《AlphaGo脑》课件

授课教师：张宝昌

bczhang@buaa.edu.cn

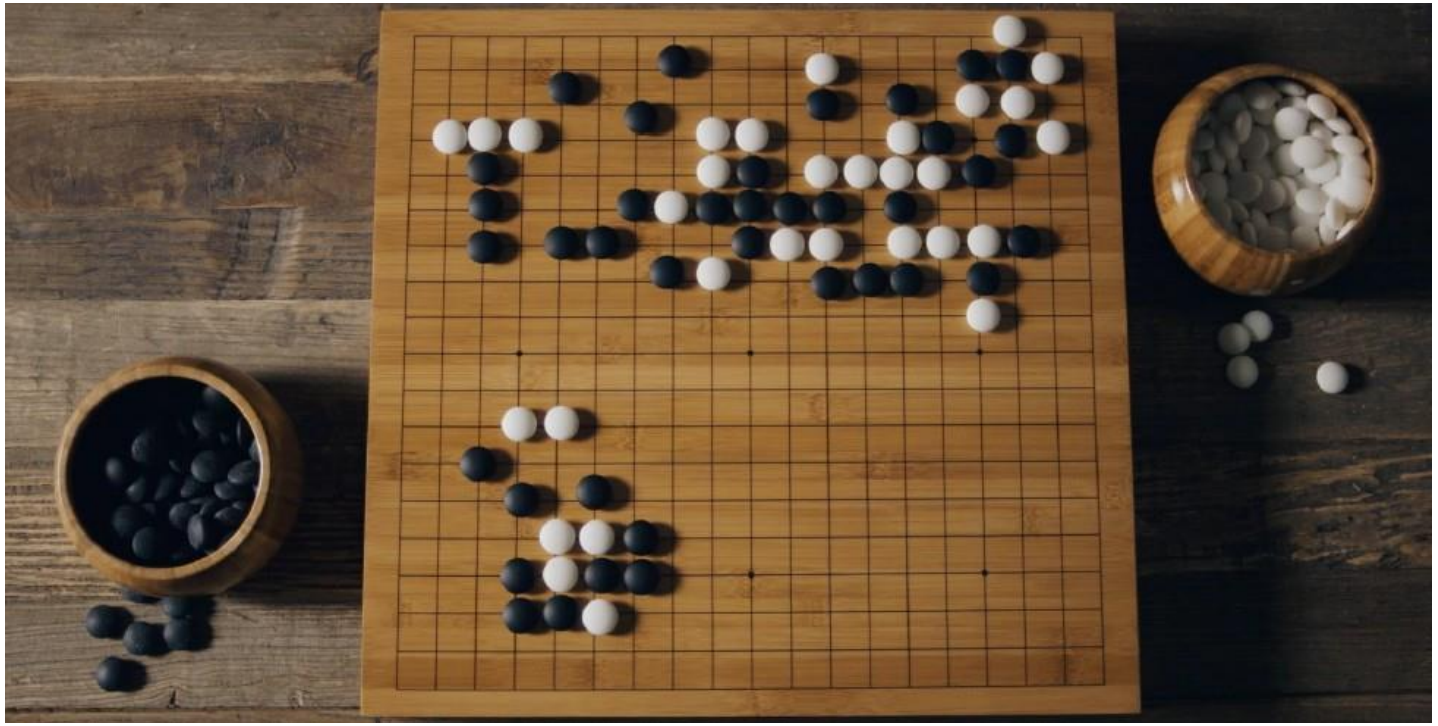
2021年

主要内容

- **介绍-AlphaGO脑**
- **博弈树**
- 蒙特卡洛树
- 策略和价值网络
- 分析

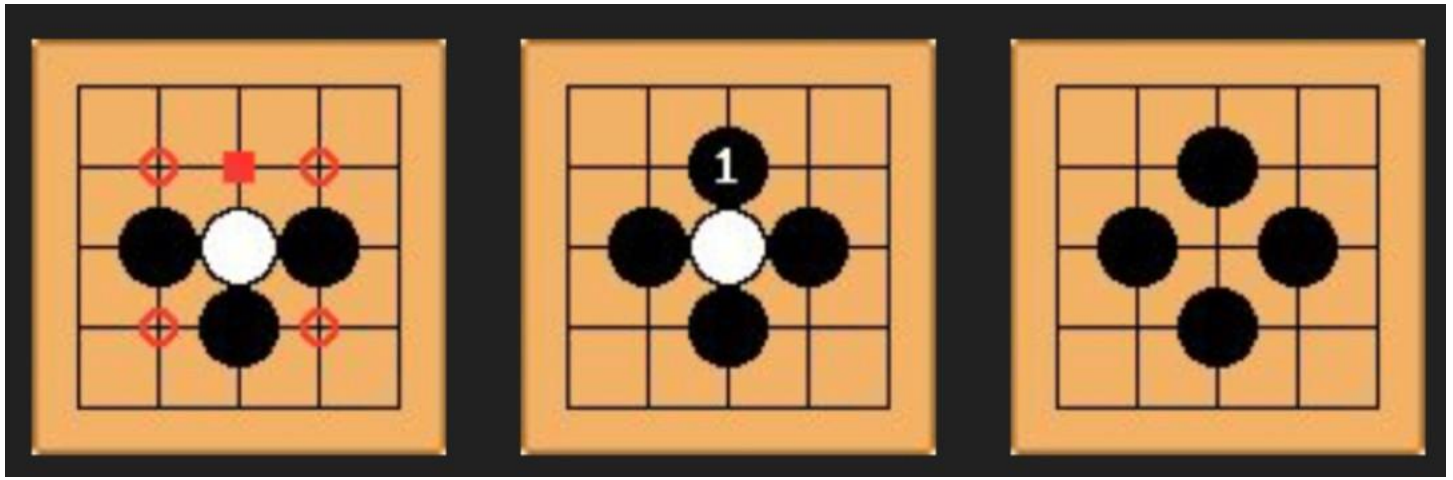
介绍

- 围棋2,500+ 历史
- 4千万爱好者，是最难的棋类游戏之一



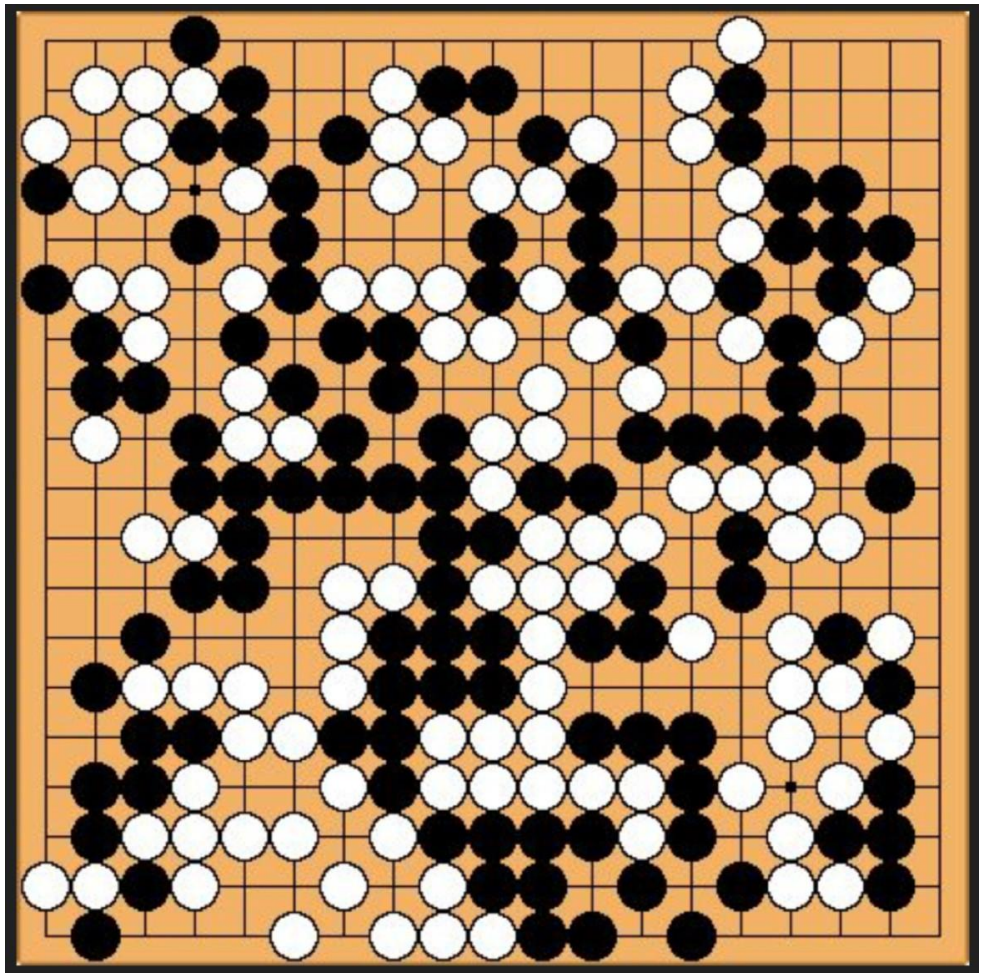
围棋规则

- 19x19 可选择
- 黑白棋子
- 通过围，战胜对手



围棋规则

- 占尽可能多的地盘



围棋为什么极具挑战性?

- 从任何一个位置都有几百种可能的下棋位置
- 需要几百次的操作才能分出胜负
- 终局不容易确定，相比之下象棋更简单

- 依赖于模式识别

博弈树

- 游戏树是一个有向图，节点是棋子，辩时移动的操作
- Tic-Tac-Toe通过博弈树可以得到最优结果
- 复杂度为O(b^d)，b是branching factor，d是深度

博弈树

- 象棋： $b \approx 35, d \approx 80, b^d \approx 10^{80}$
- 围棋： $b \approx 250, d \approx 150, b^d \approx 10^{170}$
- 围棋的搜索次数超过了宇宙的星体!
- 简单的搜索不能解决这一问题

计算机围棋(Computer Go)历史

- 1997: 超人西洋棋w/ Alpha-Beta算法 + 快速计算机
- 2005: 计算机围棋成功
- 2006: 蒙特卡洛树在9x9围棋中的应用
- 2007: 在9x9围棋中达到人类大师水平
- 2008: 在9x9围棋中达到人类特级大师水平
- 2012: Zen项目在19x19围棋中击败以四子优势击败前世界冠军
- 2015: DeepMind研发的AlphaGo以5:0击败欧洲冠军
- 2016: AlphaGo 以4:1的成绩击败世界冠军
- 2017: AlphaGo Zero 以100:0的成绩击败 AlphaGo


nature
THE INTERNATIONAL WEEKLY JOURNAL OF SCIENCE
At last — a computer program that can beat a champion Go player PAGE 484
ALL SYSTEMS GO
CONSERVATION
SONGBIRDS À LA CARTE
Illegal harvest of millions of Mediterranean birds
PAGE 452
RESEARCH ETHICS
SAFEGUARD TRANSPARENCY
Don't let openness backfire on individuals
PAGE 459
POPULAR SCIENCE
WHEN GENES GOT 'SELFISH'
Dawkins's calling card 40 years on
PAGE 462
NATUREASIA.COM
28 January 2016
Vol. 529 No. 7587

AlphaGo背后的技术

- 深度学习 + 蒙特卡洛树搜索+高性能计算机
- 学习了3000万人类专家招式和128000多次自我训练

Image: AP/ Lee Jin-man

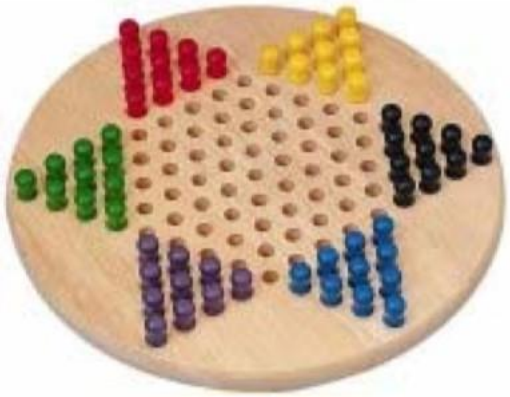
2016年三月:
AlphaGo以4-1的成绩击败李世石

主要内容

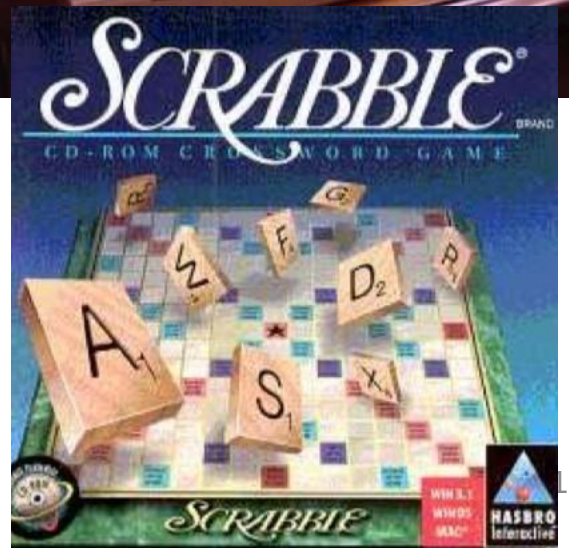
- 介绍
- **蒙特卡洛树**
- 策略和价值网络
- 分析

博弈树搜索

- 适用于具有完全信息的二人零和有限确定性博弈



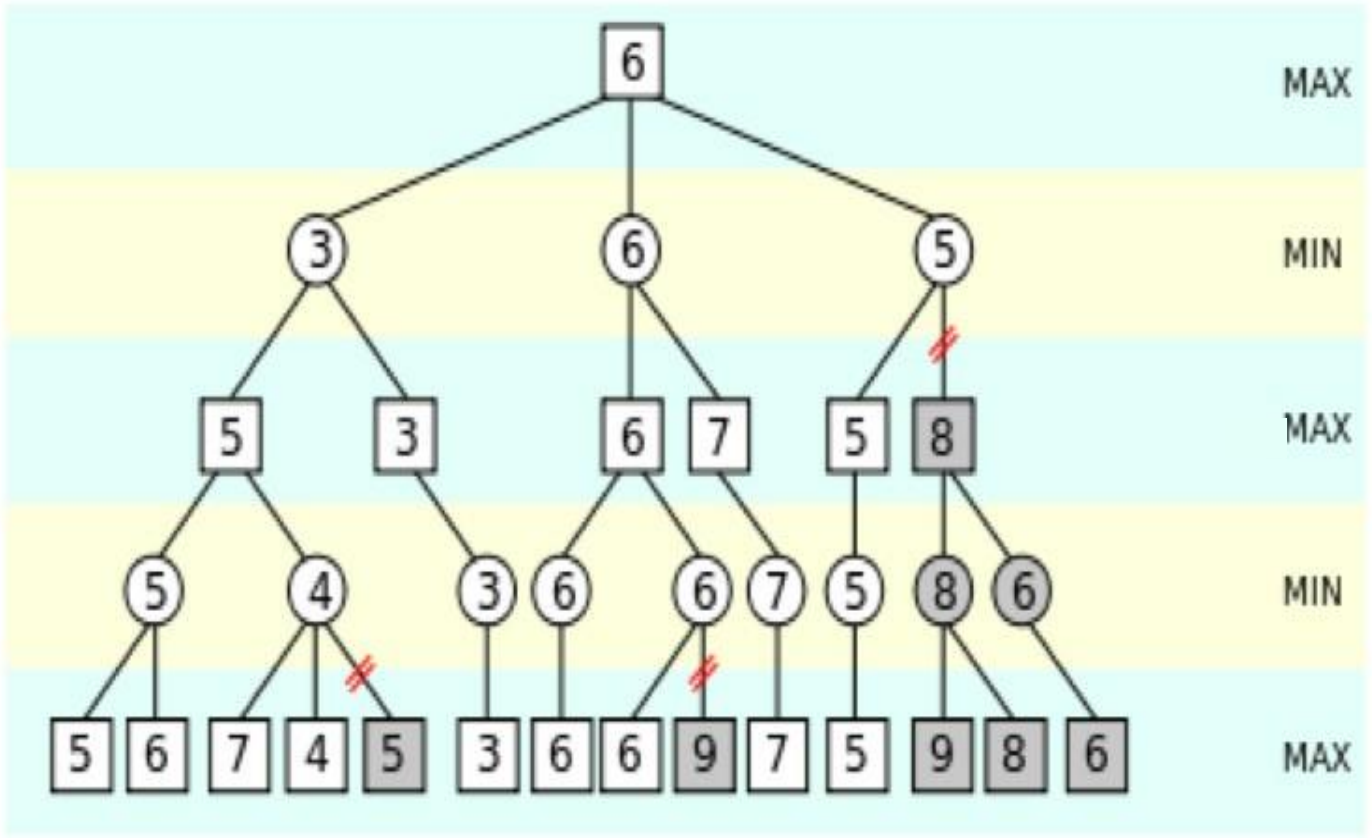


博弈树搜索

- 适用于具有完全信息的二人零和有限确定性博弈

传统博弈树搜索

- 基于Alpha-beta剪枝的极小极大算法



概要

- 博弈
- 最优决策
- 最小最大原理
- α-β 剪枝算法

博弈论

- 数学博弈论
- 经济学的一个分支，将任何多主体环境视为一个博弈，前提是每个主体对其他主体的影响都是“显著的”，而不管这些主体是合作的还是竞争的

→ 诺贝尔奖 美丽心灵 A Beautiful Mind

→ **约翰·纳什是著名数学家，生前一直在普林斯顿任教，在1994年因博弈论获得诺贝尔经济学奖**。他换有精神分裂症，**而电影《美丽心灵》就是讲述纳什一生在博弈论上取得的突破性成就及其与精神分裂症抗争的感人事迹，曾获第74届奥斯卡最佳影片和最佳导演大奖。**

AI中的博弈论 (经典例子) :

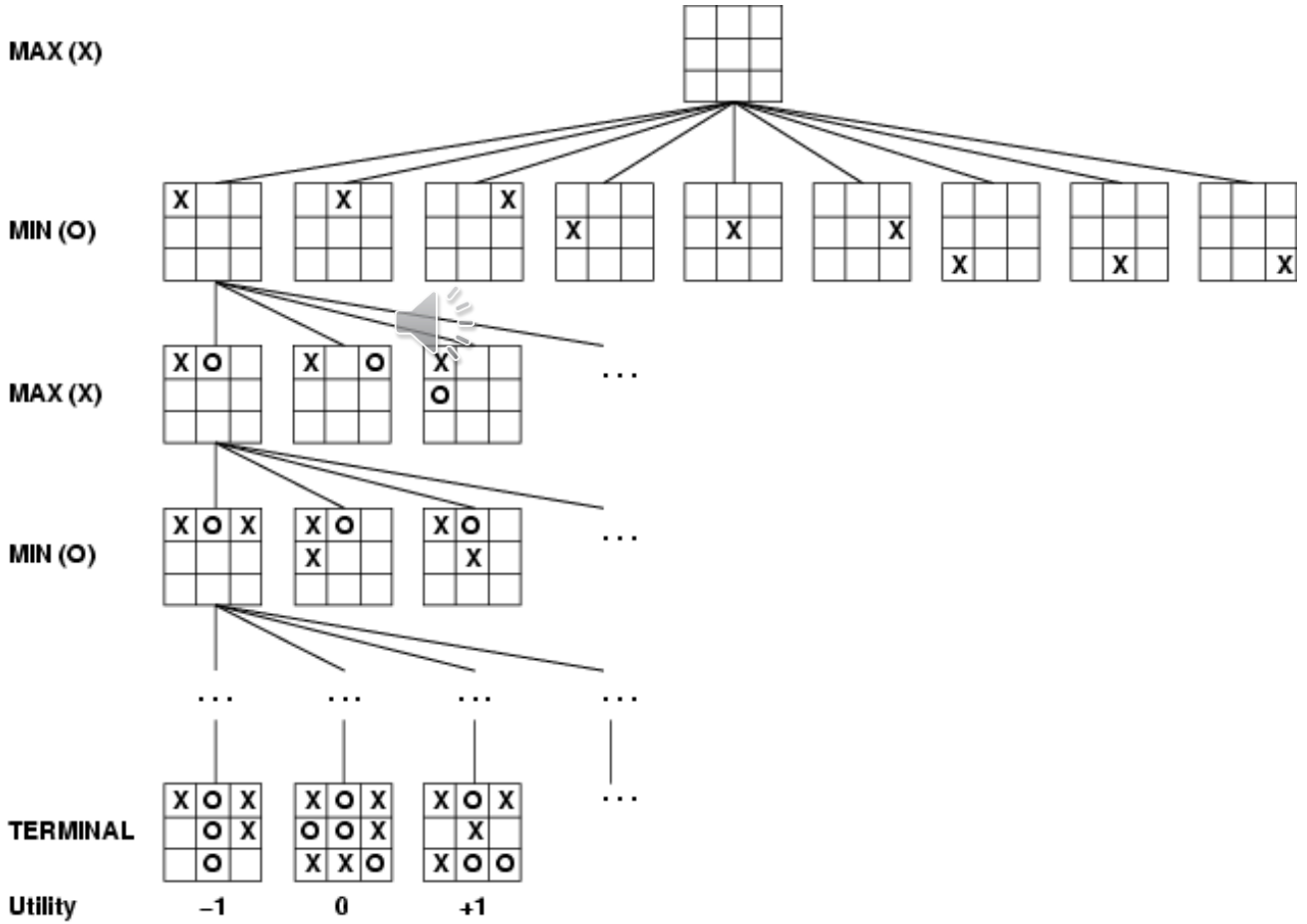
- 双参与者
- 完全信息的零和博弈

-

博弈论 vs. 搜索算法

- 博弈 vs. 搜索问题
- “不可预测”的竞争→ 对每个可能的竞争者做出指定的反应
- 时间限制→ 不可能找到确切的结果，必须求近似解

博弈树示例:
井字棋Tic-Tac-Toe



Max的角度下的树

极小极大算法

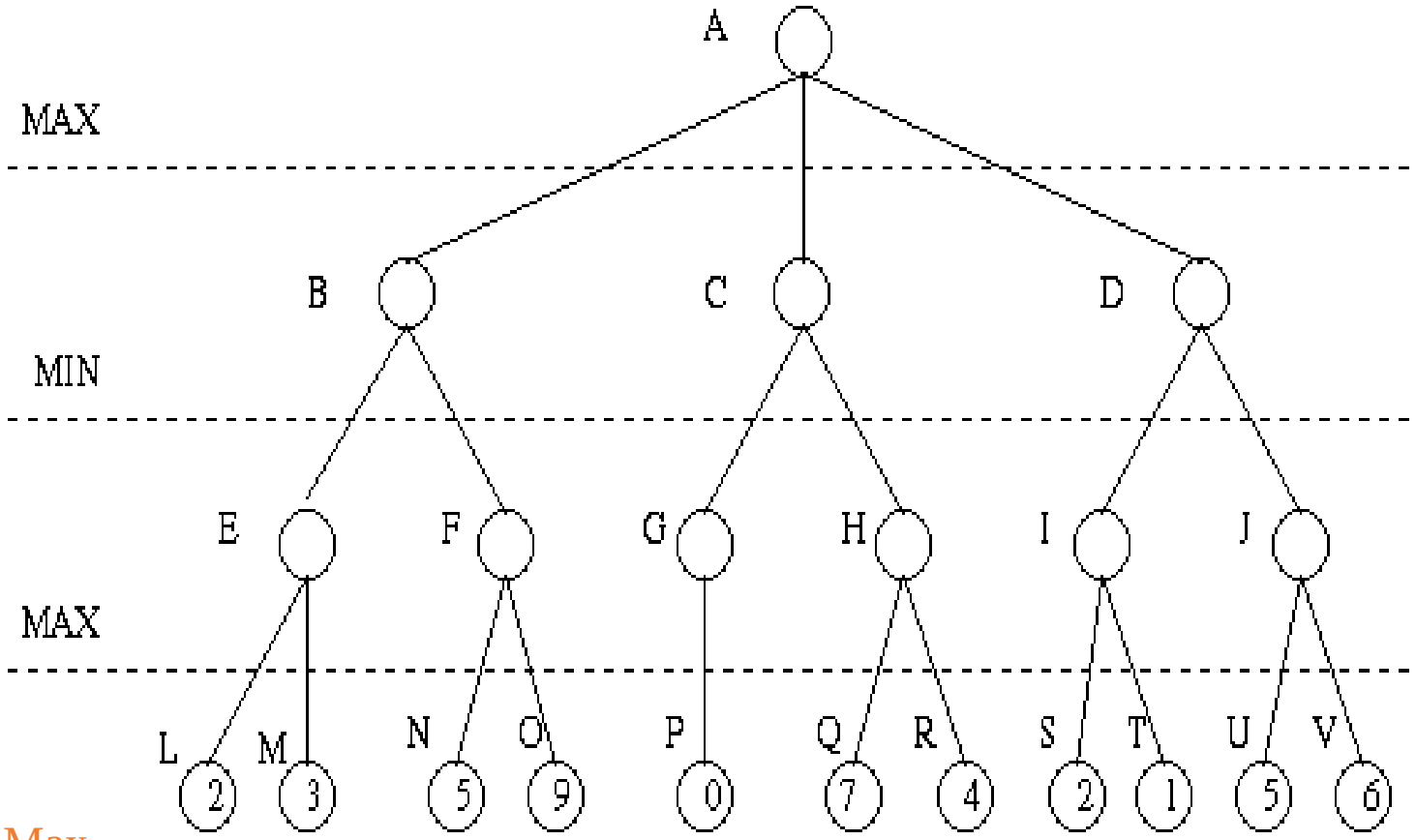
- 对决定论的完美诠释，二人博弈游戏
- Max会最大化其得分
- Min会最小化Max的得分

- 目标：达到最优的极小极大值
  →根据最佳策略确定最佳可实现收益

极小极大算法

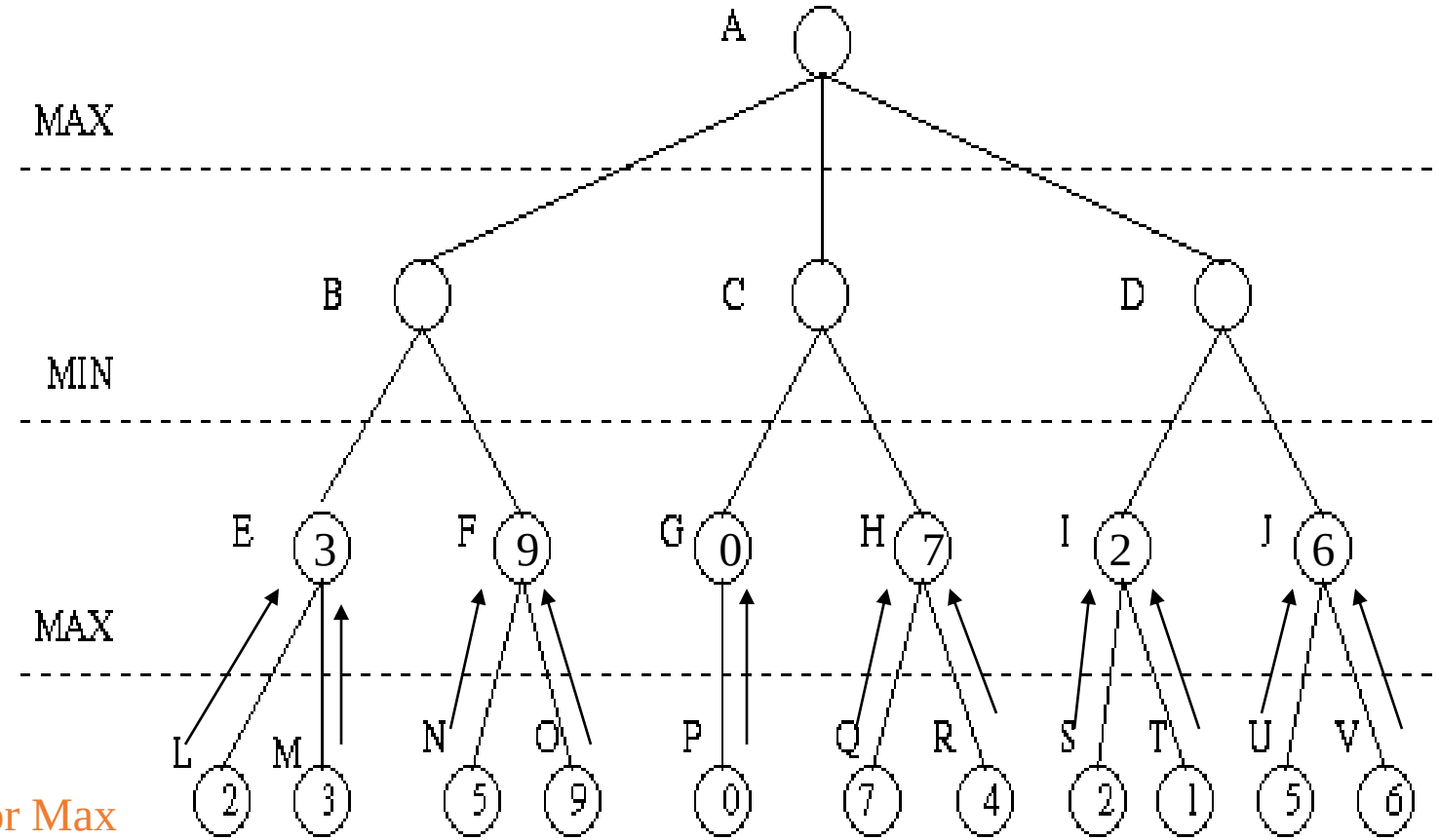
```
function MINIMAX-DECISION(state) returns an action
    v ← MAX-VALUE(state)
    return the action in SUCCESSORS(state) with value v

function MAX-VALUE(state) returns a utility value
    if TERMINAL-TEST(state) then return UTILITY(state)
    v ← −∞
    for a, s in SUCCESSORS(state) do
        v ← MAX(v, MIN-VALUE(s))
    return v

function MIN-VALUE(state) returns a utility value
    if TERMINAL-TEST(state) then return UTILITY(state)
    v ← ∞
    for a, s in SUCCESSORS(state) do
        v ← MIN(v, MAX-VALUE(s))
    return v
```

极小极大算法



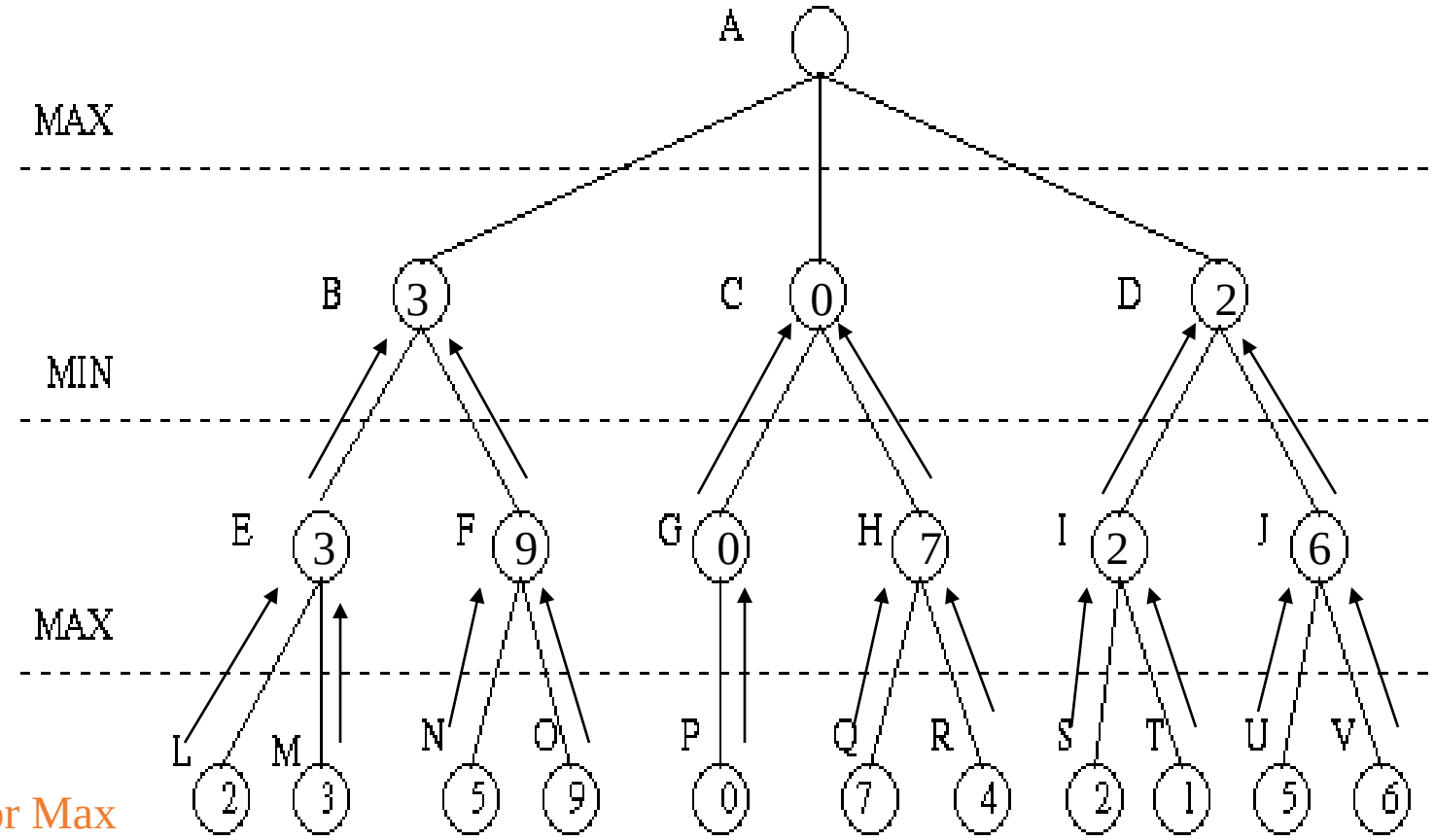
Payoff for Max

极小极大算法



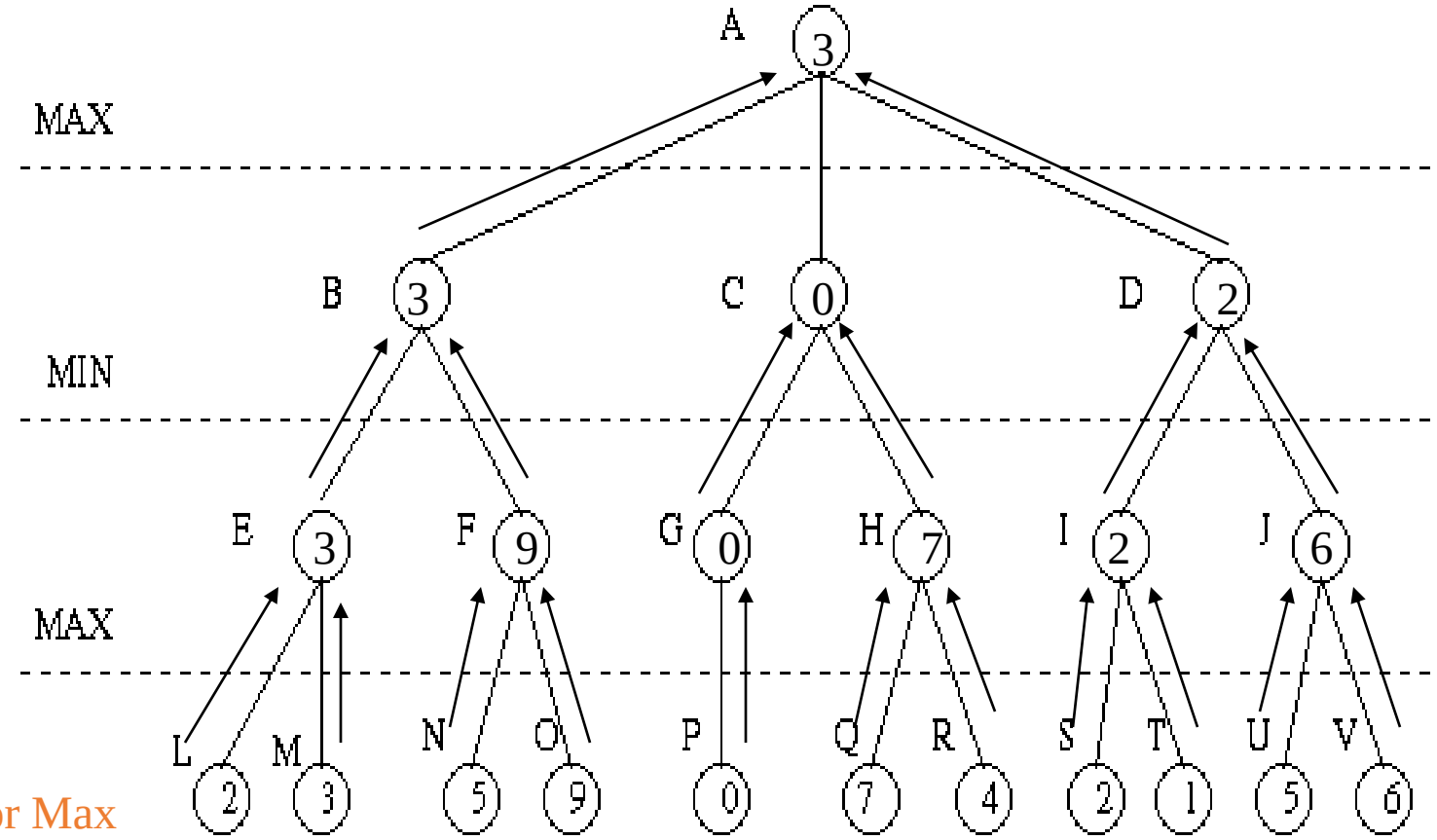
Payoff for Max

极小极大算法



Payoff for Max

极小极大算法



Payoff for Max

极小极大算法

- 极小极大算法的性质:
- 完整性?
- 是的(如果树是有限的)
- 最优性?
- 是的 (对抗最佳对手)
- 时间复杂度? $O(b^m)$　　m – 最大树深; b 分之因子
- 对国际象棋来说, $b \approx 35$, $m \approx 100$ 对“合理”的博弈
  → 确切的解不可行
- 空间复杂度? O(bm) (如果能同时产生所有后继，则深度优先探索)
- m – 树的最大深度; b –合理行动;

极小极大算法

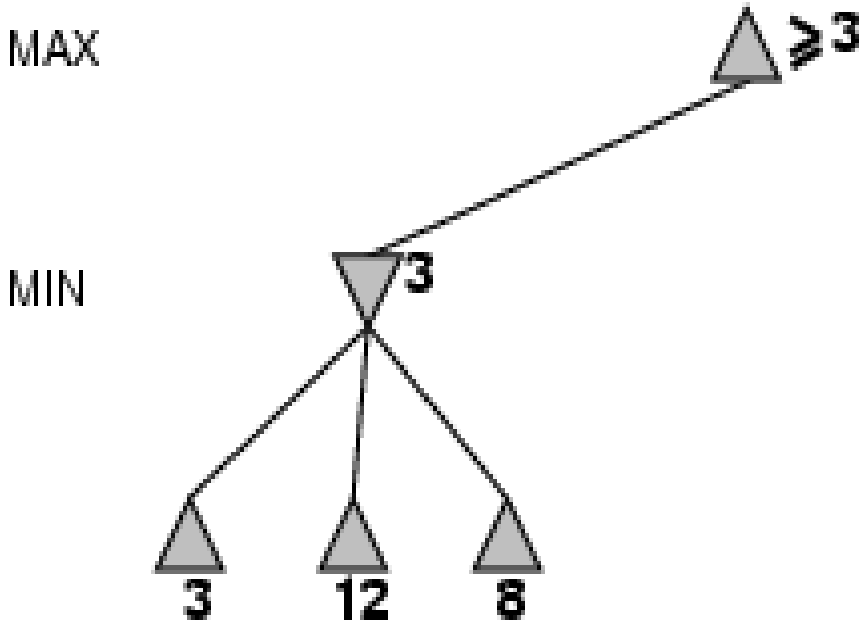
- 限制性
  - 遍历整棵树并不总是可行的
  - 时间限制
- 关键改进
  - 利用评估函数而不是效用函数
    - 评估函数提供了对给定效用的估计

α-β 剪枝

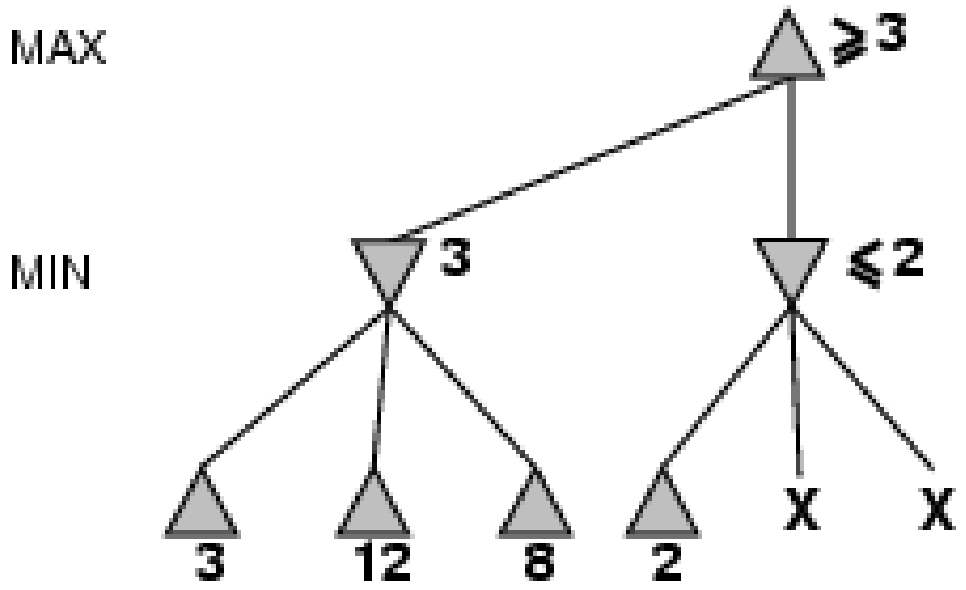
- 我们可以通过减小要检查的博弈树的大小来改进搜索吗?

→ 可以!!! 使用alpha-beta剪枝算法

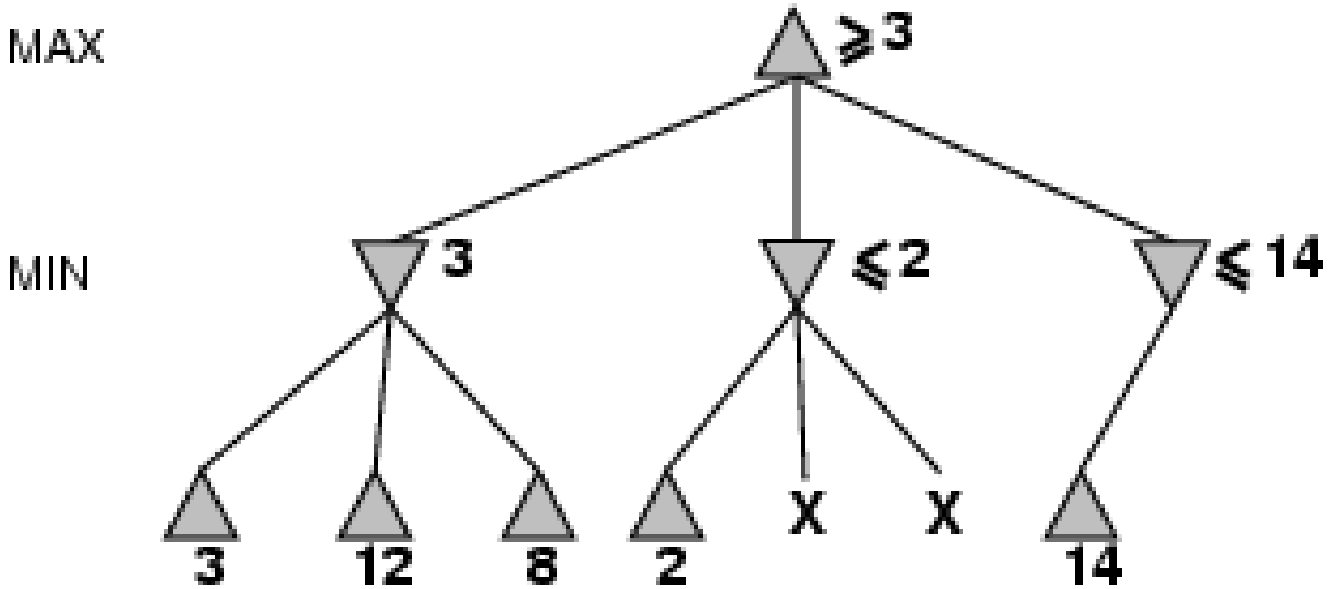
α-β剪枝示例



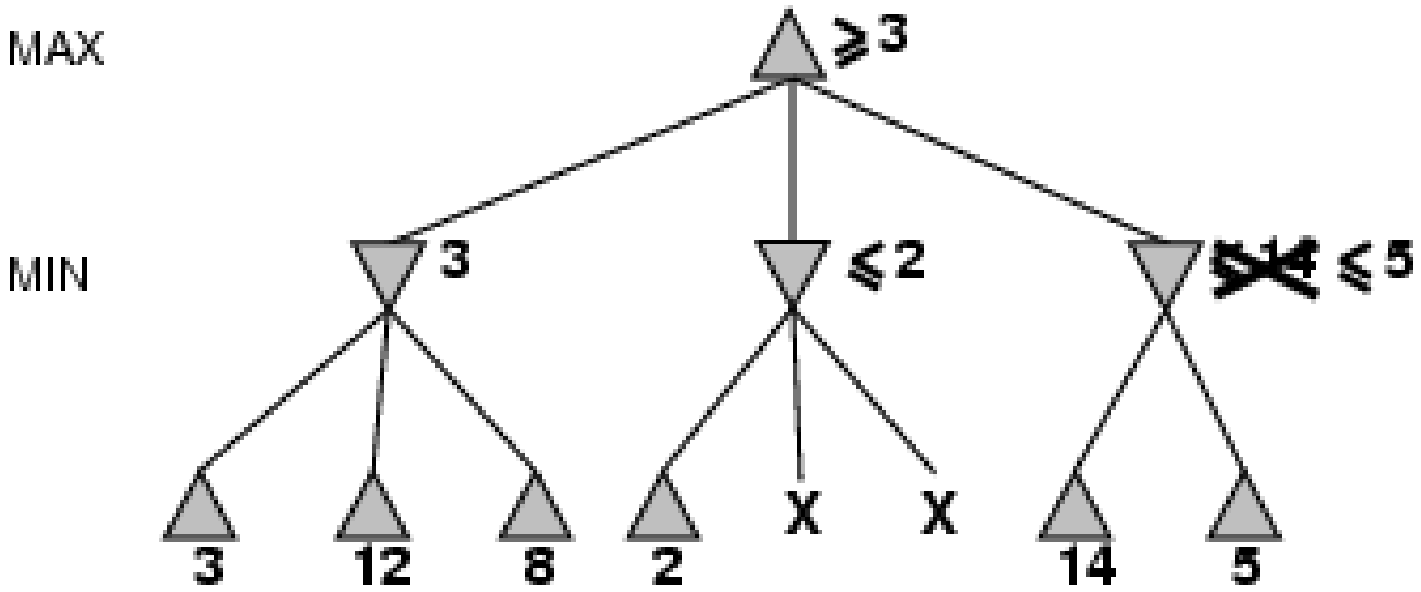
α-β剪枝示例



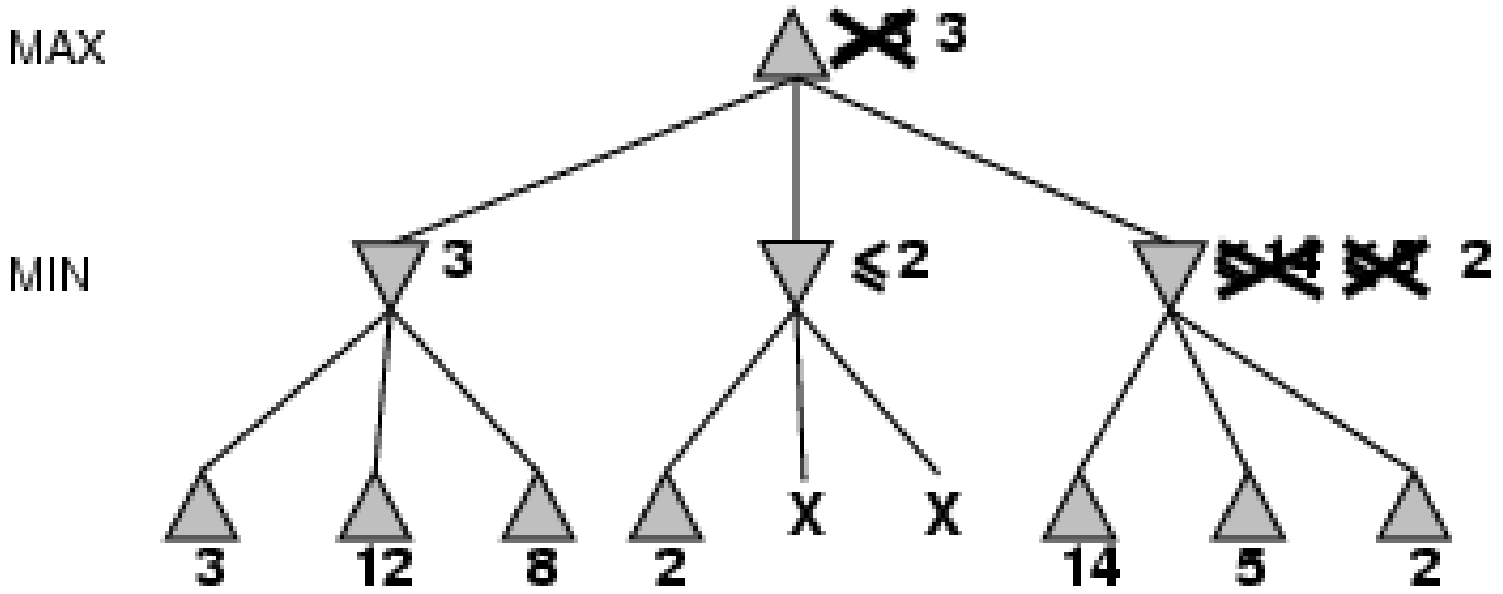
α-β剪枝示例



α-β剪枝示例



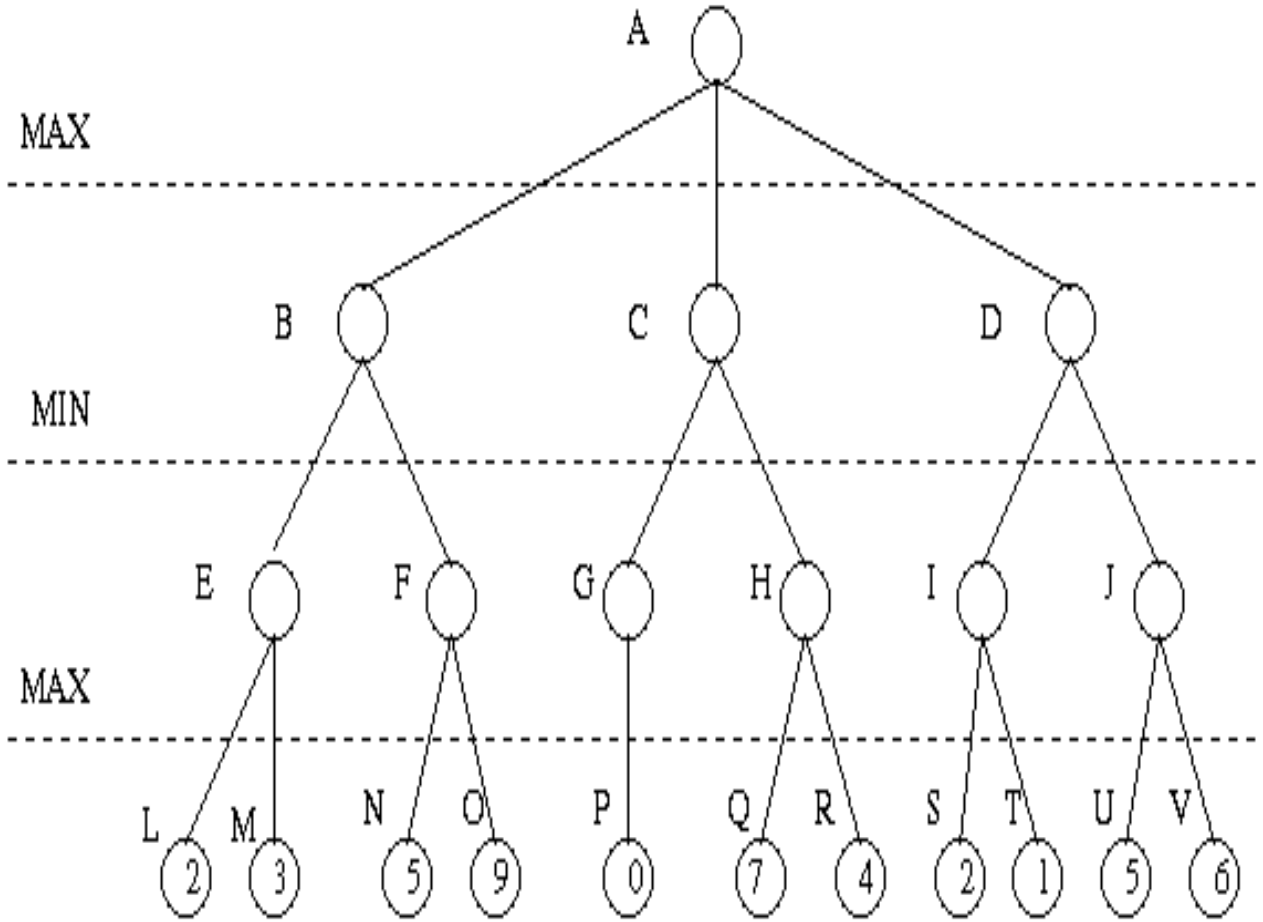
α-β剪枝示例



Alpha-Beta 剪枝示例

1.如果beta值小于等于某个MAX前继的alpha值，则MIN节点下的搜索可能会被alpha修剪。

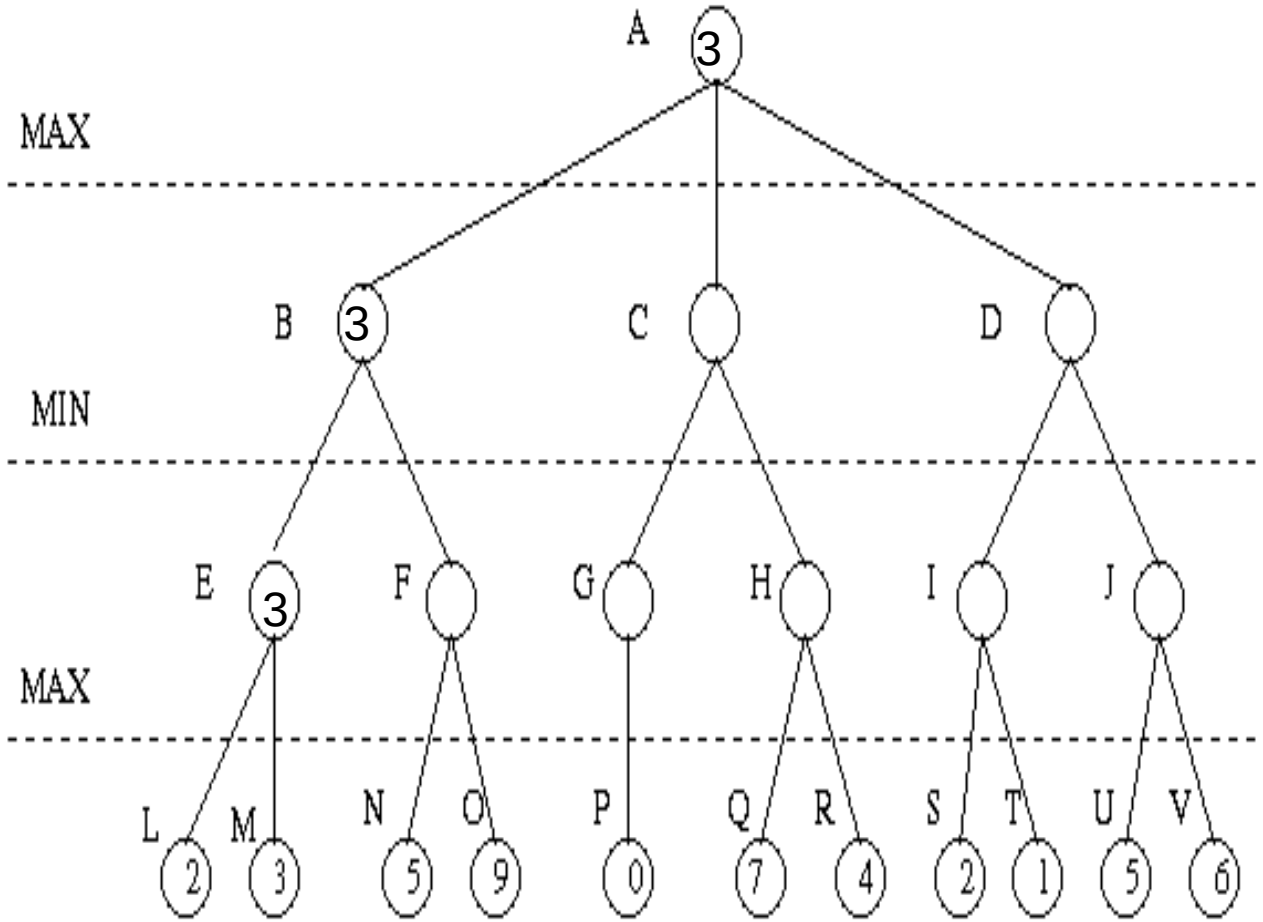
2.如果alpha值大于等于某个MIN前继的beta值，则MAX节点下的搜索可能会被beta修剪。



Alpha-Beta剪枝示例

1.如果beta值小于等于某个MAX前继的alpha值，则MIN节点下的搜索可能会被alpha修剪。

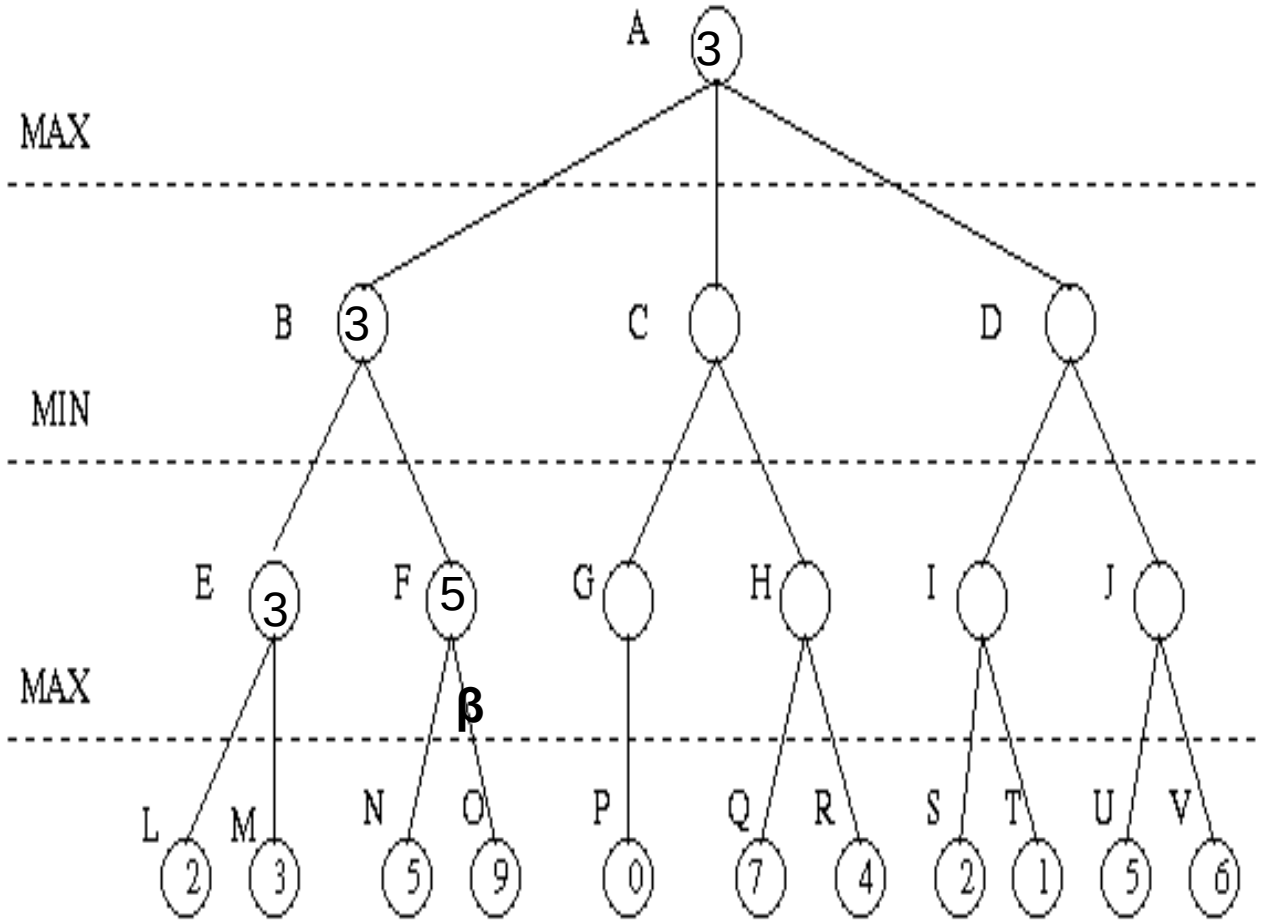
2.如果alpha值大于等于某个MIN前继的beta值，则MAX节点下的搜索可能会被beta修剪。



Alpha-Beta剪枝示例

1.如果beta值小于等于某个MAX前继的alpha值，则MIN节点下的搜索可能会被alpha修剪。

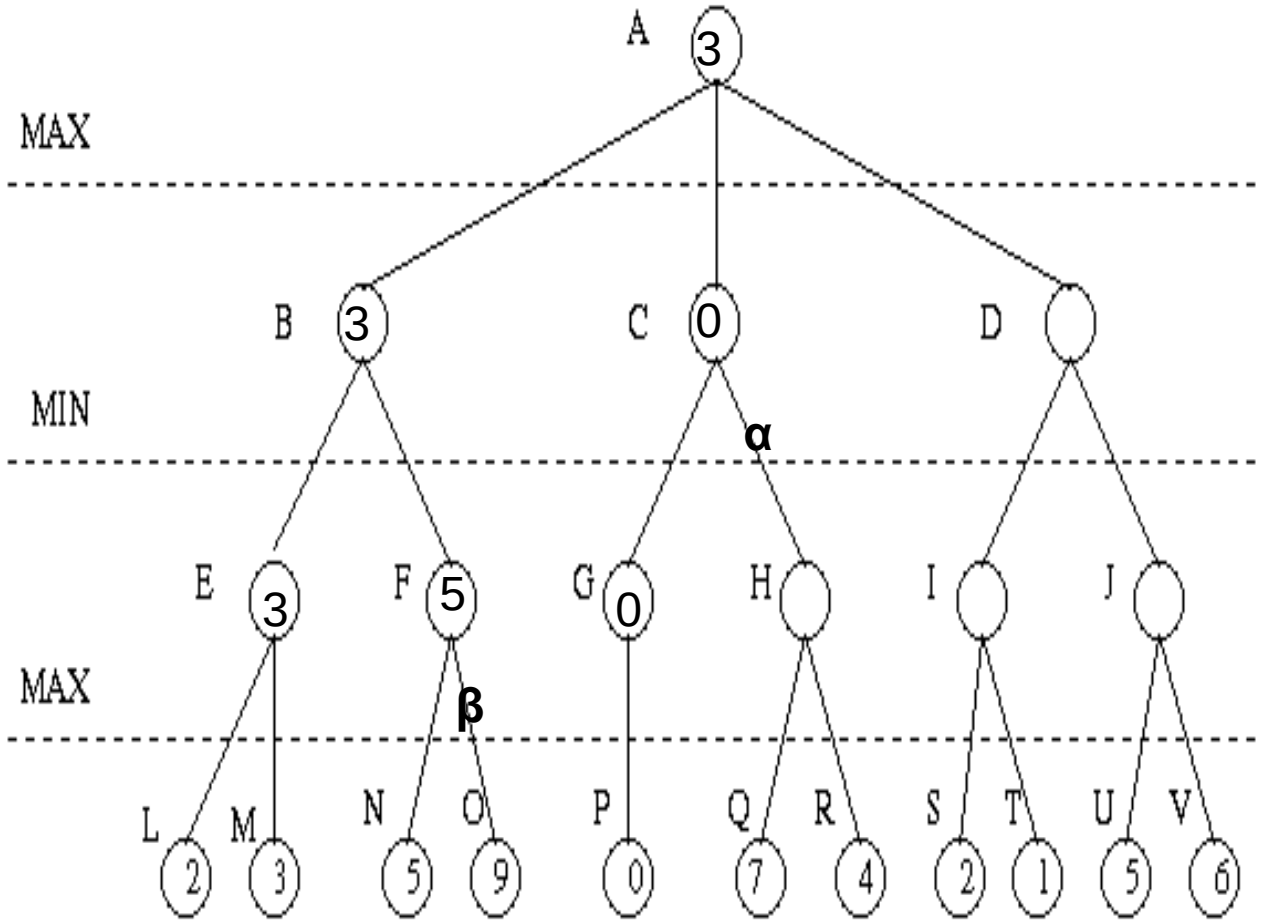
2.如果alpha值大于等于某个MIN前继的beta值，则MAX节点下的搜索可能会被beta修剪。



Alpha-Beta剪枝示例

1.如果beta值小于等于某个MAX前继的alpha值，则MIN节点下的搜索可能会被alpha修剪。

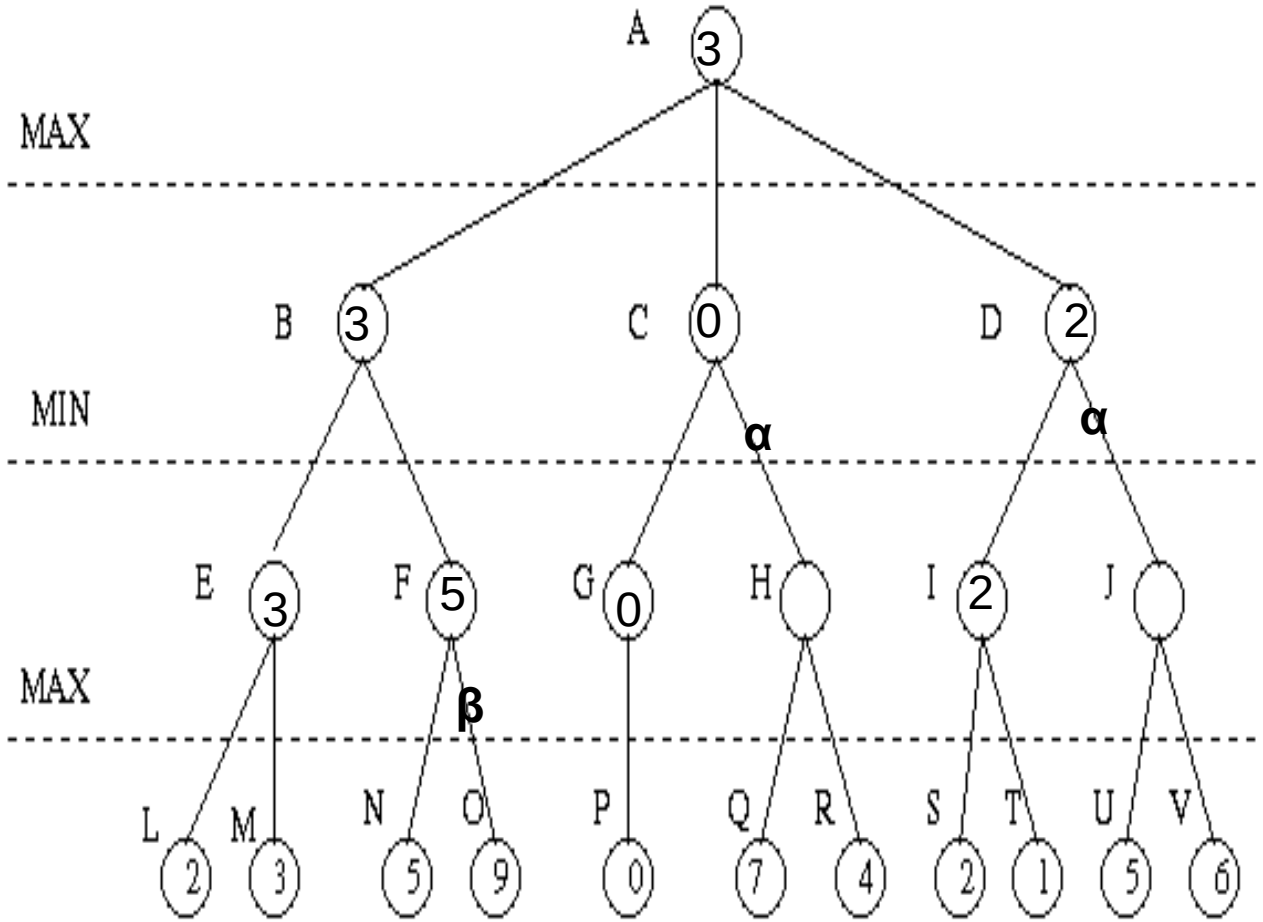
2.如果alpha值大于等于某个MIN前继的beta值，则MAX节点下的搜索可能会被beta修剪。



Alpha-Beta剪枝示例

1.如果beta值小于等于某个MAX前继的alpha值，则MIN节点下的搜索可能会被alpha修剪。

2.如果alpha值大于等于某个MIN前继的beta值，则MAX节点下的搜索可能会被beta修剪。



α-β 剪枝的性质

- 剪枝不影响最终结果
- 好的移动顺序提高了修剪的有效性（例如，国际象棋，先尝试捕获，然后威胁，向前移动，然后向后移动...）
- “完美排序”时，时间复杂度为 $O(b^{m/2})$

  →将alpha-beta修剪可以探索的搜索深度提高一倍

与计算相关的推理值示例（一种元推理形式)

总结

当计算机变得可编程时，游戏博弈是人[illegible]的首要任务之一(例如图灵、香农、维纳下棋)

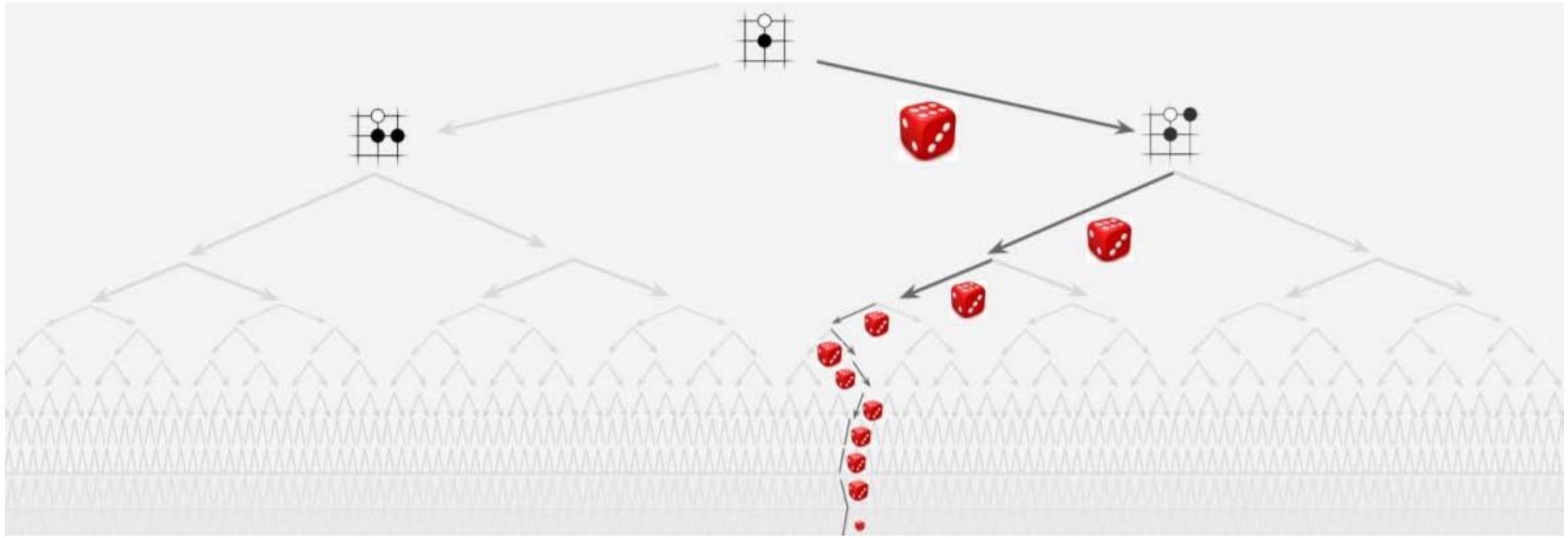
游戏博弈在搜索、数据[illegible]据库、启发式、评估功能和计算机科学的许多领域[illegible]了许多有趣的研究想法。

Alpha-beta剪枝在博弈上的应用?

- 围棋的分之因为太大
  - 平均有250次移动
  - 数量级大于国际象棋分支因子35
- 缺少合适的评估函数
  - 太敏感难以建模：相似的位置可能产生不同的结果

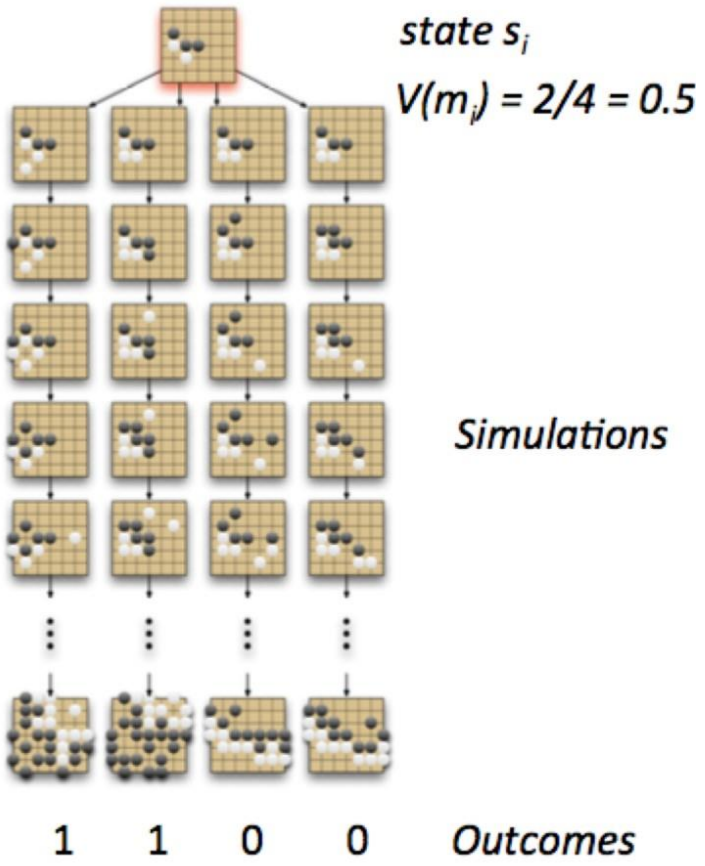
蒙特卡洛树搜索

- 决策树的启发式搜索算法
- 近年来才应用于游戏（不到十年）



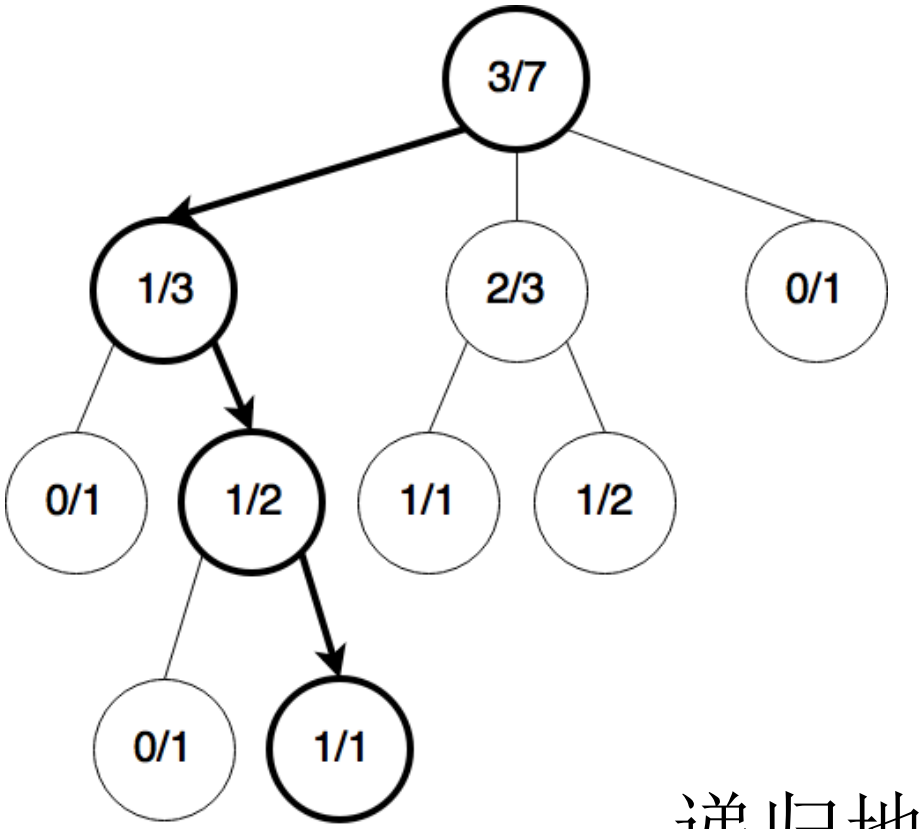
基本思想

- 没有评估函数?
  - 利用随机行动模拟游戏
  - 在游戏最终得分，保持胜利记录
  - 以最大胜率进行移动
  - 重复



蒙特卡洛树搜索

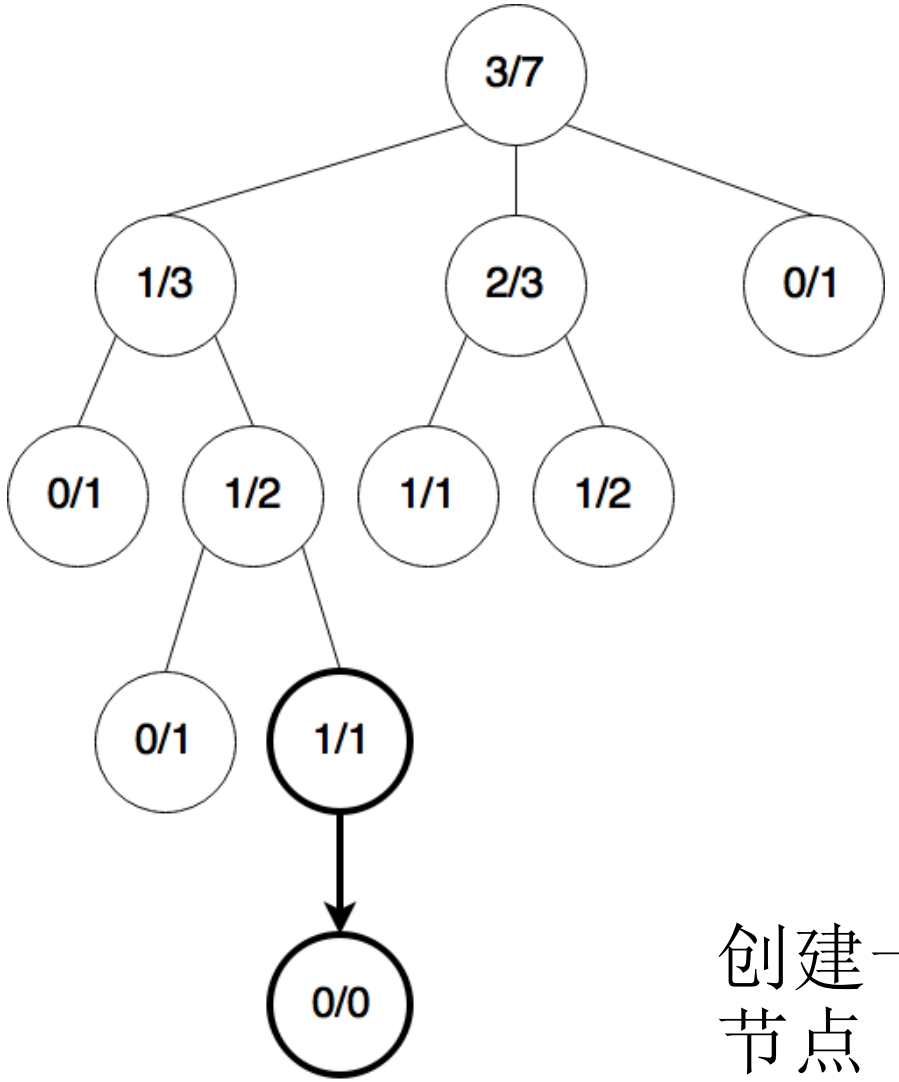
(1) 选择



递归地应用选择策略，
直到找到叶节点为止

蒙特卡洛树搜索

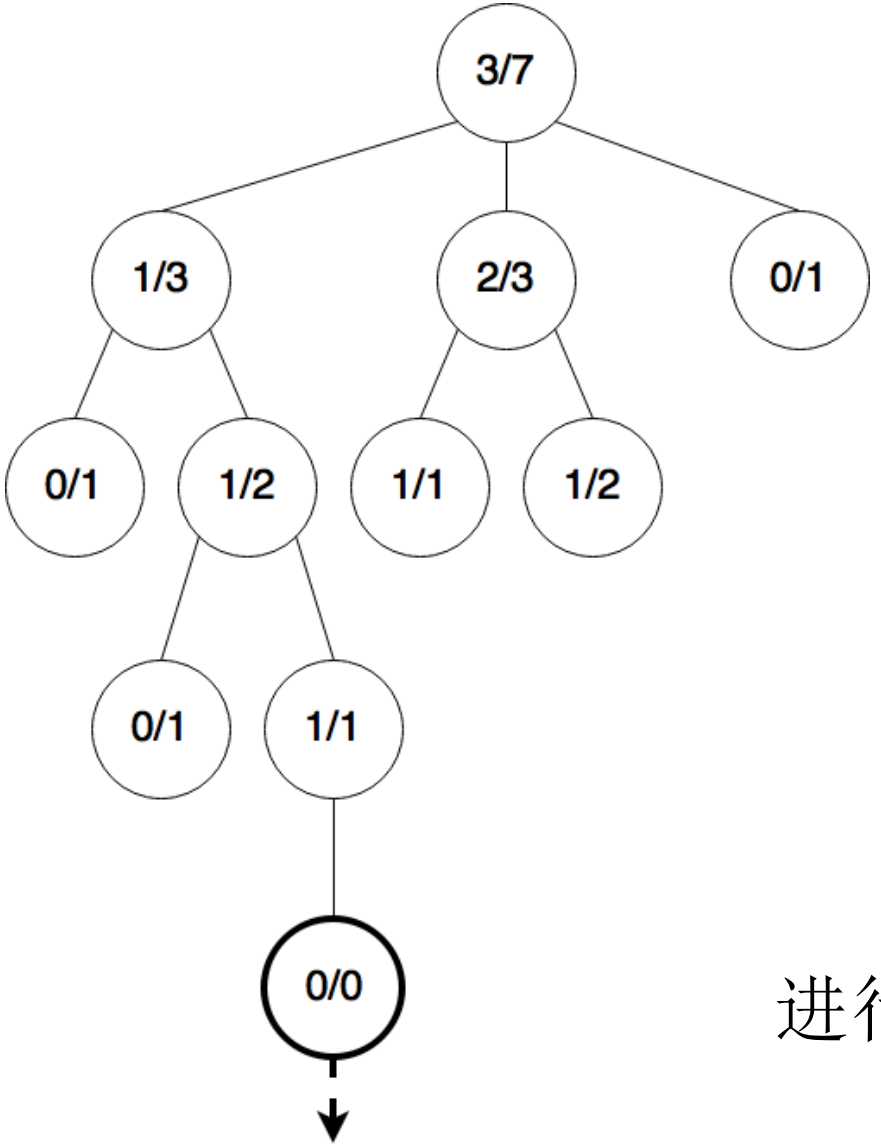
(2) 扩展



创建一个或多个节点

蒙特卡洛树搜索

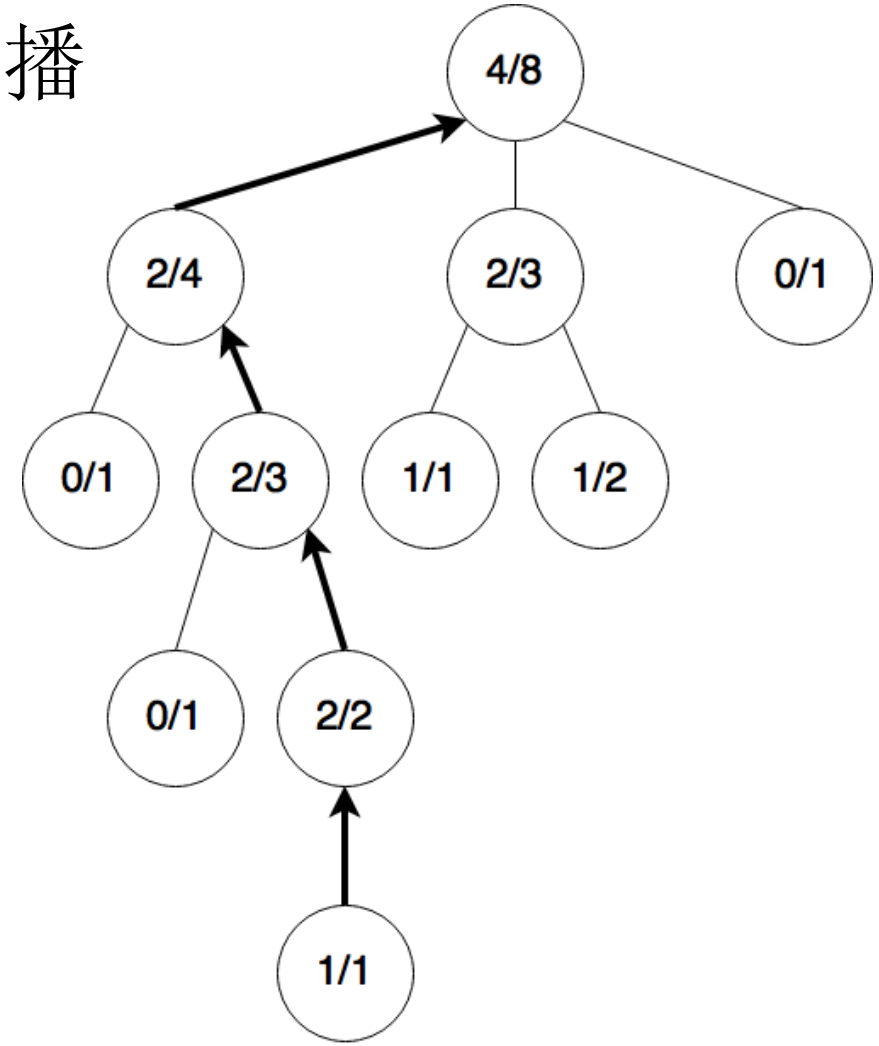
(3) 模拟



进行一次模拟游戏

蒙特卡洛树搜索

(4) 反向传播



简单蒙特卡洛树搜索

- 直接使用模拟作为alpha-beta 剪枝的评估函数
- 在围棋方面存在的问题
  - 单次模拟噪声很大，只有0/1信号
  - 为一次评估进行多次模拟非常慢，例如：国际象棋的典型评估速度为每秒100万次，而围棋每秒只有25次
- 结果: 在计算机围棋领域，MCTS被忽略了十多年

蒙特卡洛树搜索

- 利用模拟结果来引导游戏树的生长
- 我们应把什么动作作为重点?
  - 有更大希望的行动(模拟并且获胜的最多)
  - 在评估不确定度较高(模拟次数较少)情况下的行动
- 似乎有两个相互矛盾的目标
  - 老虎机理论(theory of bandits)能发挥作用

多臂老虎机问题

- 假定
  - 多种摇臂选择
  - 每个摇臂的选择都与其他选择相互独立
  - 每个摇臂都有固定的、未知的收益
- 那只摇臂的平均收益最好?

多臂老虎机问题

P(A wins)=45%

P(B wins)=47%

P(C wins)=30%

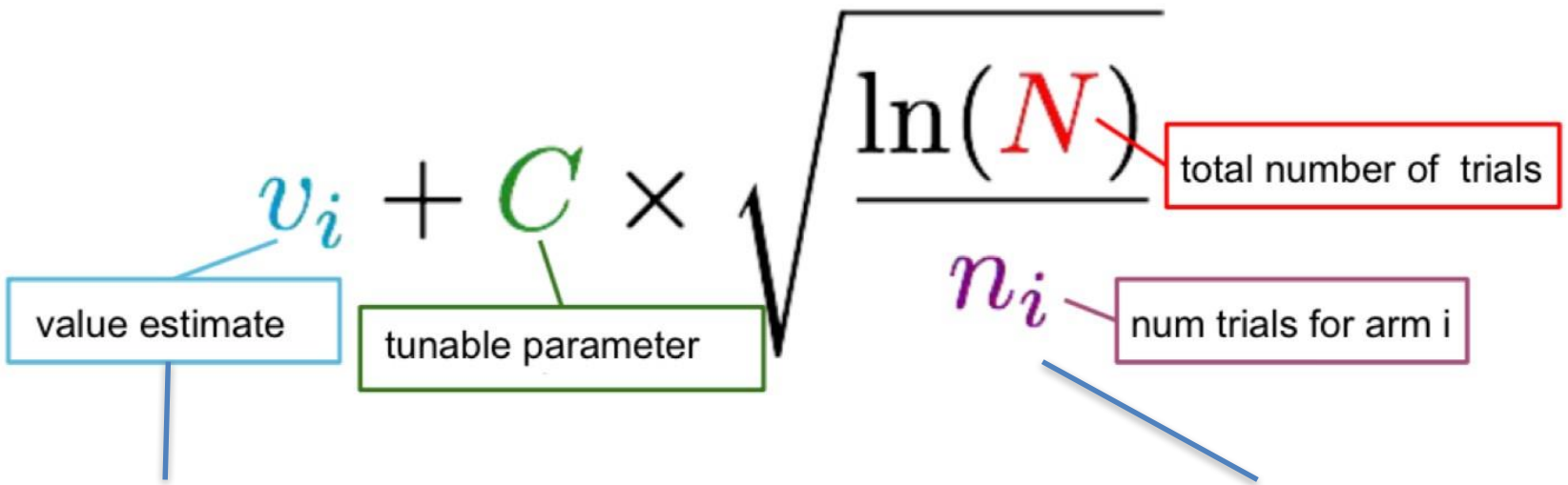
- 但是我们不知道概率，怎么选一个收益高的呢?
- 在无限的时间里，我们可以尝试无限次来估计概率
- 但实际上呢?

探索策略

- 重视经验结果，但是我们不想错过任何潜在的潜力高的摇臂
- 希望更多的利用有潜力高的摇臂，潜力高的摇臂值得进一步探索

置信区间上限算法

- 策略
  - 经验+潜力共同决定:



主要内容

- 介绍
- 蒙特卡洛树搜索
- **策略和价值网络**
- 结论

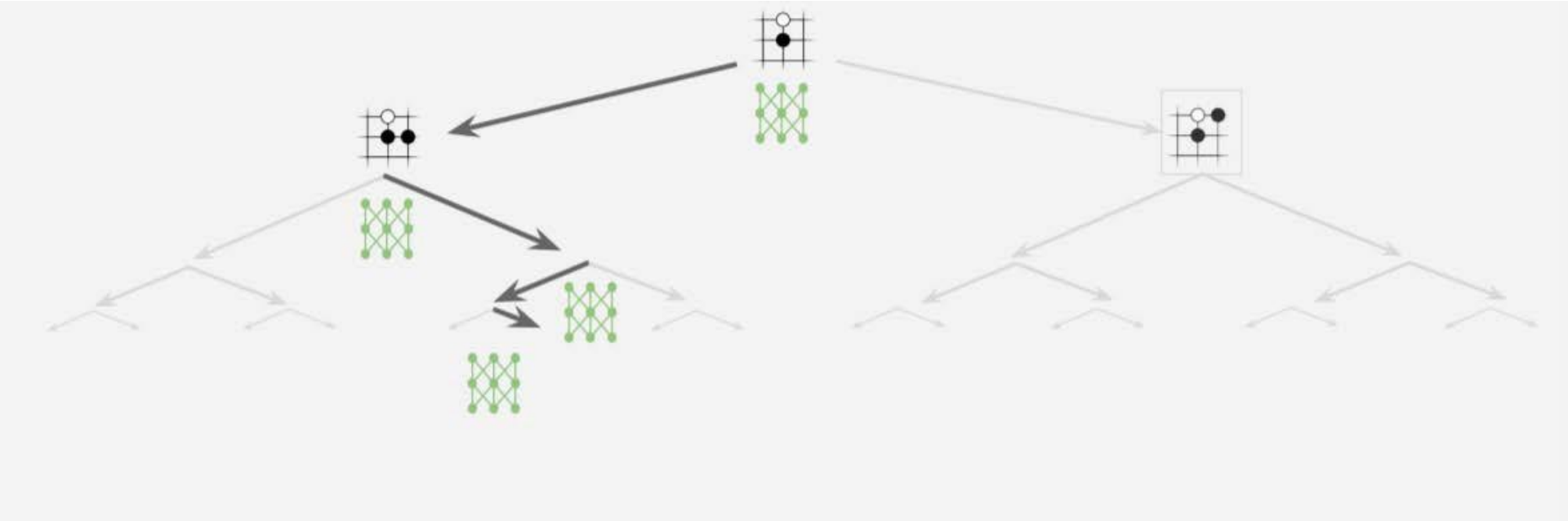
策略和价值网络

- 目标：减小搜索树的分支因子和深度
- 如何实现?
  - 利用策略网络搜索更好（更简单）的行动方式
    - 如何实现?
  - 利用价值网络估计树的下行分支（而不是模拟到最终）
    - 如何实现

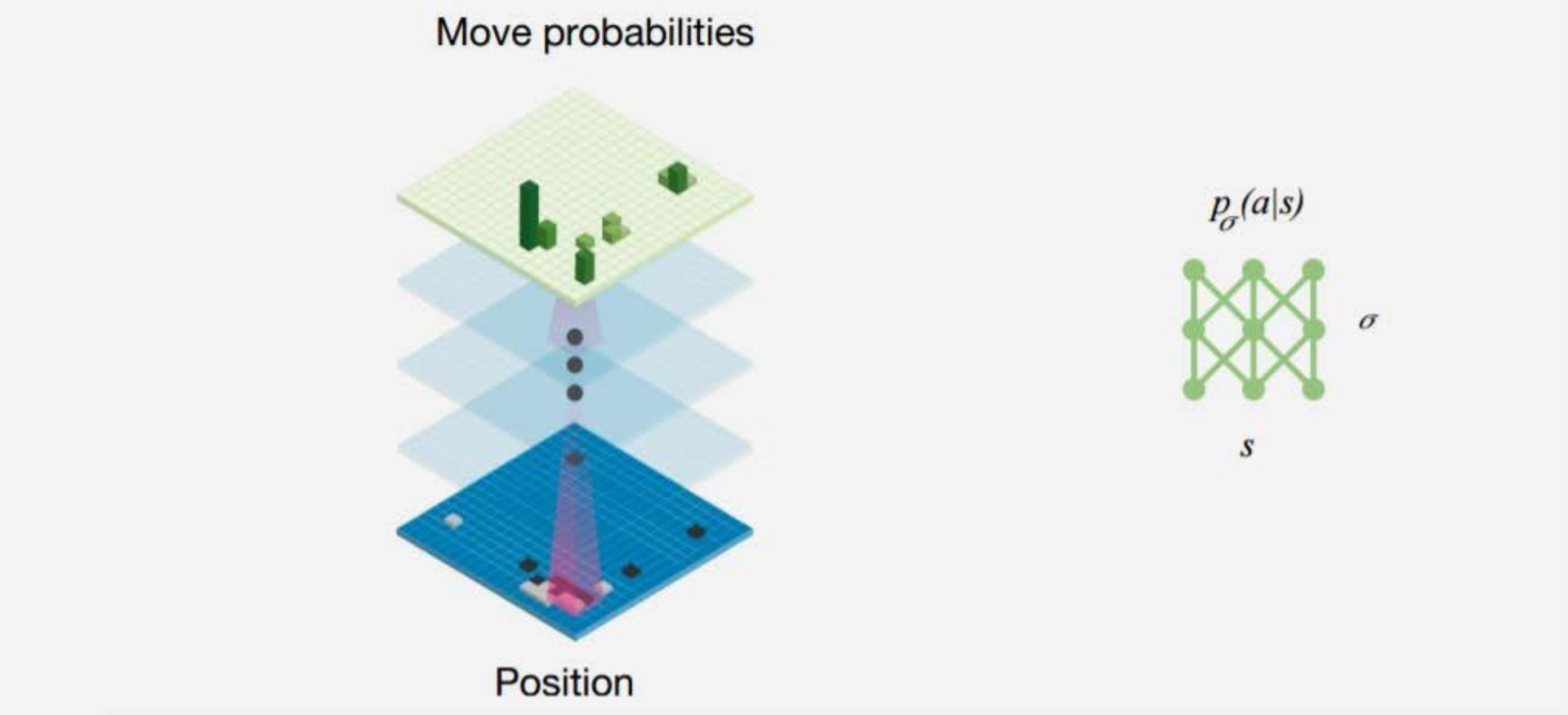
他们没有公布源代码，更没有关于方法的细节。不过我们可以简单地理解这个思路

策略和价值网络

- 减少分支因子: 策略网络



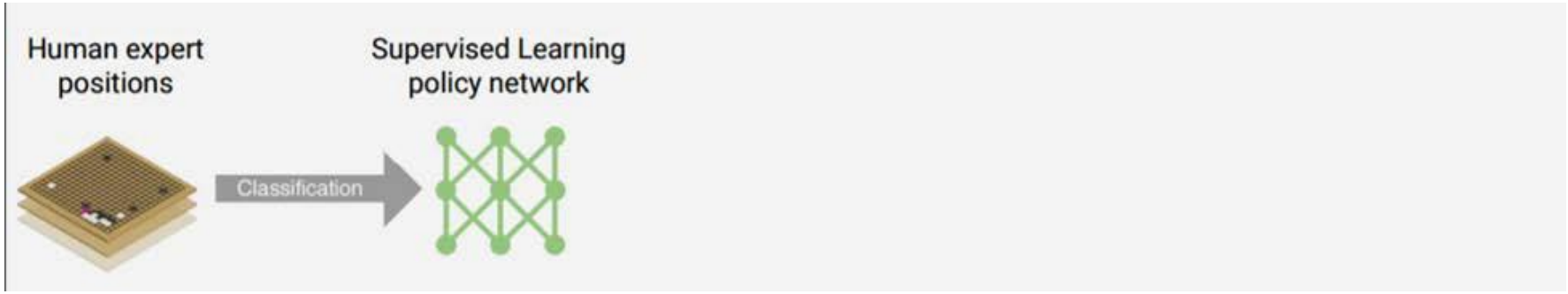
策略和价值网络



预测一次行动成为最佳选择的概率

策略和价值网络

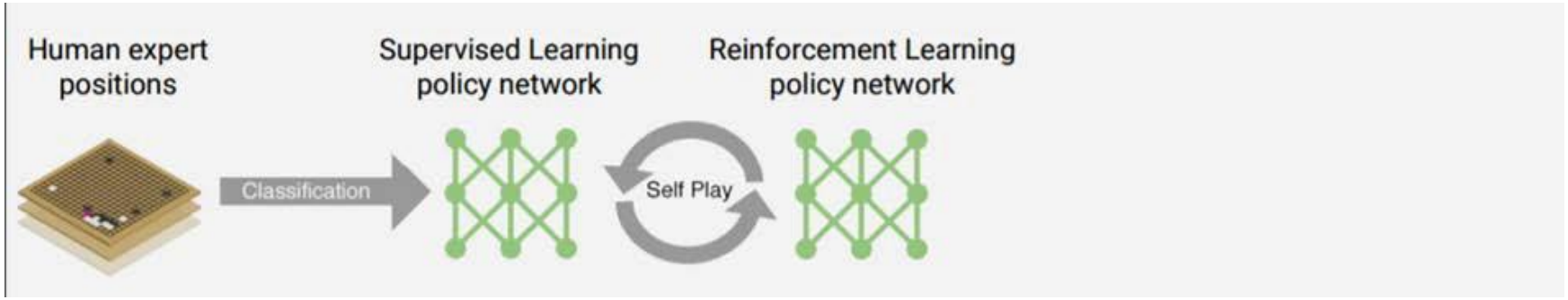
- 监督学习



- 训练数据: 来自人类专家的3000万个游戏数据
- 在某一状态下人类移动的可能性 s
- 训练时间: 4周
- 结果: 预测人类专家移动的准确度为57%

策略和价值网络

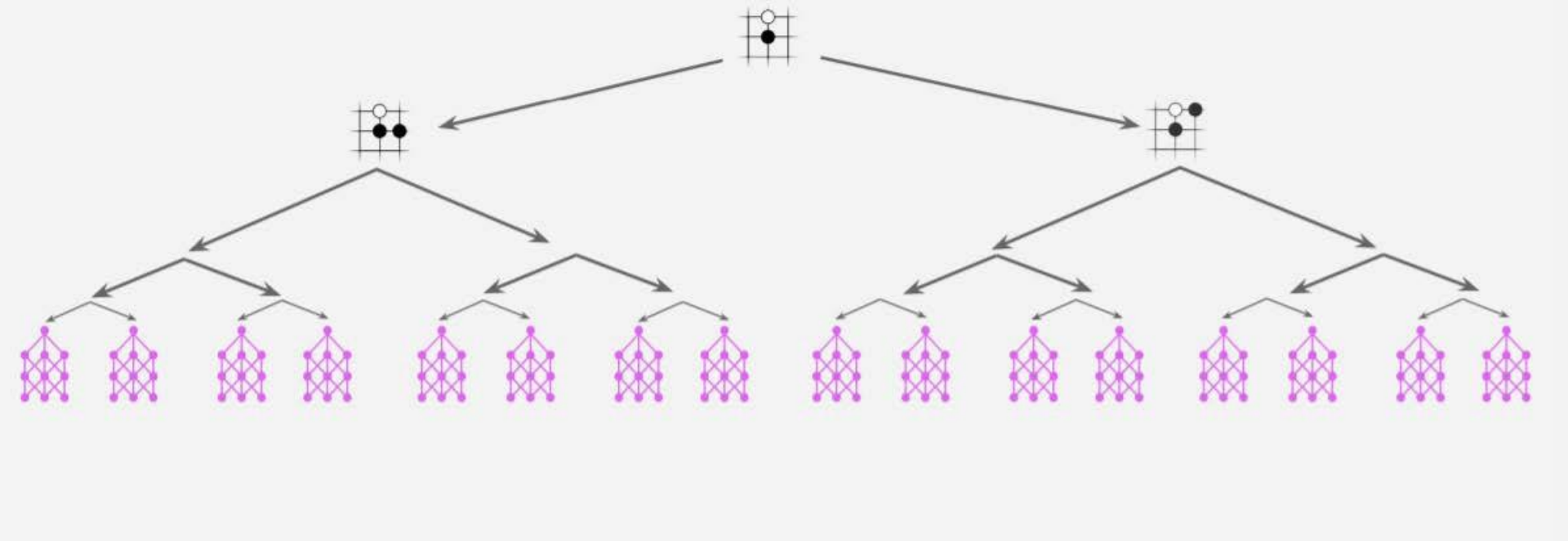
- 强化学习



- 训练数据: 利用策略网络分两阶段进行128,000多种自我训练
- 训练算法: 最大限度地赢得行动地胜利 $\Delta\sigma$
- 训练时间: 1周
- 结果: 赢得了80%以上的比赛胜利vs.监督学习

策略和价值网络

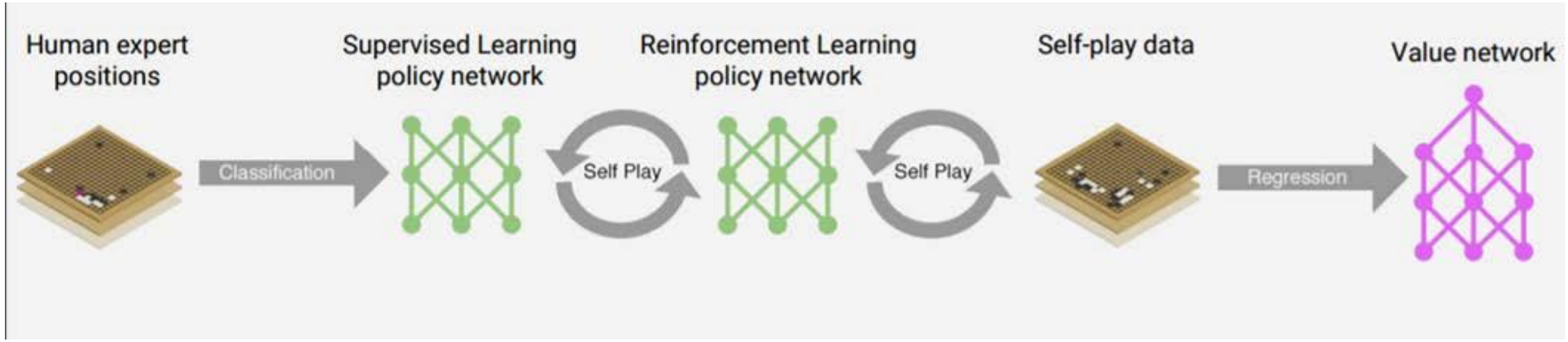
- 减小深度: 价值网络



- 给定宏观状态，估计胜利概率
- 无需模拟到游戏结束

策略和价值网络

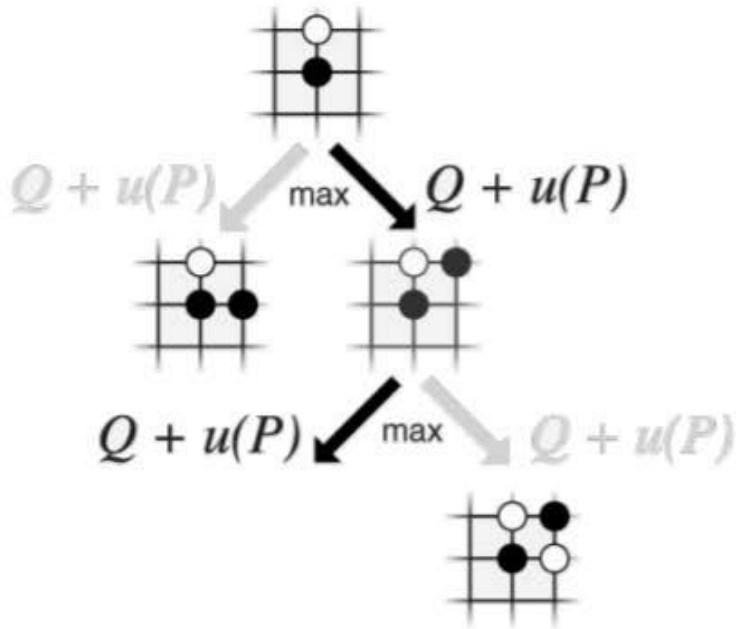
- 强化学习



- 训练数据: 三千万次的自我演算
- 训练算法: 通过随机梯度下降的方法最小化均方误差
- 训练时间：1周
- 结果: AlphaGo做好了与人类专家对战的准备

MCTS +策略/价值网络

- 选择



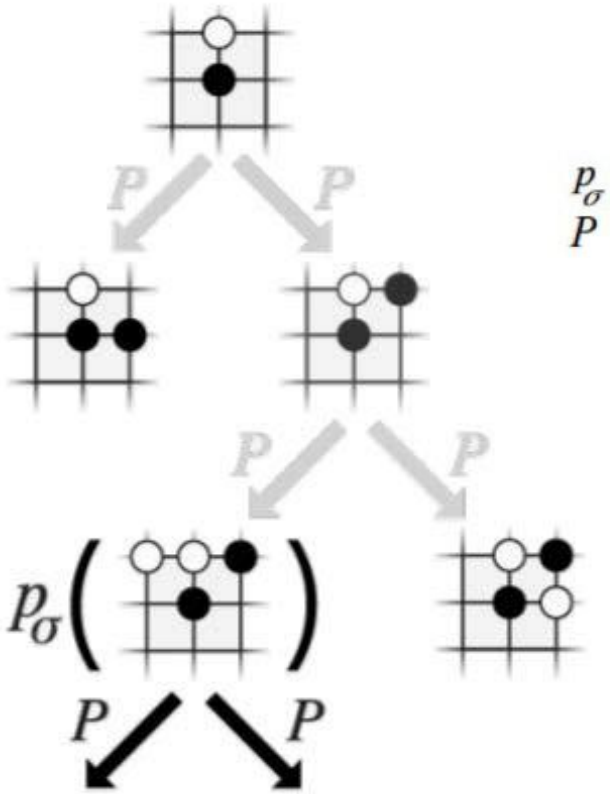
P prior probability
Q action value

Q+u(P)

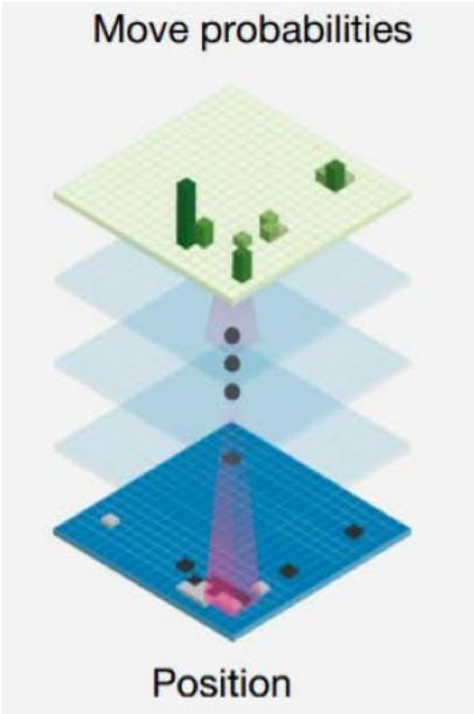
- 起初因为还没有进行模拟，所以动作值为0，则优先考虑较高的先验概率和较少的尝试次数
- 倾向于具有较高收益的行动

MCTS +策略/价值网络

- 扩展

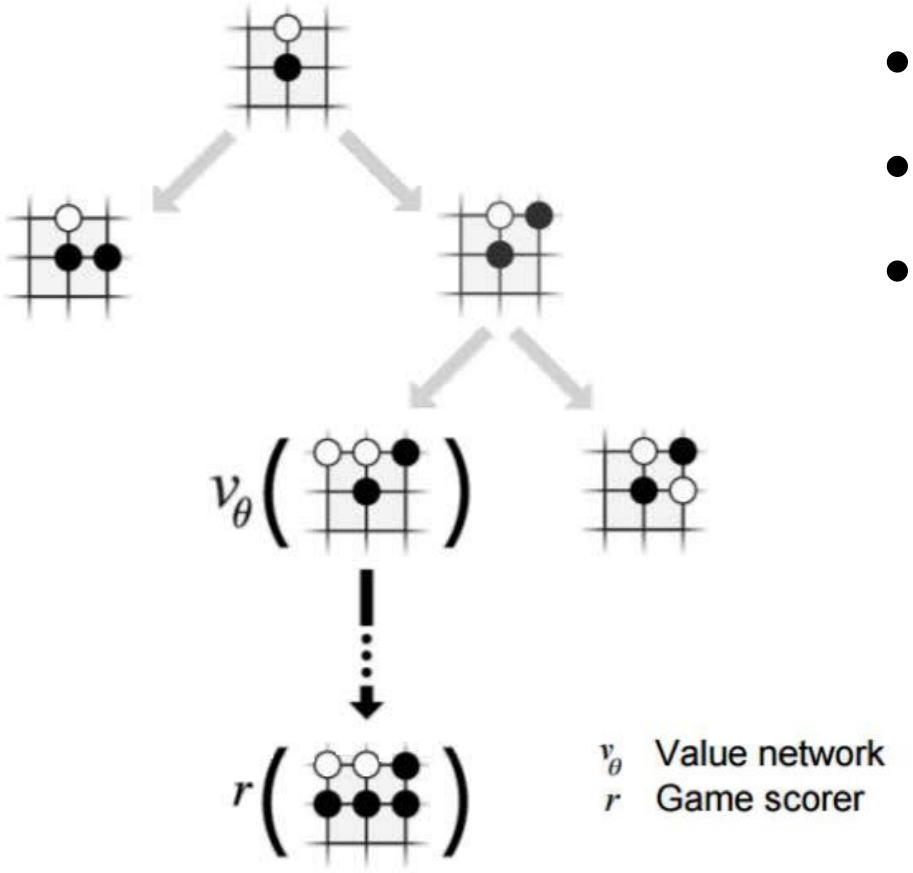


p_σ Policy network
P prior probability



MCTS +策略/价值网络

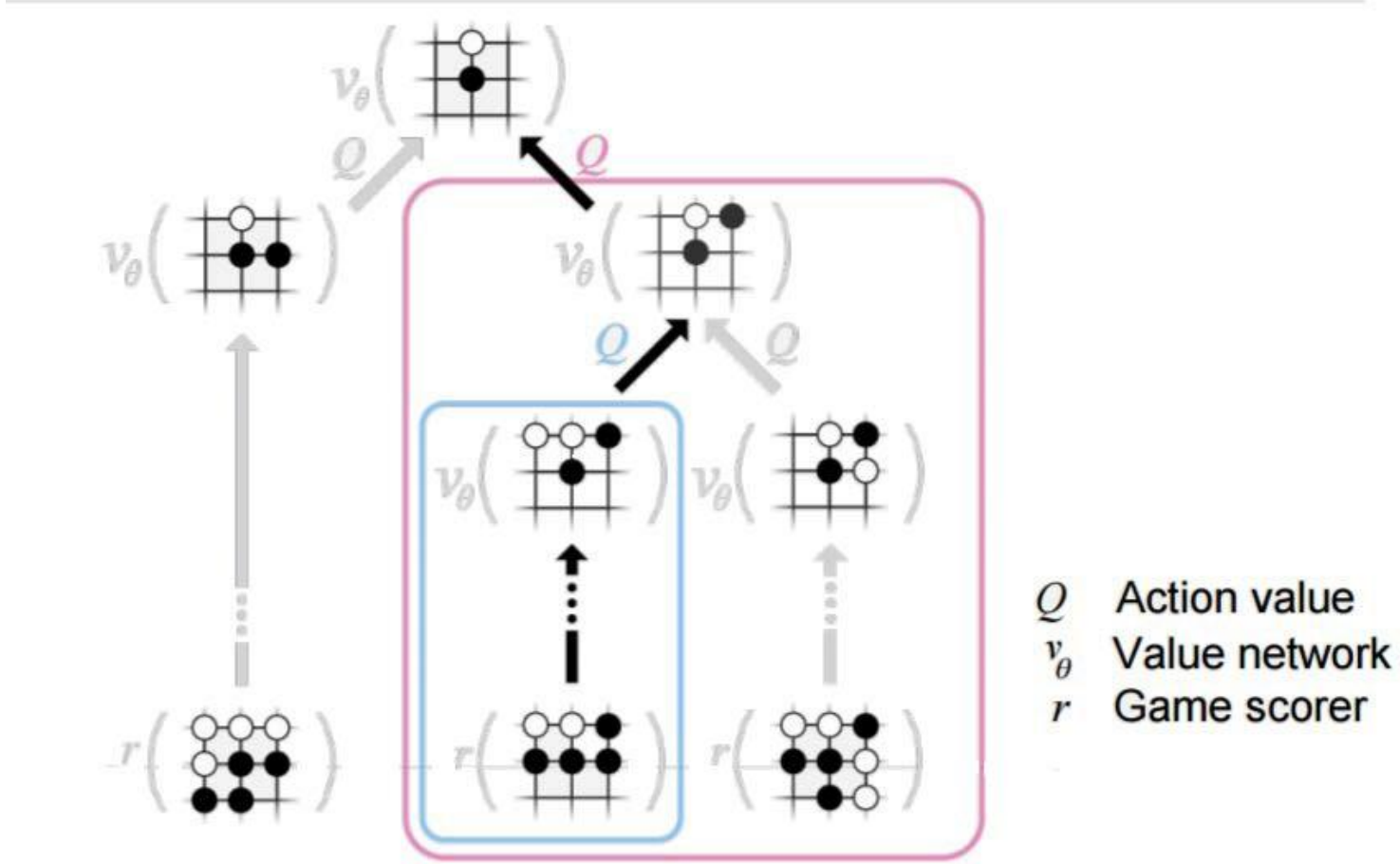
- 模拟



- 并行地进行多个模拟
- 一部分利用价值网络
- 一部分在游戏最终时结束

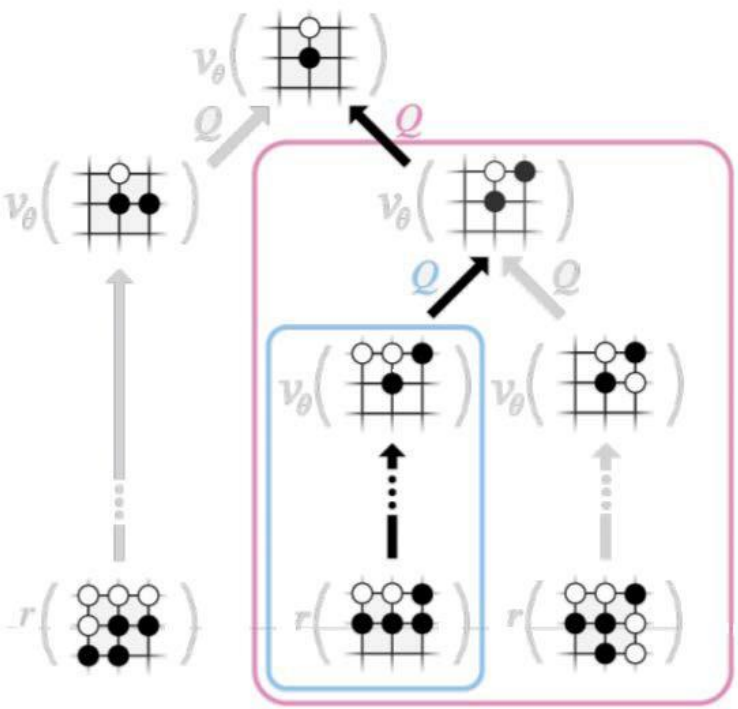
MCTS +策略/价值网络

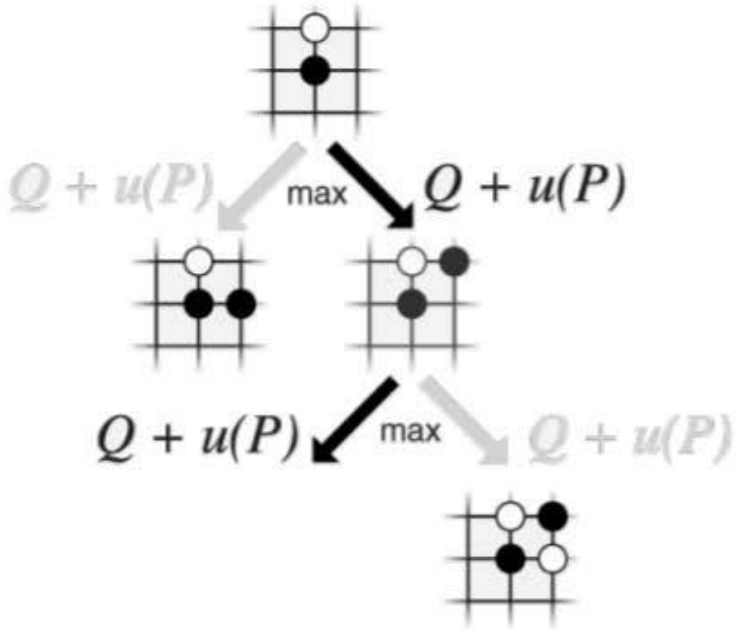
- 将结果反传回根节点



MCTS +策略/价值网络

- 重复





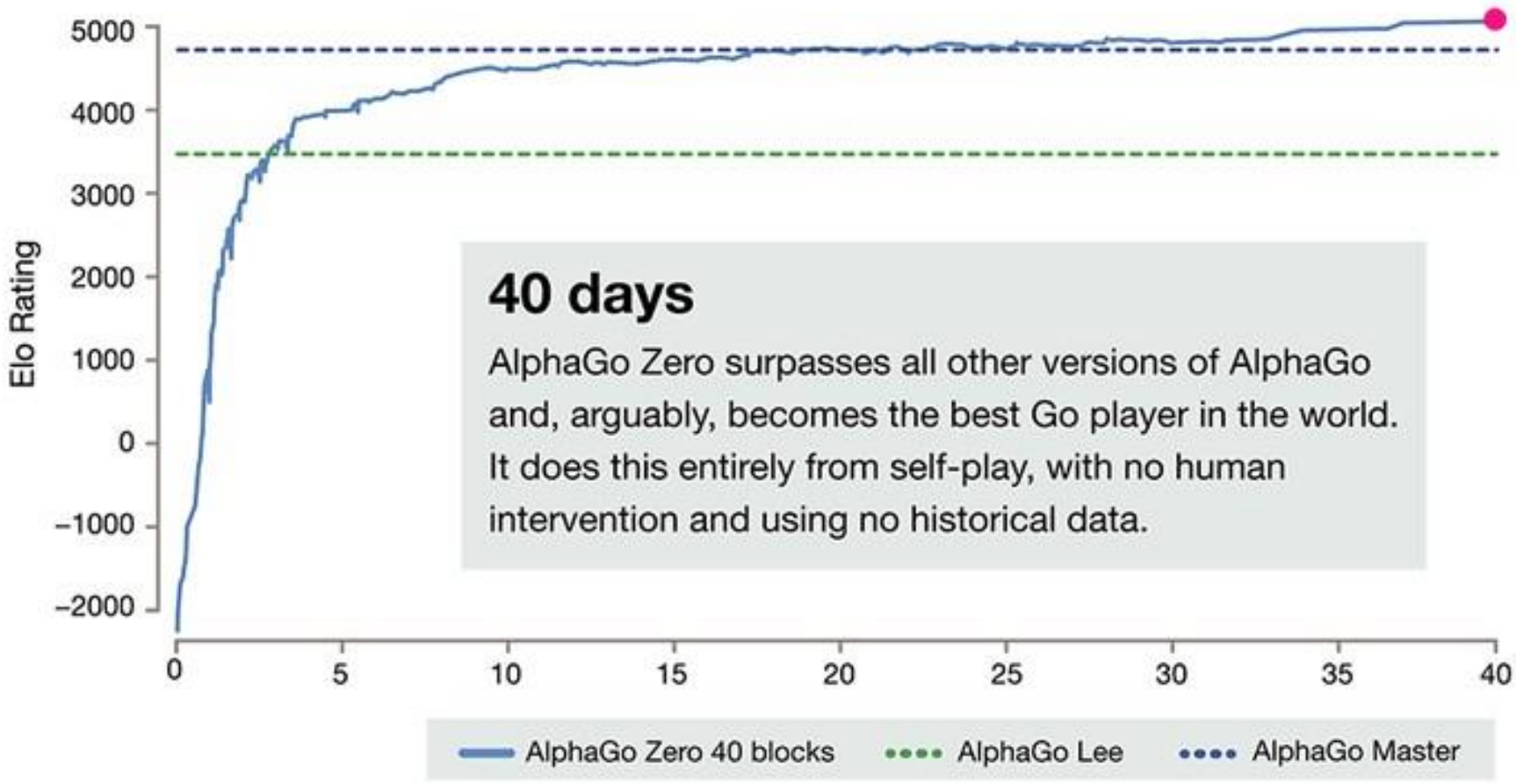
选择

AlphaGo Zero

- AlphaGo
  - 基于人类专家动作的监督学习
  - 通过自我演算进行强化学习
- AlphaGo Zero
  - 单纯地通过自我演算进行强化学习

AlphaGo Zero

- 以100:0的成绩击败 AlphaGo



人工智能未来展望

围棋仍然属于“简单”类的人工智能问题

- **Fully observable** vs. partially observable
- Single agent vs. **multiagent**
- **Deterministic** vs. stochastic
- Episodic vs. **sequential**
- **Static** vs. dynamic
- **Discrete** vs. continuous
- **Known** vs. unknown

人工智能未来展望

DeepMind's AI is Struggling to Beat Starcraft II - Bloomberg

https://www.bloomberg.com/.../deepmind-master-of-go-struggles-to-crack-its-next-mi...

人工智能未来展望

- 将搜索与学习相结合的思想是非常普遍的，并且具有广泛的适用性

参考文献

- Silver, David, et al. "Mastering the game of Go with deep neural networks and tree search." *Nature* 529.7587 (2016): 484-489.
- Silver, David, et al. "Mastering the game of go without human knowledge." *Nature* 550.7676 (2017): 354-359.
- Introduction to Monte Carlo Tree Search, by Jeff Bradberry https://jeffbradberry.com/posts/2015/09/intro-to-monte-carlo-tree-search/

作业

- 实现**蒙特卡洛树**的应用
- 电子版发送到 1810283086@qq.com