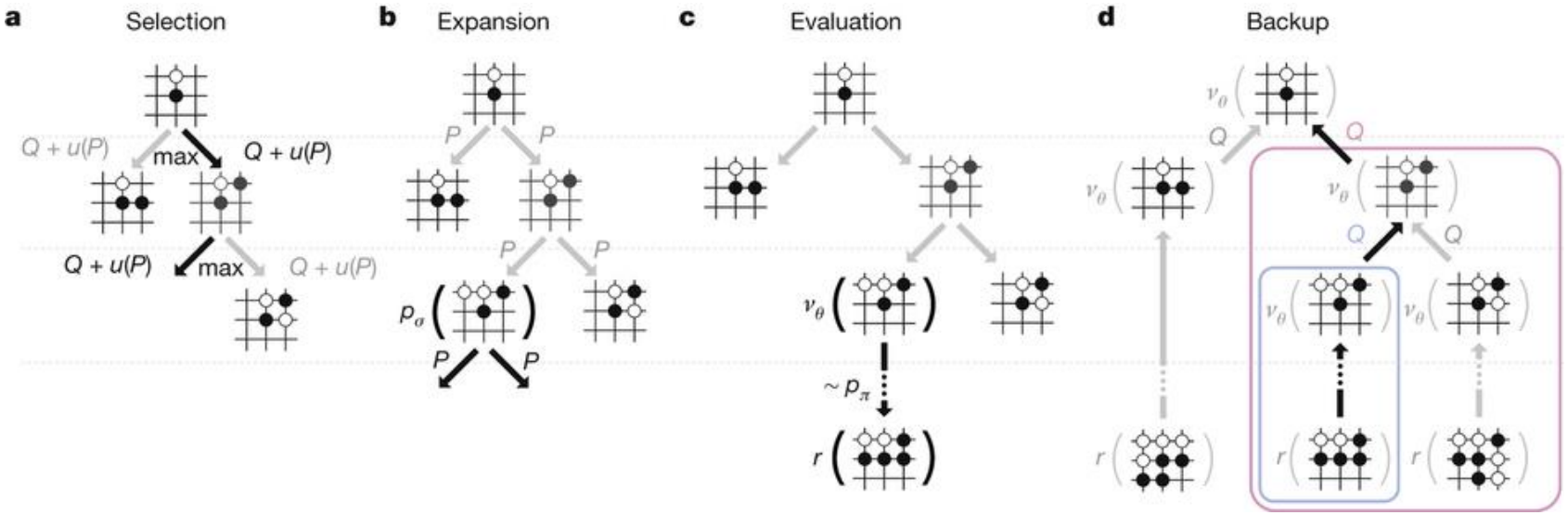
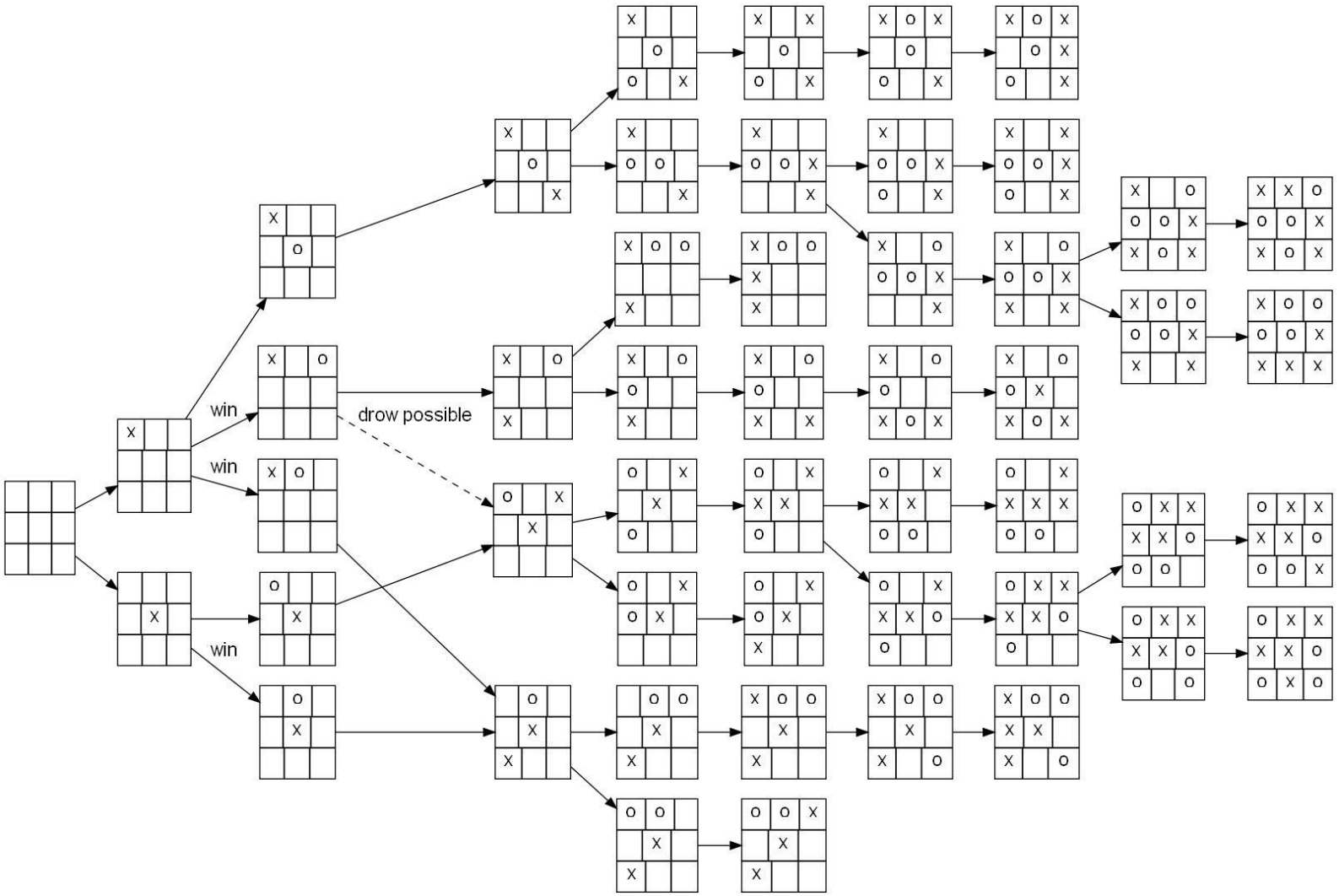
4장 게임 전략 : 상대가 있을 경우의 대처 방법



출처: https://www.nature.com/article-assets/npg/nature/journal/v529/n7587/images/nature16961-f3.jpg

4장 게임 전략 : 적용 사례



출처: https://upload.wikimedia.org/wikipedia/commons/1/1f/Tic-tac-toe-full-game-tree-x-rational.jpg

4장 게임 전략 : 적용 사례

출처: https://disruptionhub.com/wp-content/uploads/2015/09/image.jpg

4장 게임 전략 : 적용 사례

The 1996 match

Game #	White	Black	Result	Comment
1	**Deep Blue**	Kasparov	1–0	
2	**Kasparov**	Deep Blue	1–0	
3	Deep Blue	Kasparov	½–½	Draw by mutual agreement
4	Kasparov	Deep Blue	½–½	Draw by mutual agreement
5	Deep Blue	**Kasparov**	0–1	Kasparov offered a draw after the 23rd move.
6	**Kasparov**	Deep Blue	1–0	
***Result*: Kasparov – Deep Blue: 4–2**				

The 1997 rematch

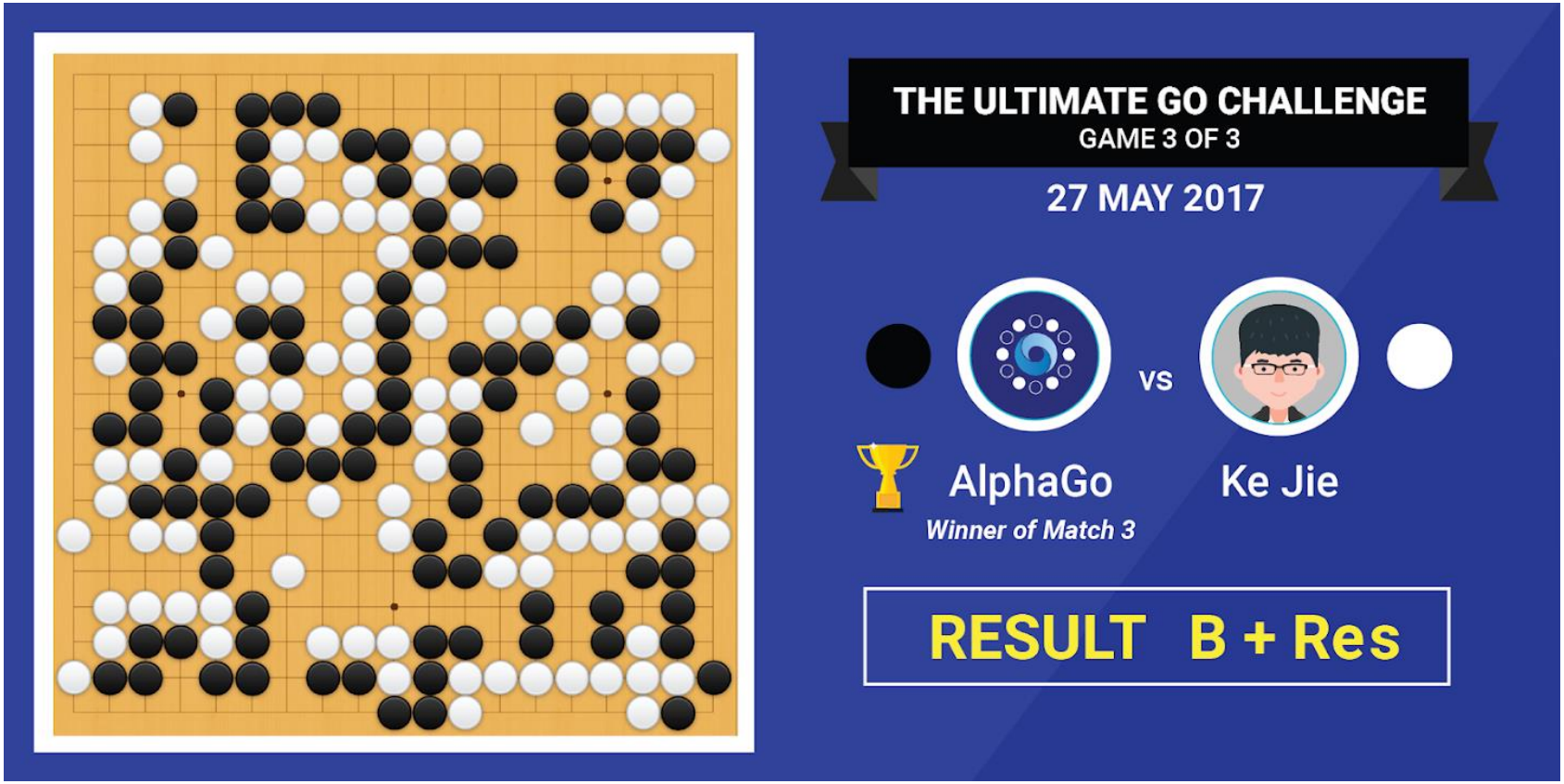
Game #	White	Black	Result	Comment
1	**Kasparov**	Deep Blue	1–0	
2	**Deep Blue**	Kasparov	1–0	
3	Kasparov	Deep Blue	½–½	Draw by mutual agreement
4	Deep Blue	Kasparov	½–½	Draw by mutual agreement
5	Kasparov	Deep Blue	½–½	Draw by mutual agreement
6	**Deep Blue**	Kasparov	1–0	
***Result*: Deep Blue–Kasparov: 3½–2½**				

출처: https://upload.wikimedia.org/wikipedia/commons/b/be/Deep_Blue.jpg

4장 게임 전략 : 적용 사례

AlphaGo is a computer program developed by Google DeepMind in London to play the board game Go.

4장 게임 전략 : 적용 사례



출처: https://deepmind.com/alphago-china

4장 게임 전략 : 적용 사례

출처: http://weekly.ascii.jp/elem/000/000/317/317173/

4장 게임 전략 : 상대가 있을 경우의 대처 방법

- **학습 목표**
  - 체험해 봅시다: Ex8_게임 전략.xlsm

    간단한 카드 게임으로 컴퓨터에 도전!
  - Min-Max 전략
  - αβ 전략

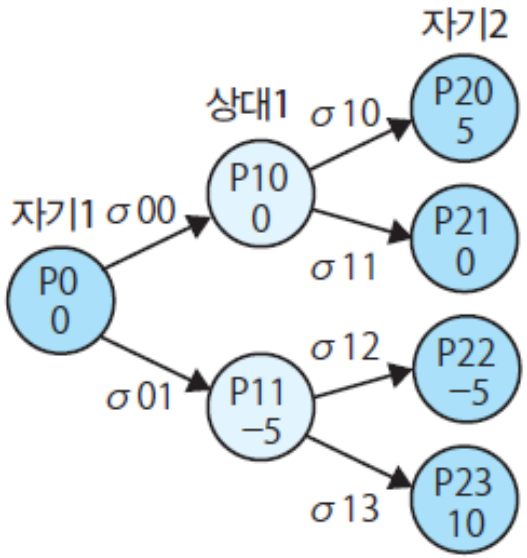
⊙ 체험해 봅시다: 간단한 카드 게임으로 컴퓨터에 도전

- 간단한 카드 게임으로 αβ 전략 시뮬레이션을 실행.
- 장기와 바둑처럼 상호 간의 수와 상태를 아는 것을 전제로 한 대전 게임으로, 다음과 같은 단순한 게임을 가정.
  - ✓ 게임 상대는 독자와 컴퓨터로, 각각 트럼프 1~13까지 13장의 카드를 가짐.
  - ✓ 서로 한 장씩 임의의 카드를 제시하고 직전에 상대가 제시한 카드와의 숫자 차이(절댓값)를 점수로 하여 합산.
  - ✓ 모든 카드를 제시해 게임이 끝났을 때 점수를 많이 얻은 쪽이 이김.
  - ✓ 먼저 하는 쪽이 최초에 카드를 제시했을 때에는 출발점이 되는 숫자를 미리 설정해 두고 그 숫자와의 차이를 점수로 함.

4.1 Min-Max 전략

- 게임 트리에서는 다음과 같은 상태 전이를 수행.
  - **자기 순서**에서는 생각할 수 있는 선택 가지 중에서 **최선**(평가값 최대)인 상태를 취함.
  - **상대 순서**에서는 생각할 수 있는 선택 가지 중에서 **상대 입장에서의 최선**(**자기 입장에서는 최악**)인 상태를 취함.
- **자기 순서에서는 최대(Maximum)**, **상대 순서에서는 최소(Minimum)**의 평가값 상태를 교대로 취하기 때문에 **이것을 Min-Max 전략**이라고 함.

4.1 Min-Max 전략



자기는 자기 순서에서 최선의 수를 둔다(Max)
상대는 상대 순서에서(자기에게) 최악의 수를 둔다(Min)

2수 앞의 가치가 '자기2'의 순서처럼 되어 있을 때,
자기가 '자기1' 순서에서 두어야 할 수는 Max(상대1)
상대가 '상대1' 순서에서 두어야 할 수는 Min(자기2)
Max(상대1) = Max(Min(자기2))=Max(0, -5)=0 → σ00
'자기2' 순서에서 100이라는 가치가 있어도 σ01은 두지 않는다.

그림 7-1 Min-Max 전략

4.1 Min-Max 전략

- 자기 순서에서 2수 앞을 내다본다는 것은, 즉 게임 트리를 2수 앞까지 전개했을 때 2수 앞의 자기 순서에서 Max인 평가값을 취하려 하지만(그림 7-1의 σ01 → σ13), 그 전(1수 앞)의 상대 순서에서는 Min인 평가값을 취하기(그림 7-1의 σ01 → σ12) 때문에 **단순히 2수 앞의 Max를 취하는 것이 좋다고는 할 수 없음**.
- **2수 앞의 Max라는 것은 1수 앞의 Min을 전제**로 한다는 것을 생각하지 않으면 안 됨.
- 즉, 우선 1수 앞의 **상대 순서에서 Min으로 취할 수 있는 후보(그림 7-1의 σ11과 σ12) 중에서 Max를 생각하게 됨**(그림 7-1의 σ00 → σ11).

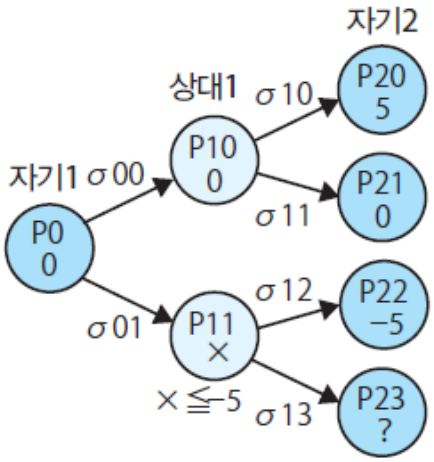
4.1 Min-Max 전략

- 이 방식이 1수 앞만 내다보는 것이라면 상대 순서에서의 평가값만 보고 Max를 취하면 문제가 없겠지만, **여러 수 앞까지의 게임 트리를 생각**하기만 해도 선택 가지 수의 곱셈 차수로 가지가 확장되므로 **탐색 공간이 방대**해짐.
- 몇 수 앞까지의 **모든 가지를 조사**하여 그 중에서 가장 좋은 수를 취한다는 것은 **상당히 어려운 일**임.

4.1 α β 전략

- Min-Max 전략에서는 게임 트리의 탐색 공간이 방대할 때 **모든 가지를 조사하는 것이 어려우므로 조사하는 가지의 개수를 줄이는 것을 생각**해 볼 수 있음.
- 각 상태에 있어서 **관심이 있는 것은 Min 또는 Max 값뿐**이므로 그 이외의 평가값을 갖는 상태는 버려도 상관없음.
- 이 점에 착안하여 **어떤 상태에서 다음 상태를 평가하지 않고 버리는 방법**을 **가지치기(pruning)**라 하며, 가지치기에는 다음과 같은 두 종류가 있음.
  - ✓ **α 가지치기**: 자기 순서에서 하한 보장 값 α보다 작은 바로 다음의 상대 순서 노드를 버린다.
  - ✓ **β 가지치기:** 상대 순서에서 상한 보장 값 β보다 큰 바로 다음의 자기 순서 노드를 버린다.

4.1 α β 전략



자기1 순서에서 P10은 최저 0으로 알고 있고(하한 보장 값), σ12가 −5로 알게 된 시점에서 P11의 평가값은 x≦−5로, Max(상대1)=Max(Min(자기2))=Max(0, x)=0이 된다. 따라서 P11 이하는 더 이상 평가할 필요가 없다(가지치기).

자기 순서에서 평가값의 하한 보장 값 α
→ α보다 작은 바로 다음의 상대 순서는 평가할 필요 없음(α 가지치기)
상대 순서에서 평가 값의 상한 보장 값 β
→ β보다 큰 바로 다음의 자기 순서는 평가할 필요 없음(β 가지치기)

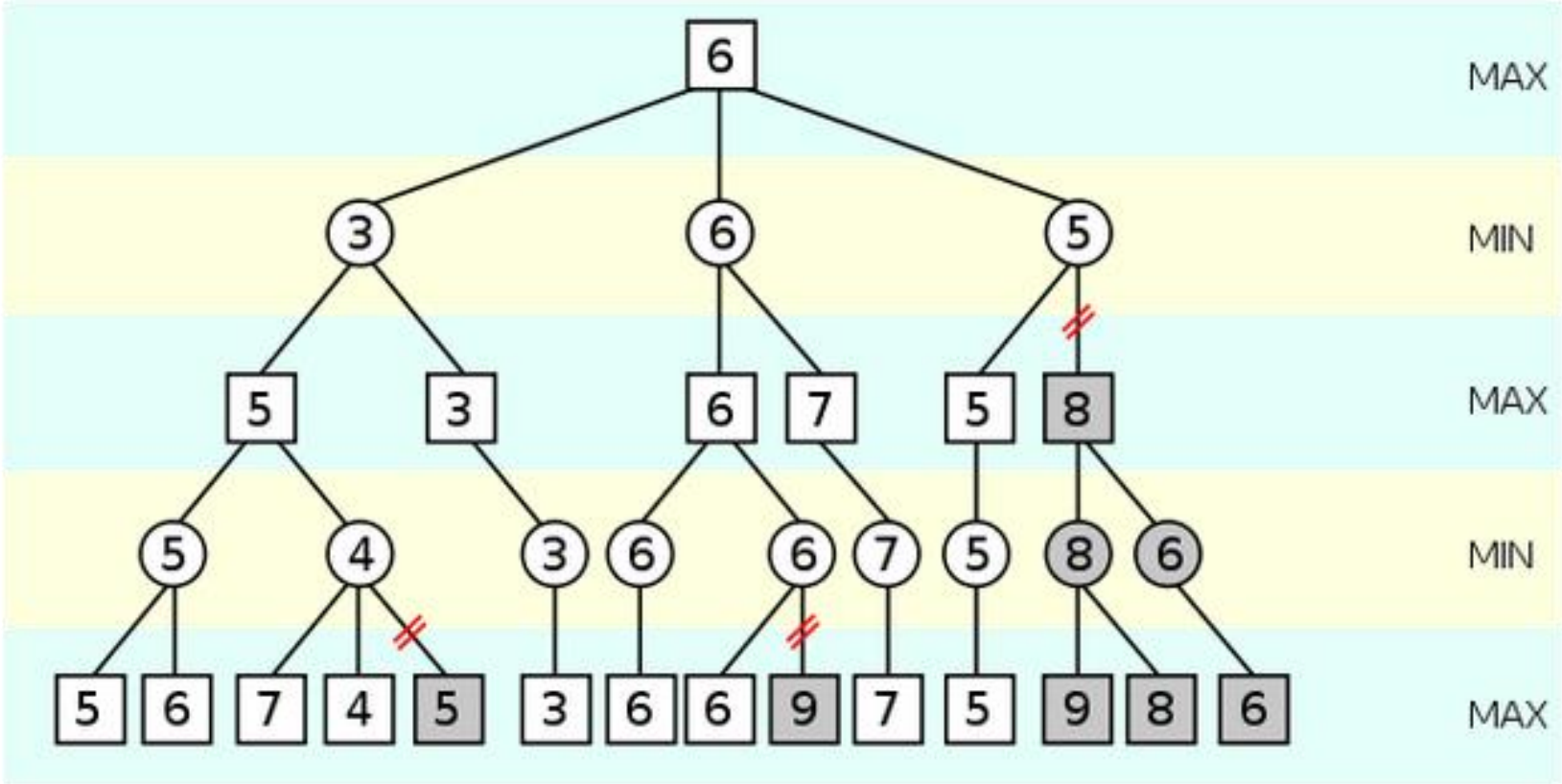
그림 7-2 αβ 전략

- 예를 들어, 위 그림의 2수 앞까지의 게임 트리에서 1수 앞(상대 순서)을 차례로 조사하면 **상태 P10에서 σ11을 취하는 것**을 알 수 있음.
- 다음으로 **상태 P11을 조사**하기 시작하면 **σ12가 P10의 σ11보다 나쁜 평가값**임을 알면 P11은 이미 σ12보다는 좋아질 수가 없기 때문에 **P10보다 반드시 나빠짐**.

4.1 α β 전략

- 결국 지금 P10의 σ11보다 좋아지지 않음. 이 경우 **σ11의 값 0을 하한 보장 값**이라고 함. 만약, P11 이외에 1수 앞의 상대 순서에서 아직 조사할 상태가 남아 있으면, σ11보다 나쁜 선택 가지가 발견되지 않은 경우에는 P10보다 좋아질 가능성이 있기 때문에 계속 조사할 필요가 있음.
- 이 상대 순서의 모든 2수 앞인 자기 순서에서 최종적으로 σ11보다 나쁜 것이 없다면 그 중에서 가장 나쁜 선택 가지를 새로운 하한 보장 값으로 설정하게 됨.
- 그러나 **하나라도 σ11보다 나쁜 것이 발견되면 그 상대 순서는 더 조사해 봐도 소용없게 됨**. 이것이 **α 가지치기**임.
- **같은 방식의 개념을 상대 순서에서 수행**하는 것이 **β 가지치기**. Min-Max 전략에서 이런 가지치기를 수행하는 것을 **αβ 전략**이라 함.

4.1 α β 전략



출처: https://jeromegoodrich.wordpress.com/2016/02/18/tail-recursive-minimax-alpha-beta-pruning/

4.1 α β 전략

- 탐색 공간은 몇 수 앞까지 내다보는지에 따라 달라짐.
- **가지치기가 실패하는 경우**도 있을 수 있음.
- 즉, '**수가 진행되고 미리 예상한 평가값이 잘못된 경우**'에는 이미 잘못된 평가값으로 가지치기를 해 버렸기 때문에 다시 처음으로 돌아갈 수는 없음.
- 이 경우가 가지치기 실패임. **미리 내다본 수의 단계가 많을수록 가지치기는 효과적인 것이 될 수 있지만, 전체 국면을 미리 다 파악할 수 없는 이상 가지치기 실패는 피할 수 없음.**