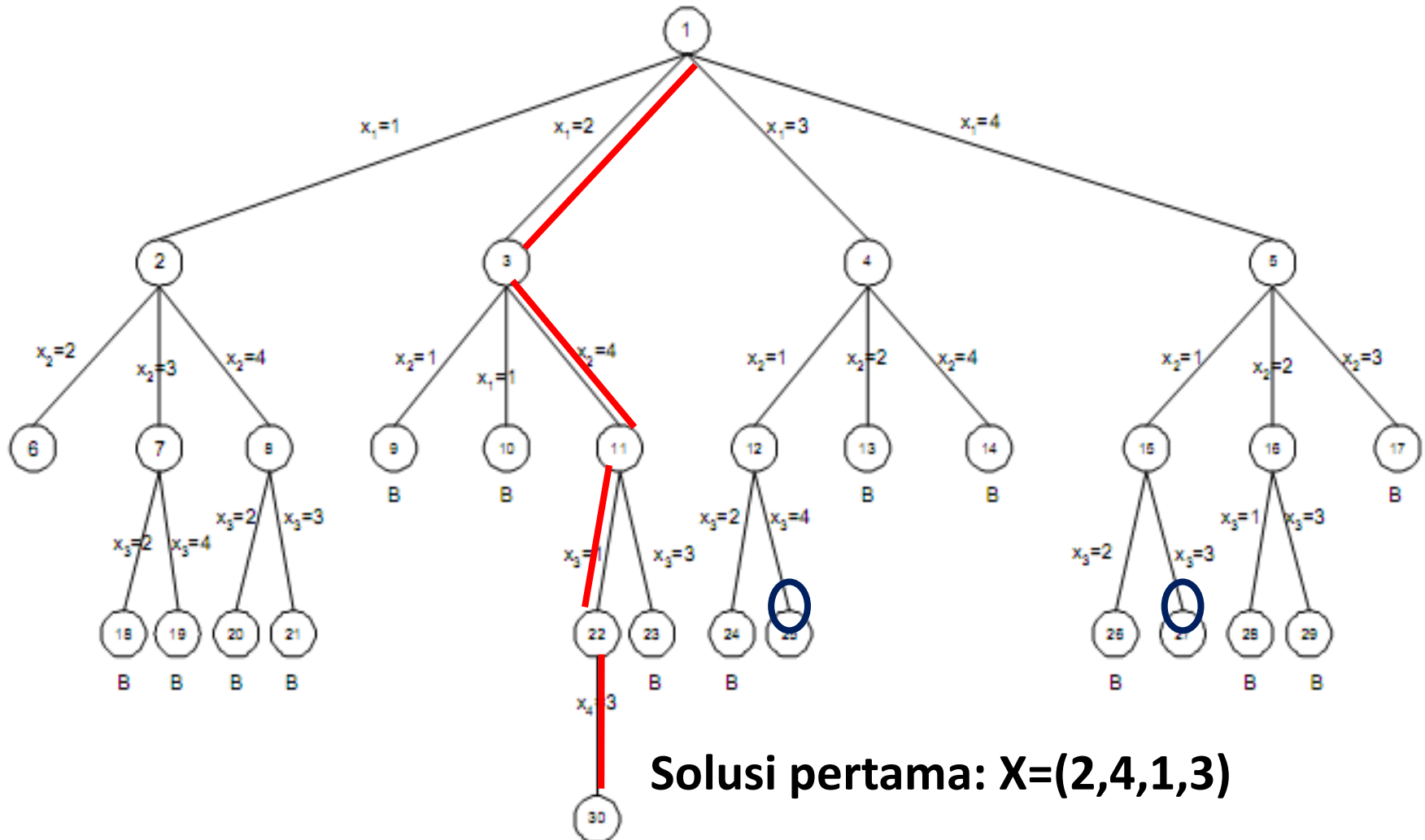
- Diberikan sebuah papan permainan yang berukuran $N \times N$ dan N buah ratu. Bagaimanakah menempatkan N buah ratu (Q) itu pada petak-petak papan permainan sedemikian sehingga tidak ada dua ratu atau lebih yang terletak pada satu baris yang sama, atau pada satu kolom yang sama, atau pada satu diagonal yang sama.



Pohon ruang status persoalan 4-Ratu: Backtracking



Solusi 4-Ratu dengan BFS-dengan Pembatasan (FIFO-Branch and Bound)

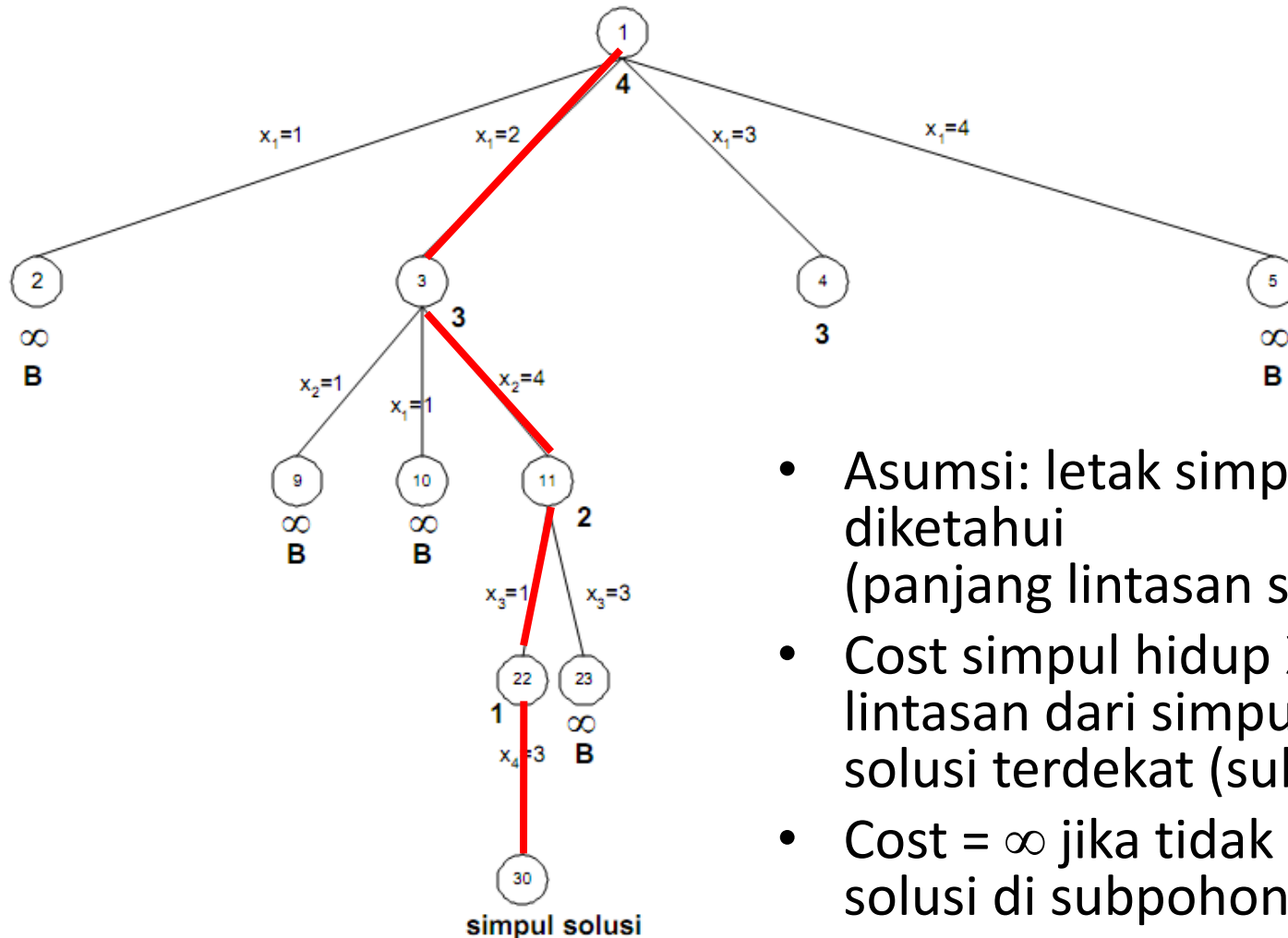


Solusi pertama: $X=(2,4,1,3)$

Strategi Pencarian B&B untuk 4-Ratu dengan *Least Cost Search*

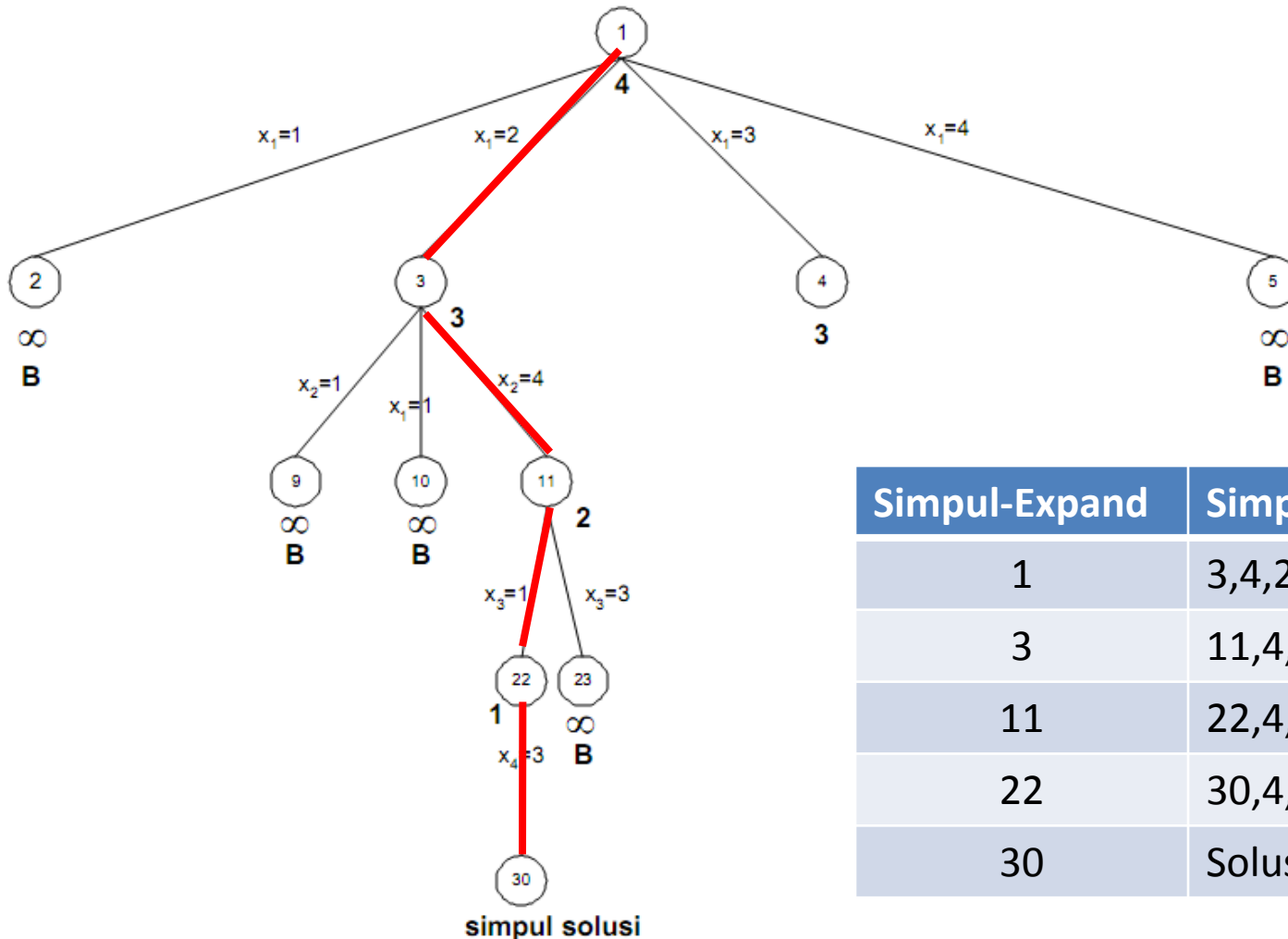
- Simpul hidup yang menjadi simpul-E(xpand) ialah simpul yang mempunyai nilai *cost* terkecil (*least cost search*) → salah satu jenis aturan
- Untuk setiap simpul X, nilai batas ini dapat berupa [HOR78]:
 1. jumlah simpul dalam upapohon X yang perlu dibangkitkan sebelum simpul solusi ditemukan
 2. panjang lintasan dari simpul X ke simpul solusi terdekat (dalam upapohon X ybs) → misal ini yang dipilih untuk persoalan 4-ratu

Solusi 4-Ratu dengan Branch & Bound dengan *Least Cost Search*



- Asumsi: letak simpul solusi diketahui (panjang lintasan solusi = 4)
- Cost simpul hidup X: panjang lintasan dari simpul X ke simpul solusi terdekat (subpohon X).
- Cost = ∞ jika tidak ada simpul solusi di subpohon tersebut.

Pembentukan Pohon Ruang Status 4-Ratu dengan Branch & Bound



Simpul-Expand	Simpul Hidup
1	3,4,2,5
3	11,4,2,5,9,10
11	22,4,2,5,9,10,23
22	30,4,2,5,9,10,23
30	Solusi ketemu

Cost dari Simpul Hidup

- Pada umumnya, untuk kebanyakan persoalan, letak simpul solusi tidak diketahui.
 - Persoalan N-Ratu: persoalan yg ideal (letak simpul solusi diketahui)
- Letak simpul solusi diketahui ?
 - knapsack problem,
 - graph colouring,
 - permainan 8-puzzle,
 - TSP

Algoritma Global Branch & Bound

1. Masukkan simpul akar ke dalam antrian Q . Jika simpul akar adalah simpul solusi (goal node), maka solusi telah ditemukan. Stop.
2. Jika Q kosong, tidak ada solusi. Stop.
3. Jika Q tidak kosong, pilih dari antrian Q simpul i yang mempunyai nilai 'cost' $\hat{c}(i)$ paling kecil. Jika terdapat beberapa simpul i yang memenuhi, pilih satu secara sembarang.
4. Jika simpul i adalah simpul solusi, berarti solusi sudah ditemukan, stop. Jika simpul i bukan simpul solusi, maka bangkitkan semua anak-anaknya. Jika i tidak mempunyai anak, kembali ke langkah 2.
5. Untuk setiap anak j dari simpul i , hitung $\hat{c}(j)$, dan masukkan semua anak-anak tersebut ke dalam Q .
6. Kembali ke langkah 2.

Permainan 15-Puzzle

1	3	4	15
2		5	12
7	6	11	14
8	9	10	13

(a) Susunan awal

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

(b) Susunan akhir

- State berdasarkan ubin kosong (*blank*)
- Aksi: up, down, left, right

Reachable Goal ?

- Terdapat $16! \approx 20,9 \times 10^{12}$ susunan ubin yang berbeda, dan hanya setengah yang dapat dicapai dari state awal sembarang.
- Teorema : Status tujuan hanya dapat dicapai dari status awal jika $\sum_{i=1}^{16} KURANG(i) + X$ bernilai genap. $X=1$ jika pada sel yg diarsir

1	3	4	15
2		5	12
7	6	11	14
8	9	10	13

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

Reachable Goal : Kurang (i)

i	Kurang (i)
1	0
2	0
3	1
4	1
5	0
6	0
7	1
8	0
9	0
10	0
11	3
12	6
13	0
14	4
15	11
16	10

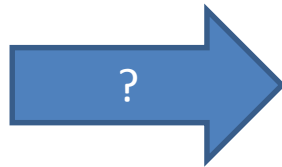
- KURANG(i) = banyaknya ubin bernomor j sedemikian sehingga $j < i$ dan $POSISI(j) > POSISI(i)$. $POSISI(i)$ = posisi ubin bernomor i pada susunan yang diperiksa.
- KURANG (4) = 1 : terdapat 1 ubin (2)
- Kesimpulan: status tujuan tidak dapat dicapai.

1	3	4	15
2		5	12
7	6	11	14
8	9	10	13

$$\sum_{i=1}^{16} Kurang(i) + X = 37 + 0 = 37$$

Reachable Goal ?

1	2	3	4
5	6		8
9	10	7	11
13	14	15	12



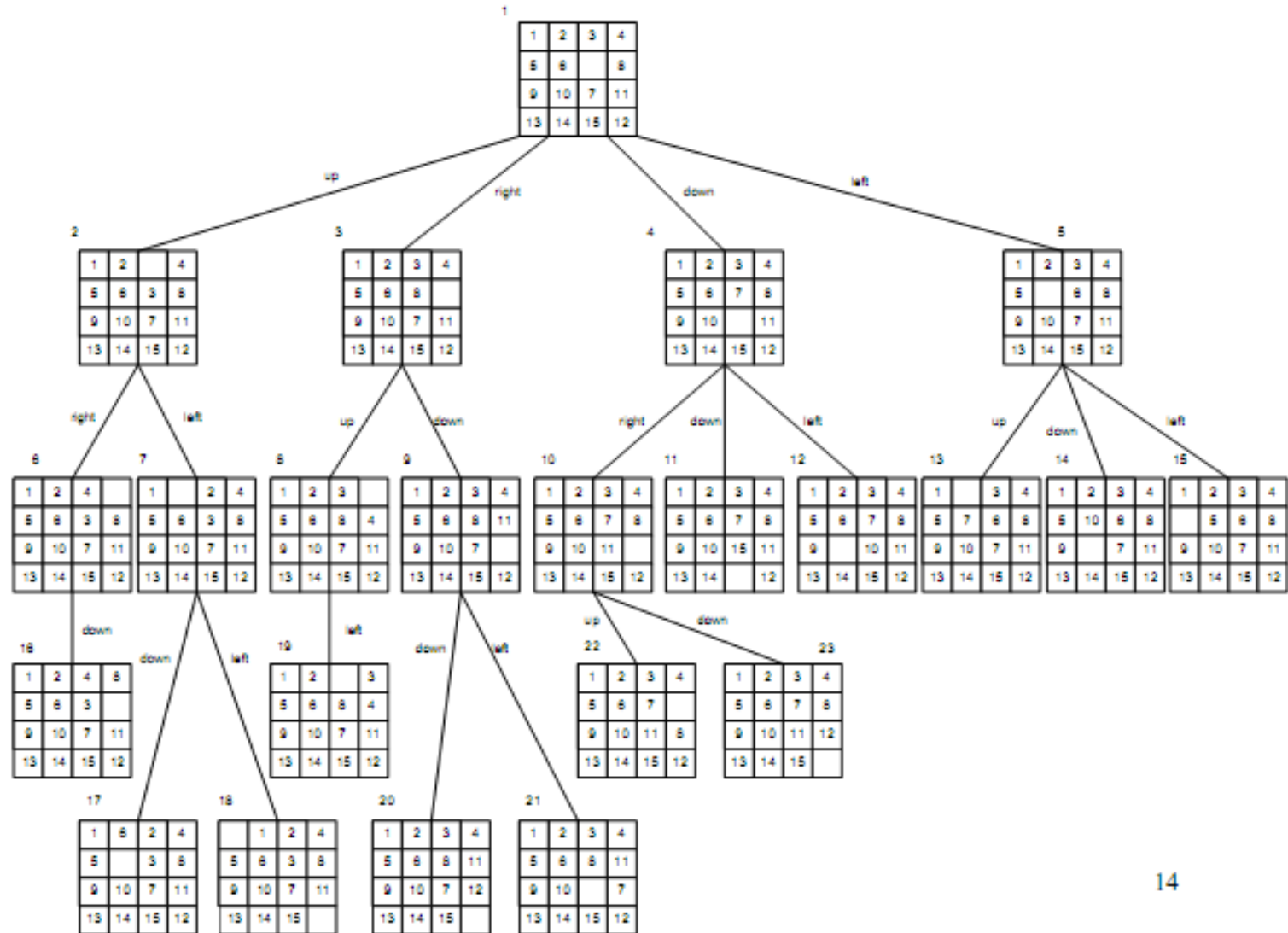
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

	■		■
■		■	
	■		■
■		■	

$$\sum_{i=1}^{16} Kurang(i) + X = 15 + 1 = 16$$

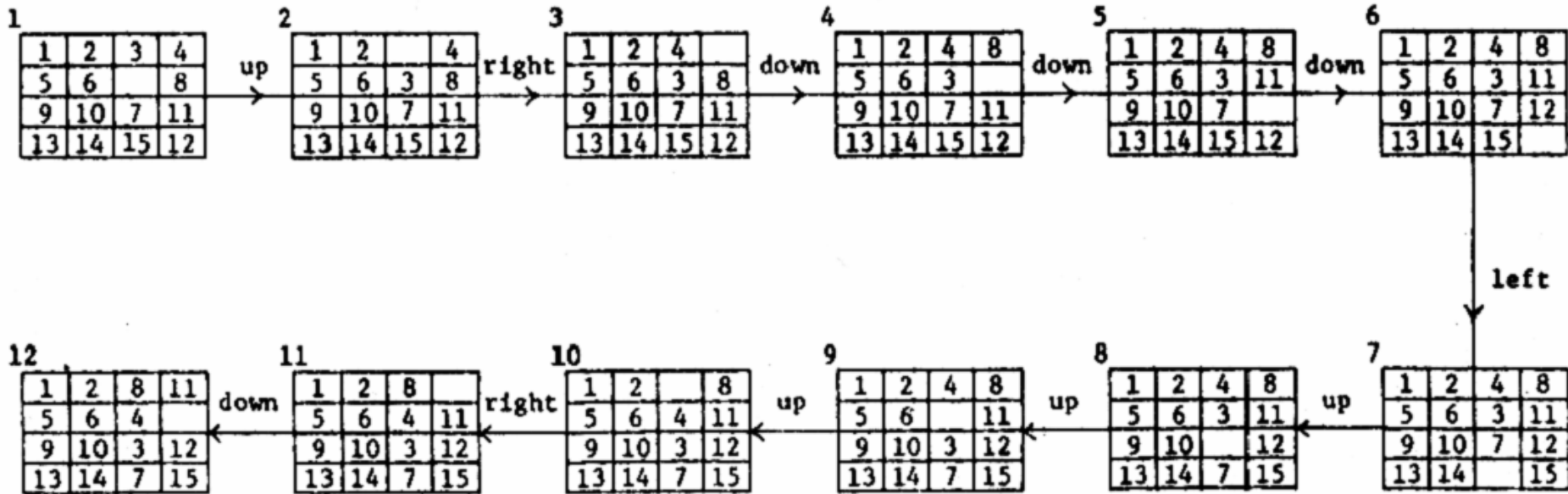
i	Kurang (i)
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	1
9	1
10	1
11	0
12	0
13	1
14	1
15	1
16	9

Pohon Ruang Status untuk 15-Puzzle



Pohon ruang status B&B ketika jalur ke solusi sudah 'diketahui'

Pohon Ruang Status untuk DFS



First ten steps in a depth first search

<http://chern.ie.nthu.edu.tw/alg2003/alg-2009-chap-7.pdf>

Cost dari Simpul Hidup (2)

- Pada umumnya, untuk kebanyakan persoalan, letak simpul solusi tidak diketahui.
- *Cost* setiap simpul umumnya berupa taksiran.

$$\hat{c}(i) = \hat{f}(i) + \hat{g}(i)$$

$\hat{c}(i)$ = ongkos untuk simpul i

$\hat{f}(i)$ = ongkos mencapai simpul i dari akar

$\hat{g}(i)$ = ongkos mencapai simpul tujuan dari simpul i .

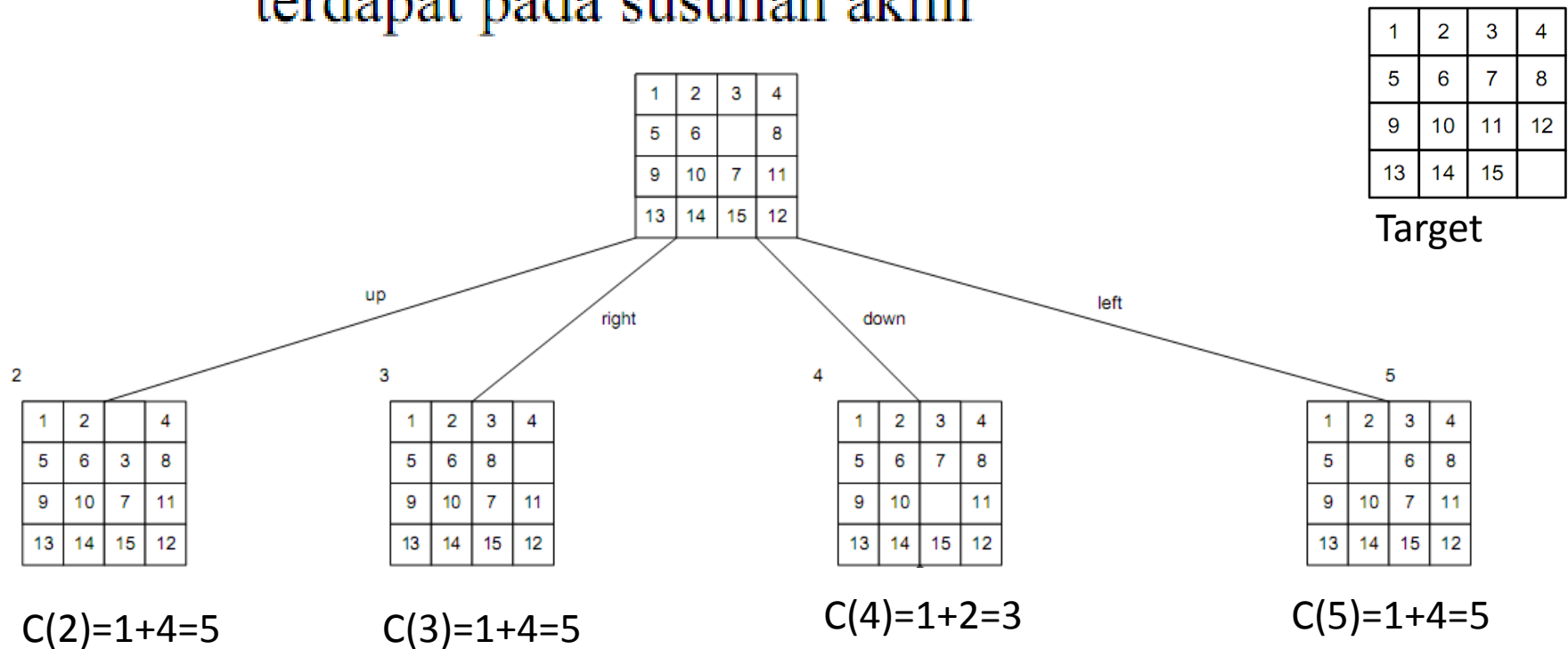
- Cost simpul P pada 15-puzzle:

$f(P)$ = adalah panjang lintasan dari simpul akar ke P

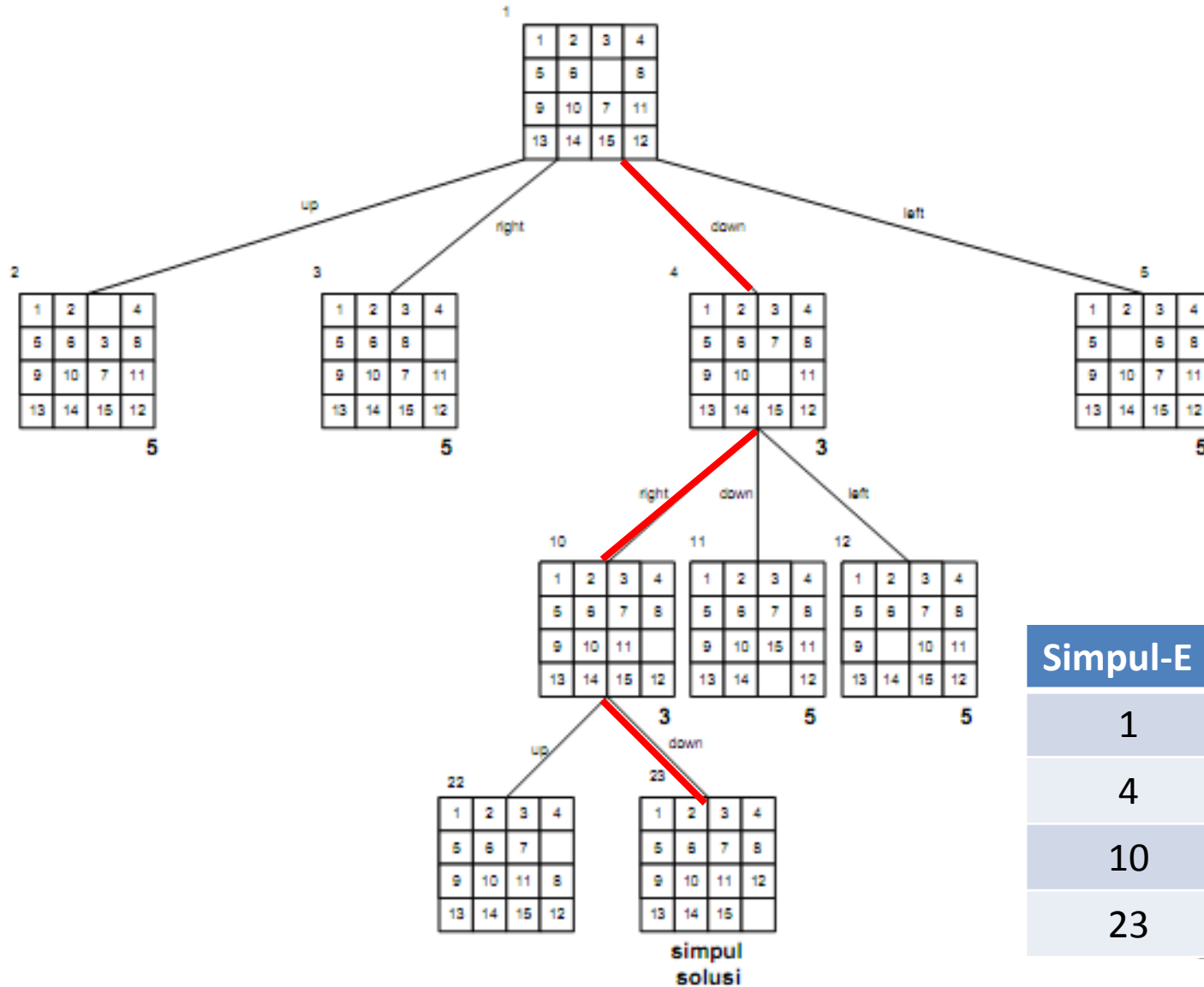
$\hat{g}(P)$ = taksiran panjang lintasan terpendek dari P ke simpul solusi pada upapohon yang akarnya P .

Cost dari Simpul Hidup 15-Puzzle

$\hat{g}(P)$ = jumlah ubin tidak kosong yang tidak terdapat pada susunan akhir



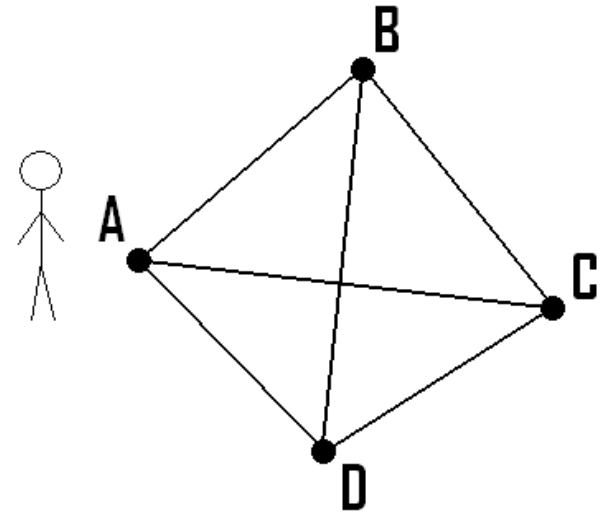
Pembentukan Pohon Ruang Status 15-Puzzle dengan Branch & Bound



Simpul-E	Simpul Hidup
1	4,2,3,5
4	10,2,3,5,11,12
10	23,2,3,5,11,12,22
23	Solusi ketemu

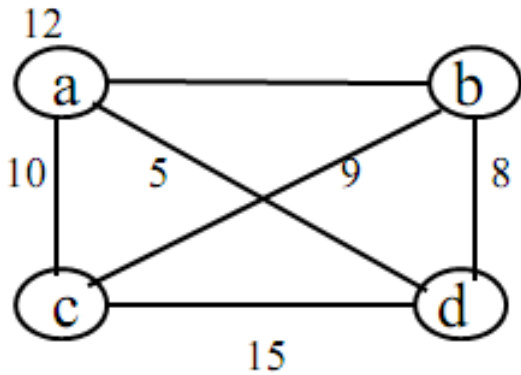
Travelling Salesperson Problem

Persoalan: Diberikan n buah kota serta diketahui jarak antara setiap kota satu sama lain. Temukan perjalanan (*tour*) terpendek yang melalui setiap kota lainnya hanya sekali dan kembali lagi ke kota asal keberangkatan.

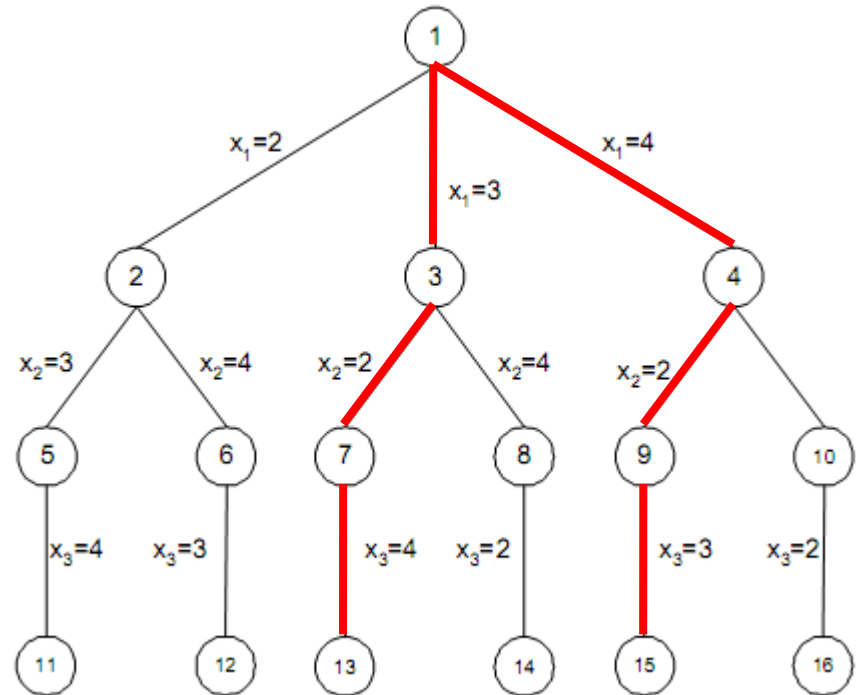


$(n-1)!$ sirkuit hamilton

Pohon Ruang Status TSP 4 Simpul



A=1; B=2; C=3; D=4
Simpul awal=1



Solusi: 1-3-2-4-1 atau 1-4-2-3-1

Bobot=5+8+9+10=32

(lihat diktat: TSP-Brute Force hal 20)

TSP dengan B & B

Contoh lain TSP 5 simpul (matriks bobot/cost matrix):

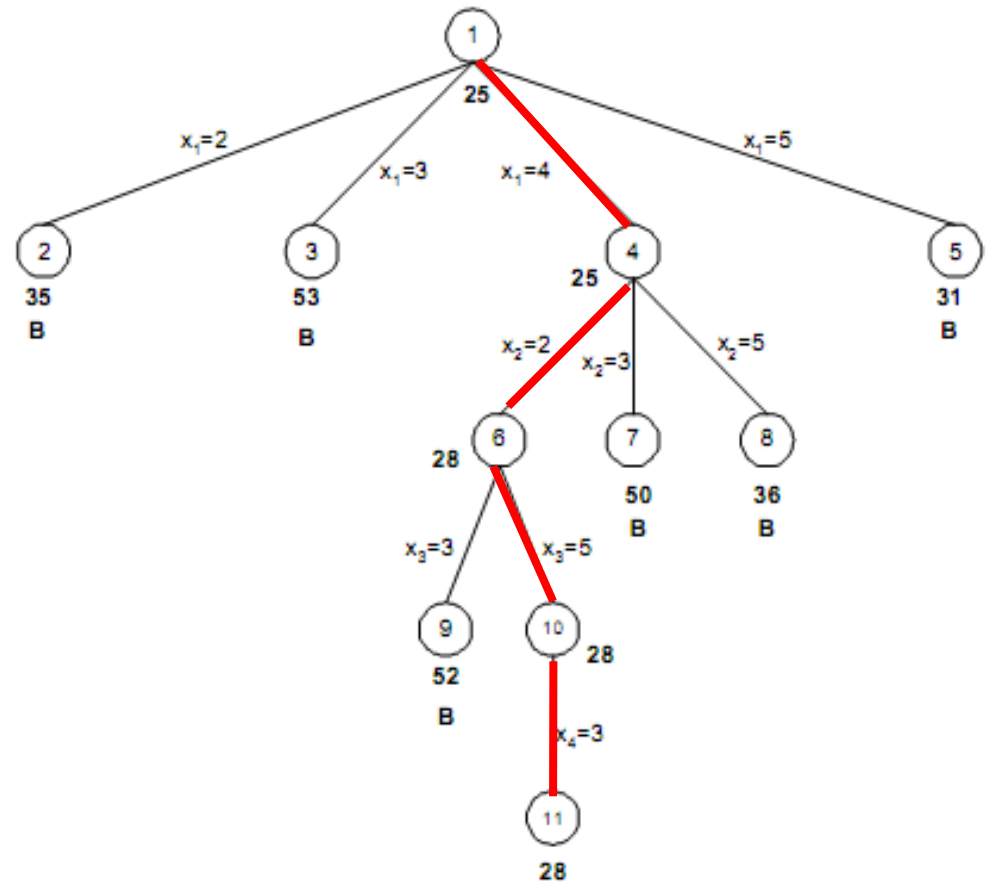
$$\begin{bmatrix}
 \infty & 20 & 30 & 10 & 11 \\
 15 & \infty & 16 & 4 & 2 \\
 3 & 5 & \infty & 2 & 4 \\
 19 & 6 & 18 & \infty & 3 \\
 16 & 4 & 7 & 16 & \infty
 \end{bmatrix}$$

Brute Force:

- $4! = 24$ sirkuit hamilton
- Solusi: 1-4-2-5-3-1
- Bobot: $10 + 6 + 2 + 7 + 3 = 28$

Greedy:

- Solusi: 1-4-5-2-3-1
- Bobot: $10 + 3 + 4 + 16 + 3 = 36$



B&B-TSP dgn Reduced Cost Matrix
 $X_0 = X_5 = 1$

Cost dari Simpul Hidup TSP

1. Matriks ongkos-tereduksi (reduced cost matrix) dari graf
 - Sebuah matriks dikatakan tereduksi jika setiap kolom dan barisnya mengandung paling sedikit satu buah nol dan semua elemen lainnya non-negatif.
 - Batas (*bound*): Jumlah total elemen pengurang dari semua baris dan kolom merupakan batas bawah dari total bobot minimum tur. (hal 159)
2. Bobot minimum tur lengkap

Reduced Cost Matrix: Contoh

R

$$\begin{bmatrix} \infty & 20 & 30 & 10 & 11 \\ 15 & \infty & 16 & 4 & 2 \\ 3 & 5 & \infty & 2 & 4 \\ 19 & 6 & 18 & \infty & 3 \\ 16 & 4 & 7 & 16 & \infty \end{bmatrix}$$



A

$$\begin{bmatrix} \infty & 10 & 17 & 0 & 1 \\ 12 & \infty & 11 & 2 & 0 \\ 0 & 3 & \infty & 0 & 2 \\ 15 & 3 & 12 & \infty & 0 \\ 11 & 0 & 0 & 12 & \infty \end{bmatrix}$$

Setiap kolom dan barisnya mengandung paling sedikit satu buah nol dan semua elemen lainnya non-negatif

Reduced Cost Matrix (3)

$$\begin{bmatrix} \infty & 20 & 30 & 10 & 11 \\ 15 & \infty & 16 & 4 & 2 \\ 3 & 5 & \infty & 2 & 4 \\ 19 & 6 & 18 & \infty & 3 \\ 16 & 4 & 7 & 16 & \infty \end{bmatrix} \begin{matrix} R_1 - 10 \\ R_2 - 2 \\ R_3 - 2 \\ R_4 - 3 \\ R_5 - 4 \end{matrix} \begin{bmatrix} \infty & 10 & 20 & 0 & 1 \\ 13 & \infty & 14 & 2 & 0 \\ 1 & 3 & \infty & 0 & 2 \\ 16 & 3 & 15 & \infty & 0 \\ 12 & 0 & 3 & 12 & \infty \end{bmatrix}$$

$$\begin{bmatrix} \infty & 10 & 20 & 0 & 1 \\ 13 & \infty & 14 & 2 & 0 \\ 1 & 3 & \infty & 0 & 2 \\ 16 & 3 & 15 & \infty & 0 \\ 12 & 0 & 3 & 12 & \infty \end{bmatrix} \begin{matrix} C_1 - 1 \\ C_3 - 3 \end{matrix} \begin{bmatrix} \infty & 10 & 17 & 0 & 1 \\ 12 & \infty & 11 & 2 & 0 \\ 0 & 3 & \infty & 0 & 2 \\ 15 & 3 & 12 & \infty & 0 \\ 11 & 0 & 0 & 12 & \infty \end{bmatrix} = A$$

B&B-TSP dgn Reduced Cost Matrix

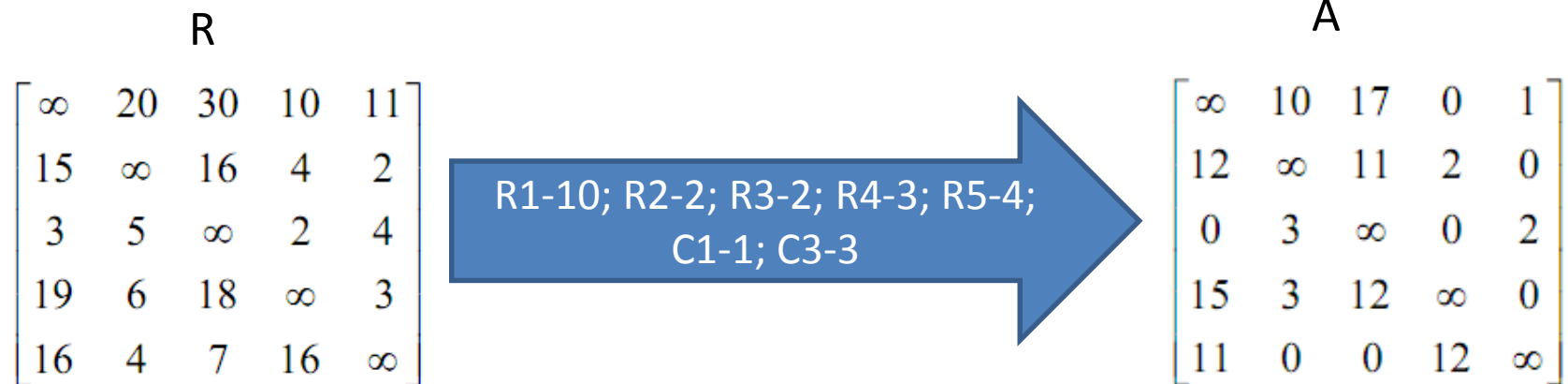
- Misalkan:
 - A : matriks tereduksi untuk simpul R .
 - S : anak dari simpul R sehingga sisi (R, S) pada pohon ruang status berkoresponden dengan sisi (i, j) pada perjalanan.
- Jika S bukan simpul daun, maka matriks bobot tereduksi untuk simpul S dapat dihitung sebagai berikut:
 - (a) ubah semua nilai pada baris i dan kolom j menjadi ∞ . Ini untuk mencegah agar tidak ada lintasan yang keluar dari simpul i atau masuk pada simpul j ;
 - (b) ubah $A(j, 1)$ menjadi ∞ . Ini untuk mencegah penggunaan sisi $(j, 1)$;
 - (c) reduksi kembali semua baris dan kolom pada matriks A kecuali untuk elemen ∞ .
 - Jika r adalah total semua pengurang, maka nilai batas untuk simpul S adalah:
$$\hat{c}(S) = \hat{c}(R) + A(i, j) + r$$
 - Hasil reduksi ini menghasilkan matriks B .

B&B-TSP dgn Reduced Cost Matrix (1)

- Misalkan:

A: matriks tereduksi untuk simpul R.

Simpul awal (R) = 1



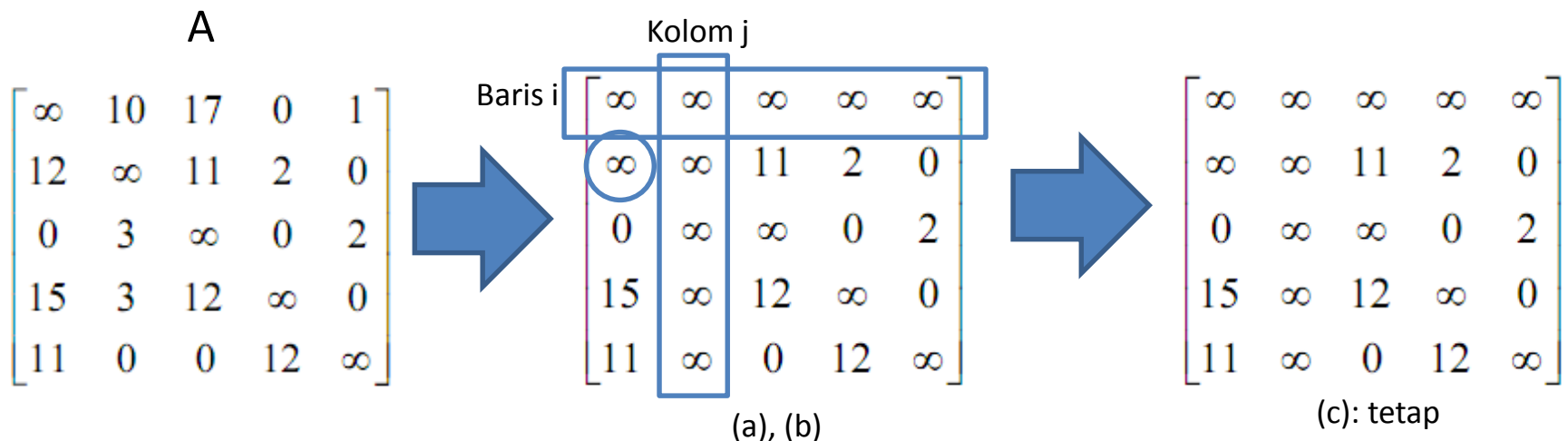
S: anak dari simpul R sehingga sisi (R, S) pada pohon ruang status berkoresponden dengan sisi (i, j) pada perjalanan.

$S \in \{2,3,4,5\}$

B&B-TSP dgn Reduced Cost Matrix (2)

- A: matriks tereduksi R; S: anak dari simpul R
- Jika S bukan simpul daun, maka matriks bobot tereduksi untuk simpul S dapat dihitung sebagai berikut (dari slide 32):
 - (a) ubah semua nilai pada baris i dan kolom j menjadi ∞ . Ini untuk mencegah agar tidak ada lintasan yang keluar dari simpul i atau masuk pada simpul j;
 - (b) ubah $A(j, 1)$ menjadi ∞ . Ini untuk mencegah penggunaan sisi (j, 1)
 - (c) reduksi kembali semua baris dan kolom pada matriks A kecuali untuk elemen ∞ .

Contoh: R=1; S=2 (bukan daun)



$$\begin{bmatrix} \infty & 20 & 30 & 10 & 11 \\ 15 & \infty & 16 & 4 & 2 \\ 3 & 5 & \infty & 2 & 4 \\ 19 & 6 & 18 & \infty & 3 \\ 16 & 4 & 7 & 16 & \infty \end{bmatrix}$$

R=1

$$\begin{bmatrix} \infty & 10 & 17 & 0 & 1 \\ 12 & \infty & 11 & 2 & 0 \\ 0 & 3 & \infty & 0 & 2 \\ 15 & 3 & 12 & \infty & 0 \\ 11 & 0 & 0 & 12 & \infty \end{bmatrix}$$

A

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 12 & \infty & 11 & \infty & 0 \\ 0 & 3 & \infty & \infty & 2 \\ \infty & 3 & 12 & \infty & 0 \\ 11 & 0 & 0 & \infty & \infty \end{bmatrix}$$

R=4

Sisi (1,4) yang sedang diperiksa, maka

$$\hat{c}(4) = \hat{c}(1) + A(1,4) + r = 25 + 0 + 0 = 25$$

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 12 & \infty & 11 & \infty & 0 \\ 0 & 3 & \infty & \infty & 2 \\ \infty & 3 & 12 & \infty & 0 \\ 11 & 0 & 0 & \infty & \infty \end{bmatrix}$$

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 11 & \infty & 0 \\ 0 & \infty & \infty & \infty & 2 \\ \infty & \infty & \infty & \infty & \infty \\ 11 & \infty & 0 & \infty & \infty \end{bmatrix}$$

Sisi (4,2) yang sedang diperiksa, maka:

$$\hat{c}(6) = \hat{c}(4) + B(4,2) + r = 25 + 3 + 0 = 28$$

Taksiran Cost dgn Reduced Cost Matrix

$$\hat{c}(S) = \hat{c}(R) + A(i, j) + r$$

$\hat{c}(S)$:

- (a) bobot perjalanan dari akar ke S (jika S daun)
- (b) Bobot perjalanan minimum yang melalui simpul S (jika S bukan daun)

$$\hat{c}(\text{akar}) = r$$

$\hat{c}(S)$ = bobot perjalanan minimum yang melalui simpul S (simpul di pohon ruang status)

$\hat{c}(R)$ = bobot perjalanan minimum yang melalui simpul R , yang dalam hal ini R adalah orangtua dari S .

$A(i, j)$ = bobot sisi (i, j) pada graf G yang berkoresponden dengan sisi (R, S) pada pohon ruang status.

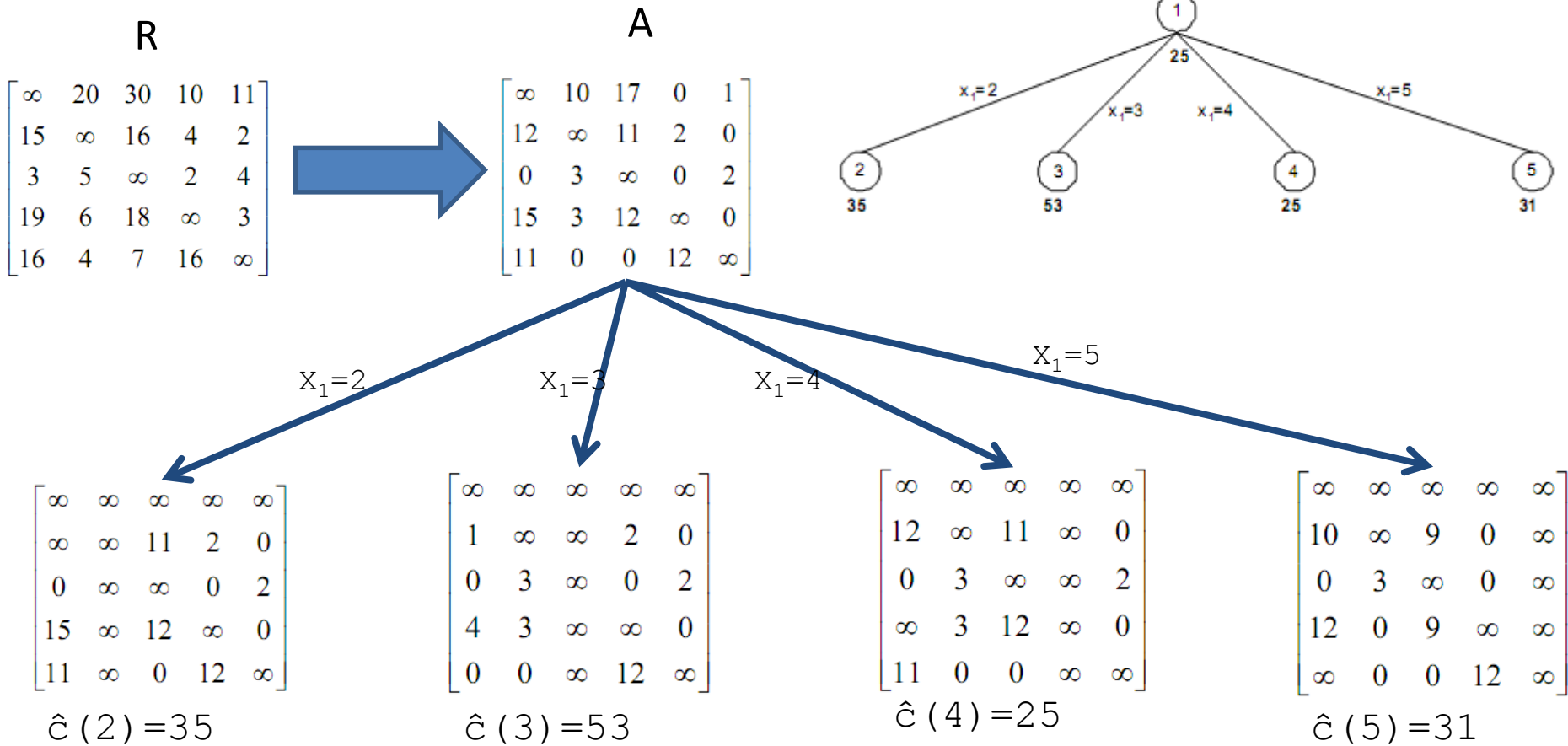
r = jumlah semua pengurang pada proses memperoleh matriks tereduksi untuk simpul S .

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 11 & 2 & 0 \\ 0 & \infty & \infty & 0 & 2 \\ 15 & \infty & 12 & \infty & 0 \\ 11 & \infty & 0 & 12 & \infty \end{bmatrix}$$

$$\hat{c}(2) = \hat{c}(1) + A(1,2) + r = 25 + 10 + 0 = 35$$

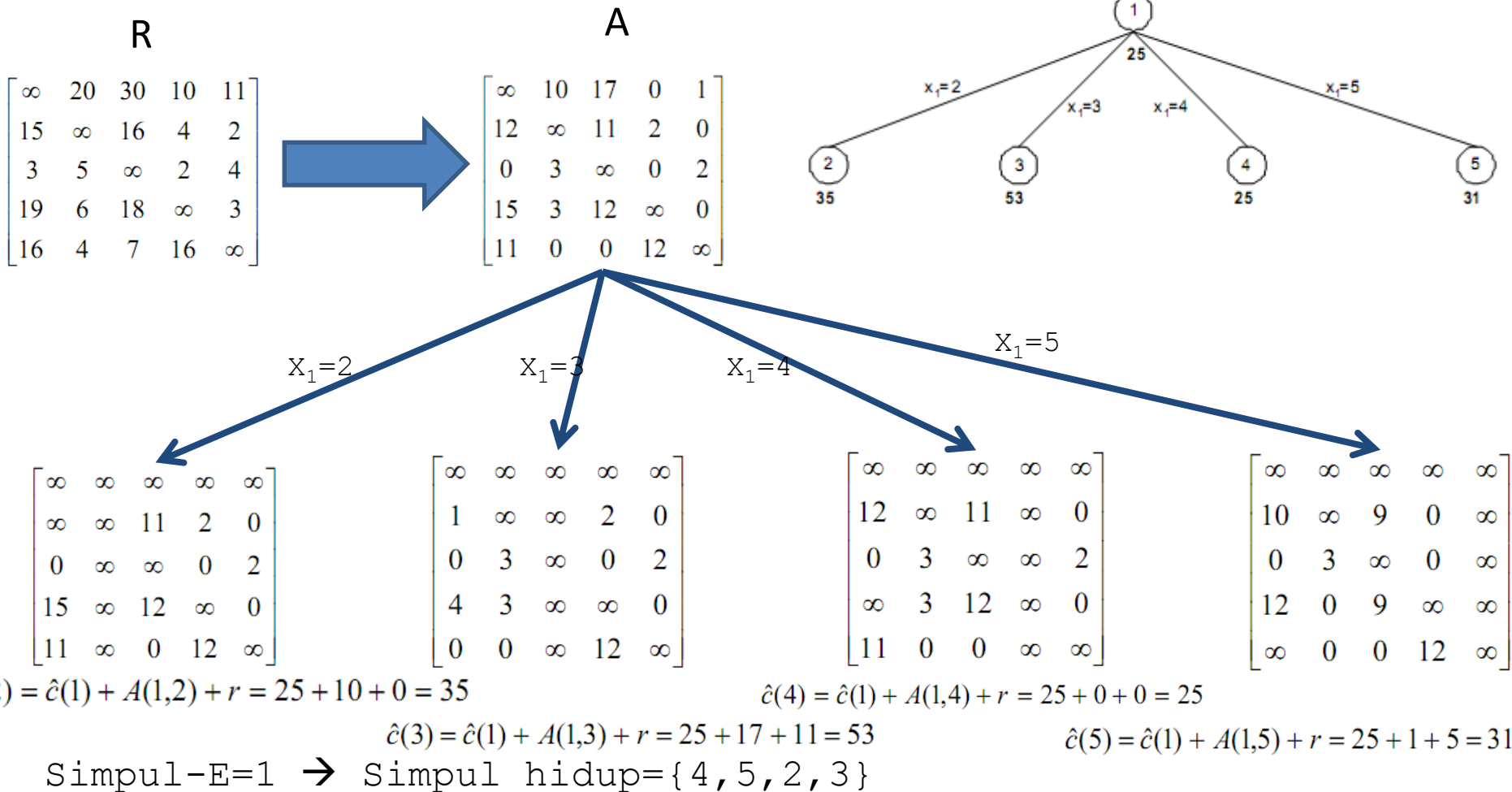
$$\hat{c}(1) = 25$$

B&B-TSP dgn Reduced Cost Matrix



Simpl-E=1 \rightarrow Simpul hidup={4, 5, 2, 3}

B&B-TSP dgn Reduced Cost Matrix



B&B-TSP dgn Reduced Cost Matrix

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 11 & \infty & 0 \\ 0 & \infty & \infty & \infty & 2 \\ \infty & \infty & \infty & \infty & \infty \\ 11 & \infty & 0 & \infty & \infty \end{bmatrix}$$

$$\hat{c}(6) = \hat{c}(4) + B(4,2) + r = 25 + 3 + 0 = 28$$

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 1 & \infty & \infty & \infty & 0 \\ \infty & 1 & \infty & \infty & 0 \\ \infty & \infty & \infty & \infty & \infty \\ 0 & 0 & \infty & \infty & \infty \end{bmatrix}$$

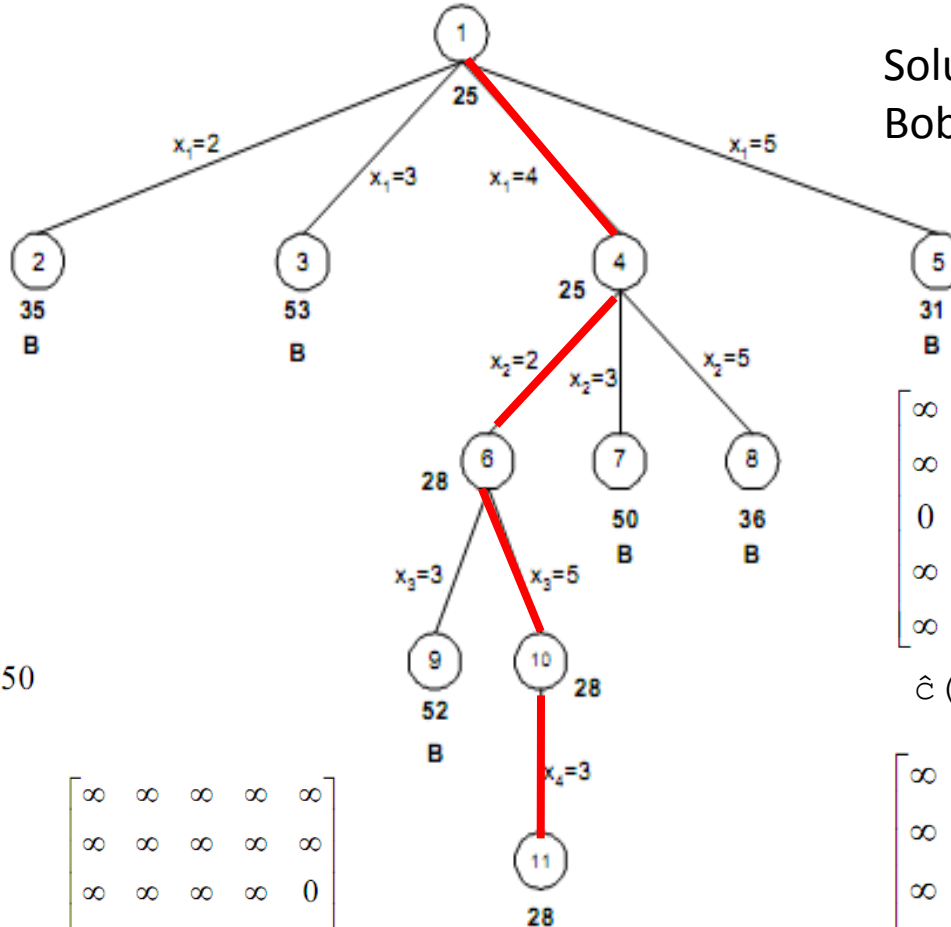
$$\hat{c}(7) = \hat{c}(4) + B(4,3) + r = 25 + 12 + 13 = 50$$

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 1 & \infty & 0 & \infty & \infty \\ 0 & 3 & \infty & \infty & 2 \\ \infty & \infty & \infty & \infty & \infty \\ \infty & 0 & 0 & \infty & \infty \end{bmatrix}$$

$$\hat{c}(8) = \hat{c}(4) + B(4,5) + r = 25 + 0 + 11 = 36$$

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & 0 \\ \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & \infty & \infty \end{bmatrix}$$

$$\hat{c}(9) = \hat{c}(6) + C(2,3) + r = 28 + 11 + 13 = 52$$



Solusi: 1,4,2,5,3,1
 Bobot: 28

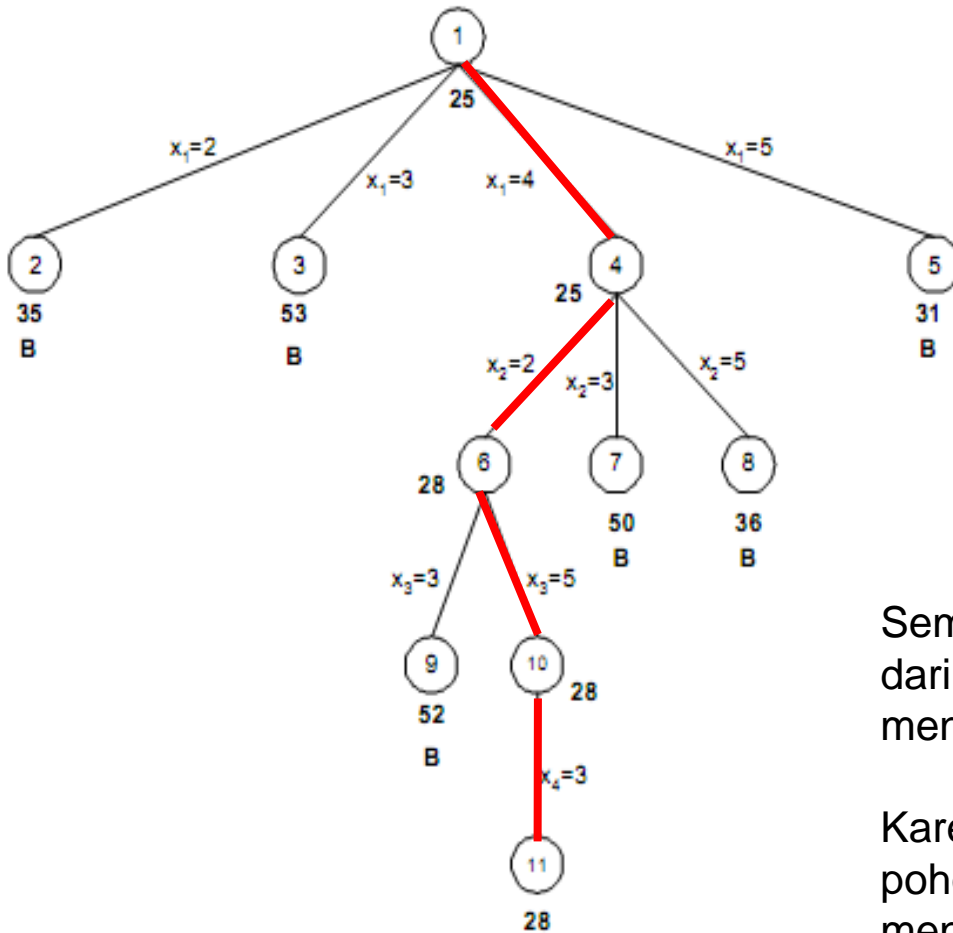
$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 0 & \infty & \infty \end{bmatrix}$$

$$\hat{c}(10) = 28$$

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \end{bmatrix}$$

$$\hat{c}(11) = 28$$

B&B-TSP dgn Reduced Cost Matrix

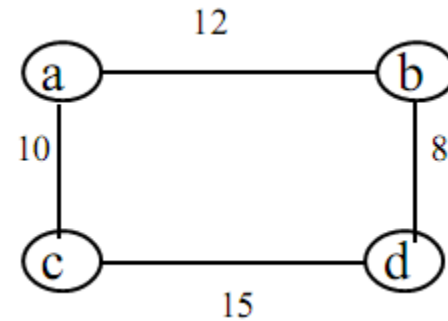
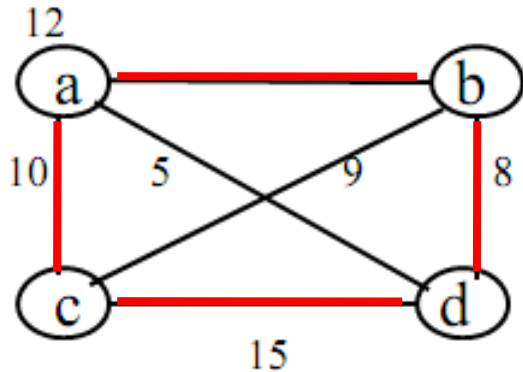


Simpul-E	Simpul Hidup
1	4,5,2,3
4	6,5,2,8,7,3
6	10,5,2,8,7,9,3
10	11,5,2,8,7,9,3
11	daun

Semua simpul hidup yang nilainya lebih besar dari 28 dibunuh (B) karena tidak mungkin lagi menghasilkan perjalanan dengan bobot < 28 .

Karena tidak ada lagi simpul hidup di dalam pohon ruang status, maka $X = (1, 4, 2, 5, 3, 1)$ menjadi solusi persoalan TSP di atas dengan bobot 28.

Bobot Tur Lengkap



Solusi: (a, i_1, i_2, i_3, a)

$$\begin{aligned}
 10 + 15 + 8 + 12 &= 45 \\
 &= 1/2 [(10 + 12) + (10 + 15) + (15 + 8) + (12 + 8)] \\
 &= 1/2 \times 90 \\
 &= 45
 \end{aligned}$$

$$\text{bobot tur lengkap} = 1/2 \sum_{i=1}^n \text{bobot sisi } i_1 + \text{bobot sisi } i_2$$

sisi i_1 dan sisi i_2 adalah dua sisi yang bersisian dengan simpul i di dalam tur lengkap.

B&B-TSP dengan Bobot Tur Lengkap

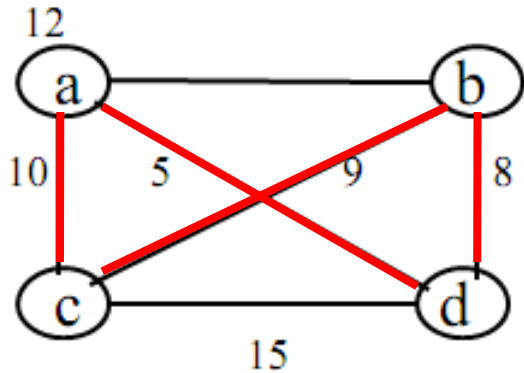
Hasil pengamatan:

$$\begin{aligned} M \equiv \text{cost} &= \text{bobot minimum tur lengkap} \\ &\geq 1/2 \sum \text{bobot sisi } i_1 + \text{bobot sisi } i_2 \end{aligned}$$

Yang dalam hal ini, sisi i_1 dan sisi i_2 adalah sisi yang bersisian dengan simpul i dengan bobot minimum.

M dapat digunakan sebagai fungsi pembatas (*bound*) untuk menghitung cost setiap simpul di dalam pohon

Cost Simpul Akar



Cost untuk simpul akar (simpul 1)

$$\begin{aligned} \text{cost} &\geq 1/2 [(5+10) + (9+8) + (9+10) + (8+5)] \\ &\geq 32 \end{aligned}$$

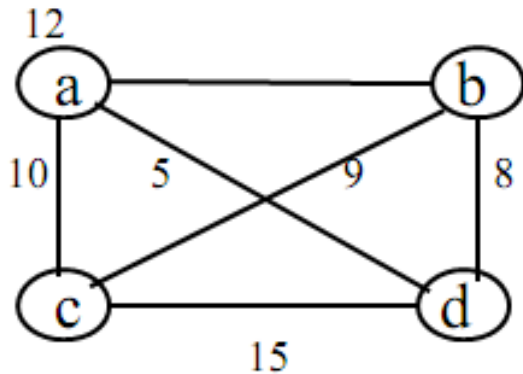
Solusi: (a, i_1, i_2, i_3, a)

Pohon dinamis:

①

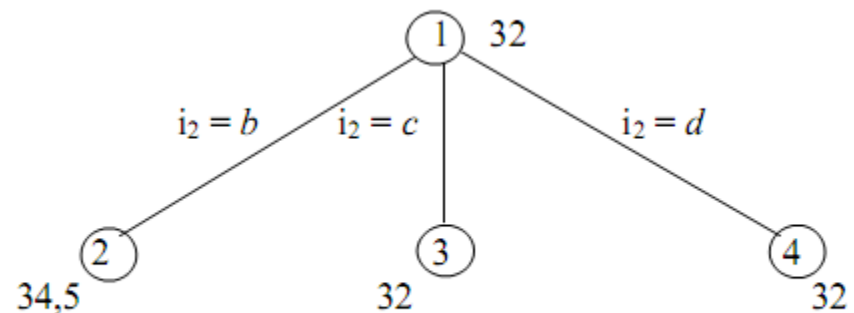
32

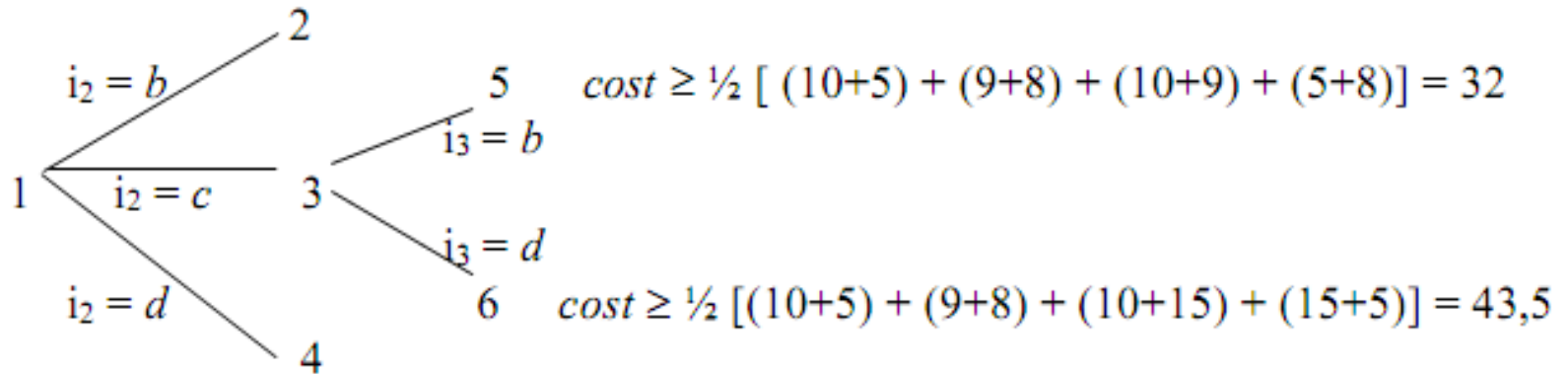
B&B-TSP dengan Bobot Tur Lengkap



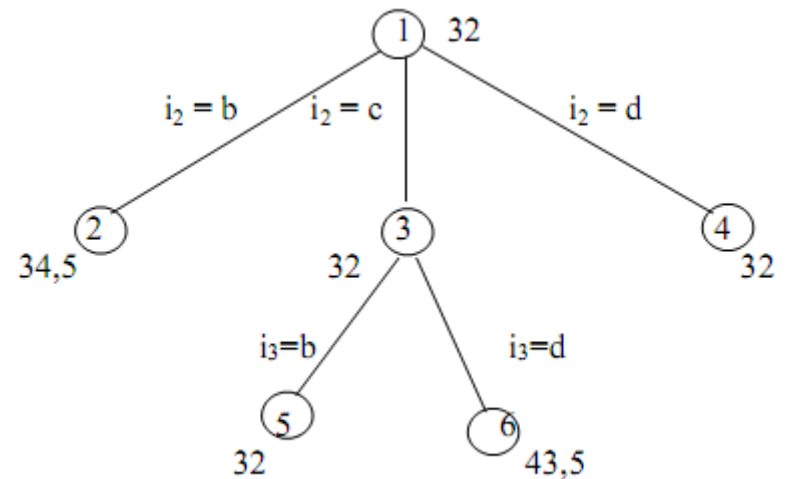
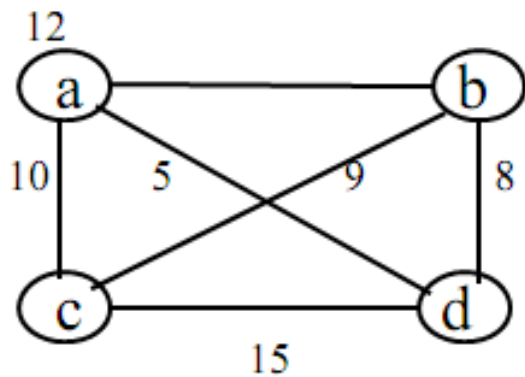
1	$i_2 = b$	2	$cost \geq 1/2 [(12+5) + (12+8) + (9+10) + (8+5)]$ $\geq 34,5$
	$i_2 = c$	3	$cost \geq 1/2 [(10+5) + (9+8) + (10+9) + (8+5)]$ ≥ 32
	$i_2 = d$	4	$cost \geq 1/2 [(5+10) + (9+8) + (10+9) + (8+5)]$ ≥ 32

Untuk $i_2=b$, sisi a-b wajib diambil.

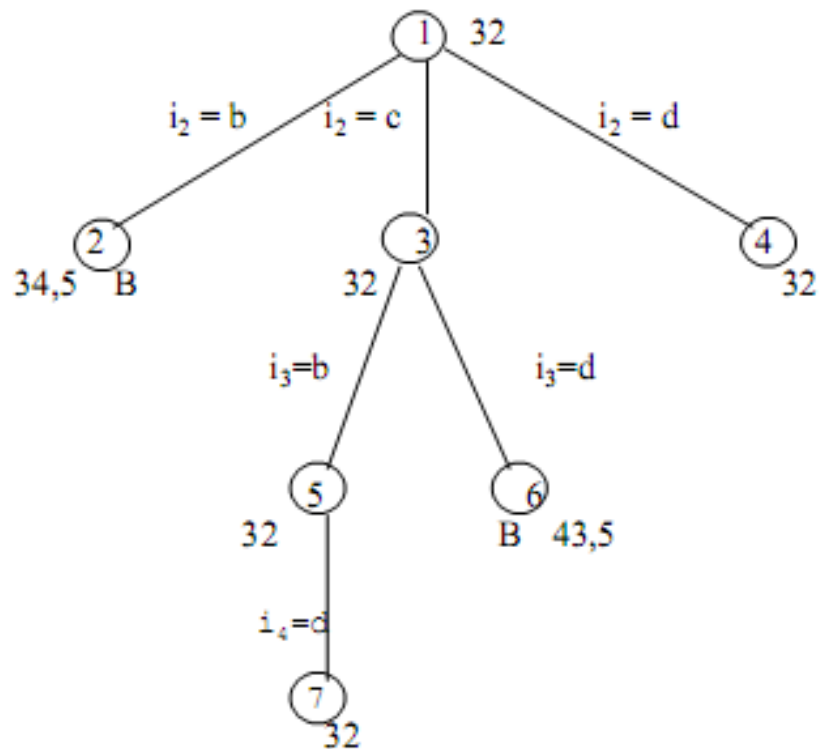




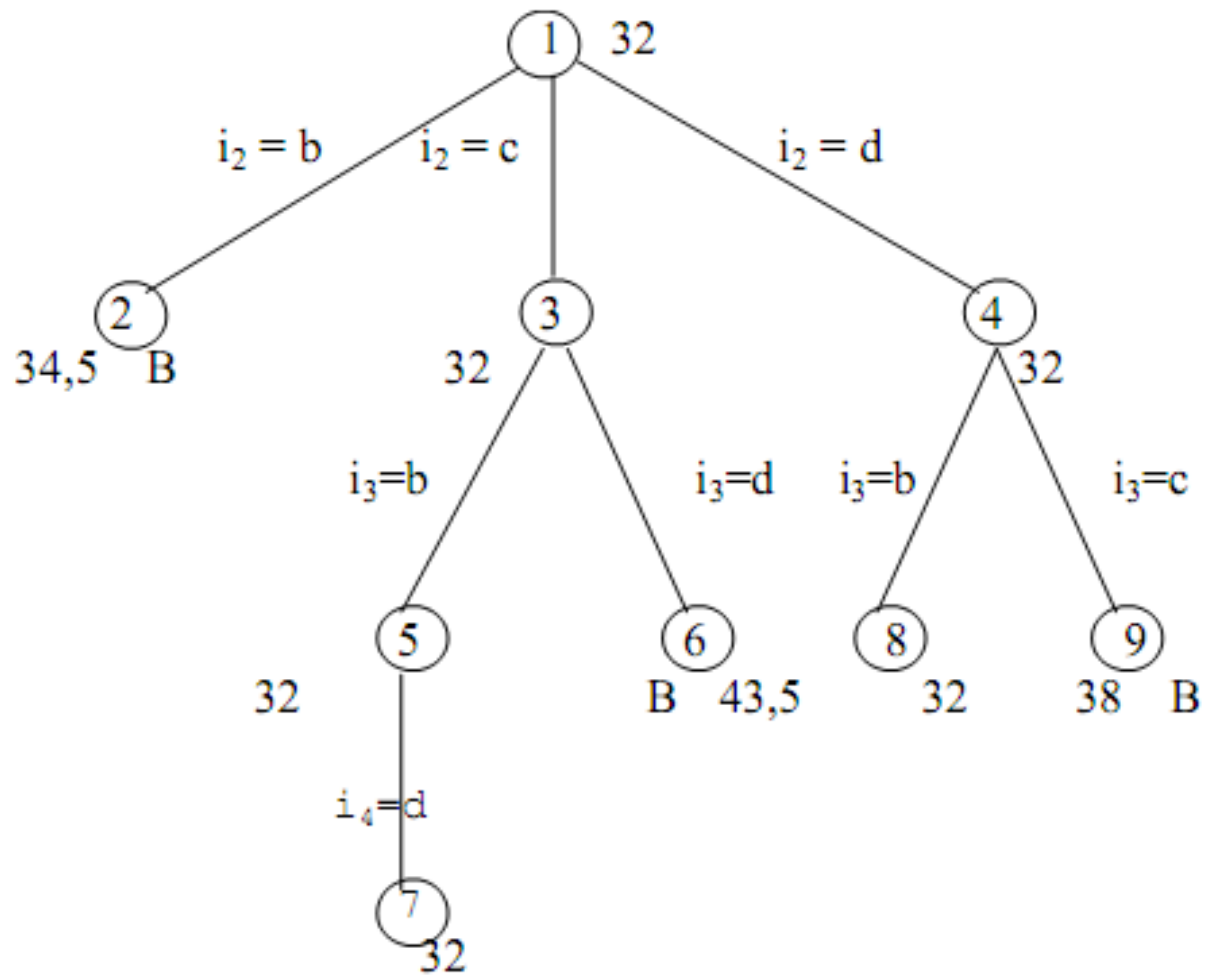
Untuk $i_3=b$, sisi a-c dan c-b wajib diambil.



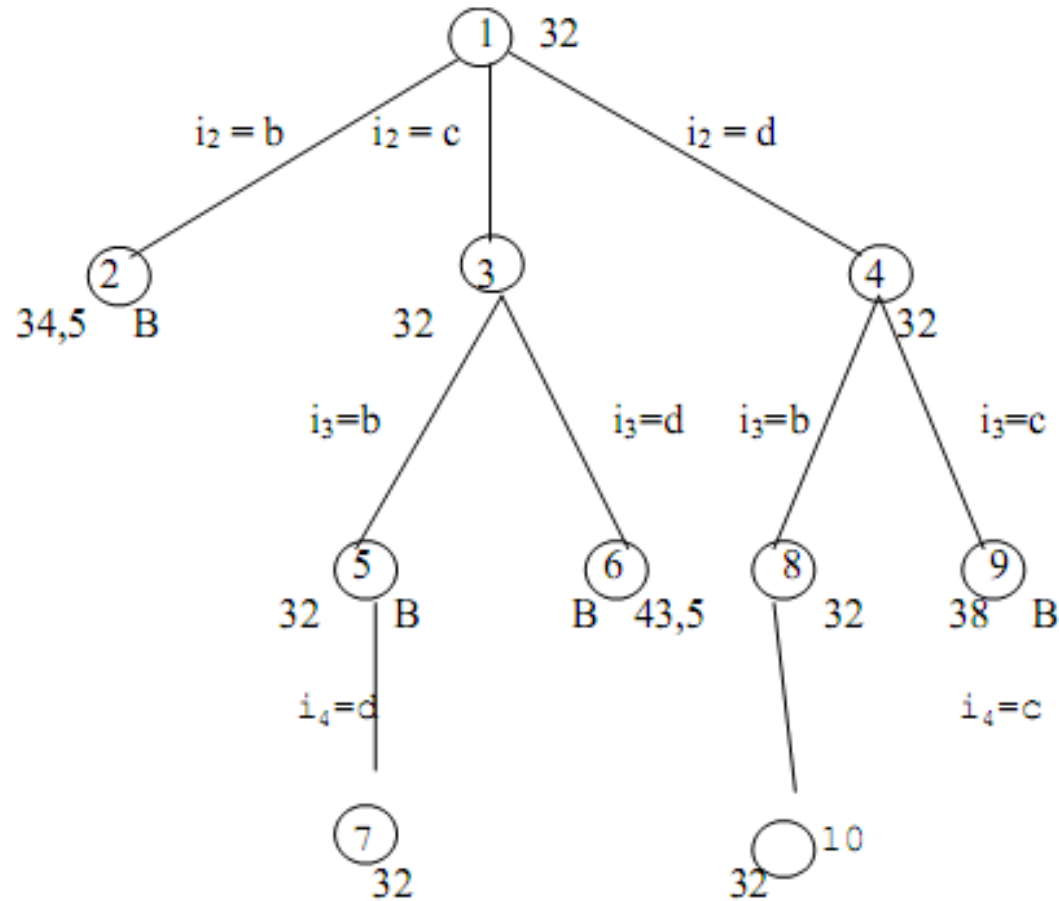
Pohon ruang status yang terbentuk:



Solusi pertama: Tur a, c, b, d, a dengan bobot 32 (*the best solution so far*). Bunuh semua simpul dengan cost > 32. (ditandai dengan B)



$$\text{Cost simpul 8} \geq \frac{1}{2}[(5+10)+(8+9)+(9+10)+(5+8)] = 32$$



$Cost$ simpul 10 $\geq \frac{1}{2}[(5+10)+(9+8)+(9+10)+(5+8)] = 32$

Solusi ke-2: tur a, d, b, c, a dengan bobot 32

Latihan: Persoalan Knapsack

- Kapasitas knapsack : 10
- Dengan B&B:

item	weight	value	$\frac{\text{value}}{\text{weight}}$
1	4	\$40	10
2	7	\$42	6
3	5	\$25	5
4	3	\$12	4

– Bagaimana percabangan pohon?

Petunjuk: setiap *item* → diikuti sertakan atau tidak?

Bagaimana pengurutan *item* pada aras pohon?

Petunjuk: ingat Greedy, bagaimana pengurutan yang memberikan keuntungan optimal?

– Perlu batas bawah, atau batas atas?

Petunjuk: tujuan meminimalkan atau memaksimalkan?

Nilai apa yang dioptimasi?

– Fungsi batas setiap simpul ?

SELAMAT BELAJAR