
PRO SHOP
HIT
THE
EAD
THIS IS NOT ‘NAM. THERE ARE RULES.

Von Neumann and Morgenstern (1944)

THEORY OF GAMES AND ECONOMIC BEHAVIOR

- "An exact description of the endeavor of the individual to obtain a maximum of utility, or, in the case of the entrepreneur, a maximum of profit."
- "The typical problems of economic behavior become strictly identical with the mathematical notions of suitable games of strategy."
- "It is not that there exists any fundamental reason why mathematics should not be used in economics. The arguments often heard that because of the human element, of the psychological factors etc., or because there is allegedly no measurement of important factors, mathematics will find no application, can all be dismissed as utterly mistaken."

Von Neumann and Morgenstern (1944)

- Chapter I: Formulation of the economic problem (utilities)
- Chapter II: General formal description of games of strategy
- Chapter III: Zero-sum two-person games: Theory
- Chapter IV: Zero-sum two-person games: Examples
  - Elementary games
  - Poker and Bluffing
- Chapters V-IX: n-person zero sum games
- Chapters X, XI, XIII: Extensions

Von Neumann and Morgenstern: Reviews

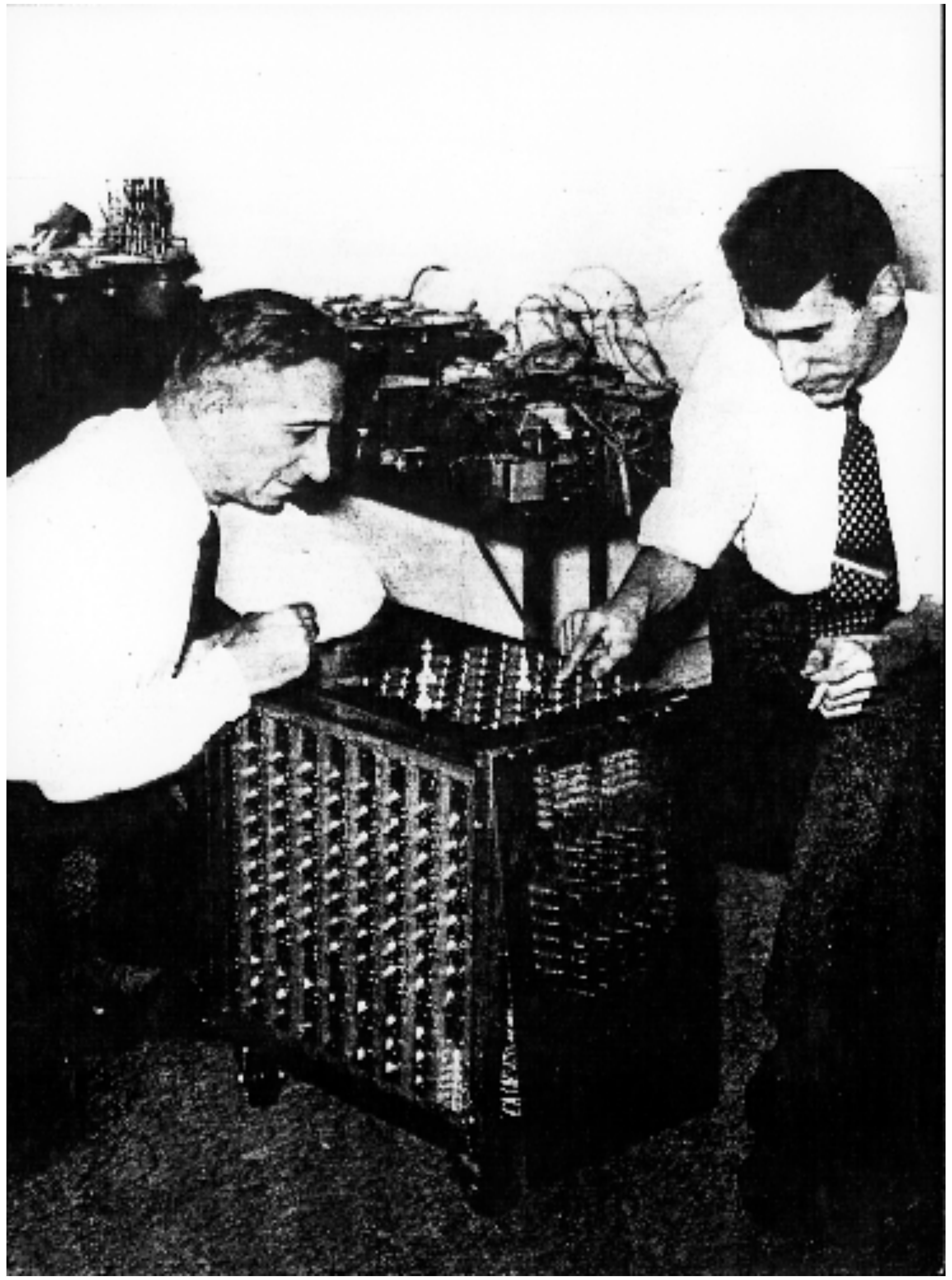
- Simon: "Although no explicit applications are made to sociology or political science, the schema is of such generality and breadth that it can undoubtedly make contributions of the most fundamental nature to those fields."

- Rowland: "…the discussion being dominated by illustrations from chess, poker, bridge, etc. and not from cartels, markets, oligopolies"

- P. Samuelson: Like tic-tac-toe, "Chess, being a game of perfect information, turns out to be trivial… If chess is trivial, penny matching is not."

Shannon

- Just because Chess is "trivial" doesn't mean computing a good strategy is trivial.
- Shannon number: $\sim 10^{120}$ variations to consider from the starting position
- **Approach 1:** Lookup table: $\sim 10^{43}$ possible boards. Just memorize the optimal move from any board.
- **Approach 2:** Approximate dynamic programming (before dynamic programming)



Shannon's minimax approach

- $f(P)$ evaluate the value of a position

$$f(P) = \begin{cases} 1 & \text{win} \\ 0 & \text{draw} \\ -1 & \text{lose} \end{cases}$$

$$\max_{W_1} \min_{B_1} \max_{W_2} \min_{B_2} \cdots \max_{W_T} \min_{B_T} f(Z_T)$$

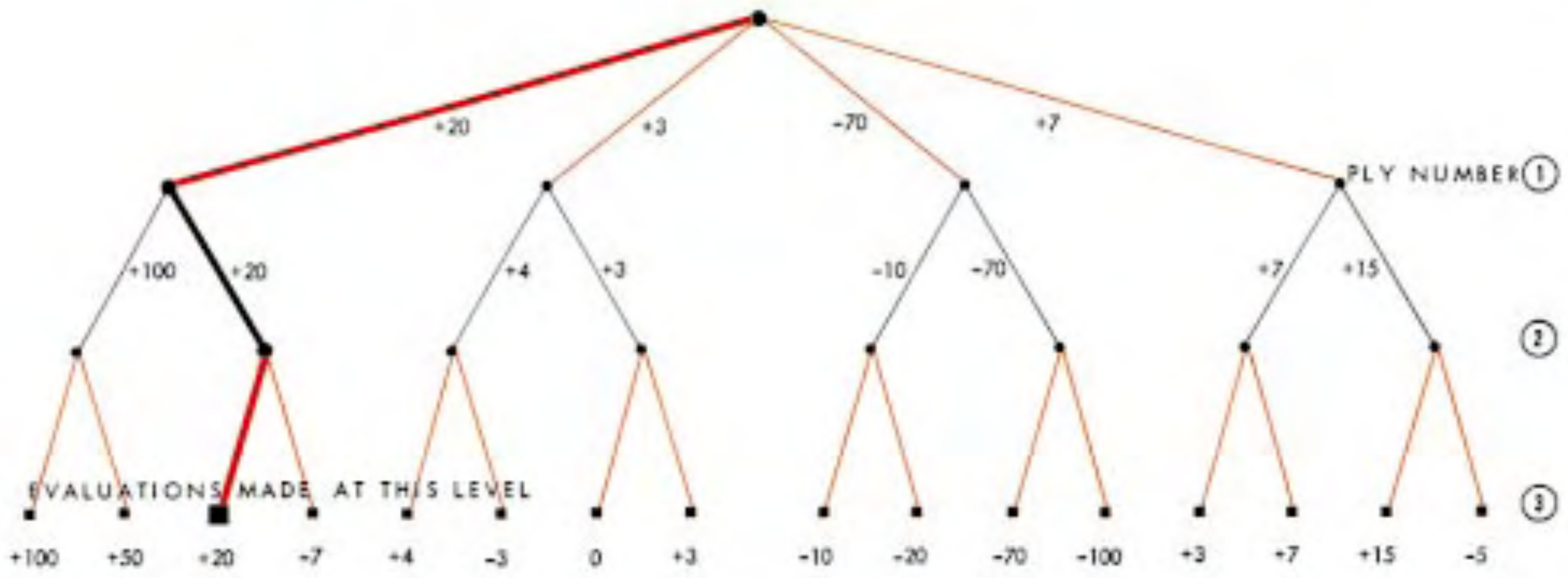
$Z_s = B_s W_s Z_{s\text{-}1}$

Z_0 is the current board position.

Shannon’s approach to the minimax game

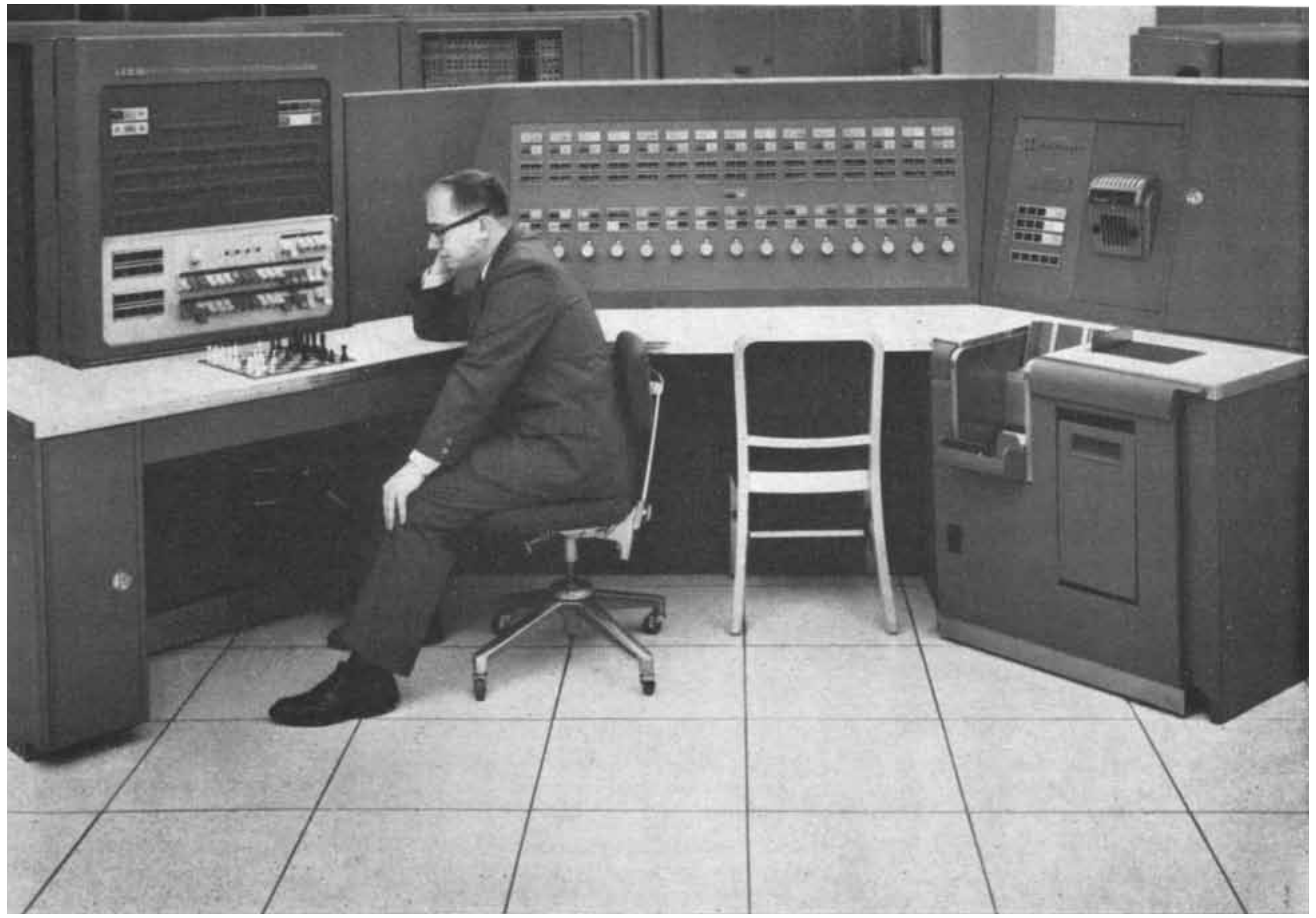
- *Function approximation*
- Type A vs Type B tree search
  - Type A: brute force
  - Type B: restrict the space of moves to save compute in tree search

1. Machine chooses branch with largest score.
2. Opponent expected to choose branch with smallest score.
3. Machine chooses branch with most positive score.



Chess through 1968

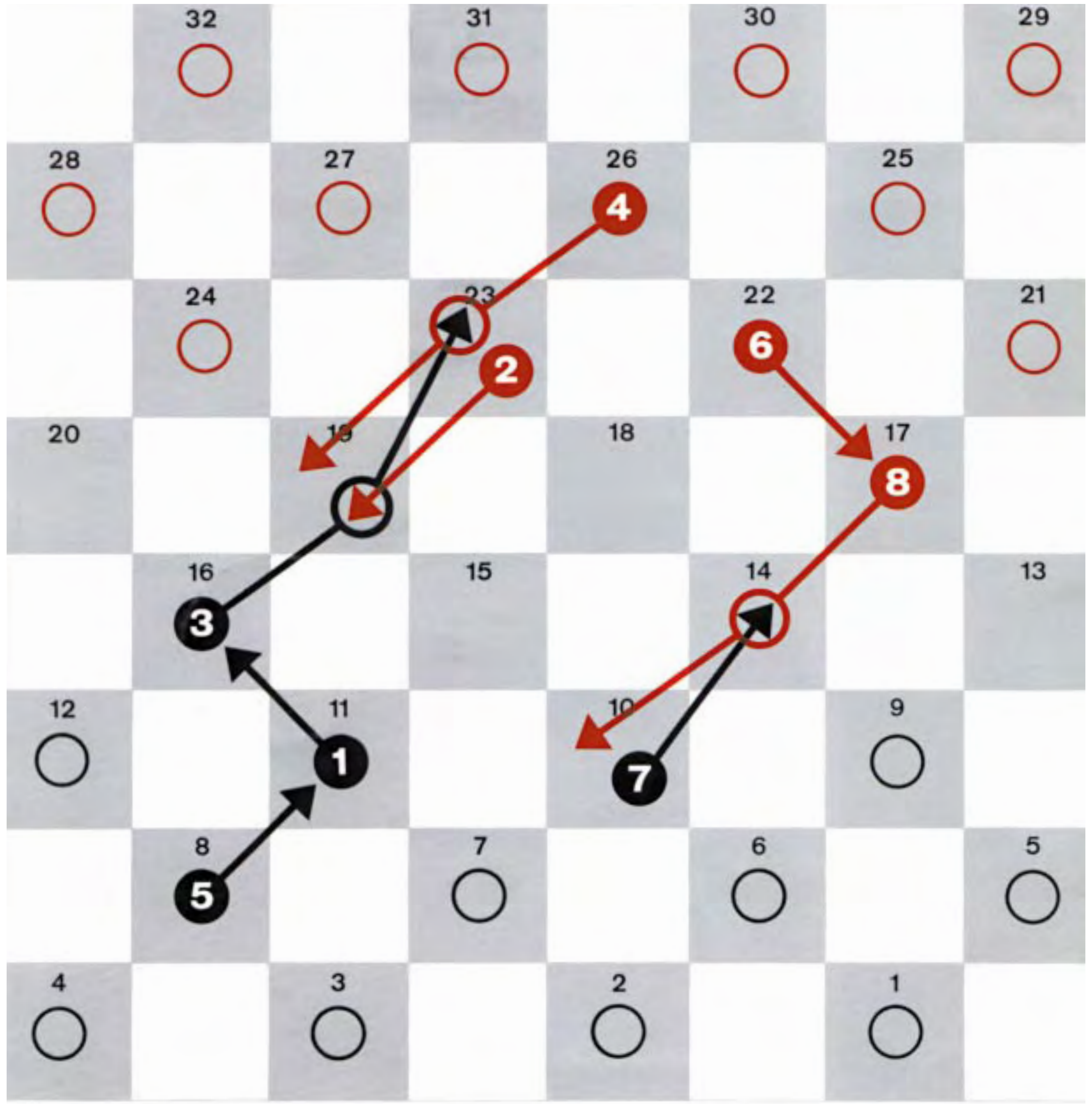
- Branch and bound applied to chess
  (Alpha-Beta)
- 1968 - John McCarthy bets David Levy that a computer will beat him in 10 years



Bernstein Chess Machine (1958)

Bernstein: "Even with much faster computers than any now in existence it will be impracticable to consider more than about six half-moves ahead, investigating eight possible moves at each stage. A more promising line of attack is to program the computer to learn from experience."

Arthur Samuel and Checkers (1959)

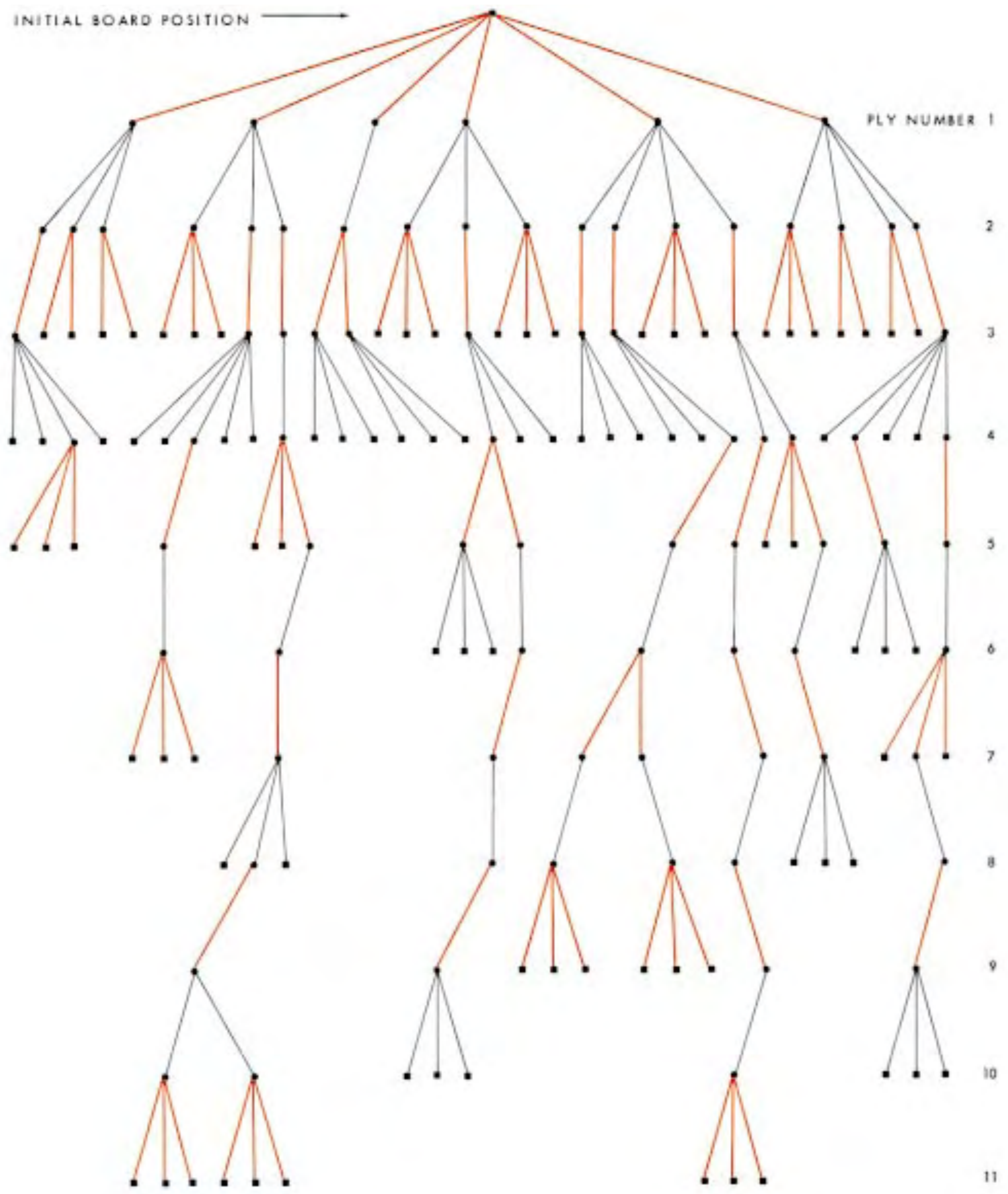


Why Checkers as an ideal *machine learning* problem?

- A definite goal must exist
- The rules of the game must be definite and known.
- There must be a background of knowledge on the task for benchmarking.
- The ability to play against people adds "spice to the study."
- Chess is too complicated; Checkers' simpler rules allow for more emphasis on learning techniques
- Search space of size $\sim 10^{20}$.

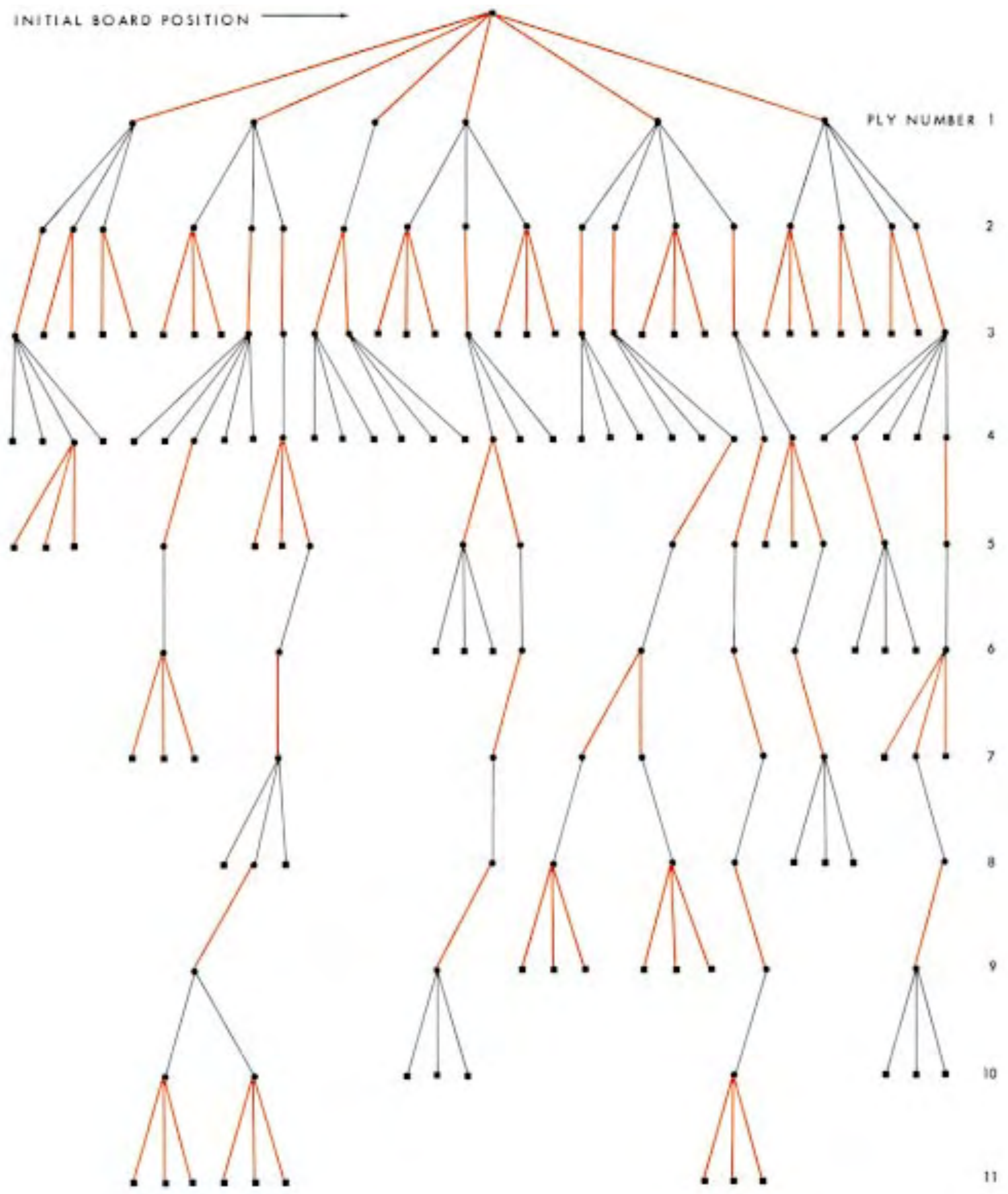
Samuel Checkers Player Approach

- Shannon Type A:
  - Defines "ply" as number of lookahead moves.
  - Function approximation
- Main addition: learning the score function.
  - Self-play
  - Primitive temporal differencing

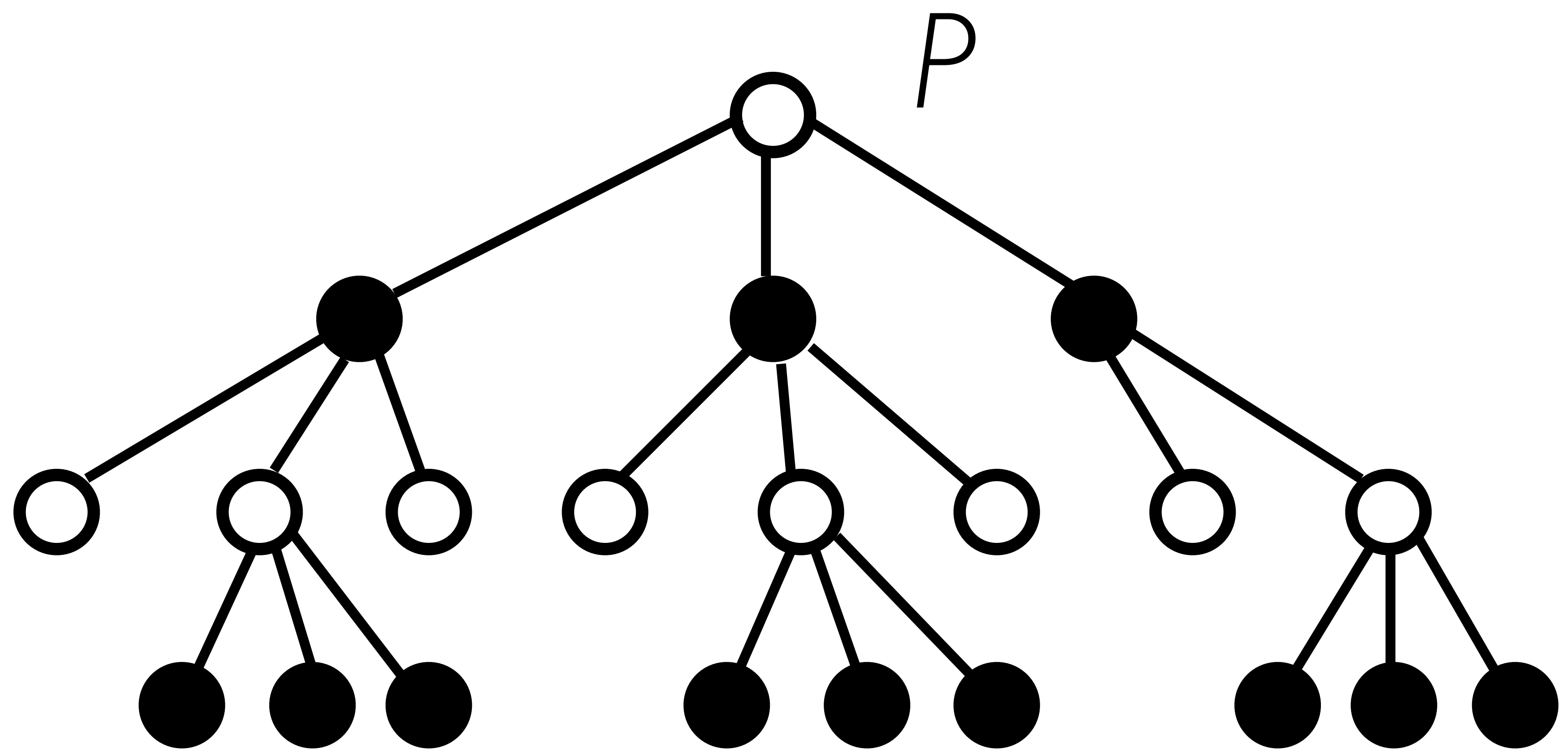


Self-play by Samuel's Checker Player

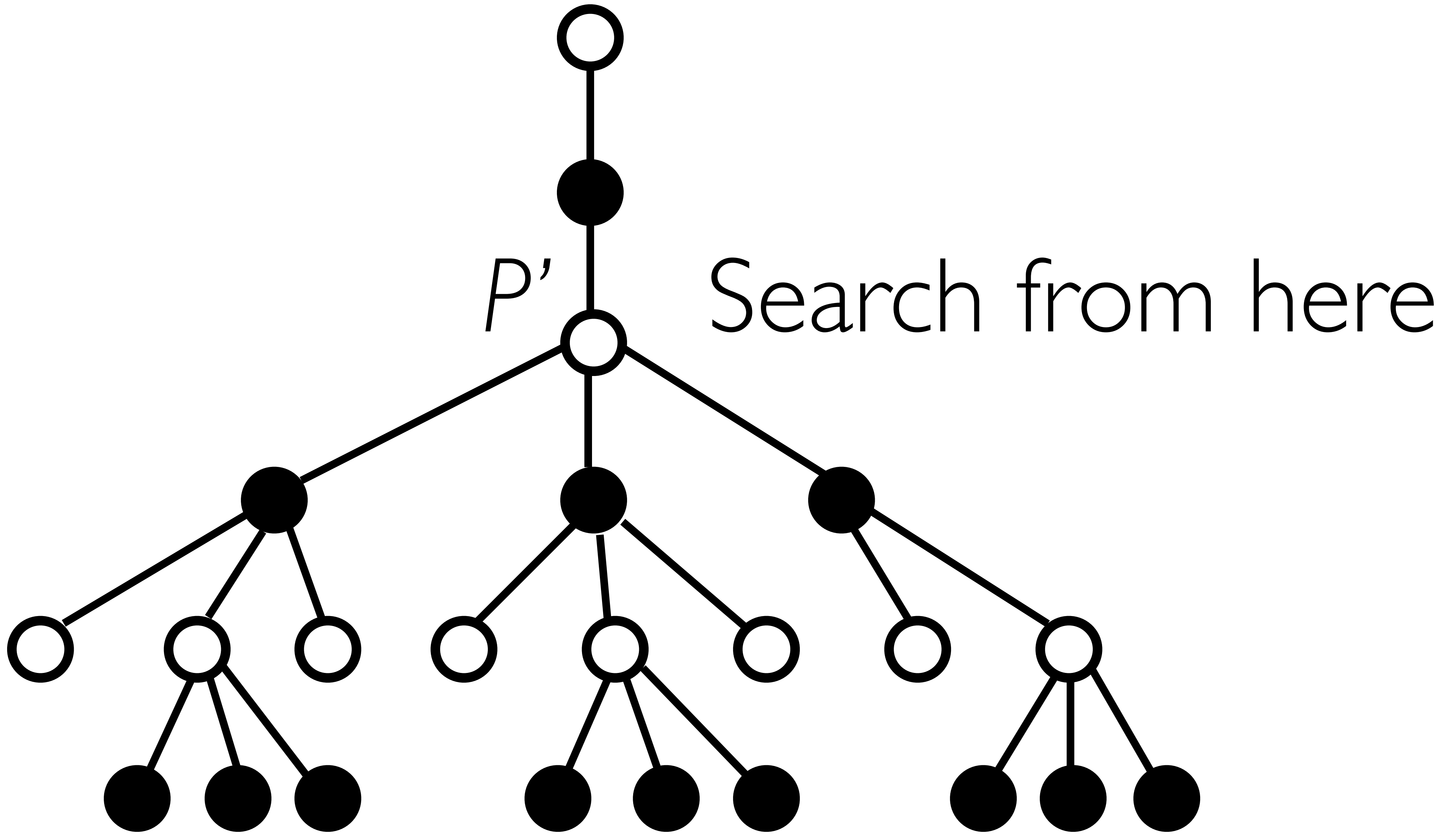
- Two players, *Alpha* and *Beta*
- Alpha updates its scoring function after every move, Beta stays constant.
- If Alpha wins, Beta uses Alpha's final scoring function. Otherwise, Alpha is given a strike. After 3 strikes, parts of Alpha's polynomial are reset.
- Score is updated by recording the current score and the score up the tree of all nodes visited.



Samuel's primitive temporal differencing



Take branch with maximum cost

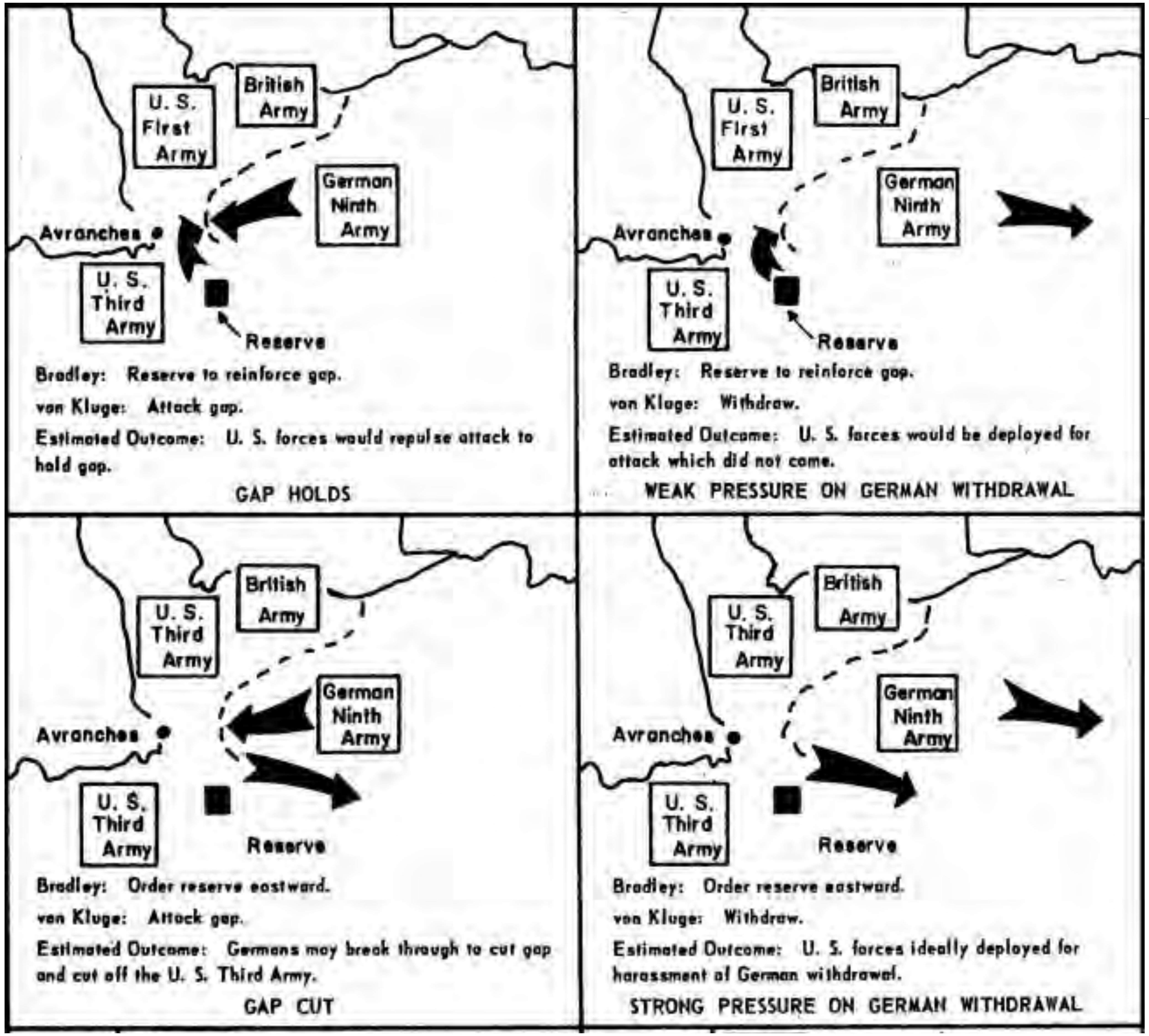


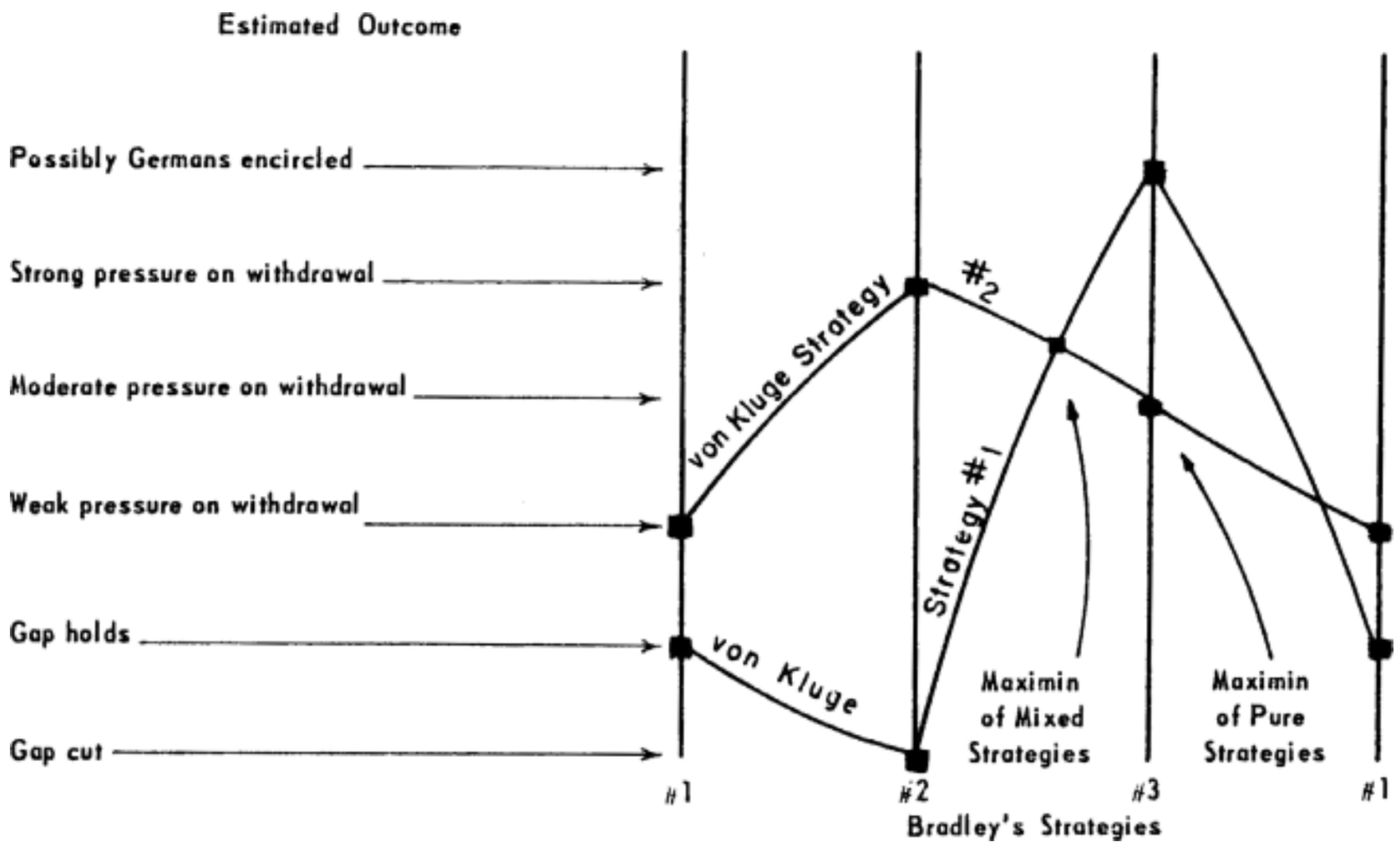
Gives new estimate of cost at previous level

Update $f(P)$ based on $\Delta = f(P) - f(P')$

Meanwhile at the RAND corporation…

- Game theorists in residence from founding through 1960s
- von Neumann, Bellman, Shapley, regular visits from Kuhn, Nash, and Tucker
- Wanted to put this to the test, published countless papers on games for war games.
- Shapley: "A Hidden Target Model," "The Silent Duel, One Bullet Versus Two, Equal Accuracy"
- Bellman: "A Bomber–Fighter Duel", "Application of Theory of Games to Identification of Friend and Foe", "On Games Involving Bluffing"





From thesis of Oliver Haywood "Military Doctrine of Decision and the von Neumann Theory of Games"

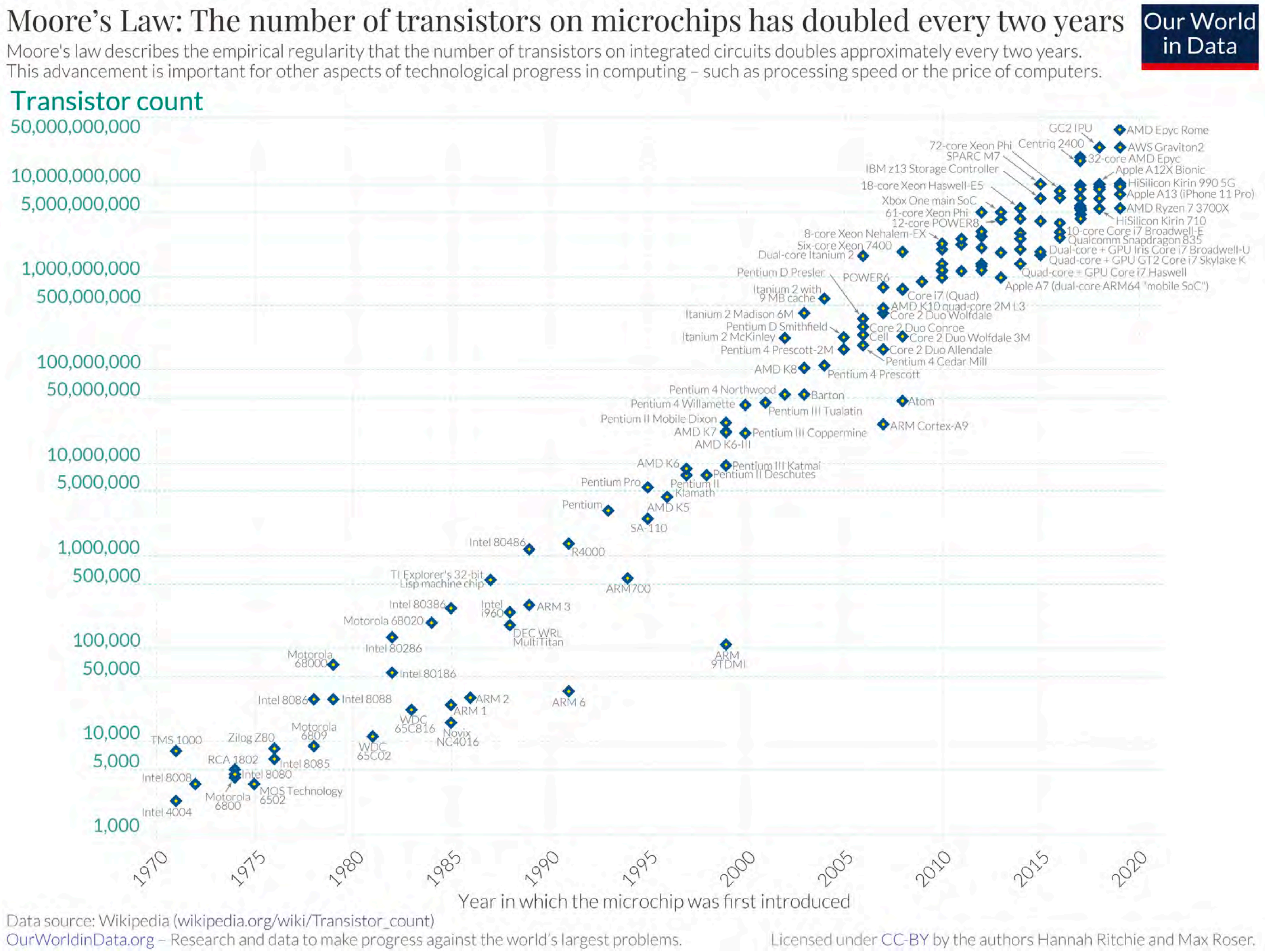
The trouble with games as a model of human behavior

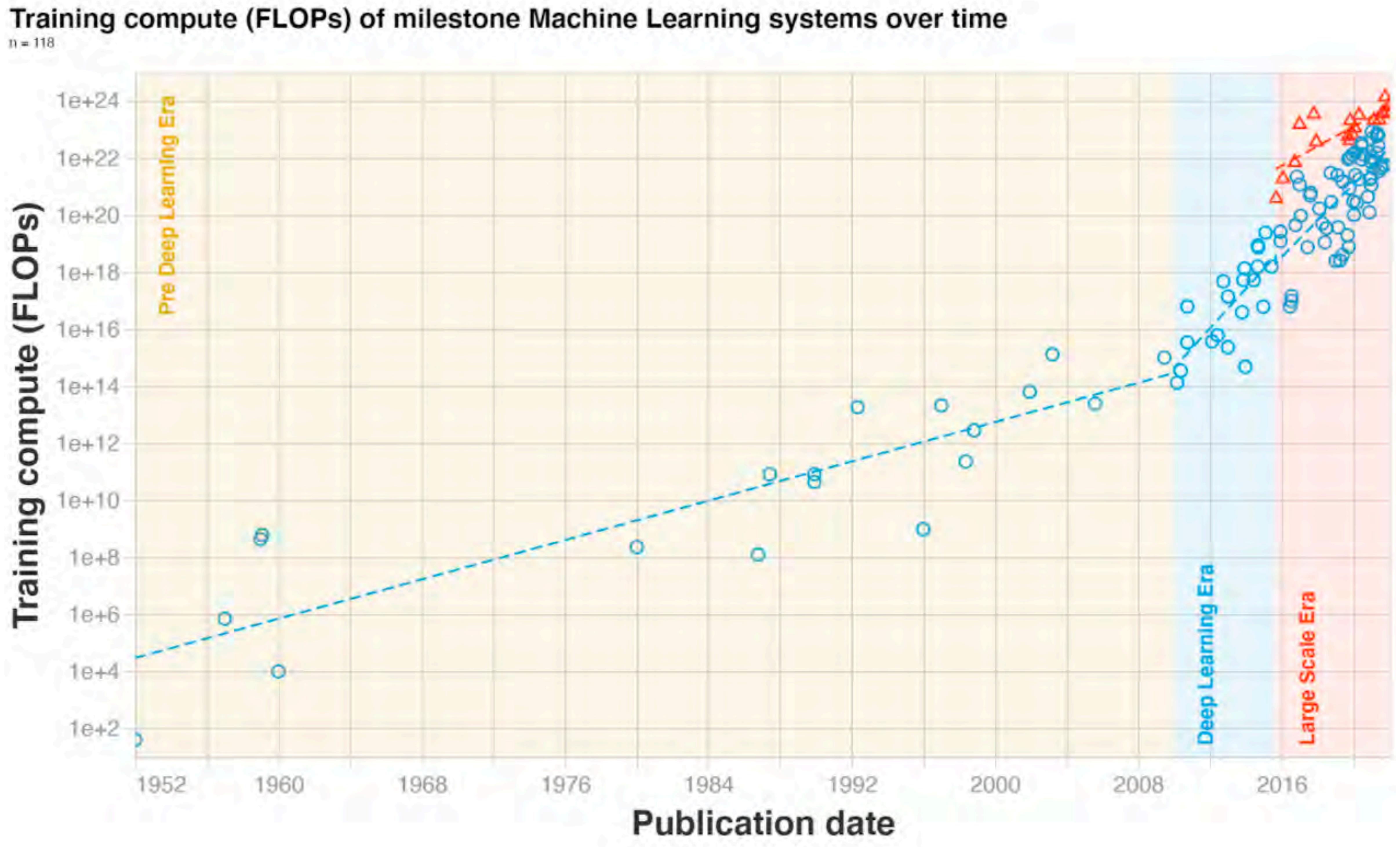
- Even by early 1950s, realized that the vNM program wouldn't work.
- Didn't seem to apply to communism, non-zero-sum games, bounded rationality, folk theorems (known at RAND in early 50s)
- Complexity Issues, lack of predictability and generalizability.
- But RAND still invested in "war games" as way of understanding individual and organizational behavior. Was metaphorical, not literal, and the language of game theory—-strategies, equillibria, etc.—-framed their research.

Game winter

- Not a whole lot happened between 1959 and 1997!
- David Levy won his bet against McCarthy in 1978.

- Meanwhile, we sent a man to the moon, developed large-scale aviation, scaled OR for infrastructure management, widely deployed automation in chemical process engineering. Oh, and computers got a lot faster…


Moore's Law: The number of transistors on microchips has doubled every two years
Our World in Data
Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years.
This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.
Transistor count
50,000,000,000
10,000,000,000
5,000,000,000
1,000,000,000
500,000,000
100,000,000
50,000,000
10,000,000
5,000,000
1,000,000
500,000
100,000
50,000
10,000
5,000
1,000
1970
1975
1980
1985
1990
1995
2000
2005
2010
2015
2020
Year in which the microchip was first introduced
Data source: Wikipedia (wikipedia.org/wiki/Transistor_count)
OurWorldinData.org – Research and data to make progress against the world's largest problems.
Licensed under CC-BY by the authors Hannah Ritchie and Max Roser.


Training compute (FLOPs) of milestone Machine Learning systems over time
n = 118
Training compute (FLOPs)
1e+24
1e+22
1e+20
1e+18
1e+16
1e+14
1e+12
1e+10
1e+8
1e+6
1e+4
1e+2
Pre Deep Learning Era
Deep Learning Era
Large Scale Era
1952
1960
1968
1976
1984
1992
2000
2008
2016
Publication date

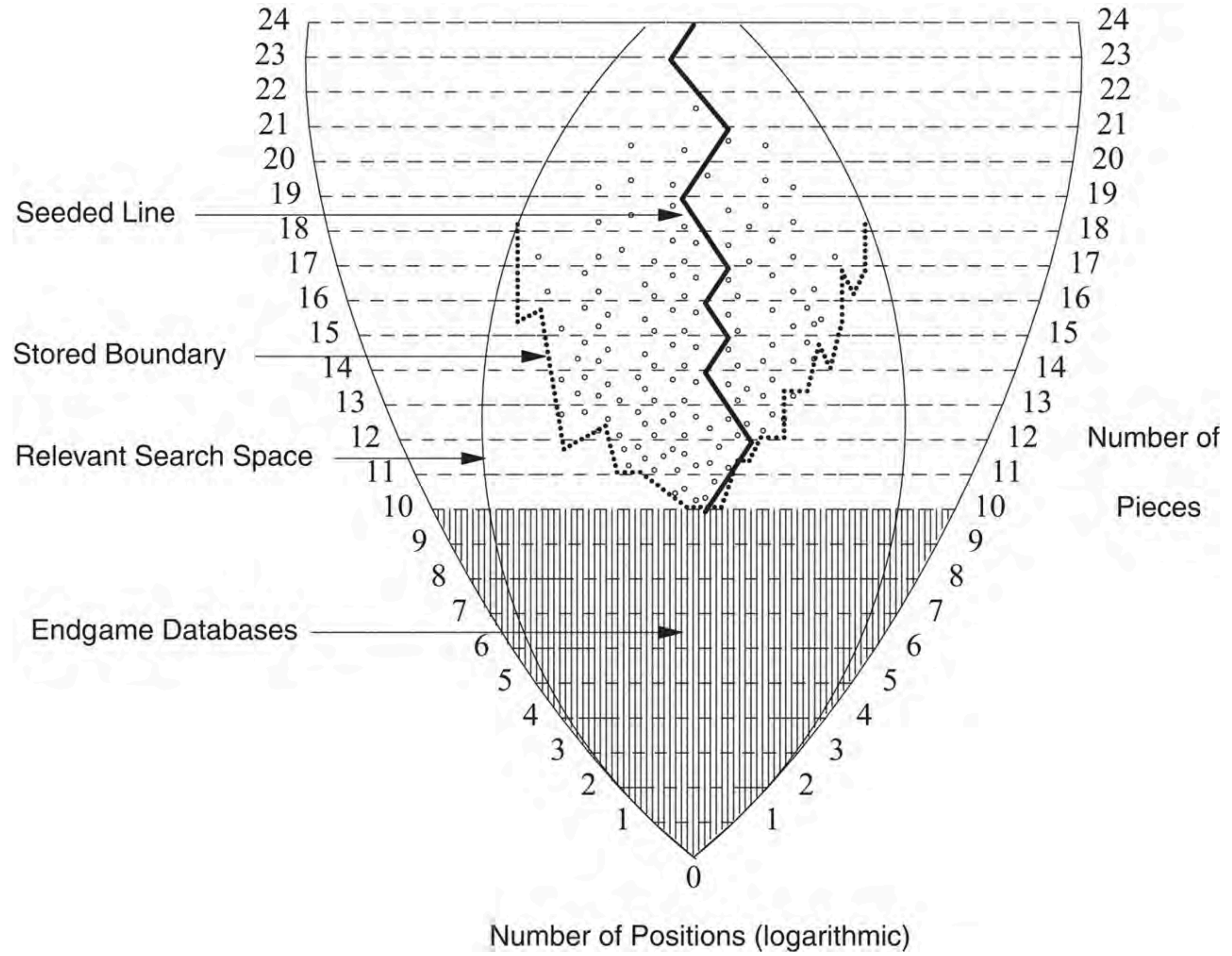
Deep Blue (1997)

- Shannon Type A
- Branch and Bound
- ~10 gigaflop custom supercomputer
  (FYI: a PS5 is 10 teraflops)

Checkers is solved (Schaeffer et al., *Science* 2007)

- "Since 1989, almost continuously, dozens of computers have been working on solving checkers, applying state-of-the-art artificial intelligence techniques to the proving process. This paper announces that checkers is now solved: Perfect play by both sides leads to a draw."

- Theorem proving via complex tree search.
- Run in parallel over many computers for 27 years.
- Evaluated ~10^{14} positions.

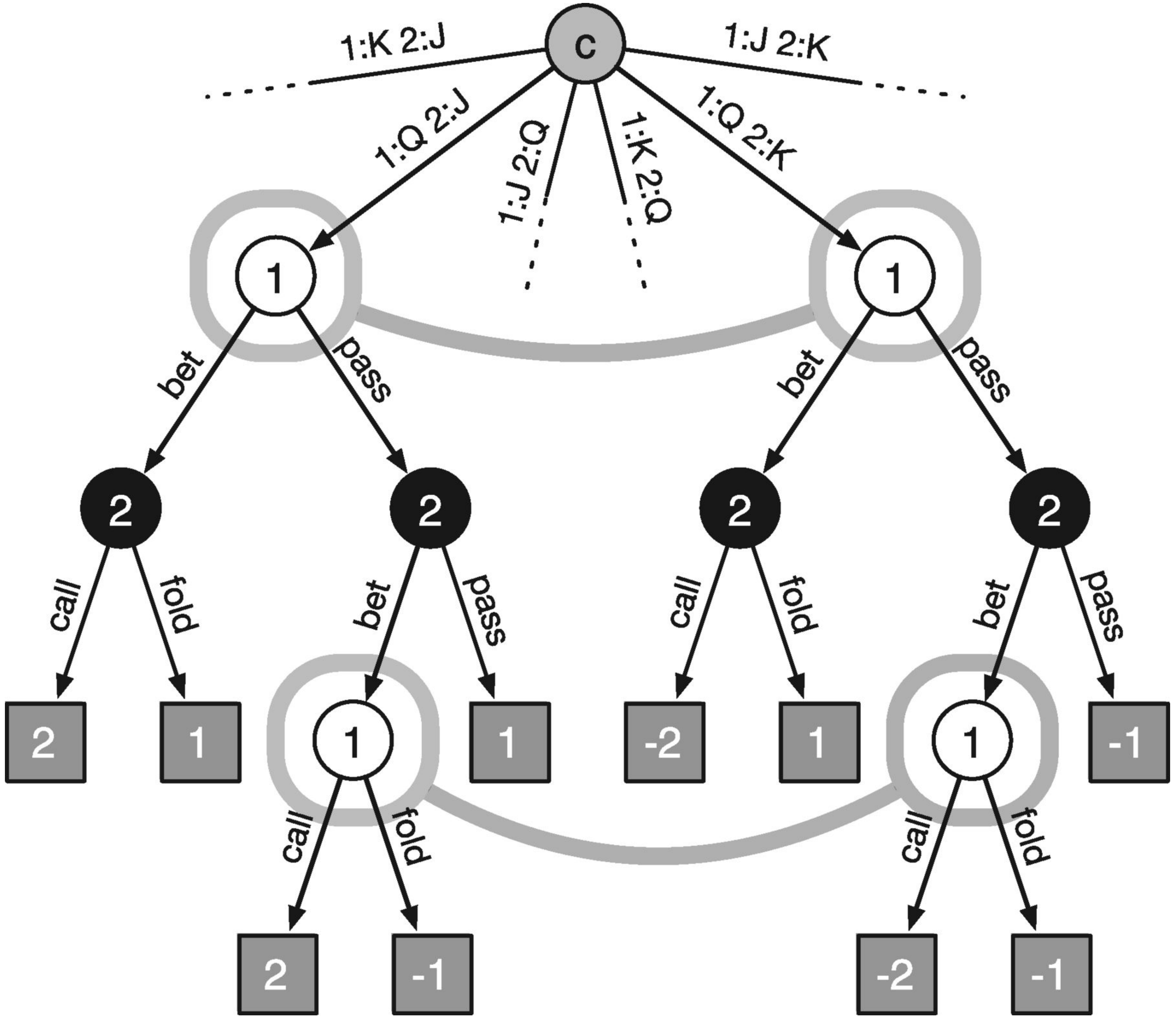


AlphaGo (2016)

- Function approximation with convnets
- Self-play with Monte Carlo Tree Search
- Trained on 10^8 moves from 160,000 games
- ~1 petaflop custom supercomputer (1920 CPUs and 280 GPUs)

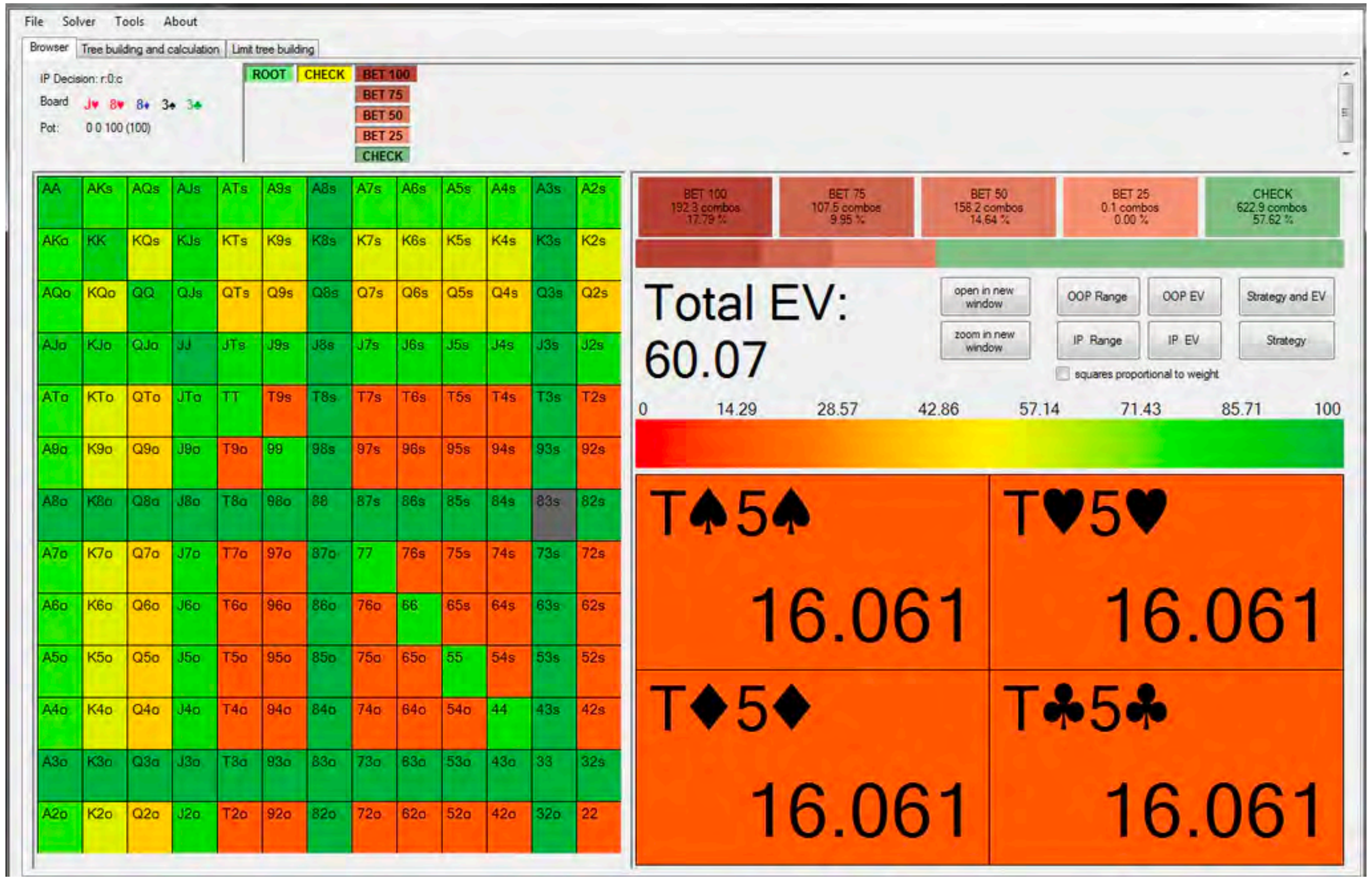
GTO Poker (Bowling et al., 2007-2015)

- One of the nontrivial games (imperfect information)
- Major breakthrough: **counterfactual regret minimization.**
- Regret: summed differences of played response to best response.
- ϵ-regret$\Rightarrow$$\epsilon$-Approximate Nash Eq.
- Upper bound regret using an idea similar to Samuel.
- Run standard regret minimization.



Game Theory Optimal Gameplay

- Poker solvers now available. Solve restricted games.
- Professional poker players train memorizing solver outputs.
- Professional chess players train themselves following AlphaZero.
- *Solving games with computers improves human play*



PIO Solver

Games?

- The ideas were there, we had to wait for computers to catch up.
- Game theory, as a means to understanding humans and human economies, was a total failure.
- Game theory, as a scheme for rational mechanism design, a success, but rationality is a choice.
- Game theory has been hugely impactful in understanding games.
- Games are best-case for policy optimization

Finding what works

$$\underset{\text{policy}}{\text{maximize}} \quad \text{outcome}$$

- For simple models, can be solved by optimization methods
- For games, everything rigid, can solve by brute force search
- For more complex models, can solve by DFO

  this is what we do in randomized controlled trials
- To understand people, what happens when the distribution changes?, when the moves change their meaning?, when the desired outcome changes?, when the outcome is unclear?, when outcomes are not comparable?…
- Some problems are perhaps not best solved with optimization.