
MatsuLaB

A*でウキウキ*Nbest!

A*アルゴリズム

Astar algorithm

2013年8月２日発表

形態素解析でN-bestを出したい！

- **Viterbiアルゴリズムを使えば，1bestを効率よく求めることができる.**
- **But…**
  - Viterbiでは「解析結果の上位N件」といったN-bestの解が求められない…

  - A*探索を使う！

基本的にこの論文を読めと書いてある

- **逆向きに単語列を辿る際にA*アルゴリズムを用いると，確率最大の解だけでなく，確率が大きい順番に一つずつ任意個の形態素解析候補を求めるN-best探索(N-best search)を実現できる[31].**
  - [31] M.Nagata, “A Stochastic Japanese Morphological Analyzer Using a Forward-DP Backward A* N-Best Search Algorithm,” COLING 94, pp.201-207 (1994).

もっと詳しいジャーナル版を読んでみた

Information Processing Society of Japan

Vol. 40 No. 9 情報処理学会論文誌 Sep. 1999

統計的言語モデルと N-best 探索を用いた日本語形態素解析法

永田昌明†

本論文では，統計的言語モデルと N-best 探索アルゴリズムを用いた新しい日本語形態素解析法を提案する．本方法は，未知語の確率モデルを持つことにより任意の日本語文を高精度に解析し，確率が大きい順に任意個の形態素解析候補を求められる．EDR コーパスの部分集合（約 19 万文，約 470 万語）を用いて言語モデルの学習を行い，オープンテキスト 100 文に対してテストを行ったところ，単語分割の精度は第 1 候補で再現率 94.6%適合率 93.5%，上位五候補で再現率 97.8%適合率 88.3%であった．

A Japanese Morphological Analysis Method Using a Statistical Language Model and an N-best Search Algorithm

MASAAKI NAGATA†

We present a novel method for Japanese morphological analysis which uses a statistical language model and an N-best search algorithm. It has a probabilistic model for unknown words to parse unrestricted Japanese sentences accurately and it can get N-best morphological analysis hypotheses. When the statistical Japanese morphological analyzer was trained on the subset of the EDR corpus (about 190 thousand sentences, 4.7 million words) and tested on 100 sentences of open text, it achieved 94.6% recall and 93.5% precision for the top candidate, and 97.8% recall and 88.3% precision for the top five candidates.

```
1   open ← {w_{n+1}}
2   closed ← φ
3   g(w_{n+1}) ← 0
4   f(w_{n+1}) ← h(w_{n+1})
5   while open ≠ φ do
6     w_i ← arg min_{w∈open} f(w)
7     if w_i = w_0 then return SUCCESS
8     open ← open − {w_i}
9     closed ← closed ∪ {w_i}
10    for w_{i−1} ∈ T_start(w_i) do
11      g'(w_i, w_{i−1}) ← |logP(w_i|w_{i−1})| + g(w_i)
12      f'(w_i, w_{i−1}) ← g'(w_i, w_{i−1}) + h(w_{i−1})
13      if w_{i−1} ∈ open then
14        if f'(w_i, w_{i−1}) < f(w_{i−1}) then
15          g(w_{i−1}) ← g'(w_i, w_{i−1})
16          f(w_{i−1}) ← f'(w_i, w_{i−1})
17          q(w_{i−1}) ← w_i
18        endif
19      else if w_{i−1} ∈ closed then
20        if f'(w_i, w_{i−1}) < f(w_{i−1}) then
21          g(w_{i−1}) ← g'(w_i, w_{i−1})
22          f(w_{i−1}) ← f'(w_i, w_{i−1})
23          q(w_{i−1}) ← w_i
24          closed ← closed − {w_{i−1}}
25          open ← open ∪ {w_{i−1}}
26        endif
27      else
28        g(w_{i−1}) ← g'(w_i, w_{i−1})
29        f(w_{i−1}) ← f'(w_i, w_{i−1})
30        q(w_{i−1}) ← w_i
31        open ← open ∪ {w_{i−1}}
32      endif
33    end
34  end
35  return FAILURE
```

図3　A* アルゴリズムを用いた後向き探索アルゴリズム
Fig. 3　Backward search algorithm.

疑似コードが載ってた！

- **でもコレ，ただのA*…**
  - これだとN-best出せない…
  - 地の文にもちゃんとN-bestの出し方は載ってない…
- **ネット上の解説：**
  - 「これを何回も繰り返せばいい」

- **微妙に間違い．図のように単にラティスをいじっているだけではN-bestを出せません．**

大きなヒント発見！

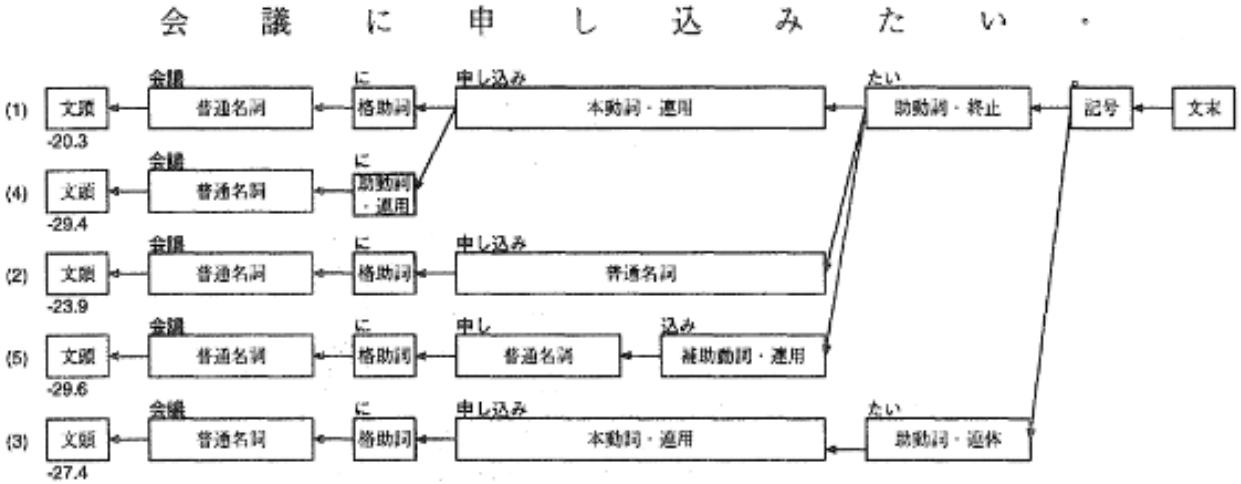


図 4　A^* アルゴリズムを用いた後向き探索
Fig. 4　Backward search using A^* algorithm.

木探索か！
…というか載ってるの図だけか！！

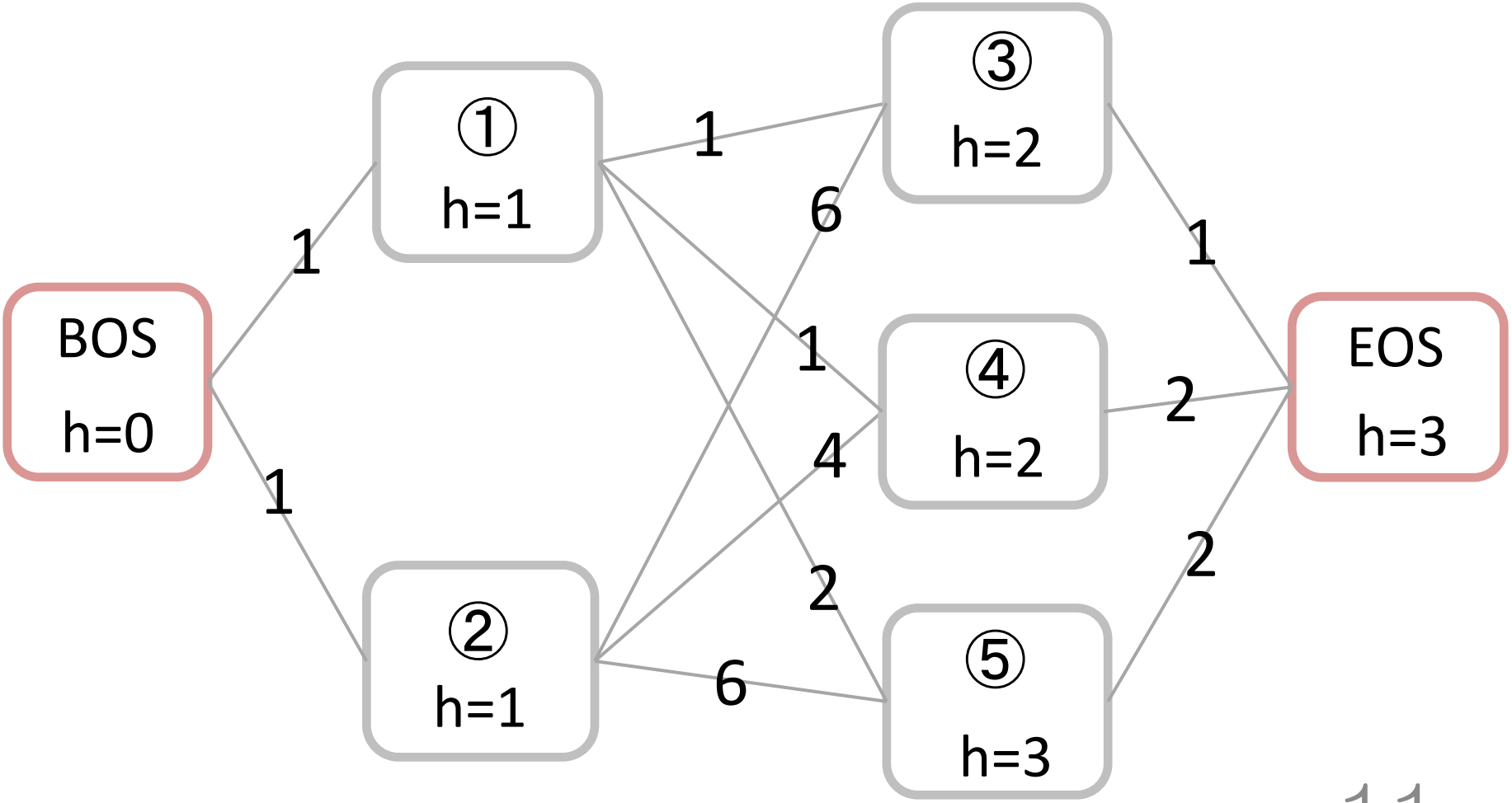
A*でN-bestを出す方法

- Viterbiの前向き探索を実施.
- 前向き探索で各ノードに付けた「BOSまでの最小コスト」をゴールまでの距離hとして使用.
- 木探索的にA*サーチを実施.

Notation

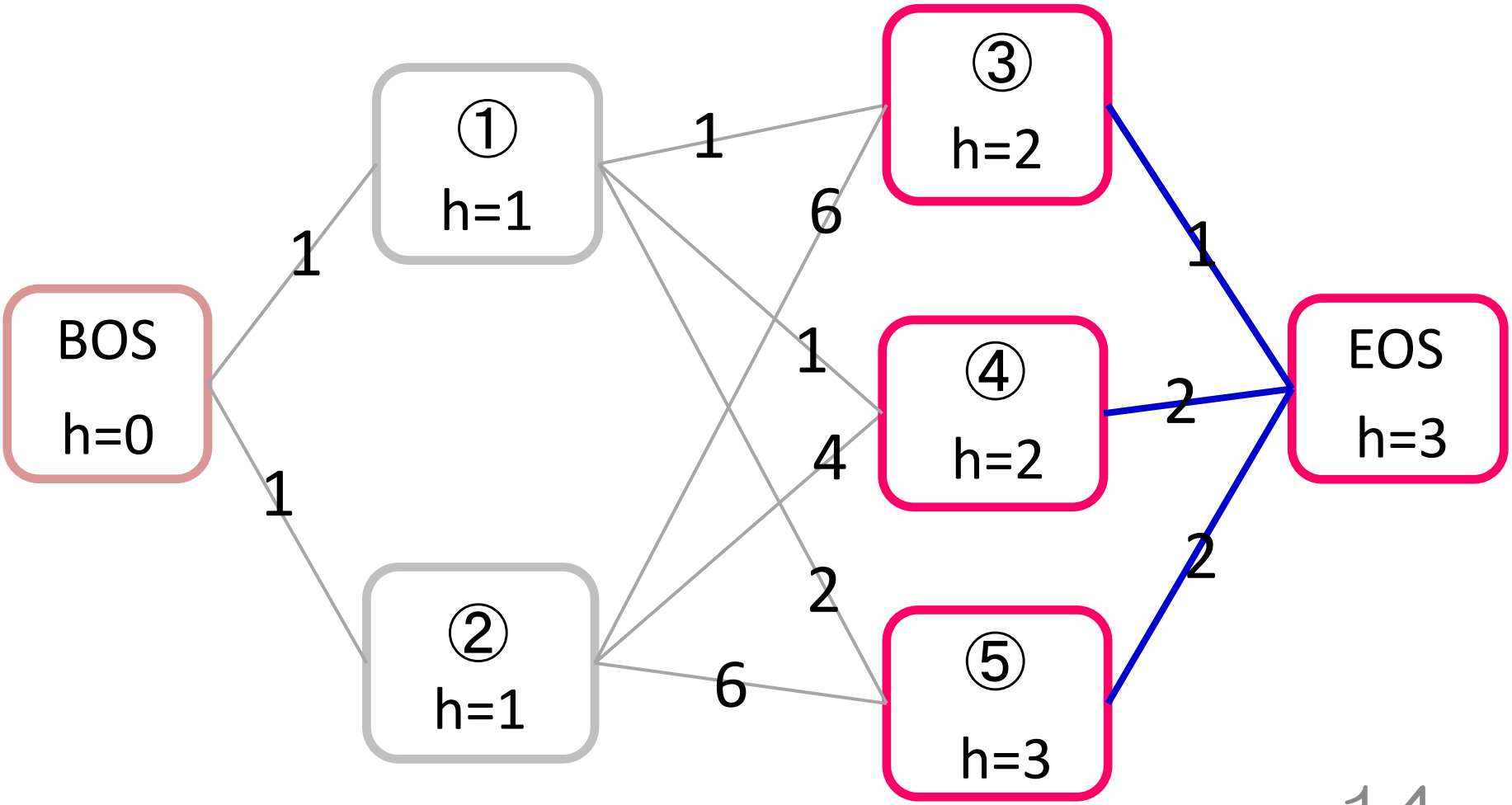
- h：当該ノードからBOSまでのコスト最小の経路のコスト.
- g：EOSから当該ノードに至るまでにかかっているコスト（最小でない場合もある）.
- f = h + g

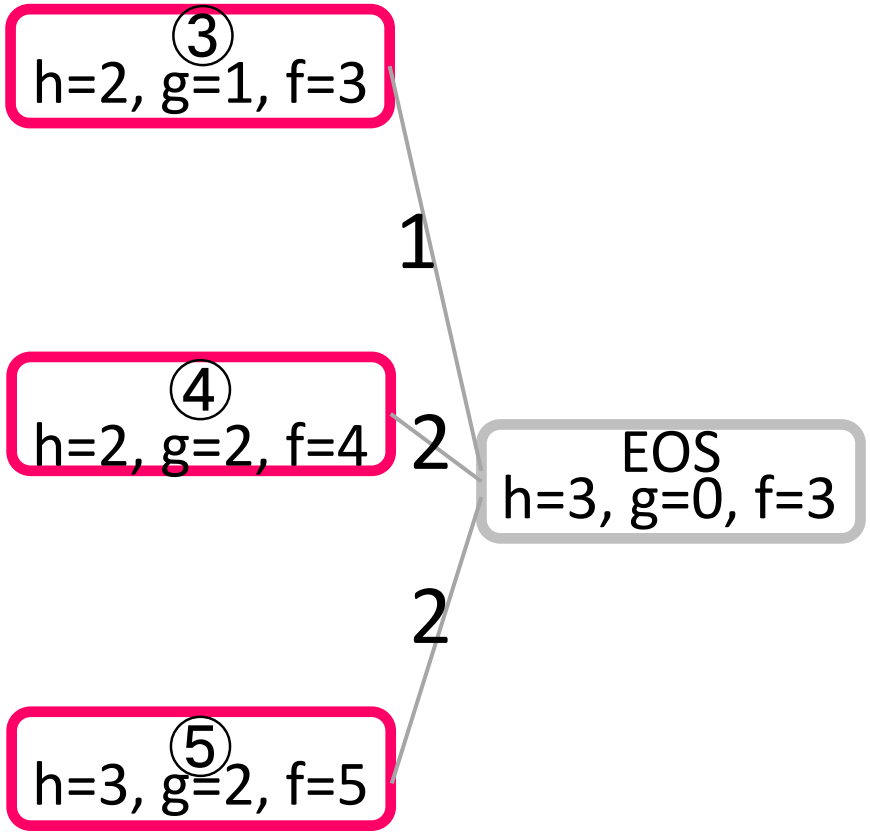
やってみよう

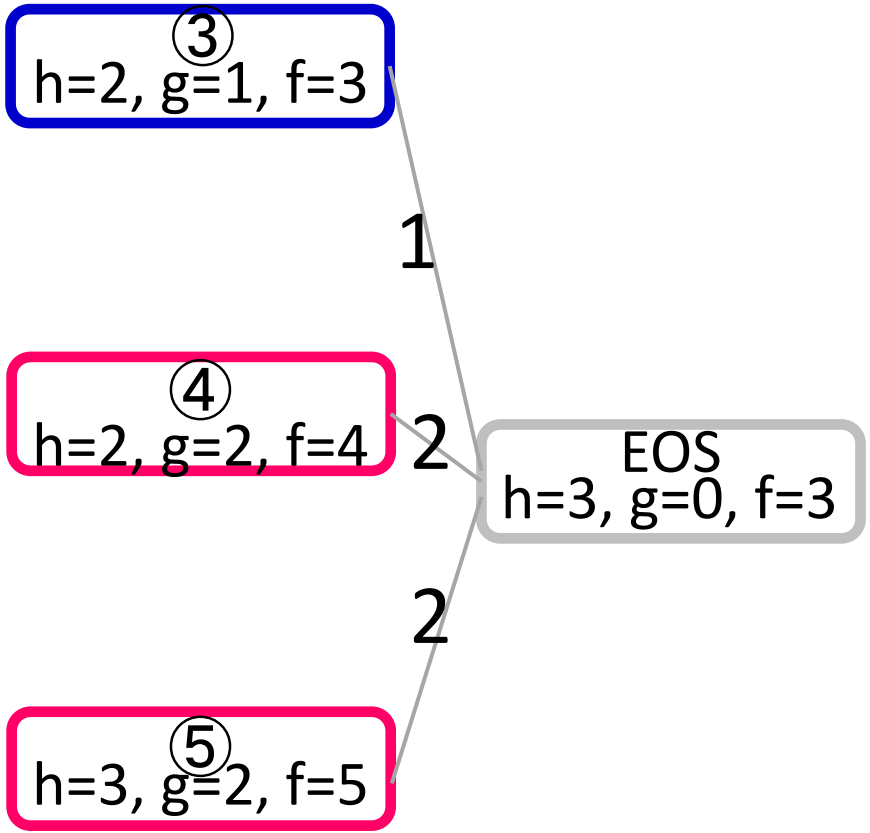

BOS
h=0
①
h=1
②
h=1
③
h=2
④
h=2
⑤
h=3
EOS
h=3
1
1
1
6
1
4
2
6
1
2
2

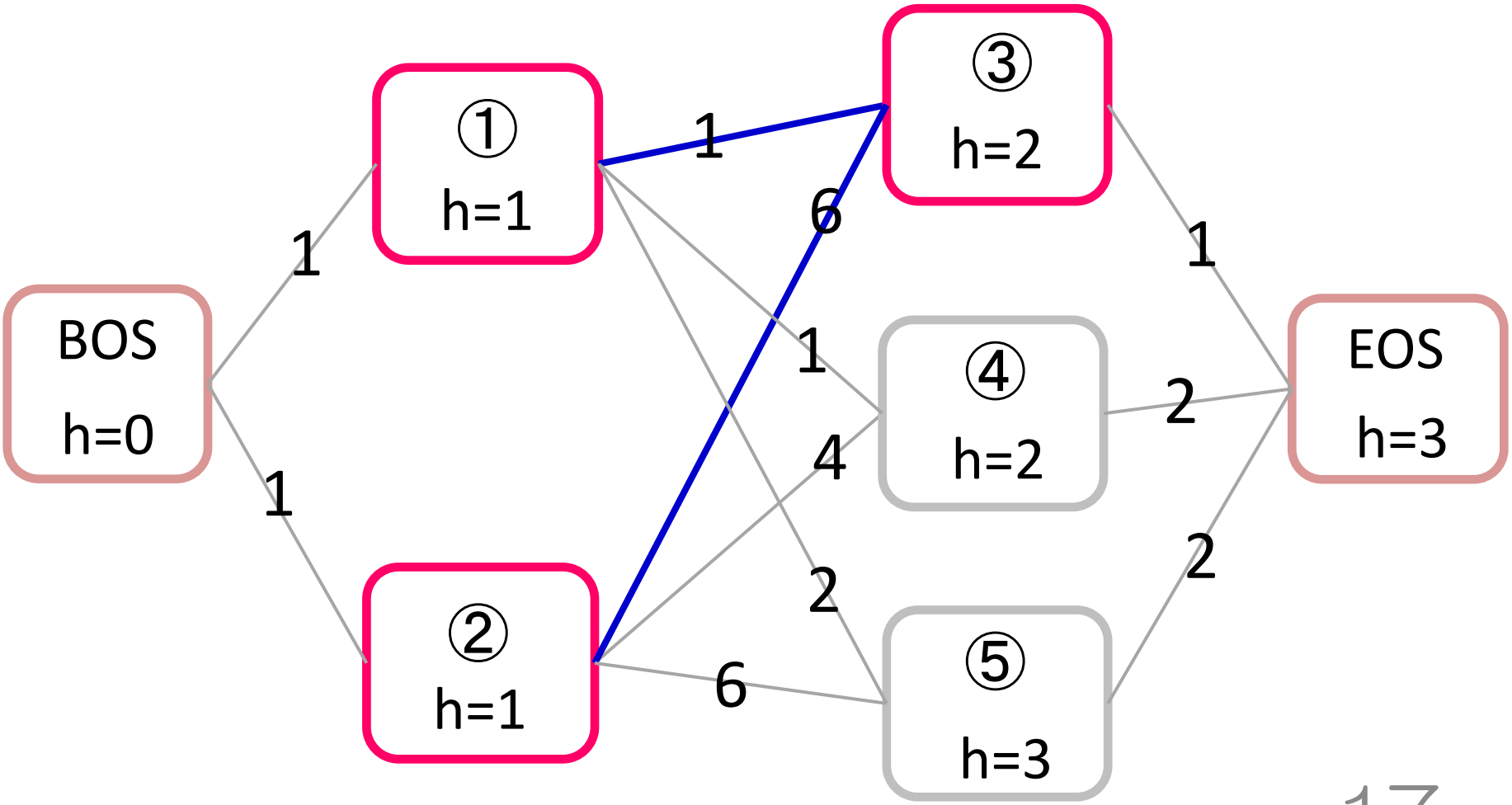
EOS
h=3, g=0, f=3

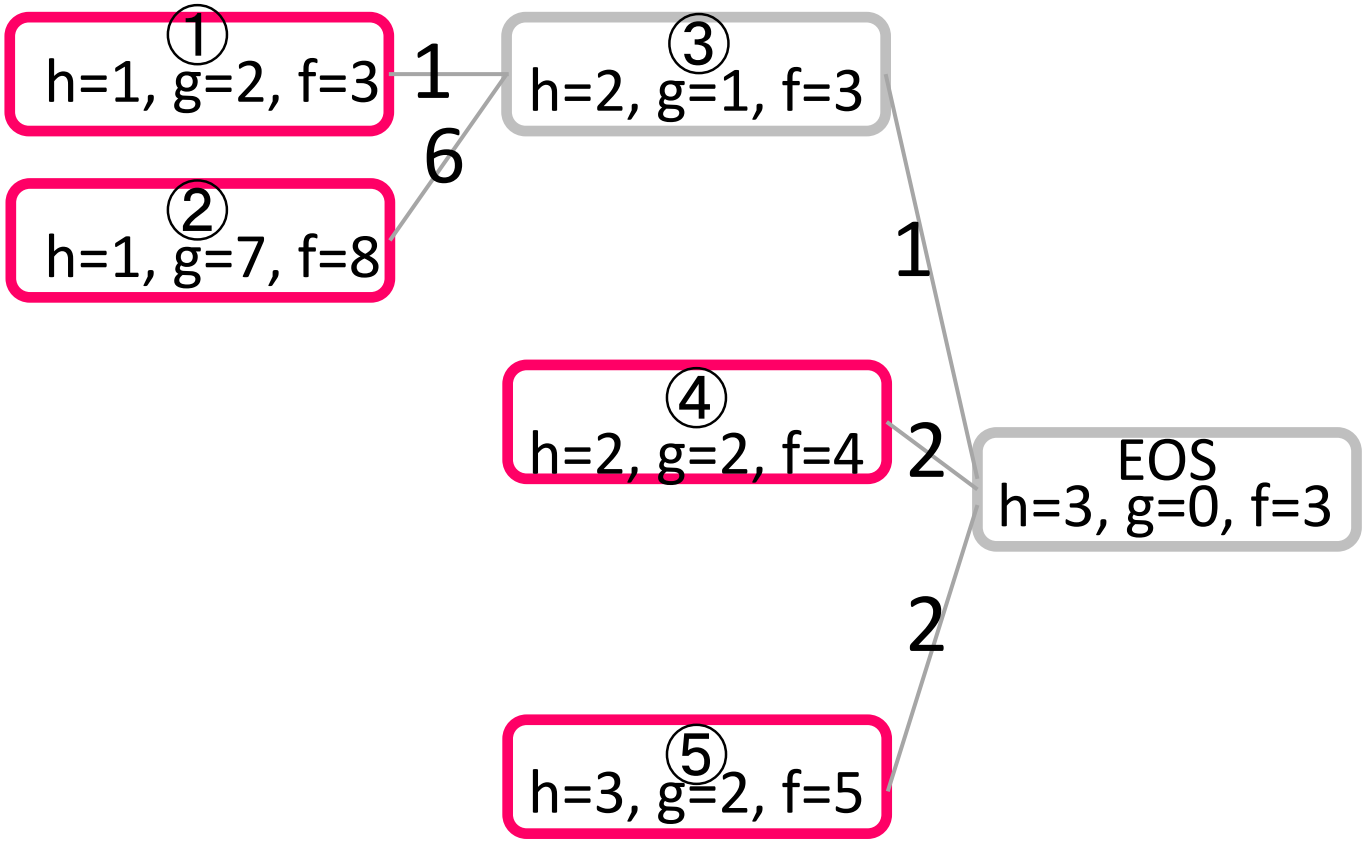
EOS
h=3, g=0, f=3

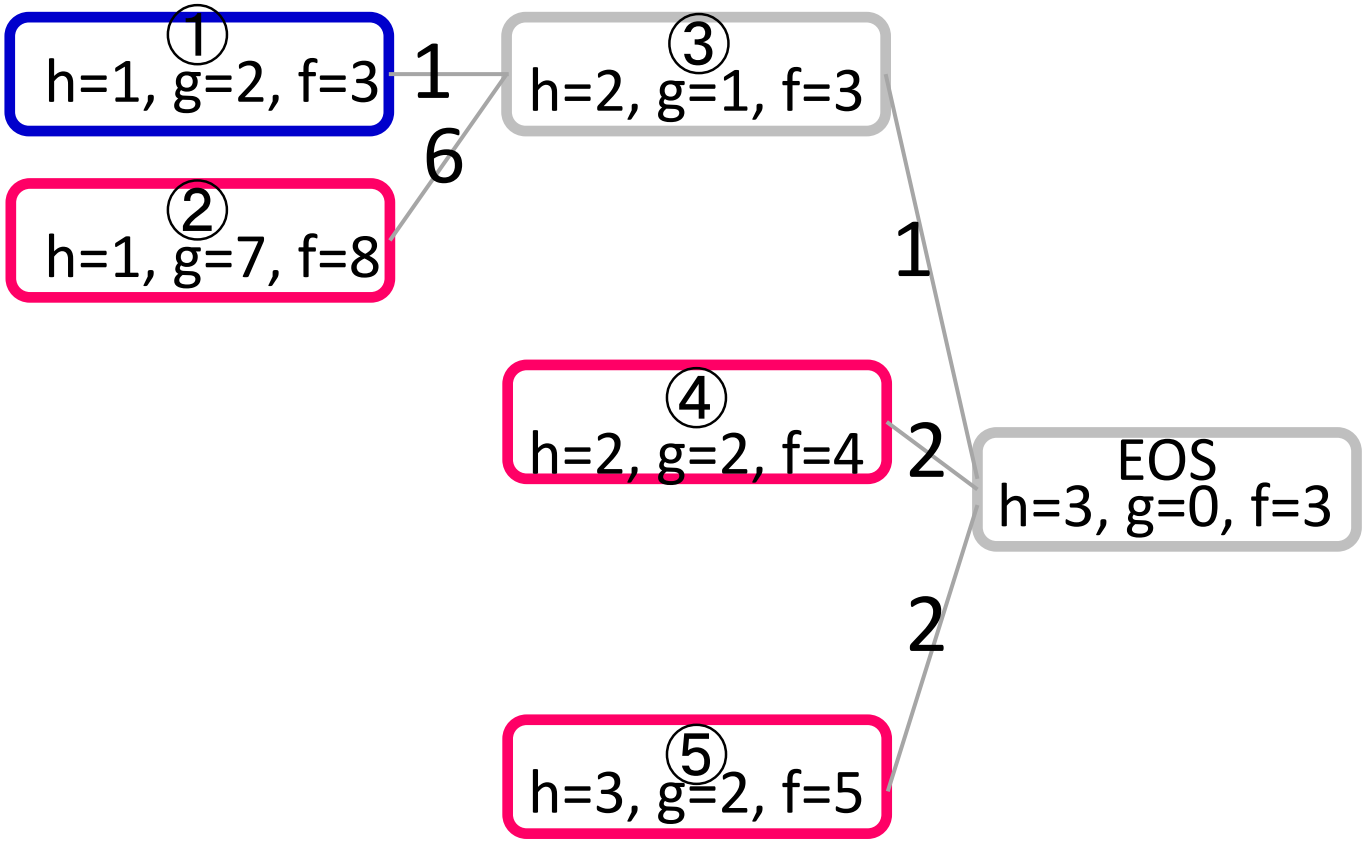

BOS
h=0
①
h=1
②
h=1
③
h=2
④
h=2
⑤
h=3
EOS
h=3
1
1
1
6
1
4
2
6
1
2
2

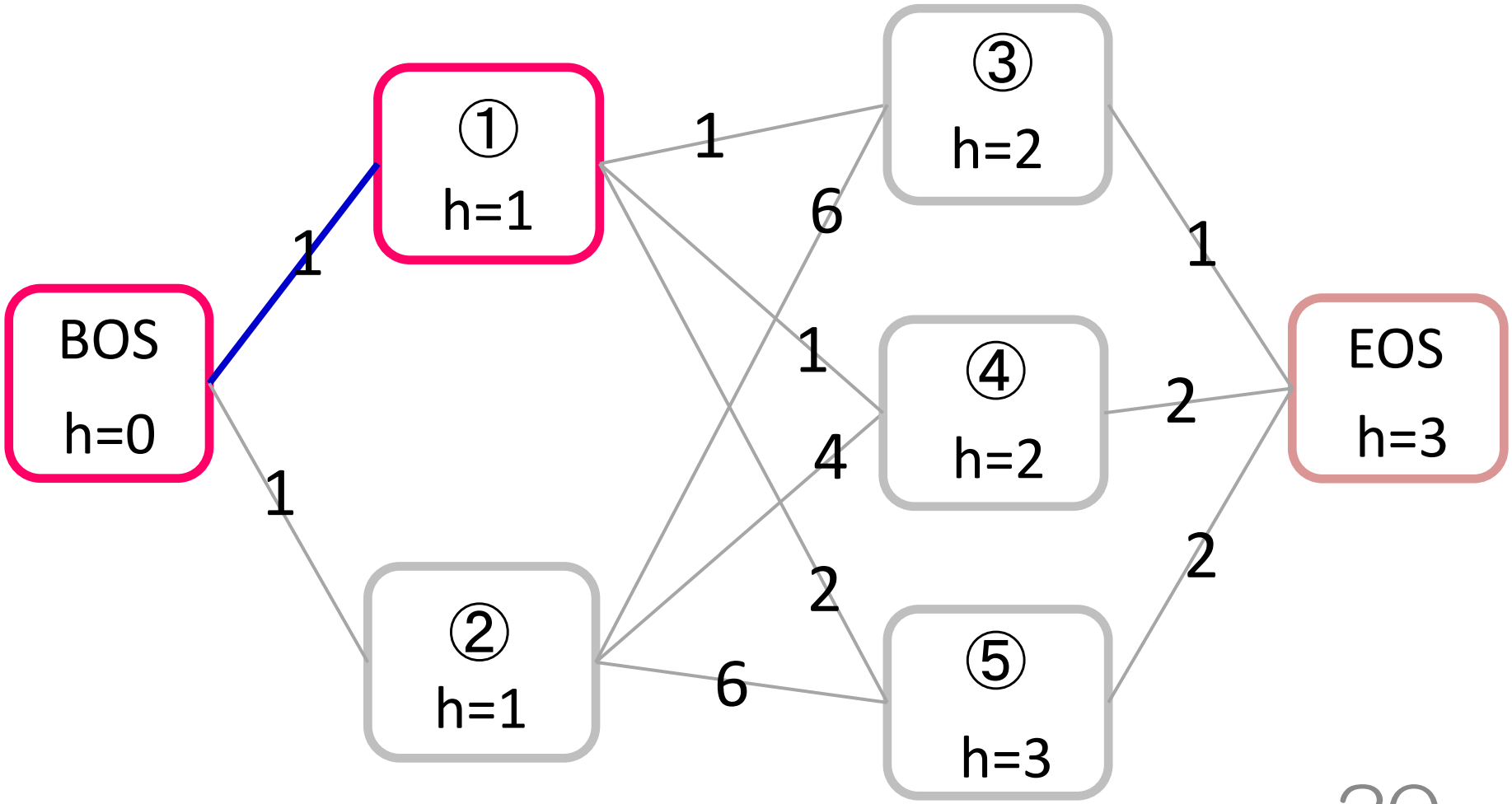

③
h=2, g=1, f=3
1
④
h=2, g=2, f=4
2
EOS
h=3, g=0, f=3
2
⑤
h=3, g=2, f=5

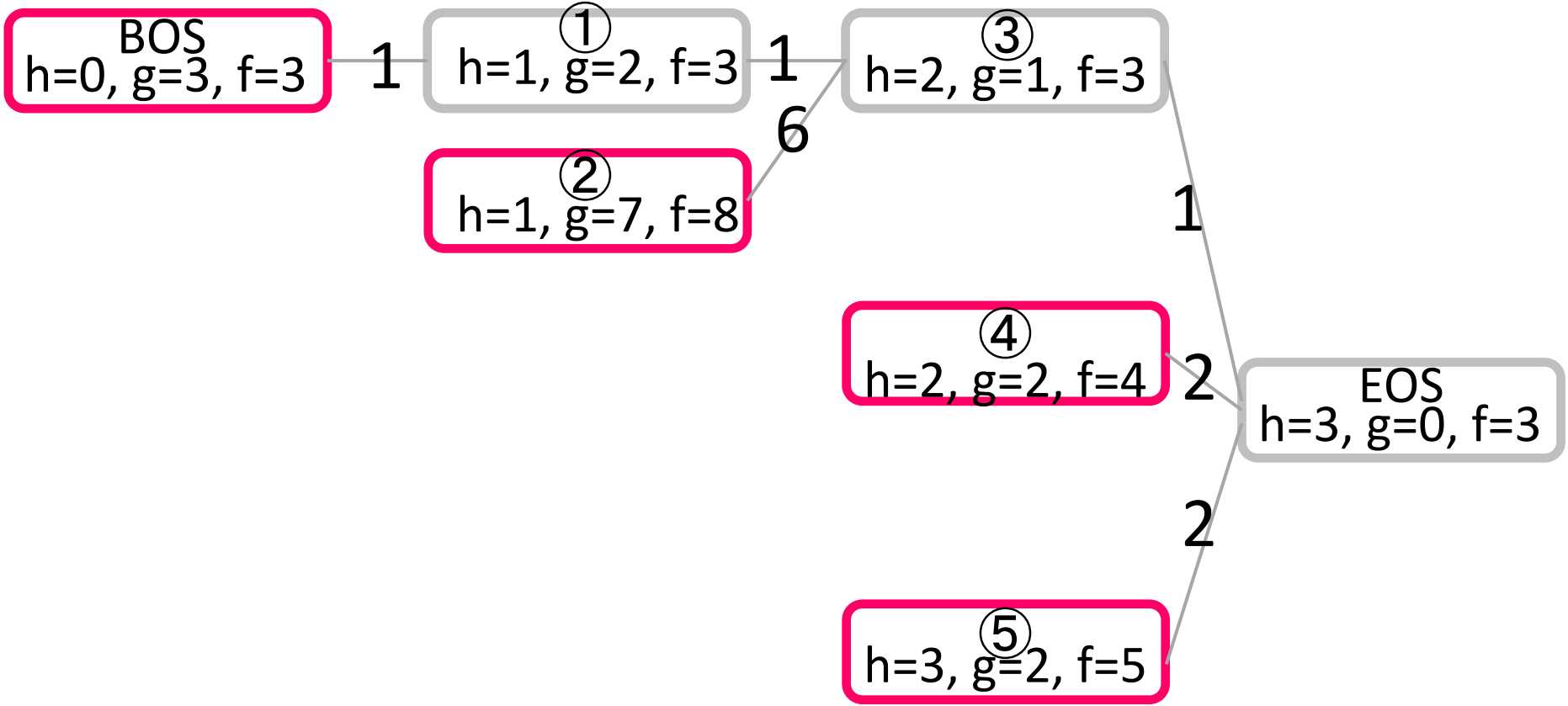

③
h=2, g=1, f=3
1
④
h=2, g=2, f=4
2
EOS
h=3, g=0, f=3
2
⑤
h=3, g=2, f=5

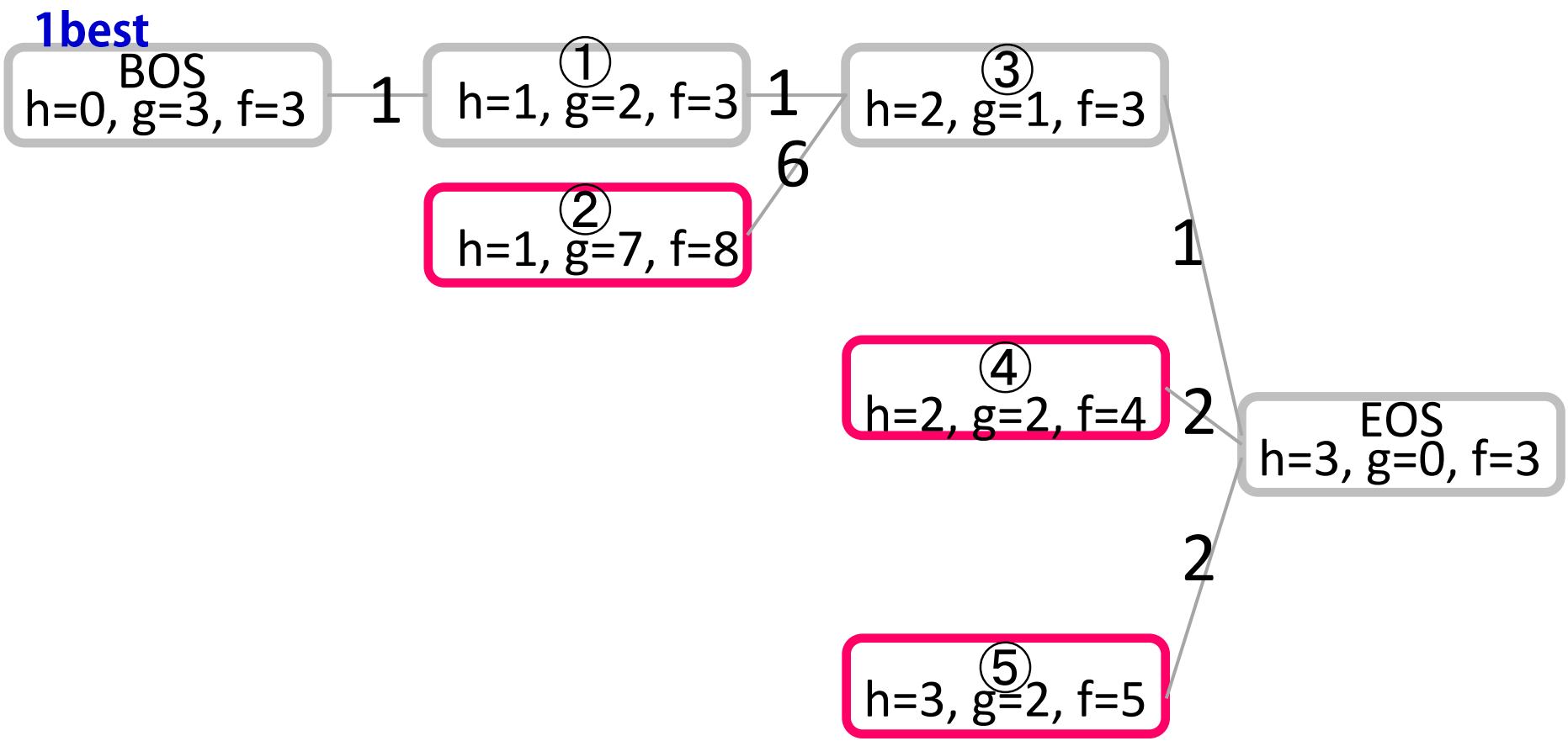

BOS
h=0
①
h=1
②
h=1
③
h=2
④
h=2
⑤
h=3
EOS
h=3
1
1
1
6
1
4
2
6
1
2
2

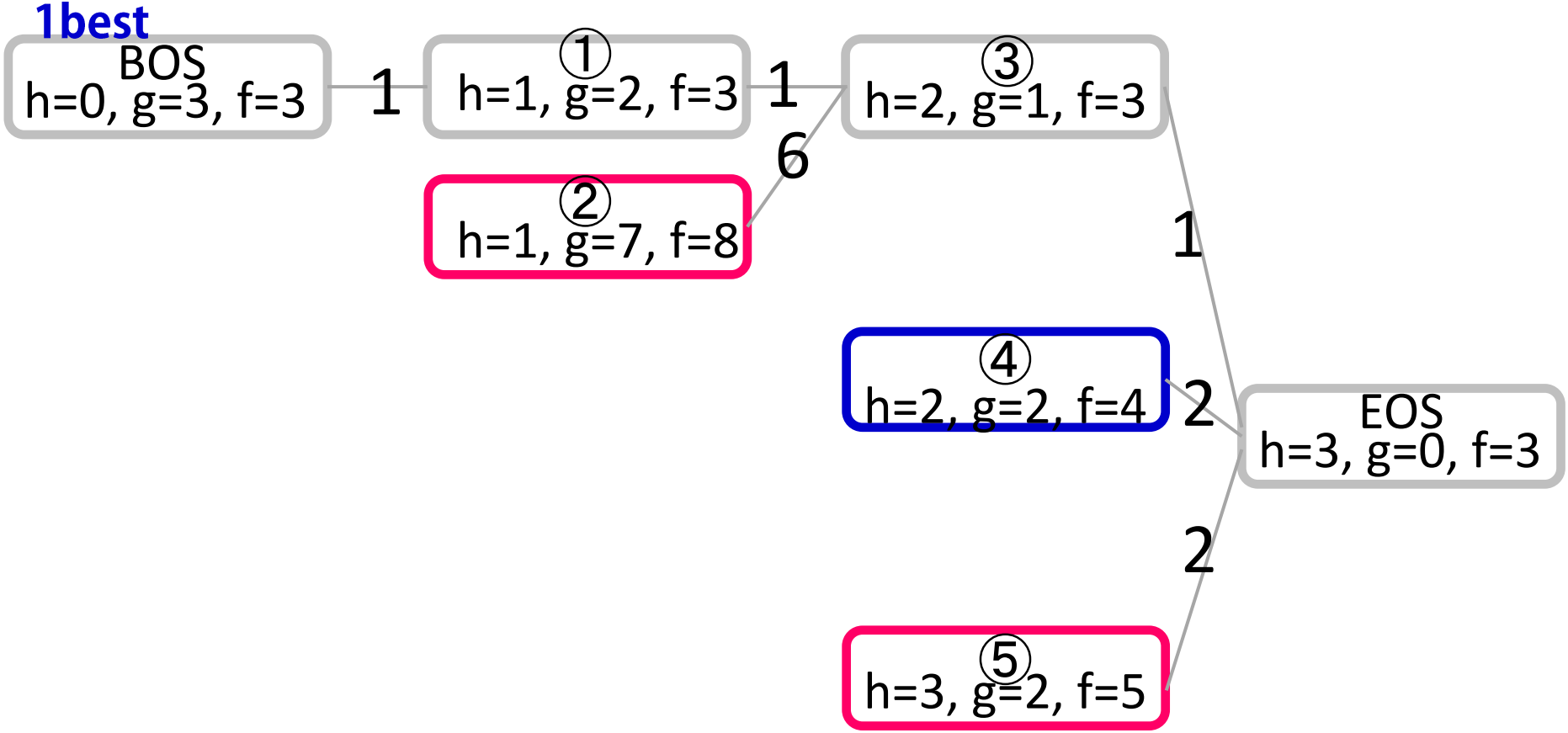

①
h=1, g=2, f=3
②
h=1, g=7, f=8
③
h=2, g=1, f=3
④
h=2, g=2, f=4
⑤
h=3, g=2, f=5
EOS
h=3, g=0, f=3
1
6
1
2
2

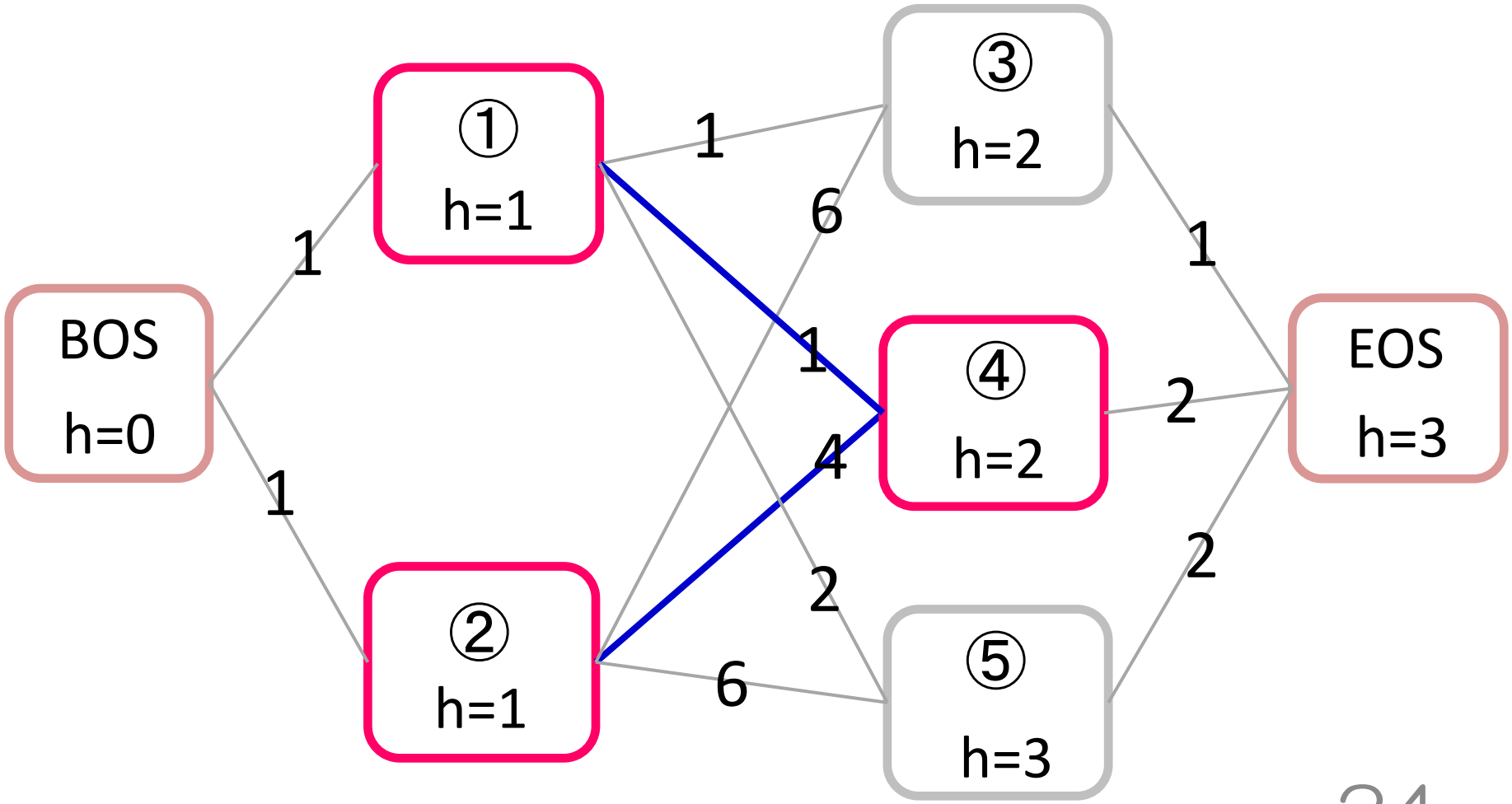

①
h=1, g=2, f=3
②
h=1, g=7, f=8
③
h=2, g=1, f=3
④
h=2, g=2, f=4
⑤
h=3, g=2, f=5
EOS
h=3, g=0, f=3
1
6
1
2
2

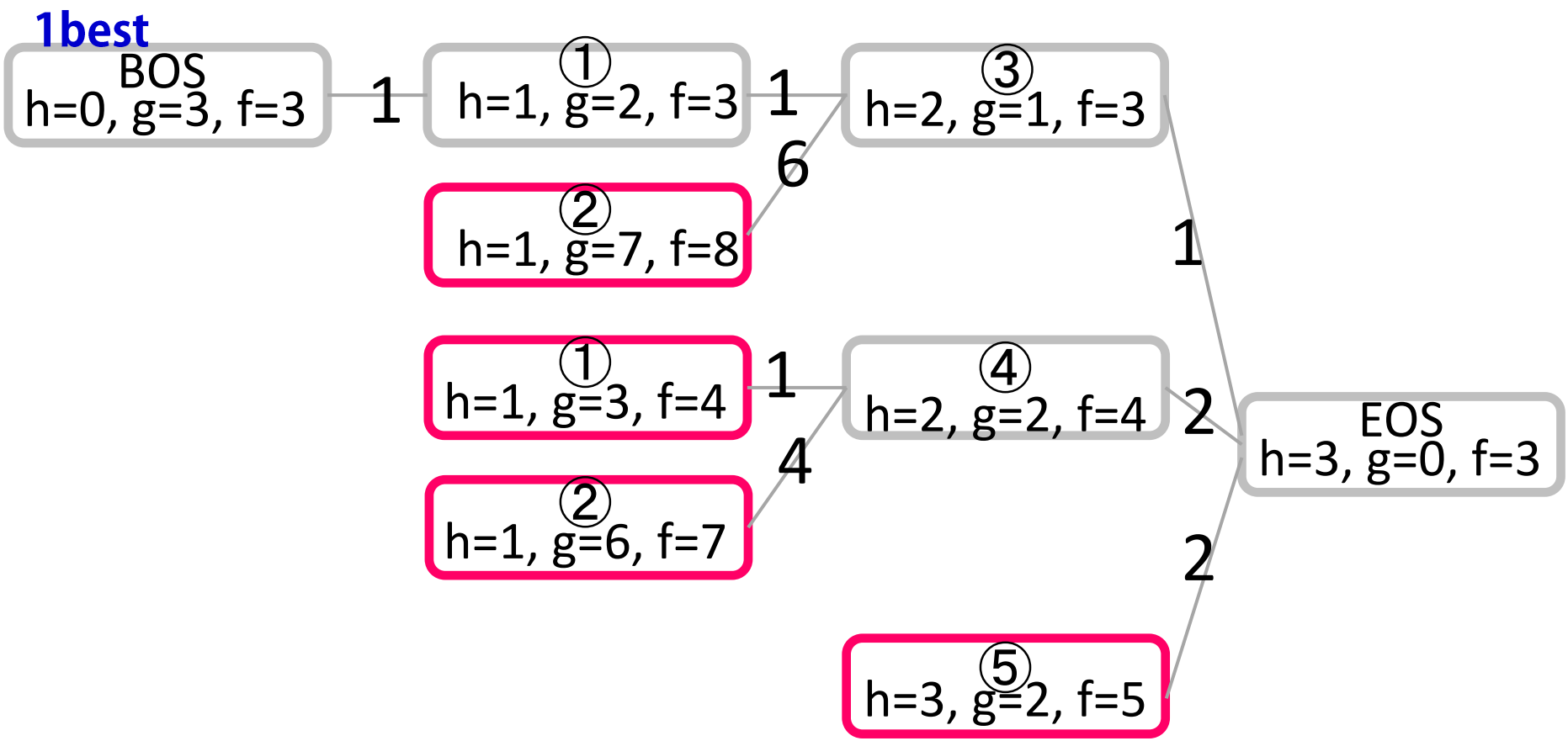

BOS
h=0
①
h=1
②
h=1
③
h=2
④
h=2
⑤
h=3
EOS
h=3
1
1
1
6
1
4
2
6
1
2
2

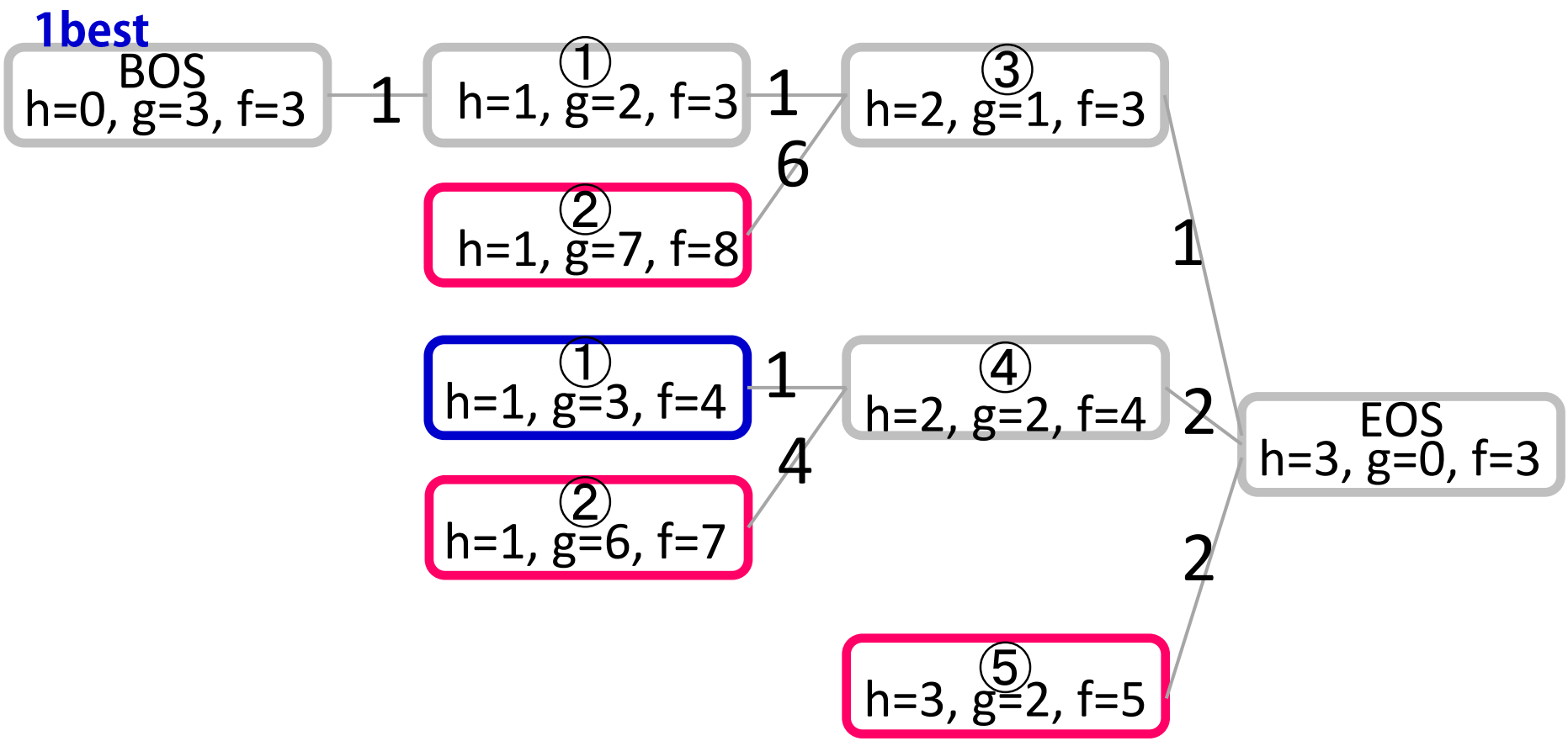

BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
④
h=2, g=2, f=4
2
EOS
h=3, g=0, f=3
2
⑤
h=3, g=2, f=5

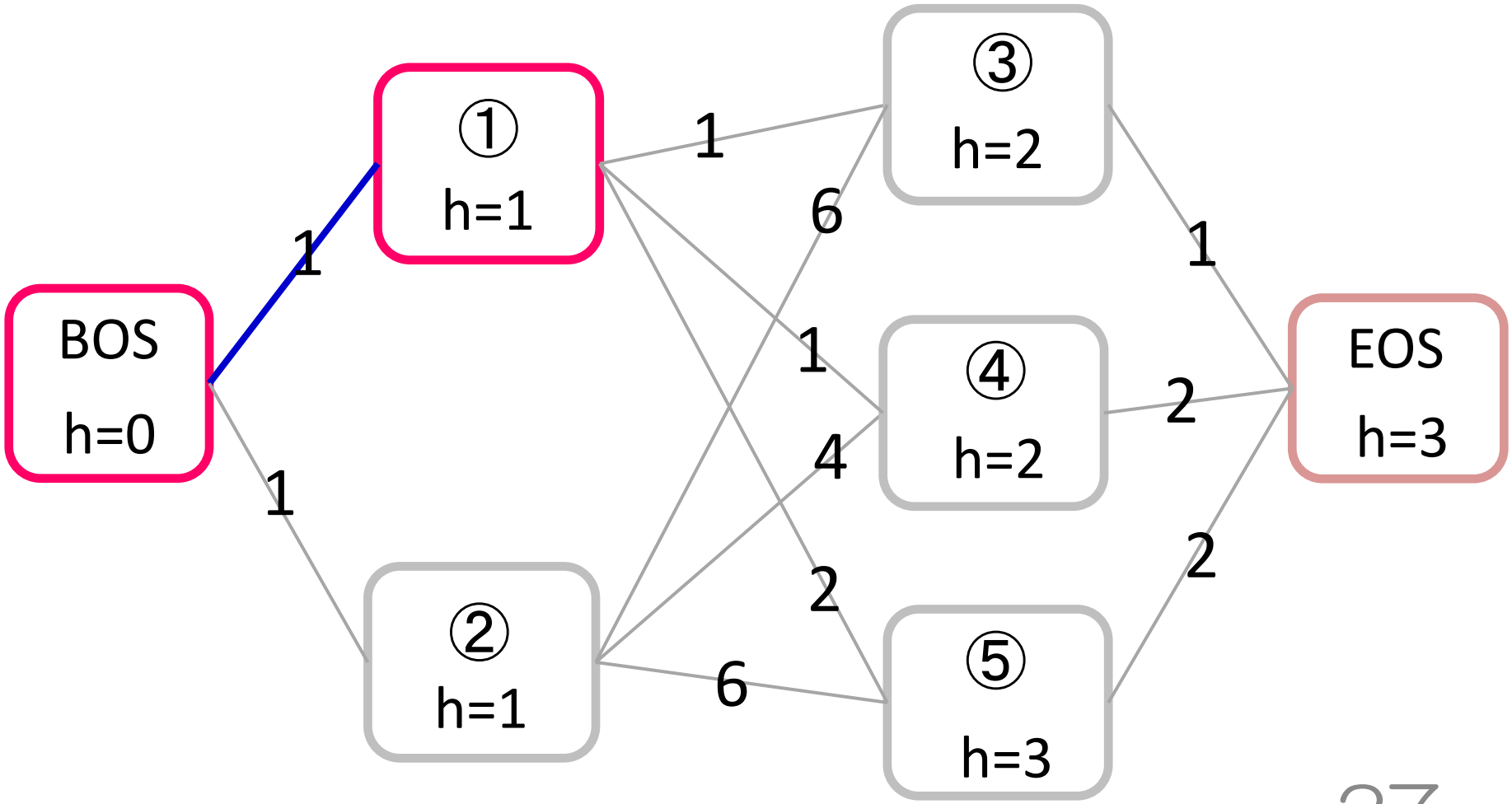

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
④
h=2, g=2, f=4
2
EOS
h=3, g=0, f=3
2
⑤
h=3, g=2, f=5

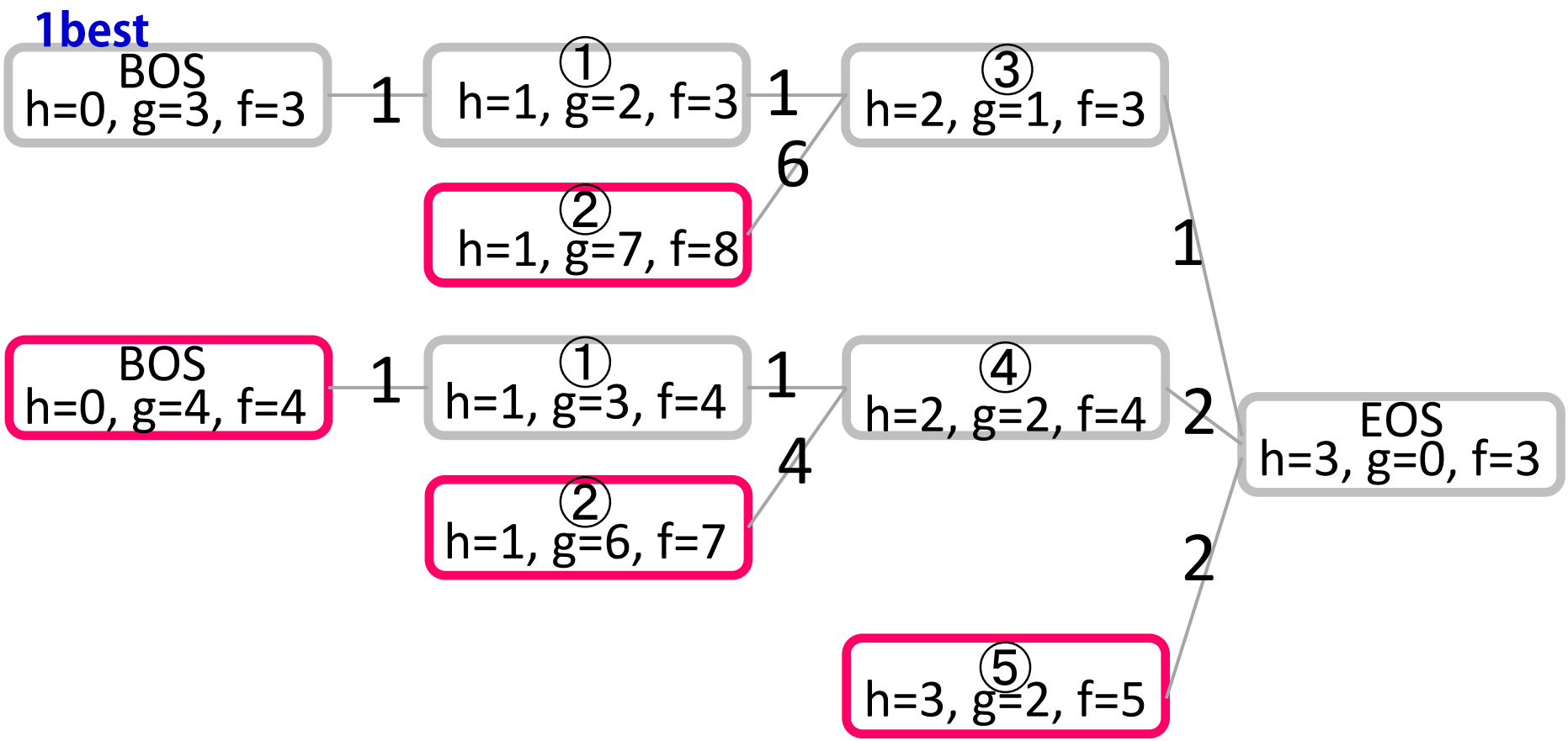

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
④
h=2, g=2, f=4
2
EOS
h=3, g=0, f=3
2
⑤
h=3, g=2, f=5

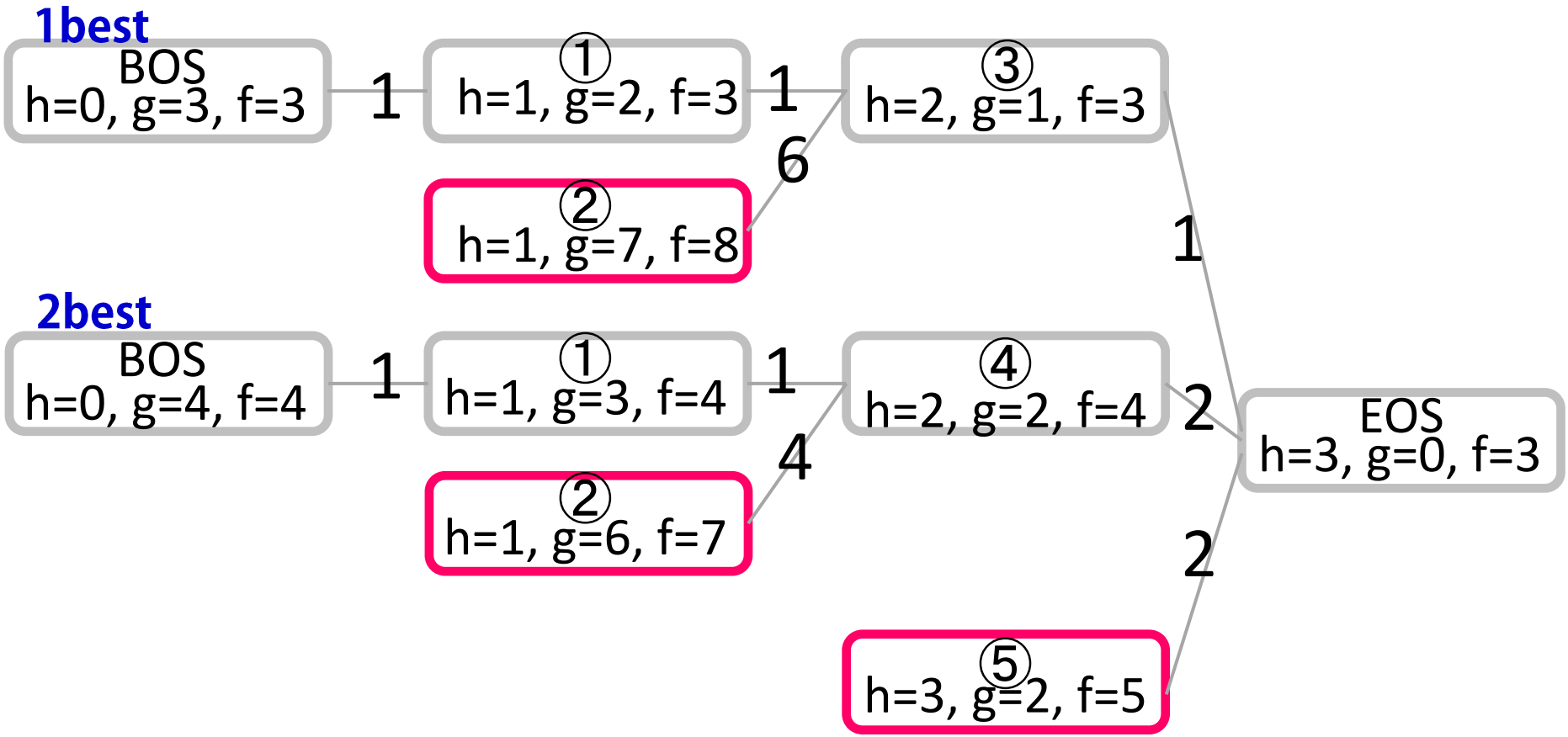

BOS
h=0
①
h=1
②
h=1
③
h=2
④
h=2
⑤
h=3
EOS
h=3
1
1
1
6
1
4
2
6
2
1
2

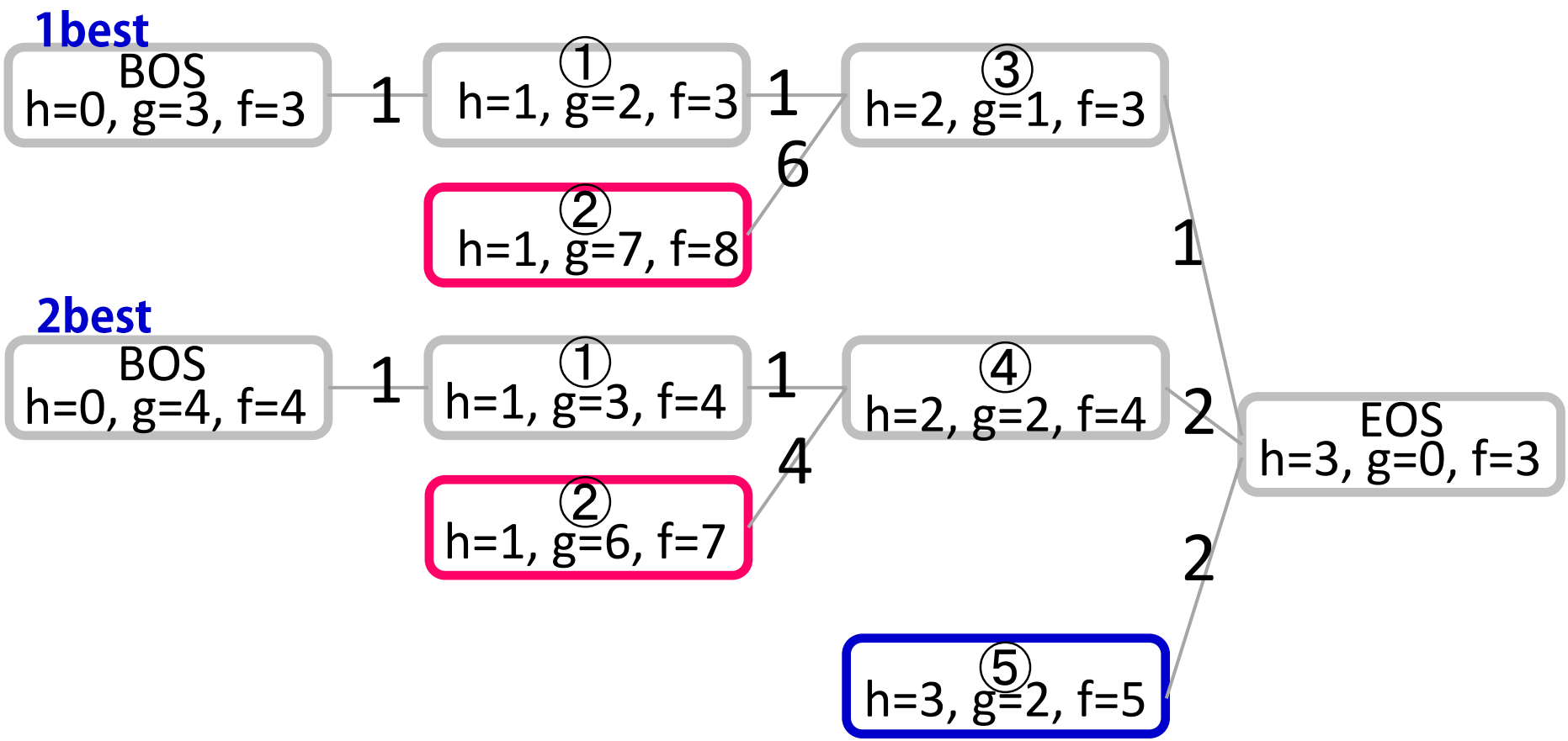

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
EOS
h=3, g=0, f=3
4
②
h=1, g=6, f=7
2
⑤
h=3, g=2, f=5

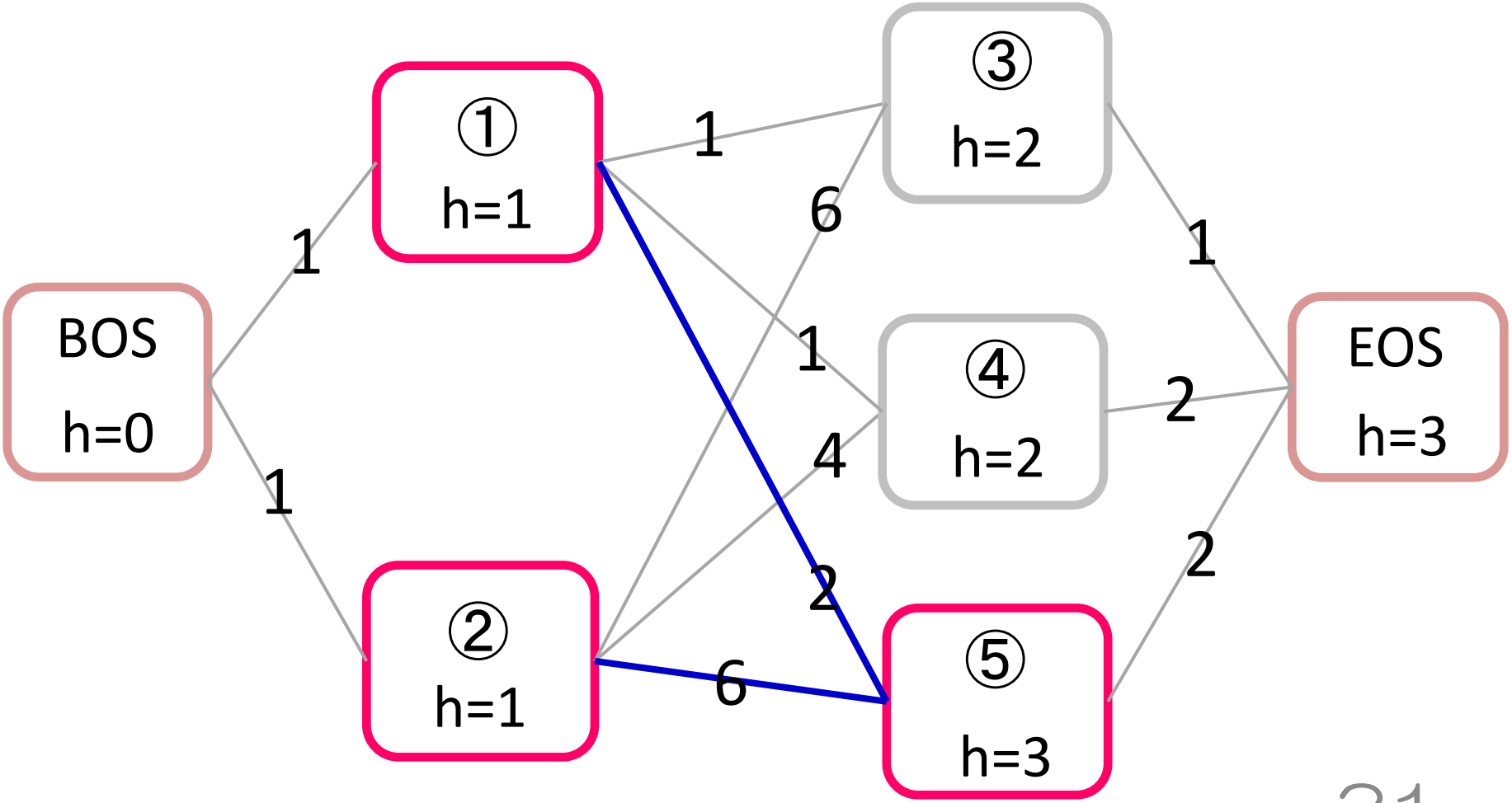

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
EOS
h=3, g=0, f=3
4
②
h=1, g=6, f=7
2
⑤
h=3, g=2, f=5

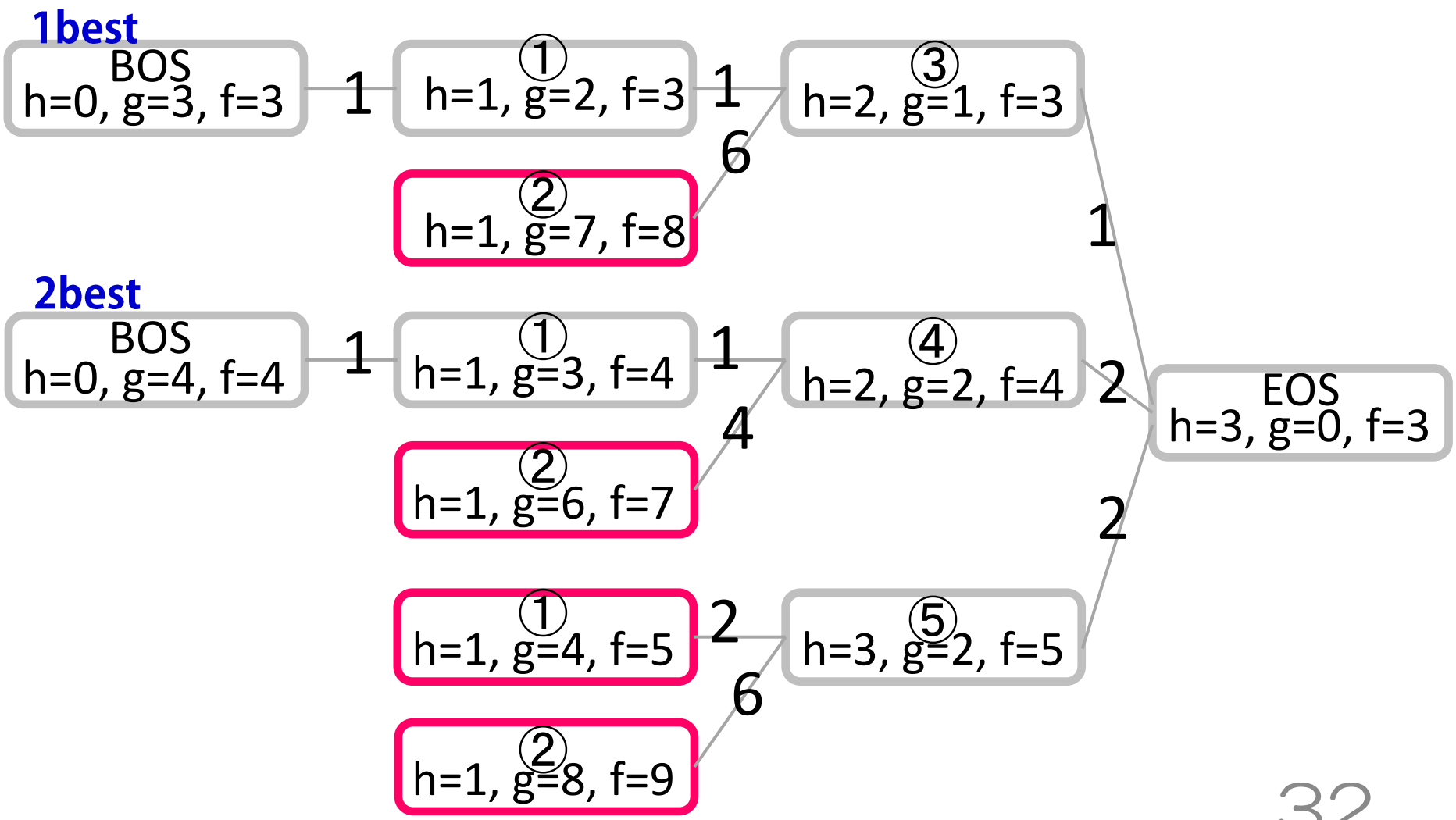

BOS
h=0
①
h=1
②
h=1
③
h=2
④
h=2
⑤
h=3
EOS
h=3
1
1
1
6
1
4
2
6
1
2
2

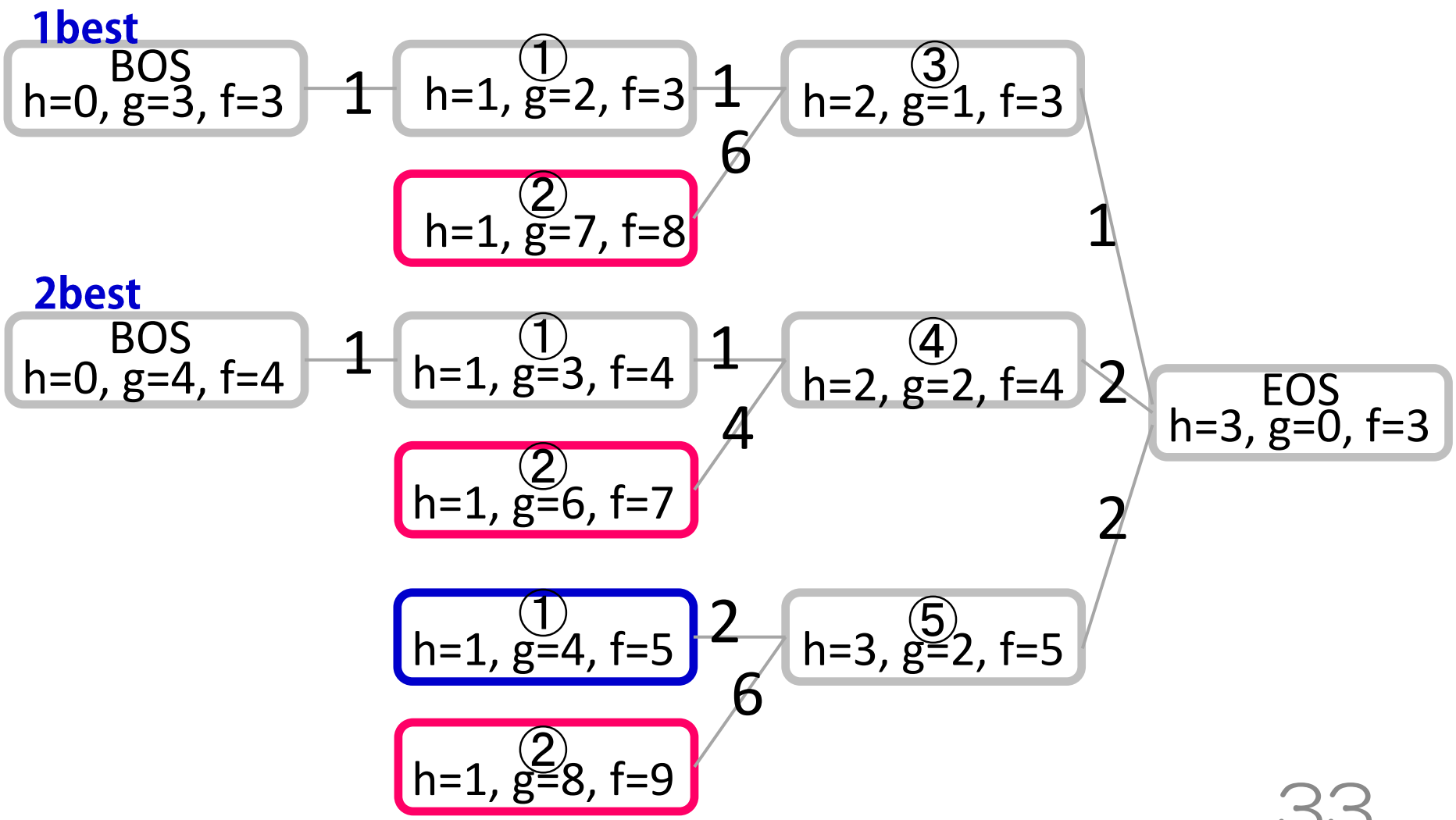

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
EOS
h=3, g=0, f=3
4
②
h=1, g=6, f=7
2
⑤
h=3, g=2, f=5

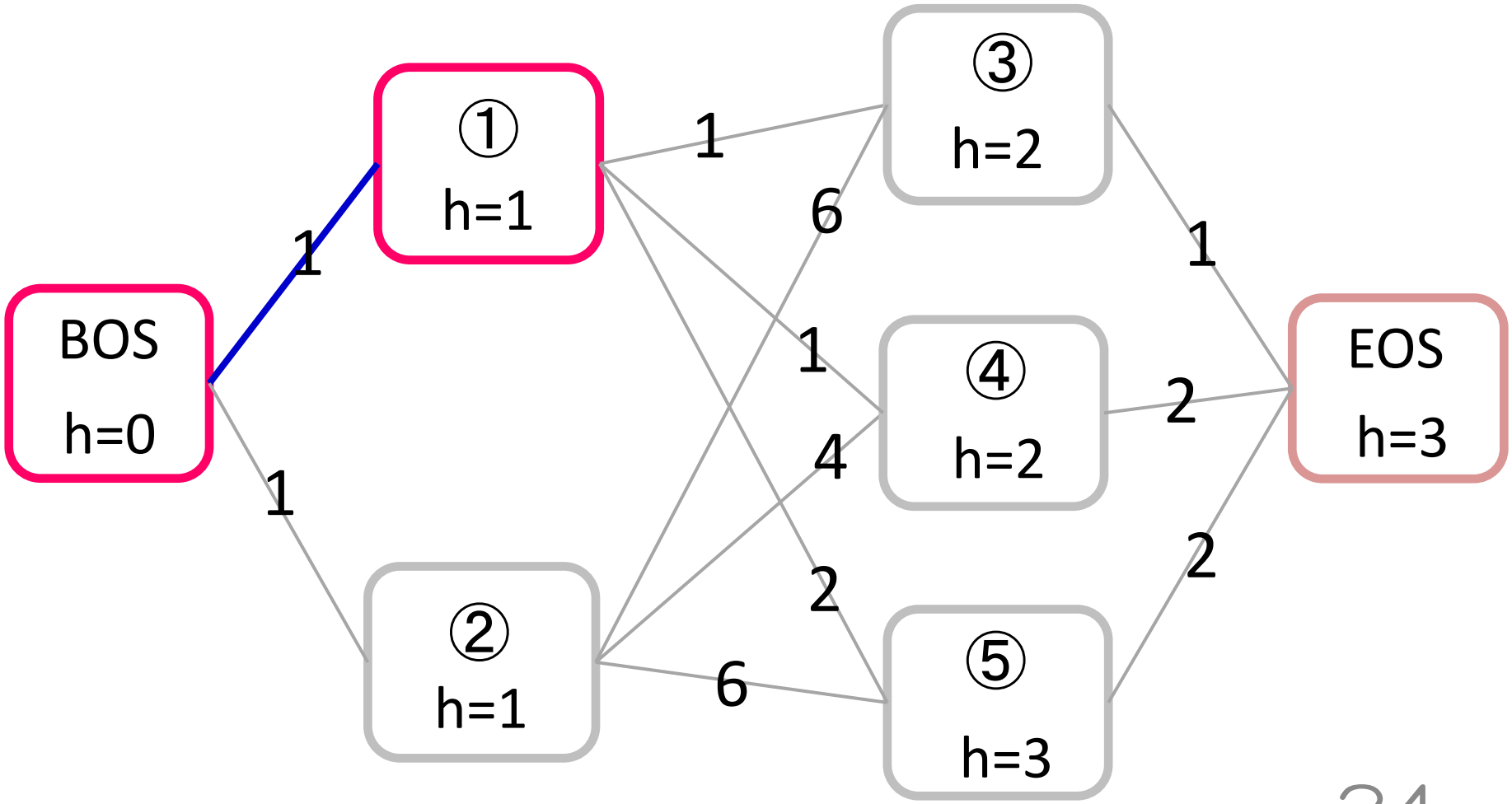

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4
②
h=1, g=6, f=7
EOS
h=3, g=0, f=3
2
⑤
h=3, g=2, f=5

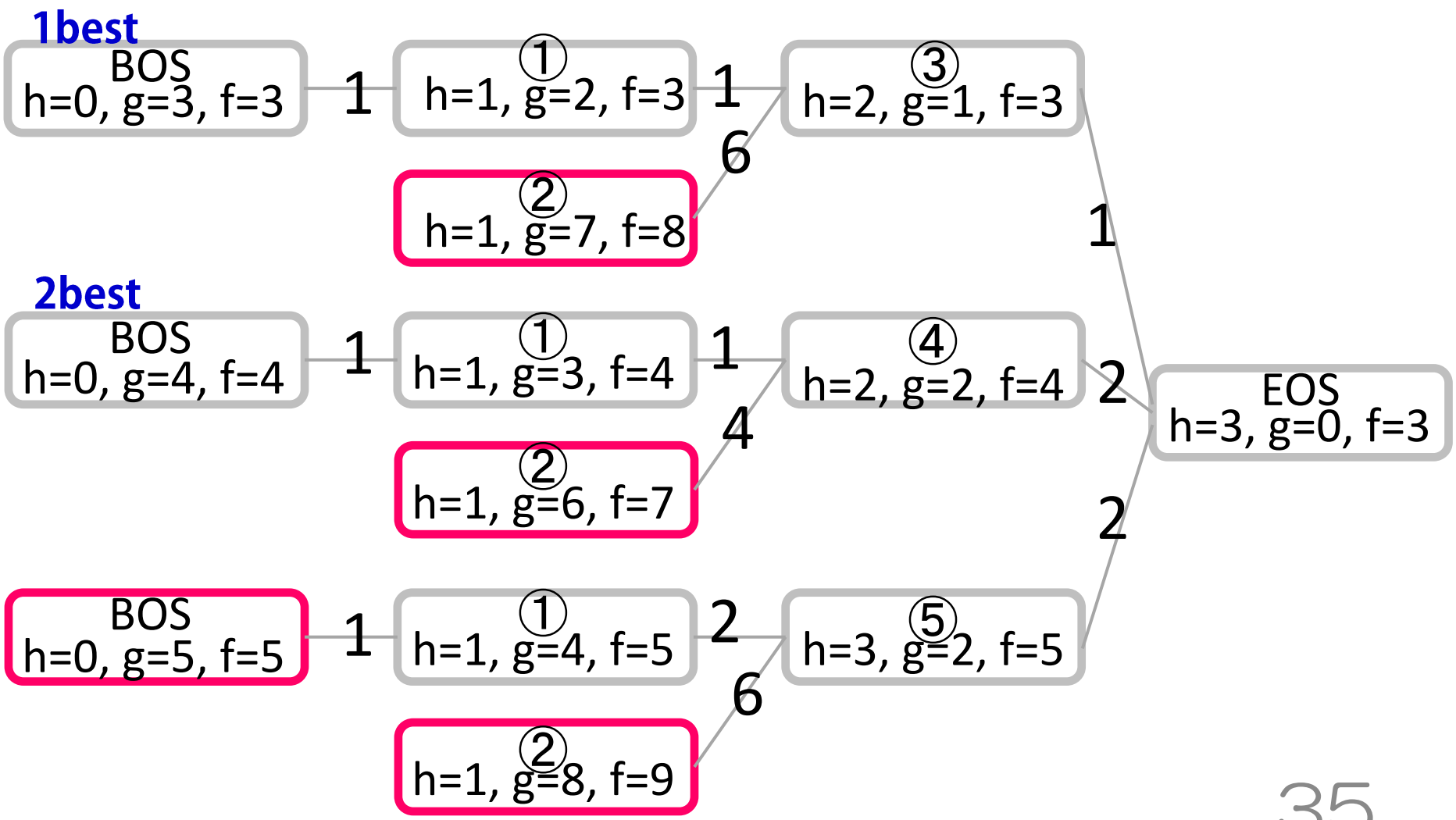

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4
②
h=1, g=6, f=7
EOS
h=3, g=0, f=3
2
⑤
h=3, g=2, f=5

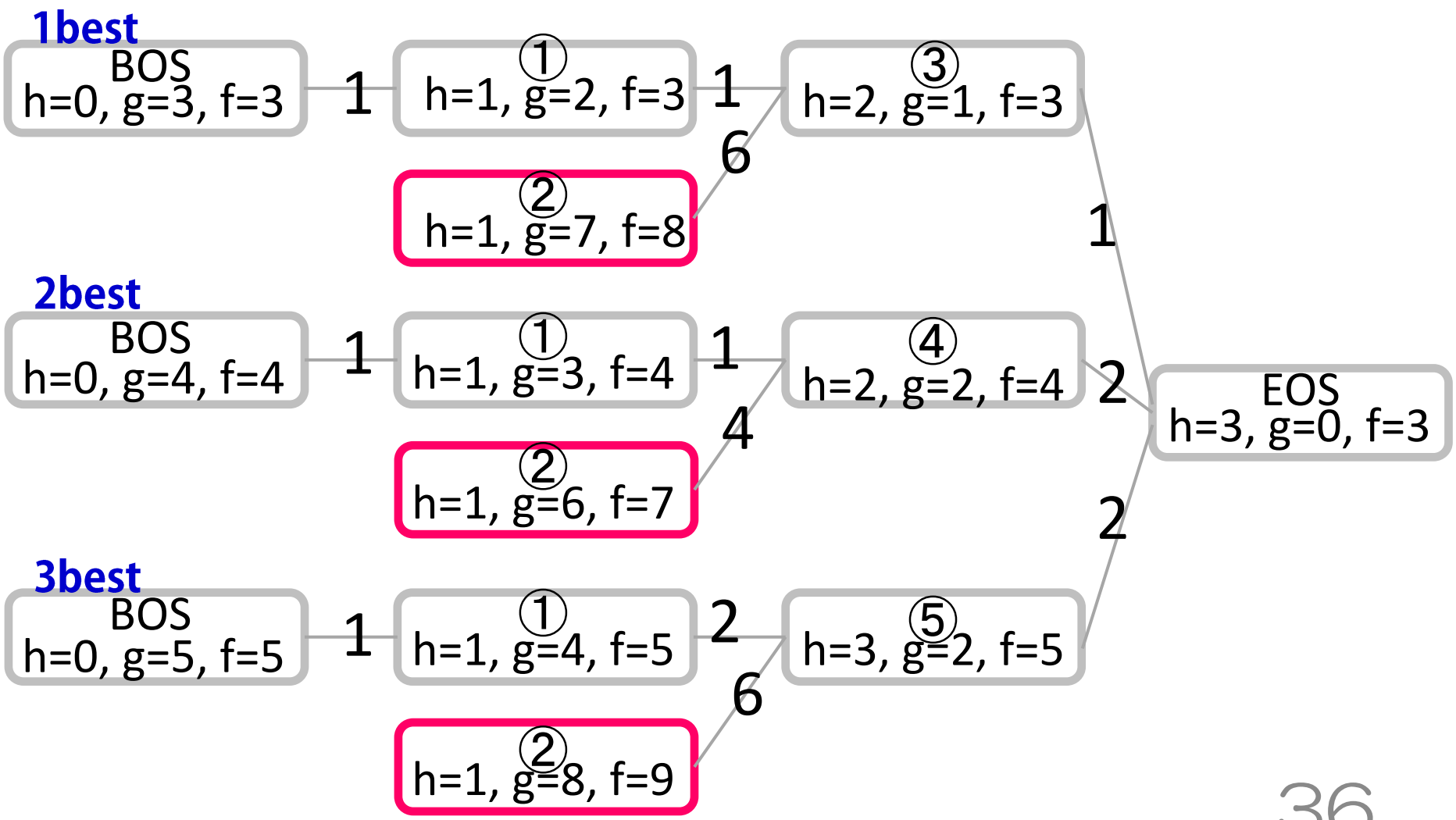

BOS
h=0
①
h=1
②
h=1
③
h=2
④
h=2
⑤
h=3
EOS
h=3
1
1
1
6
1
4
2
6
1
2
2

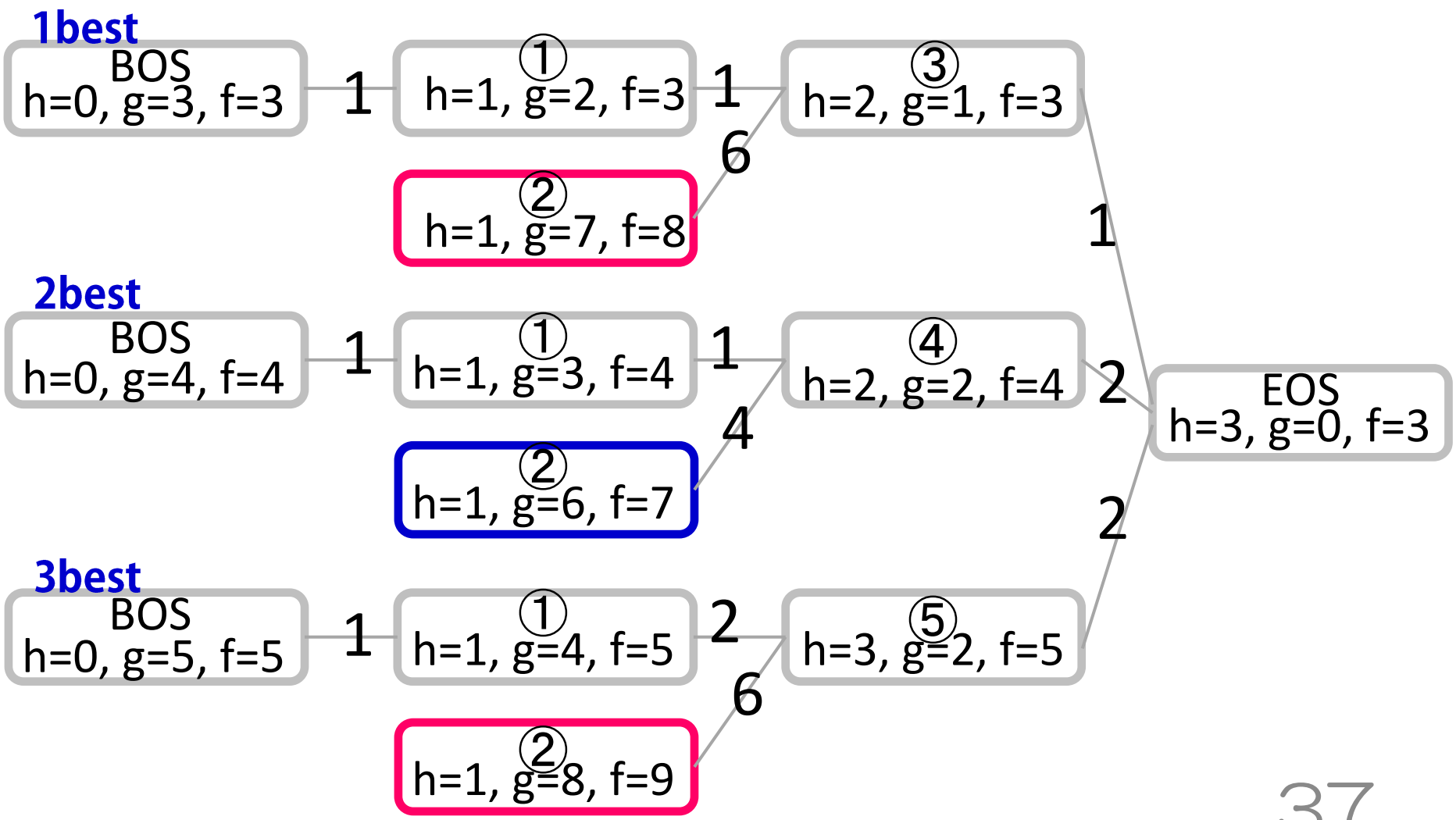

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4
②
h=1, g=6, f=7
EOS
h=3, g=0, f=3
2
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
6
②
h=1, g=8, f=9

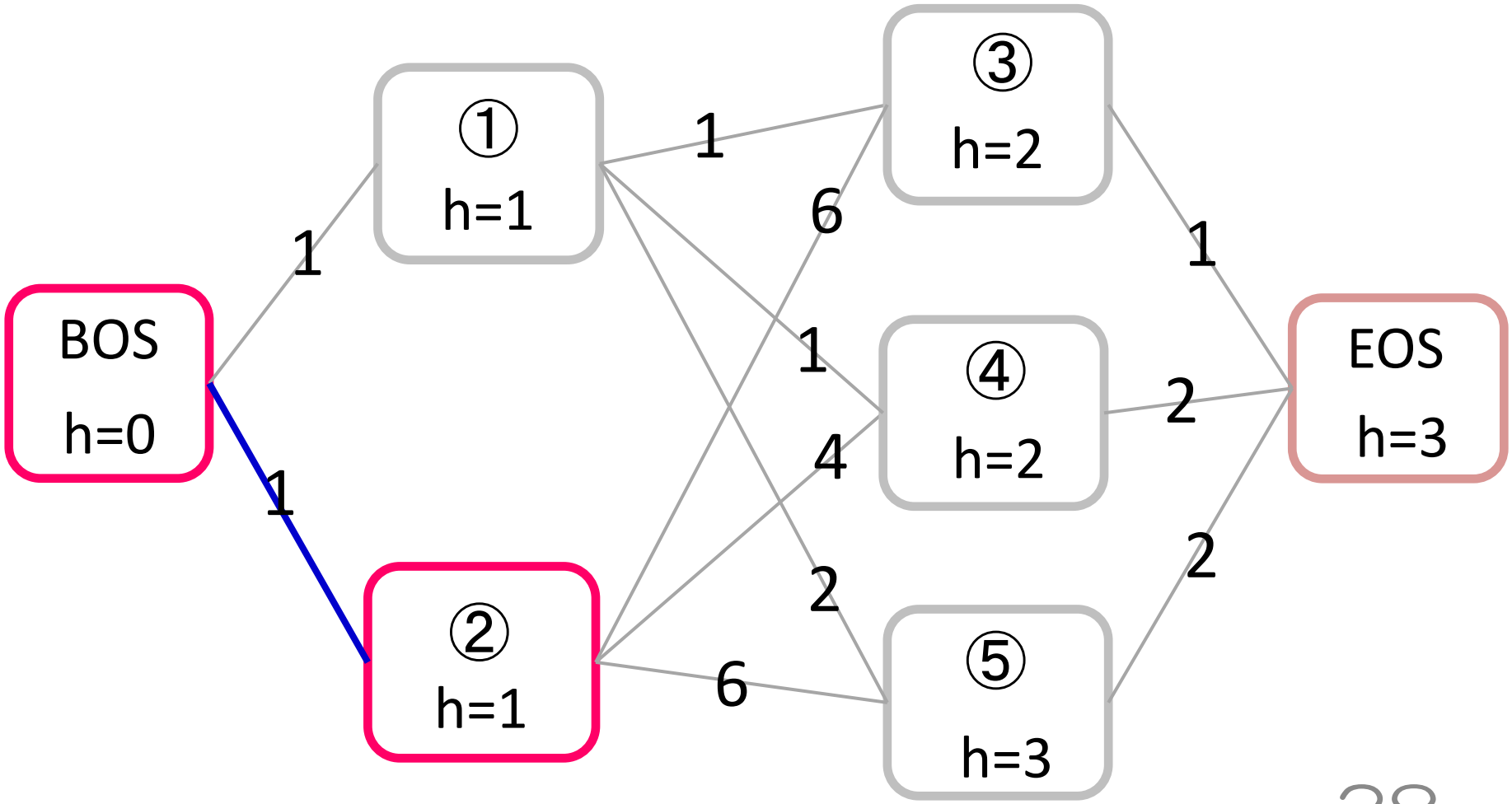

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
EOS
h=3, g=0, f=3
4
②
h=1, g=6, f=7
2
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
6
②
h=1, g=8, f=9

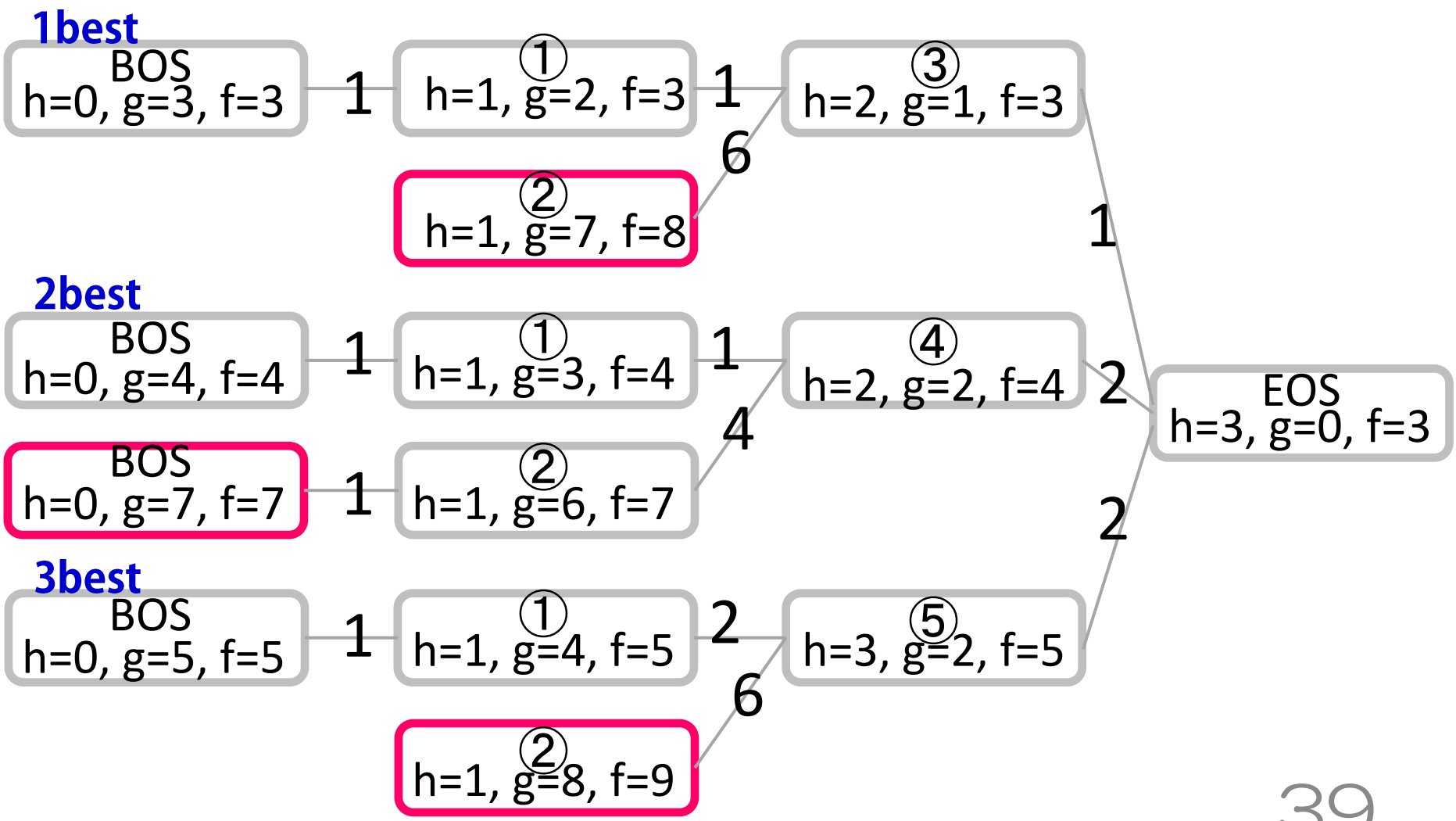

BOS
h=0
①
h=1
②
h=1
③
h=2
④
h=2
⑤
h=3
EOS
h=3
1
1
1
6
1
4
2
6
1
2
2

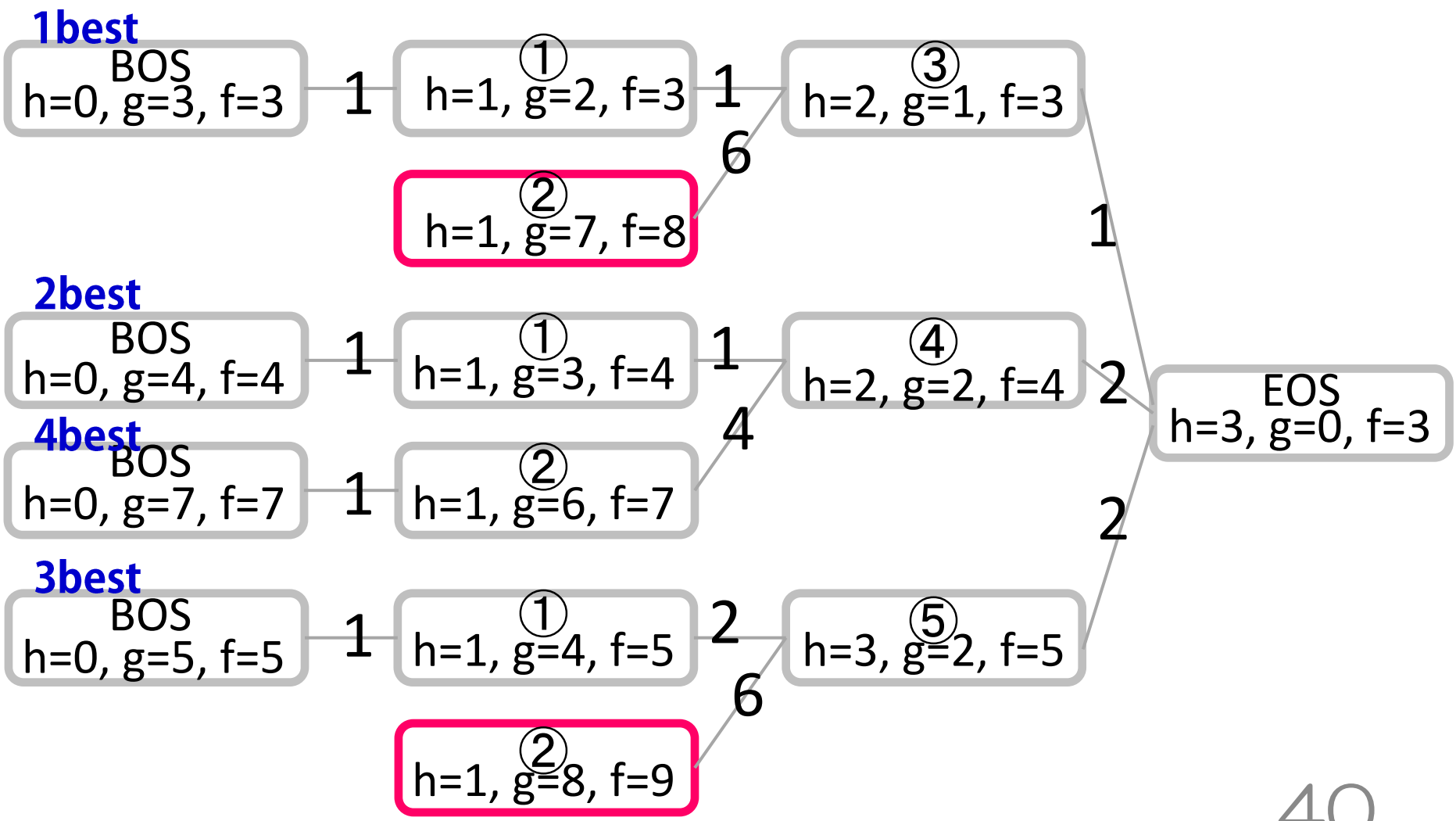

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4
②
h=1, g=6, f=7
EOS
h=3, g=0, f=3
2
BOS
h=0, g=5, f=5
1
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
6
②
h=1, g=8, f=9

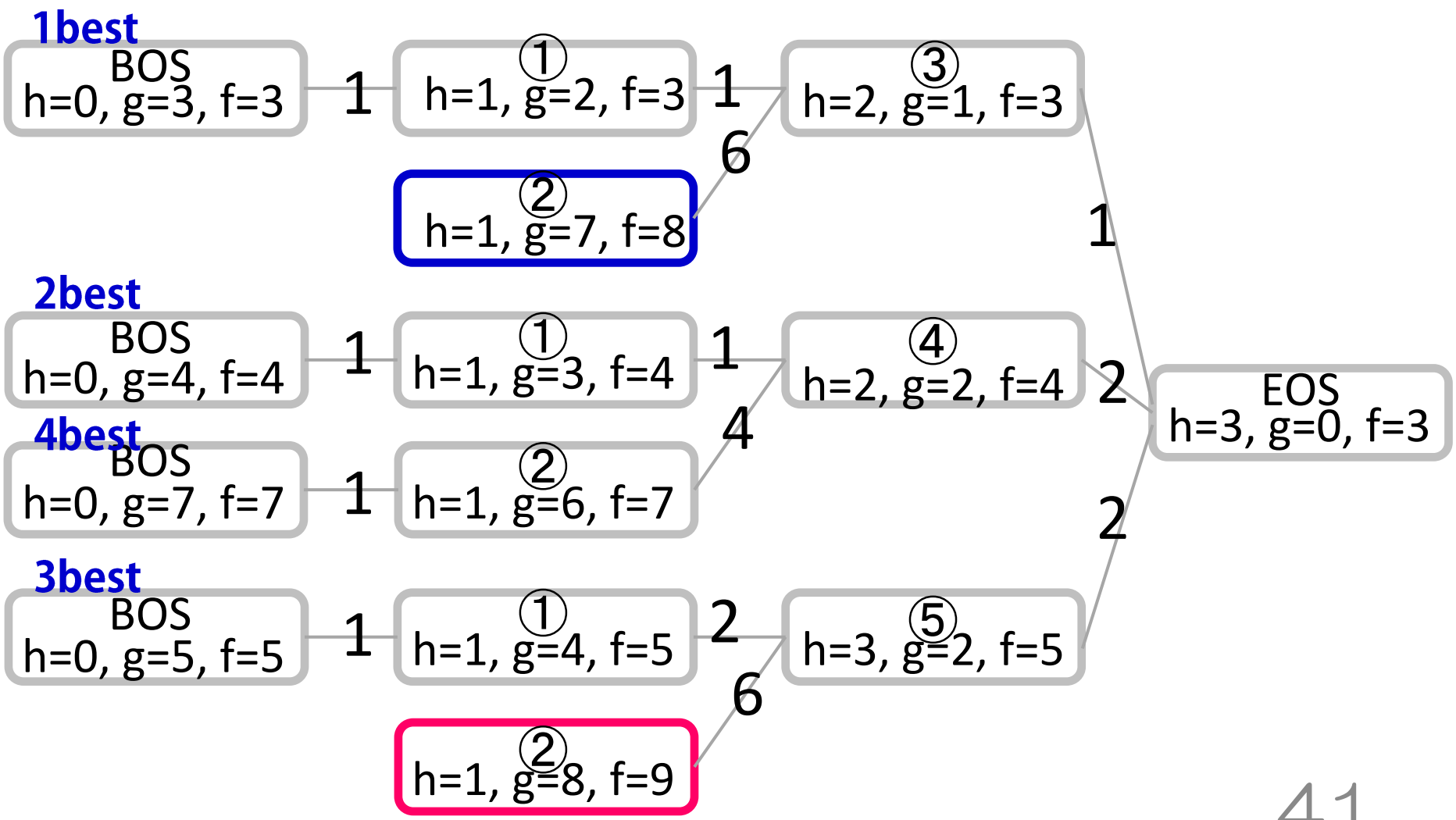

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4
②
h=1, g=6, f=7
EOS
h=3, g=0, f=3
2
3best
BOS
h=0, g=5, f=5
1
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
6
②
h=1, g=8, f=9

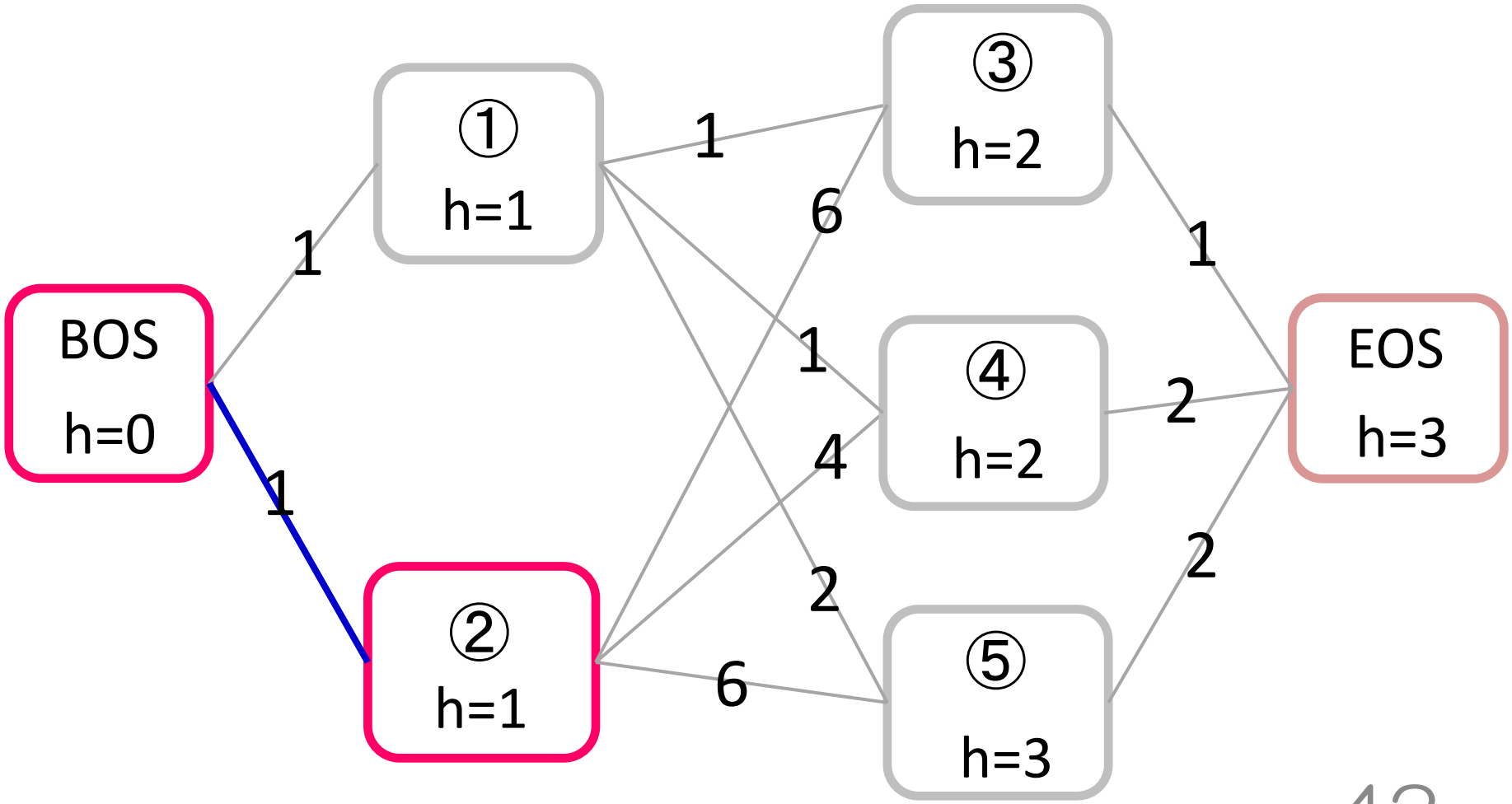

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4
②
h=1, g=6, f=7
EOS
h=3, g=0, f=3
2
3best
BOS
h=0, g=5, f=5
1
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
6
②
h=1, g=8, f=9

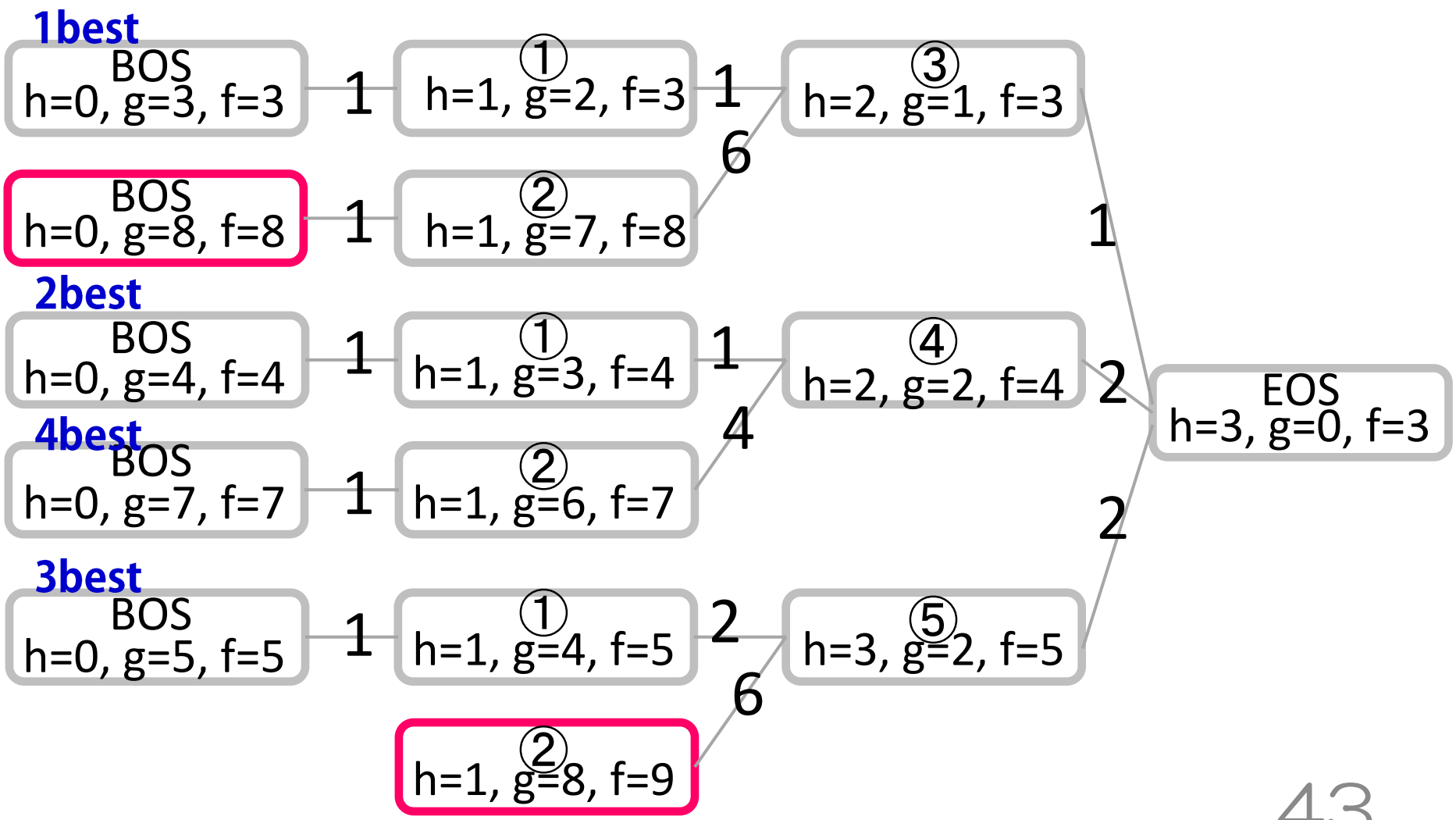

BOS
h=0
①
h=1
②
h=1
③
h=2
④
h=2
⑤
h=3
EOS
h=3
1
1
1
6
1
4
2
6
1
2
2

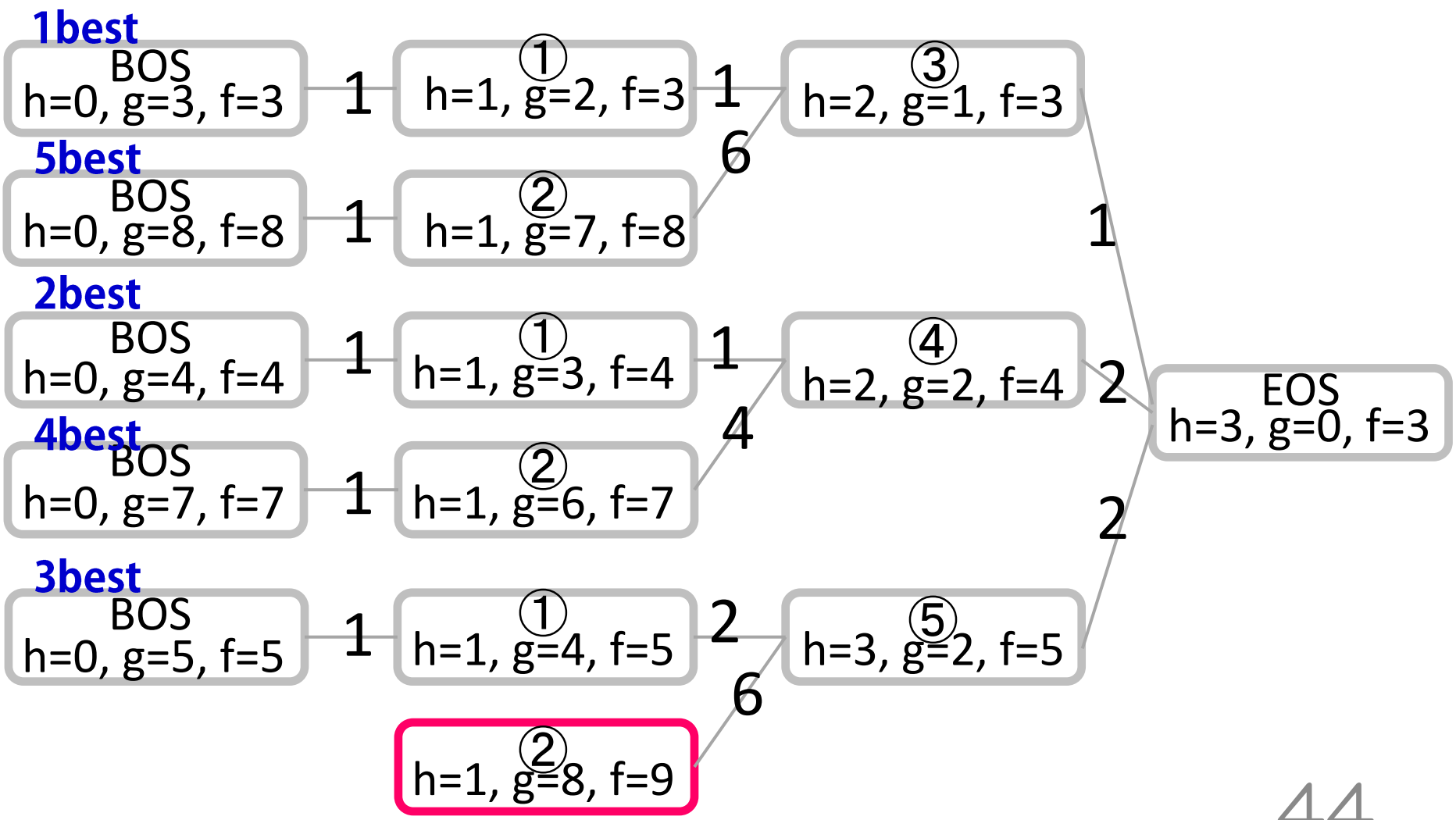

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
BOS
h=0, g=7, f=7
1
②
h=1, g=6, f=7
4
EOS
h=3, g=0, f=3
2
3best
BOS
h=0, g=5, f=5
1
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
6
②
h=1, g=8, f=9

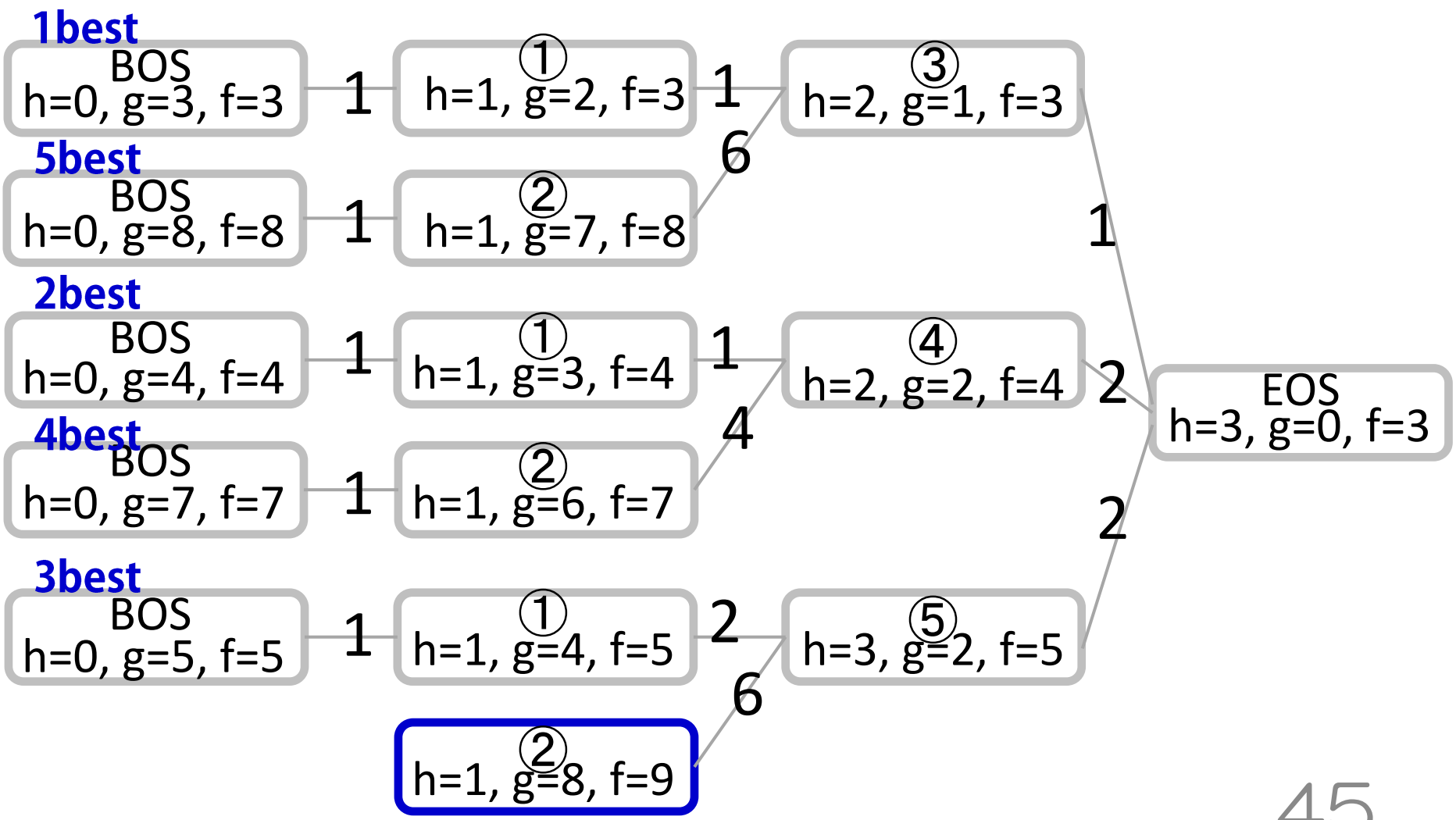

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4best
BOS
h=0, g=7, f=7
1
②
h=1, g=6, f=7
4
EOS
h=3, g=0, f=3
2
3best
BOS
h=0, g=5, f=5
1
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
6
②
h=1, g=8, f=9

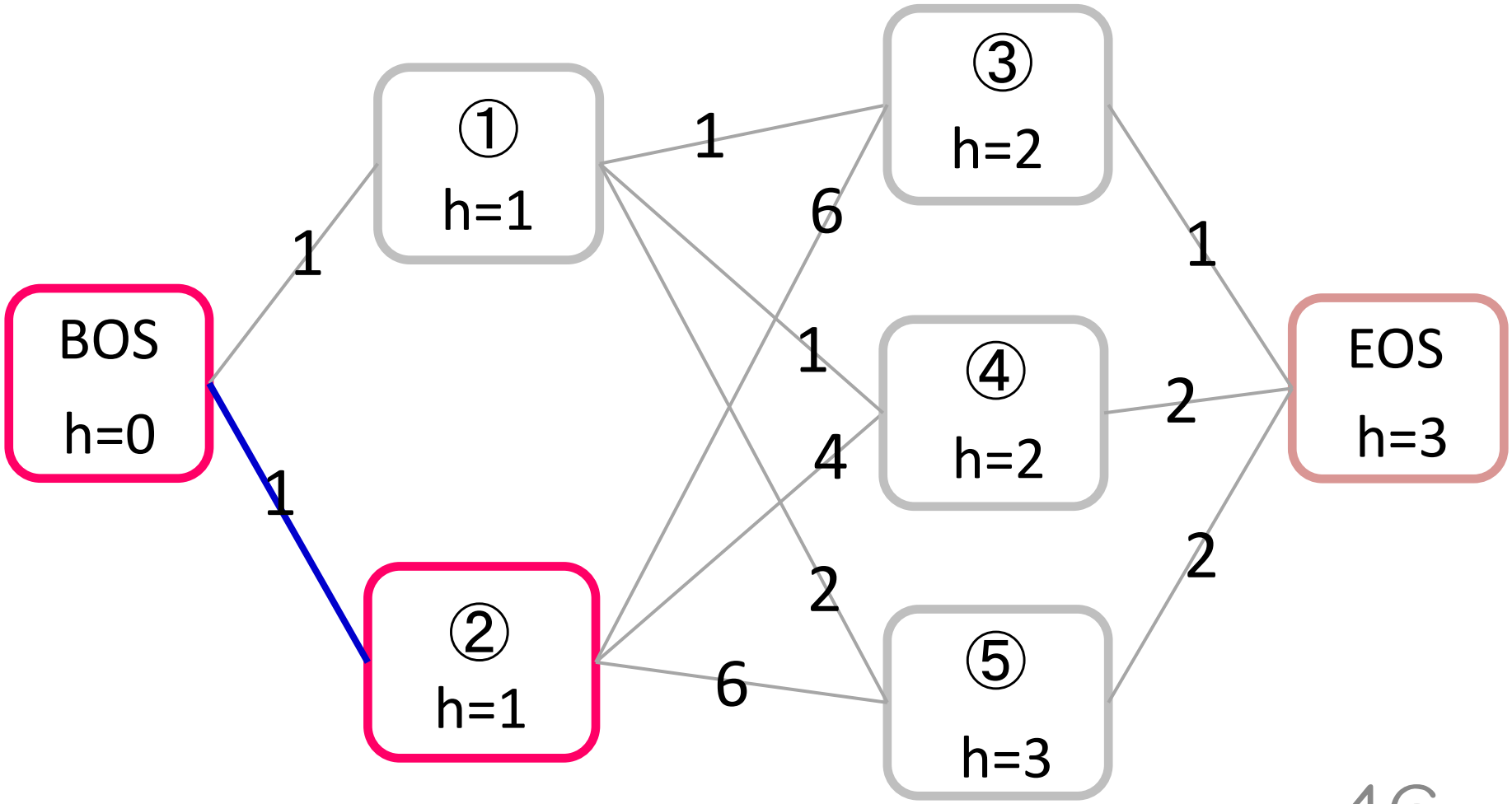

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
②
h=1, g=7, f=8
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4best
BOS
h=0, g=7, f=7
1
②
h=1, g=6, f=7
4
EOS
h=3, g=0, f=3
2
3best
BOS
h=0, g=5, f=5
1
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
6
②
h=1, g=8, f=9

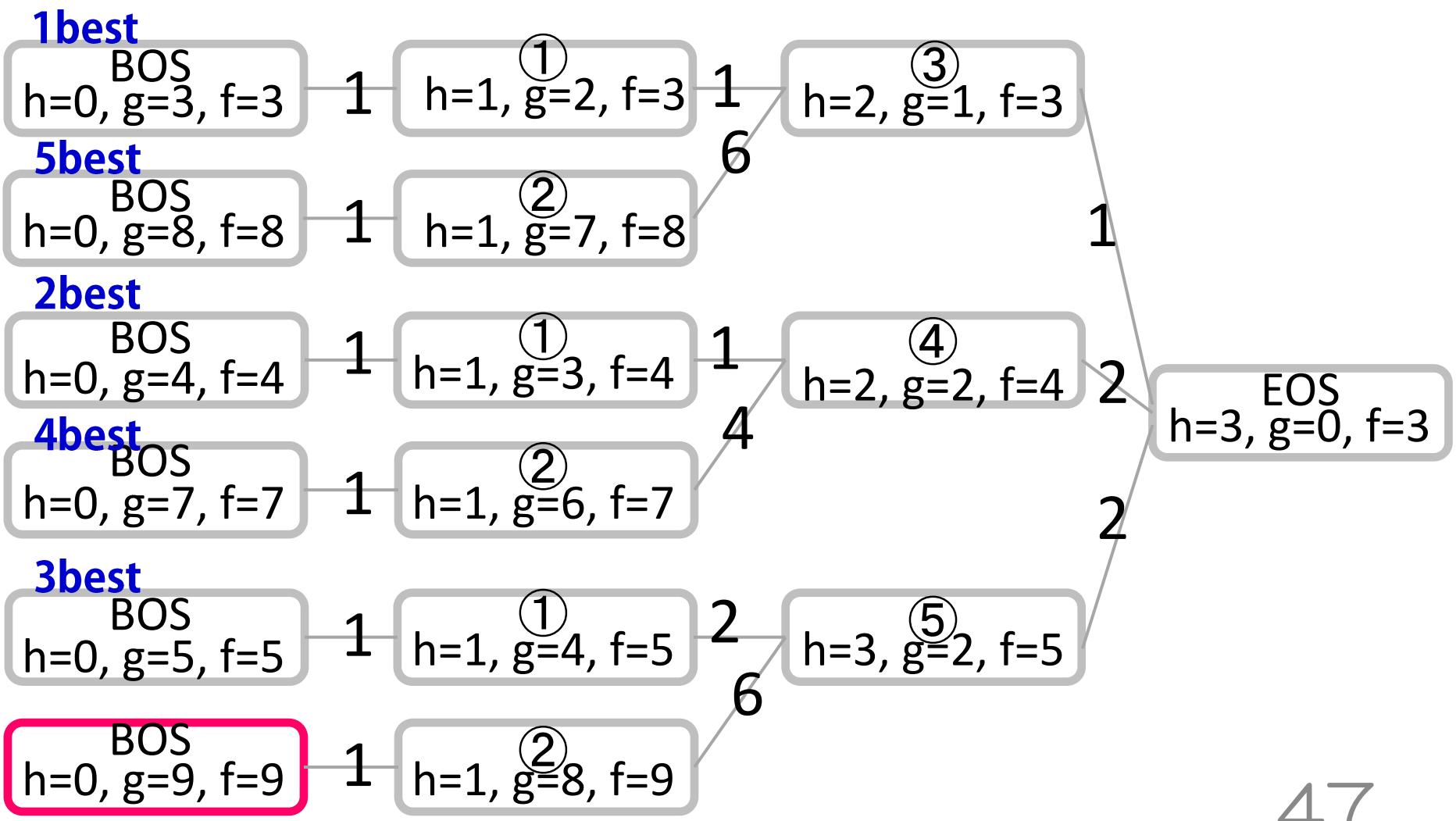

BOS
h=0
①
h=1
②
h=1
③
h=2
④
h=2
⑤
h=3
EOS
h=3
1
1
1
6
1
4
2
6
1
2
2

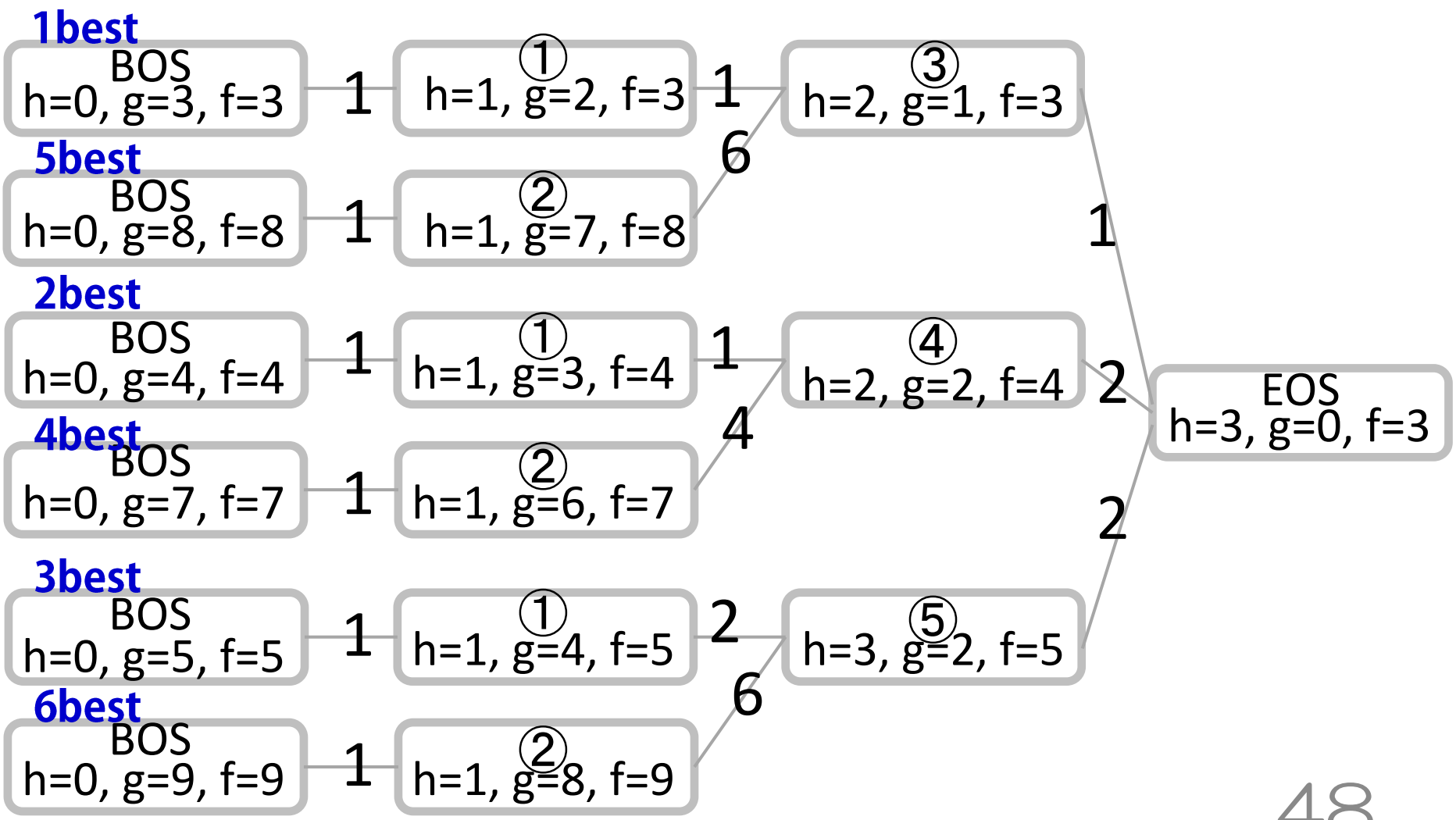

1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
6
BOS
h=0, g=8, f=8
1
②
h=1, g=7, f=8
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4
4best
BOS
h=0, g=7, f=7
1
②
h=1, g=6, f=7
EOS
h=3, g=0, f=3
2
3best
BOS
h=0, g=5, f=5
1
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
6
②
h=1, g=8, f=9


1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
5best
BOS
h=0, g=8, f=8
1
②
h=1, g=7, f=8
6
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4best
BOS
h=0, g=7, f=7
1
②
h=1, g=6, f=7
4
EOS
h=3, g=0, f=3
2
3best
BOS
h=0, g=5, f=5
1
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
6
②
h=1, g=8, f=9


1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
5best
BOS
h=0, g=8, f=8
1
②
h=1, g=7, f=8
6
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4best
BOS
h=0, g=7, f=7
1
②
h=1, g=6, f=7
4
EOS
h=3, g=0, f=3
2
3best
BOS
h=0, g=5, f=5
1
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
②
h=1, g=8, f=9
6


BOS
h=0
①
h=1
②
h=1
③
h=2
④
h=2
⑤
h=3
EOS
h=3
1
1
1
6
1
4
2
6
1
2
2


1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
5best
BOS
h=0, g=8, f=8
1
②
h=1, g=7, f=8
6
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4best
BOS
h=0, g=7, f=7
1
②
h=1, g=6, f=7
4
EOS
h=3, g=0, f=3
2
3best
BOS
h=0, g=5, f=5
1
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
BOS
h=0, g=9, f=9
1
②
h=1, g=8, f=9
6


1best
BOS
h=0, g=3, f=3
1
①
h=1, g=2, f=3
1
③
h=2, g=1, f=3
5best
BOS
h=0, g=8, f=8
1
②
h=1, g=7, f=8
6
1
2best
BOS
h=0, g=4, f=4
1
①
h=1, g=3, f=4
1
④
h=2, g=2, f=4
2
4best
BOS
h=0, g=7, f=7
1
②
h=1, g=6, f=7
4
EOS
h=3, g=0, f=3
2
3best
BOS
h=0, g=5, f=5
1
①
h=1, g=4, f=5
2
⑤
h=3, g=2, f=5
6best
BOS
h=0, g=9, f=9
1
②
h=1, g=8, f=9
6

ポイント

- **前向き探索を行う際に，（第１位を除いて）第N位の経路は必ず，1～N-1位の経路と比較されて負けている.**
  - そうでないと，N位よりも上位になっている.

- **1～N-1の経路を開いていくうちに，必ずN位の経路が負けたノード（N位のノードが負けたノード）が開かれる.**

まとめ

- **単なるA*だけではN-bestは出せない.**
  - [永田94]には図以外まともに載ってないという事実…
- **木探索しないといけない.**
- **わかってみたら，すごく簡単だった.**

実は…

- この本に実装法が載ってた…
  - 流石！