

TD-Gammon

Gerald Tesauro

1992, 1994, 1995

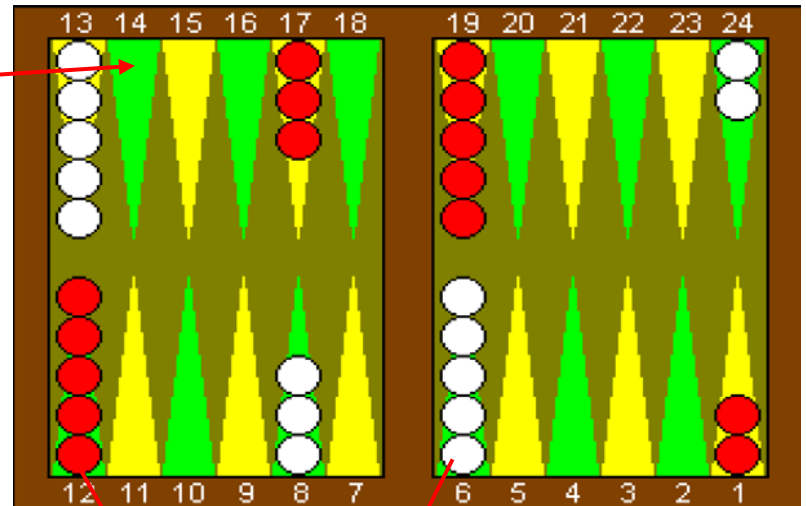
1 Rules

Backgammon

- is an ancient two-player game
- at least a thousand years older than chess
- two dice
- number of possible states (about 10^{20})
 - far more than the number of memory elements one could have in any physically realizable computer
 - the number of states is so large that a lookup table cannot be used
 - Chess about 10^{46}
- branching factor (about 420)
 - at each ply there are 21 dice combinations possible
 - about 20 legal moves per dice combination
- 8-10 for checkers
- 30-40 for chess

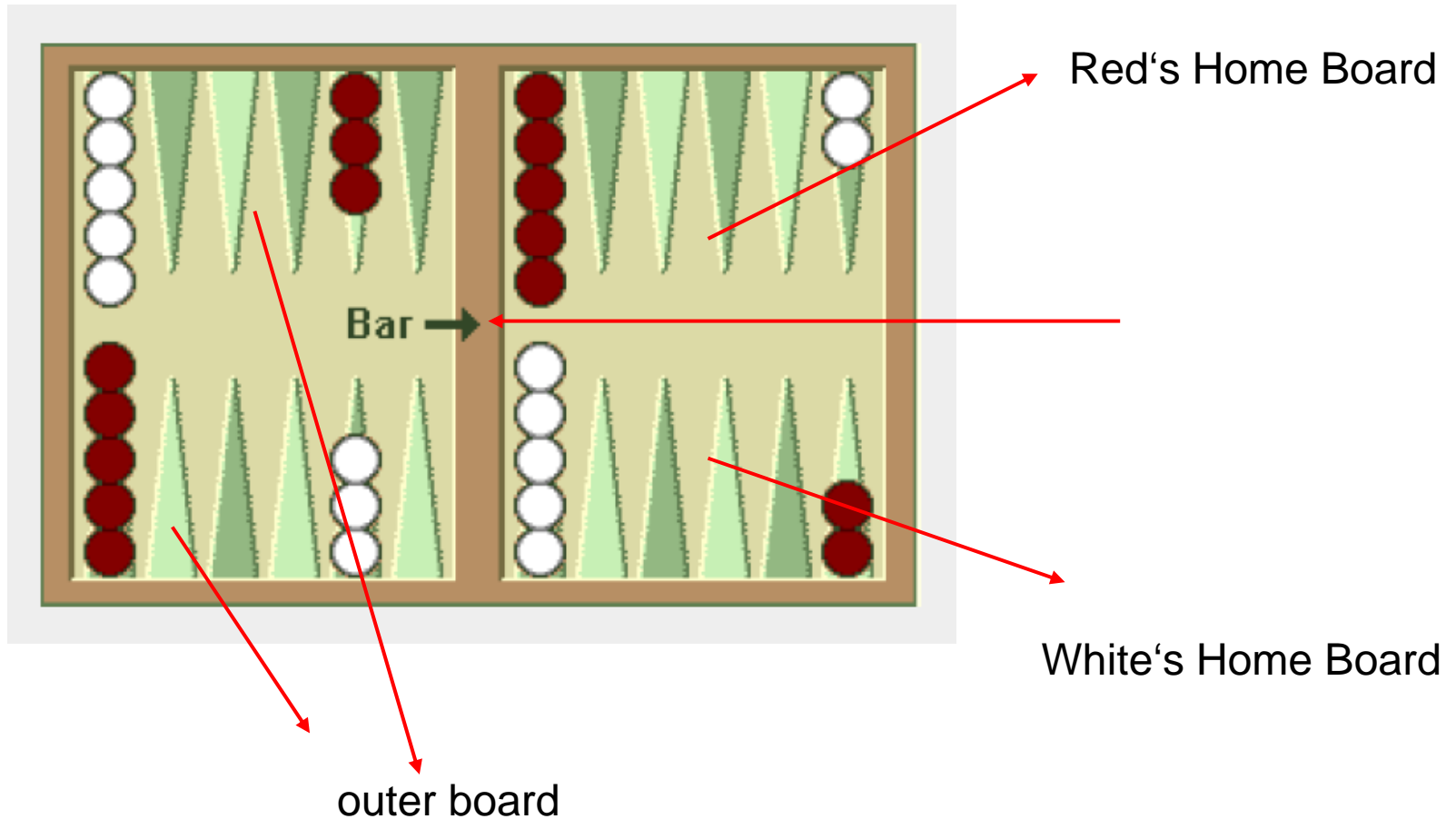
Board

- board consisting of twenty-four narrow triangles called **points**
- the triangles alternate in color and are grouped into four quadrants of six triangles each



checkers (pieces)

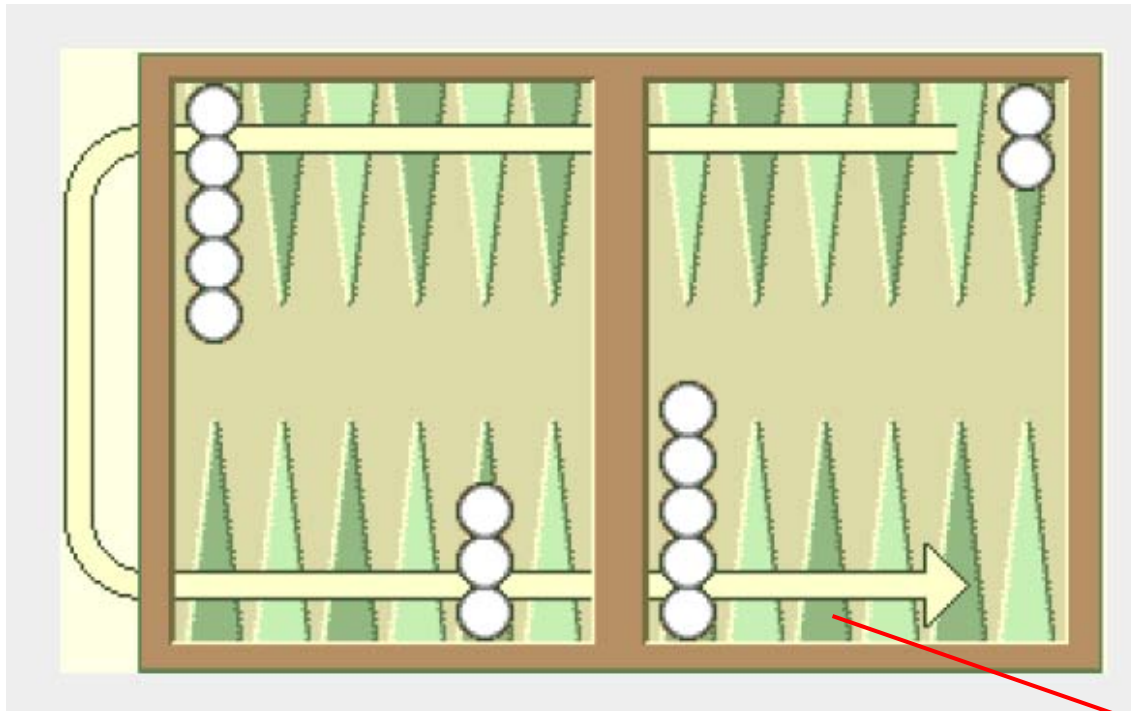
A board with the checkers in their initial position.



Object of the Game

- The object of the game is for a player to move all of his checkers into his own home board and then bear them off.
- The first player to bear off all of his checkers wins the game.
- no draw

Direction of movement

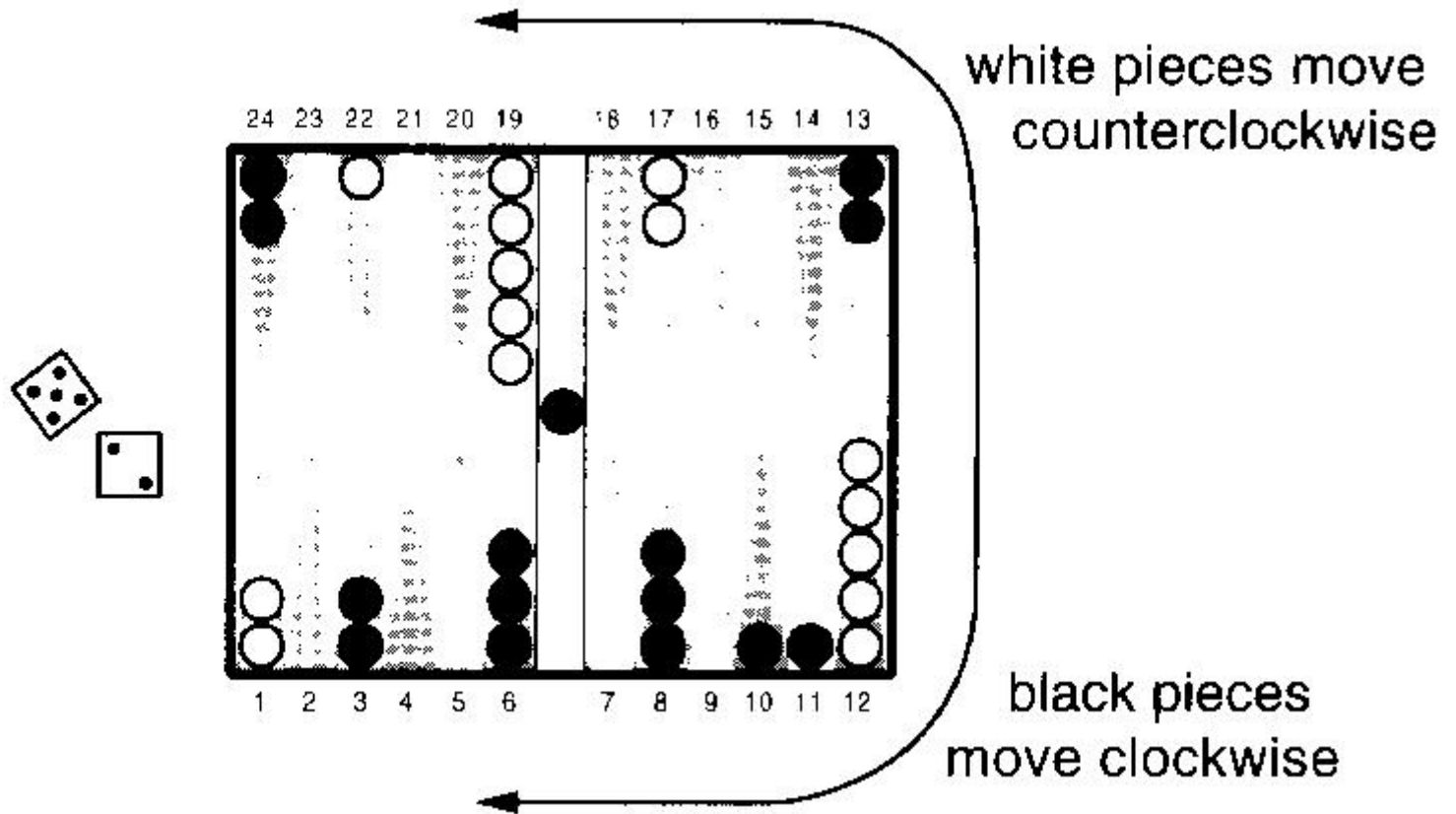


Direction of movement of White's checkers.
Red's checkers move in the opposite direction.

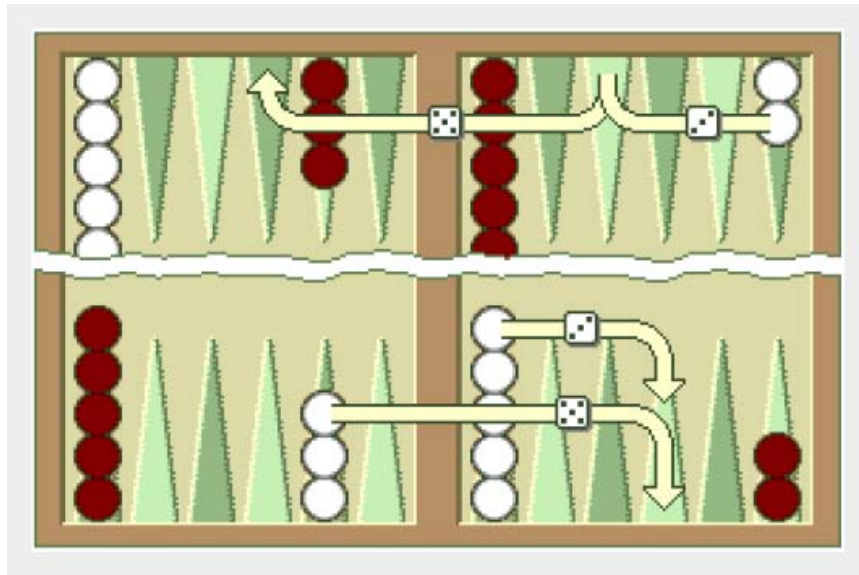
White's Home Board

TD-Gammon

Tesauro 1992, 1994, 1995, ...



Movement of the Checkers

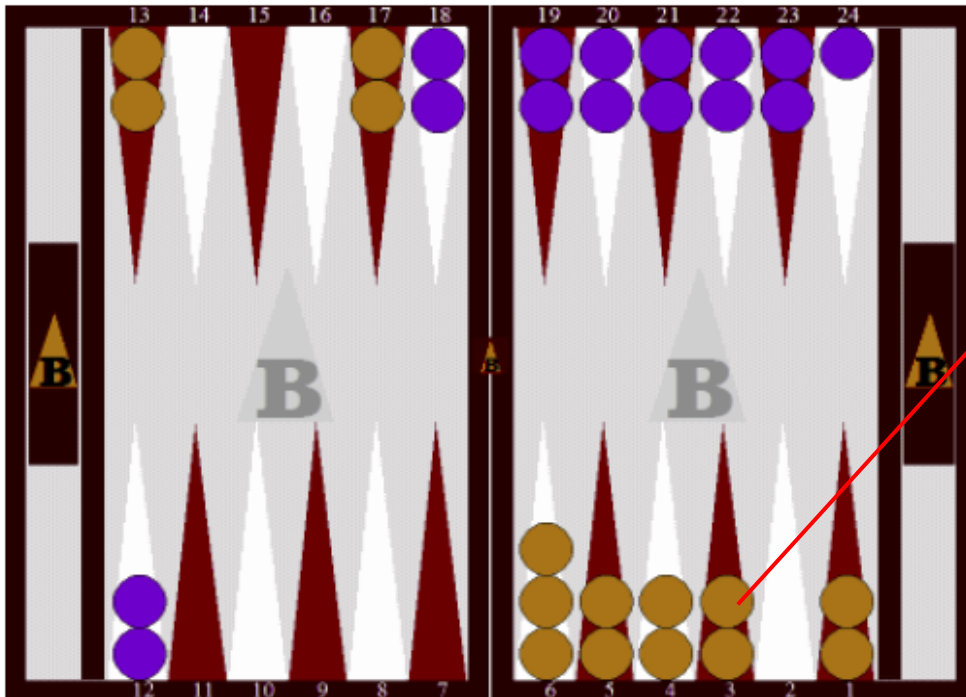


White opens the game with 53.

Movement of the Checkers

- A checker may be moved only to an open point, one that is not occupied by two or more opposing checkers.
- The numbers on the two dice constitute separate moves. For example, if a player rolls 5 and 3, he may move one checker five spaces to an open point and another checker three spaces to an open point, or he may move the one checker a total of eight spaces to an open point, but only if the intermediate point (either three or five spaces from the starting point) is also open.
- A player who rolls doubles plays the numbers shown on the dice twice. A roll of 6 and 6 means that the player has four sixes to use, and he may move any combination of checkers he feels appropriate to complete this requirement.
- A player must use both numbers of a roll if this is legally possible (or all four numbers of a double). When only one number can be played, the player must play that number. Or if either number can be played but not both, the player must play the larger one. When neither number can be used, the player loses his turn. In the case of doubles, when all four numbers cannot be played, the player must play as many numbers as he can.

Example

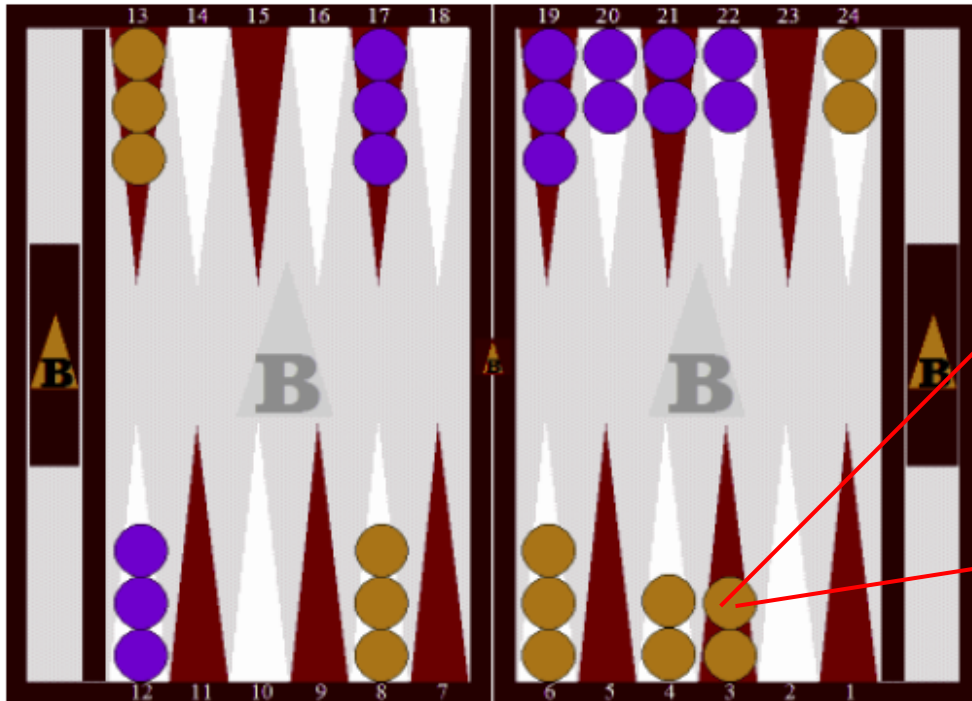


Move: 3/3

1. 17/11 17/11

2. 13/4 13/10
(better)

Example



Move: 5/6

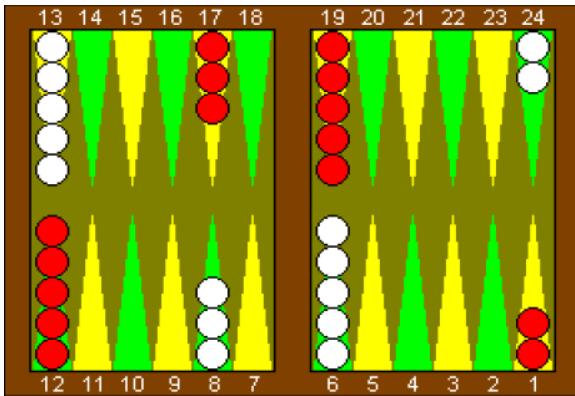
1. 24/13

2. 13/7 8/3
(better)

Move: 6/6

24/18 24/18 8/2 8/2

Opening position



2/3

1. 13/8

2. 24/21 13/11
(better)

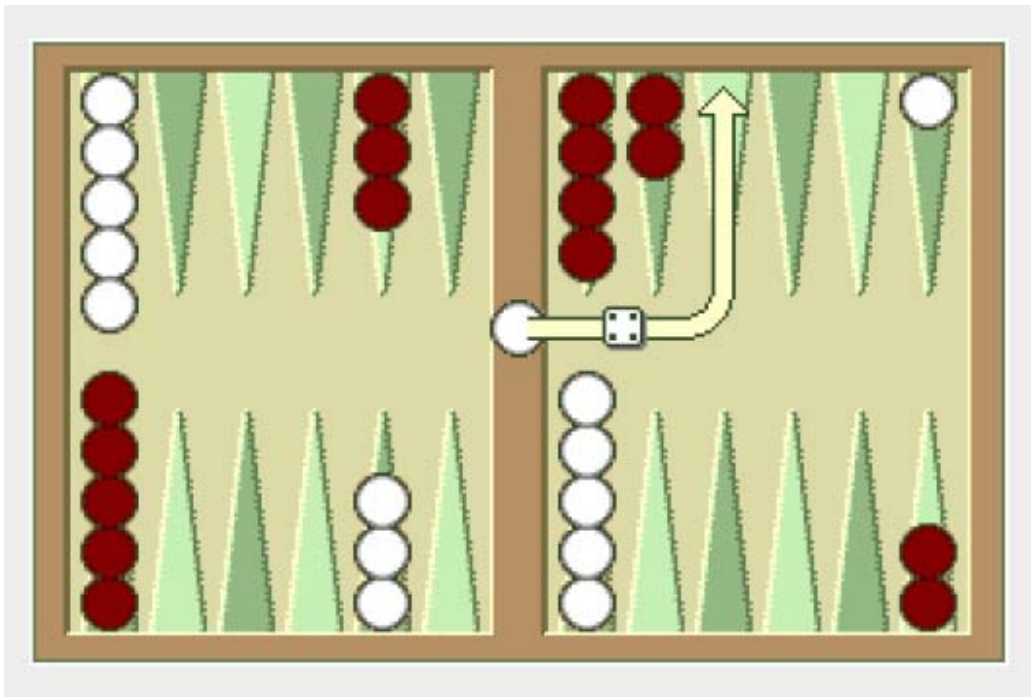
3. 13/11 13/10
(better)

Hitting and Entering

- A point occupied by a single checker of either color is called a **blot**.
- If an opposing checker lands on a blot, the blot is hit and placed on the **bar**.
- Any time a player has one or more checkers on the bar, his first obligation is to enter those checker(s) into the opposing home board. A checker is entered by moving it to an open point corresponding to one of the numbers on the rolled dice.
- For example, if a player rolls 4 and 6, he may enter a checker onto either the opponent's four point or six point, so long as the prospective point is not occupied by two or more of the opponent's checkers.
- If neither of the points is open, the player loses his turn.

Hitting and Entering

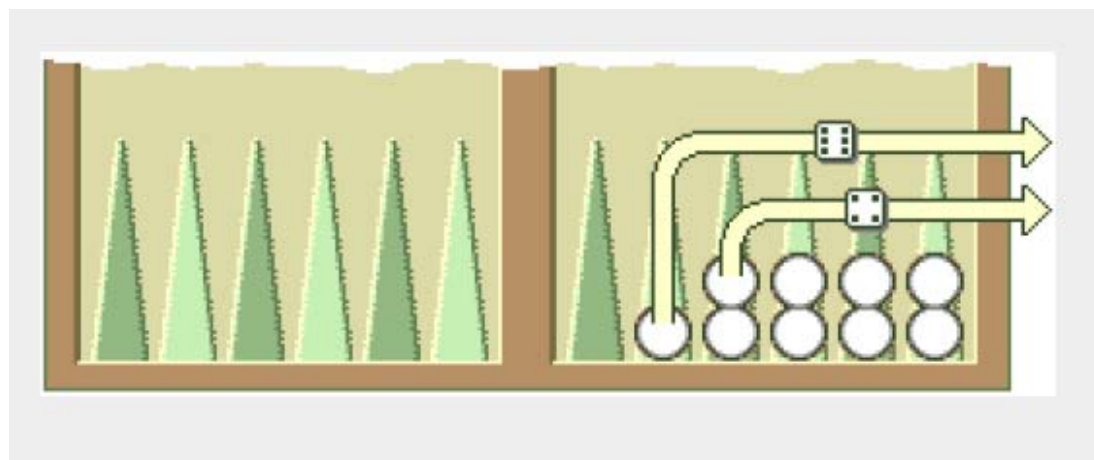
- If White rolls [64] with a checker on the bar, he must enter the checker onto Red's four point since Red's six point is not open.



Bearing Off

- Once a player has moved all of his fifteen checkers into his home board, he may commence bearing off.
- A player bears off a checker by rolling a number that corresponds to the point on which the checker resides, and then removing that checker from the board.
- Thus, rolling a 6 permits the player to remove a checker from the six point.

Bearing Off



White rolls 64 and bears off two checkers.

Bearing Off

- If there is no checker on the point indicated by the roll, the player must make a legal move using a checker on a higher-numbered point. If there are no checkers on higher-numbered points, the player is permitted (and required) to remove a checker from the highest point on which one of his checkers resides. A player is under no obligation to bear off if he can make an otherwise legal move.
- A player must have all of his active checkers in his home board in order to bear off.
- If a checker is hit during the bear-off process, the player must bring that checker back to his home board before continuing to bear off.
- The first player to bear off all fifteen checkers wins the game.

2 TD – Gammon

TD-Gammon

- one of the most impressive applications of reinforcement learning
- required little backgammon knowledge
- yet learned to play extremely well, near the level of the world's strongest grandmasters
- GNU Backgammon: <http://www.gnubg.org/>

TD-Gammon

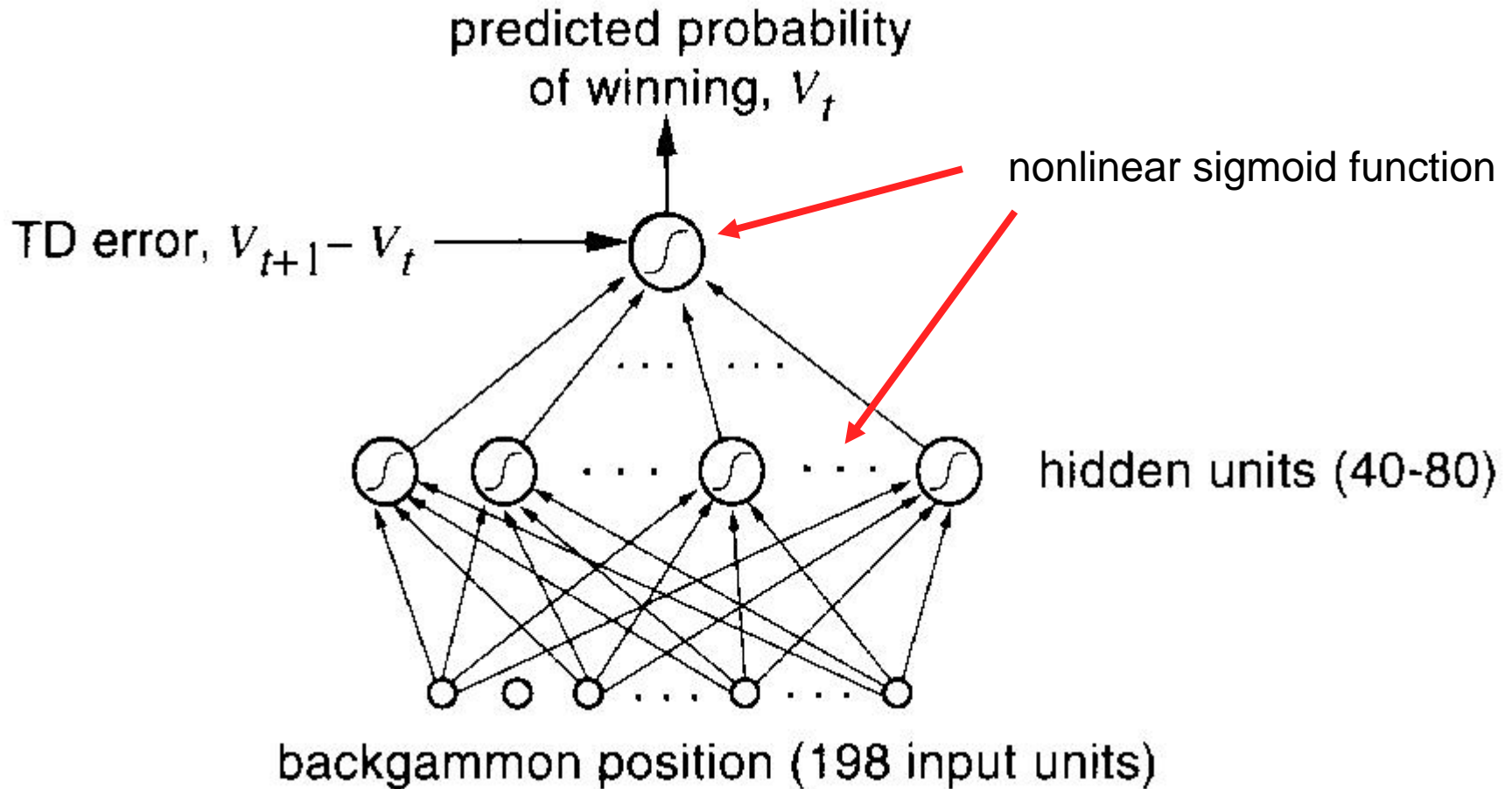
- straightforward combination of the TD(λ) algorithm and nonlinear function approximation using a multilayer neural network trained by backpropagating TD errors
- game = episode, undiscounted
- rewards were defined as 0 for all time steps except those on which the game is won (1)

3 Neural network

Neural network

- 198 input units to the network (representation of a backgammon position)
- final output unit, estimate of the value of that position - value function (probability)
- hidden units (40 – 80)

Multi-layer Neural Network



Input units to the network

- for each point on the backgammon board, four units indicated the number of white pieces on the point
- if there were no white pieces, then all four units took on the value zero
- if there was one piece, then the first unit took on the value 1
- if there were two pieces, then both the first and the second unit were 1
- if there were three or more pieces on the point, then all of the first three units were 1
- the fourth unit took on the value $(n-3)/2$ ($n > 3$)
- with four units for white and four for black at each of the 24 points, that made a total of 192 units

Input units to the network

- two additional units encoded the number of white and black pieces on the bar ($n/2$)
- two more encoded the number of black and white pieces already successfully removed from the board ($n/15$)
- finally, two units indicated in a binary fashion whether it was white's or black's turn to move

On-Line Gradient-Descent TD(λ)

Initialize $\vec{\theta}$ arbitrarily

Repeat (for each episode):

$$\vec{e} = 0$$

$s \leftarrow$ initial state of episode

Repeat (for each step of episode):

$a \leftarrow$ action given by π for s

Take action a , observe reward, r , and next state, s'

$$\delta \leftarrow r + \gamma V(s') - V(s)$$

$$\vec{e} \leftarrow \gamma \lambda \vec{e} + \nabla_{\vec{\theta}} V(s)$$

$$\vec{\theta} \leftarrow \vec{\theta} + \alpha \delta \vec{e}$$

$$s \leftarrow s'$$

until s is terminal

TD-Gammon – Learning

Output (hidden unit j):

$$h(j) = \sigma \left(\sum_i w_{ij} \cdot \Phi(i) \right) = \frac{1}{1 + e^{-\sum_i w_{ij} \cdot \Phi(i)}}$$

weight of ist connection
to the jth hidden unit

value of the ith input unit

nonlinear sigmoid function
(between 0 and 1)
(natural interpretation as a probability)

The computation from hidden units to the output unit
was entirely analogous.

TD-Gammon – Learning

gradient-descent form of the TD(λ) algorithm

update rule:

$$\vec{\theta}_{t+1} = \vec{\theta}_t + \alpha [r_{t+1} + \gamma \cdot V_t(s_{t+1}) - V_t(s_t)] \cdot \vec{e}_t$$

$$\gamma = 1$$

the vector of all modifiable parameters
(in this case, the weights of the network)

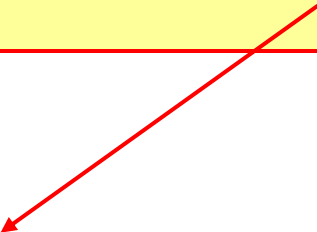
vector of eligibility traces

TD-Gammon – Learning

vector of eligibility traces:

$$\vec{e}_t = \gamma \cdot \lambda \cdot \vec{e}_{t-1} + \nabla_{\vec{\theta}_t} V_t(s_t)$$

$$\vec{e}_0 = \vec{0}$$



The gradient in this equation can be computed efficiently by the backpropagation procedure.

TD-Gammon – Learning

- by playing his learning backgammon player against itself (300000 – 1500000)
- TD-Gammon considered each of the 20 or so ways it could play its dice roll and the corresponding positions that would result
- the network was consulted to estimate each of their values
- the move was then selected that would lead to the position with the highest estimated value
- Continuing in this way, with TD-Gammon making the moves for both sides, it was possible to easily generate large numbers of backgammon games

TD-Gammon – Learning

- the weights of the network were set initially to small random values
- the initial evaluations were thus entirely arbitrary
- Since the moves were selected on the basis of these evaluations, the initial moves were inevitably poor, and the initial games often lasted hundreds or thousands of moves before one side or the other won, almost by accident.
- After a few dozen games however, performance improved rapidly.

4 Results

TD-Gammon 0.0

- about 300000 games against itself
- TD-Gammon 0.0 as described above learned to play approximately as well as the best previous backgammon computer programs
- This was a striking result because all the previous high-performance computer programs had used extensive backgammon knowledge.
- TD-Gammon 0.0, on the other hand, was constructed with essentially zero backgammon knowledge.
- 40 hidden units

TD-Gammon 1.0

- was clearly substantially better than all previous backgammon programs and found serious competition only among human experts
- 80 hidden units

Later versions

- TD-Gammon 2.0 (40 hidden units)
- TD-Gammon 2.1 (80 hidden units)
- augmented with a selective two-ply search procedure
- To select moves, these programs looked ahead not just to the positions that would immediately result, but also to the opponent's possible dice rolls and moves.
- The most recent version of the program, TD-Gammon 3.0, uses 160 hidden units and a selective three-ply search.

Summary of TD-Gammon Results

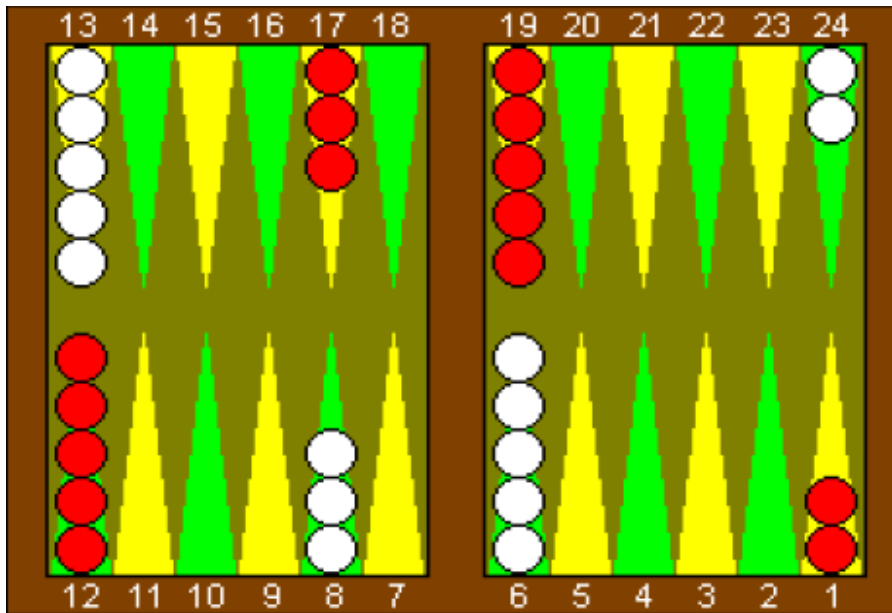
Program	Hidden Units	Training Games	Opponents	Results
TD-Gam 0.0	40	300,000	other programs	tied for best
TD-Gam 1.0	80	300,000	Robertie, Magriel, . . .	-13 points / 51 games
TD-Gam 2.0	40	800,000	various Grandmasters	-7 points / 38 games
TD-Gam 2.1	80	1,500,000	Robertie	-1 point / 40 games
TD-Gam 3.0	80	1,500,000	Kazaros	+6 points / 20 games

Results in play against world-class human opponents

Program	Training Games	Opponents	Results
TDG 1.0	300,000	Robertie, Davis, Magriel	-13 pts/51 games (-0.25 ppg)
TDG 2.0	800,000	Goulding, Woolsey, Snellings, Russell, Sylvester	-7 pts/38 games (-0.18 ppg)
TDG 2.1	1,500,000	Robertie	-1 pt/40 games (-0.02 ppg)

Table 1. Results of testing TD-Gammon in play against world-class human opponents. Version 1.0 used 1-play search for move selection; versions 2.0 and 2.1 used 2-ply search. Version 2.0 had 40 hidden units; versions 1.0 and 2.1 had 80 hidden units.

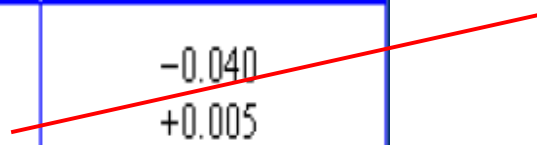
Opening position



4/1

Move	Estimate	Rollout
13-9, 6-5	-0.014	-0.040
13-9, 24-23	+0.005	+0.005

TD – Gammon



TD – Gammon / Expert thinking

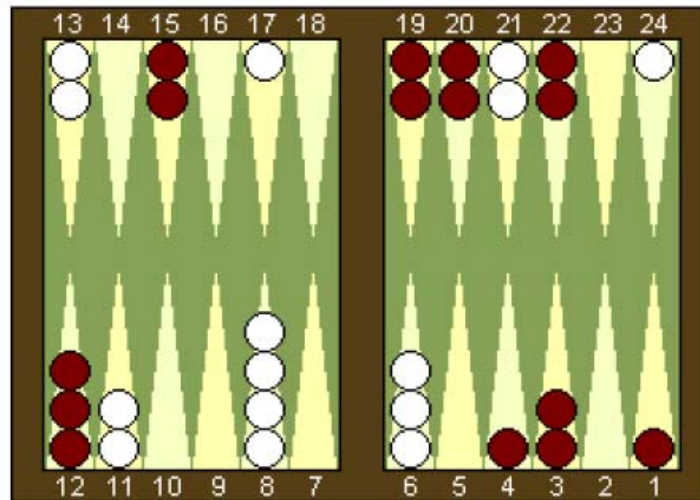


Figure 3. A complex situation where TD-Gammon's positional judgment is apparently superior to traditional expert thinking. White is to play 4-4. The obvious human play is 8-4*, 8-4, 11-7, 11-7. (The asterisk denotes that an opponent checker has been hit.) However, TD-Gammon's choice is the surprising 8-4*, 8-4, 21-17, 21-17! TD-Gammon's analysis of the two plays is given in Table 3.

Move	Estimate	Rollout
8-4*, 8-4, 11-7, 11-7	+0.184	+0.139
8-4*, 8-4, 21-17, 21-17	+0.238	+0.221

Table 3. TD-Gammon's analysis of the two choices in Figure 3. The estimated equity is the neural network's output at the 1-ply level (i.e., no lookahead). The rollout is actual outcome playing each position out 10,000 times to completion with different random dice sequences (see the appendix). Standard deviation in the rollout results is approximately 0.01.

Stochastic Environment

- A second key ingredient is the stochastic nature of the task coming from the random dice rolls.
- One important effect of the stochastic dice rolls is that they produce a degree of variability in the positions seen during training.
- As a result, the learner explores more of the state space than it would in the absence of such a stochastic noise source, and a possible consequence could be the discovery of new strategies and improved evaluations.

Stochastic Environment

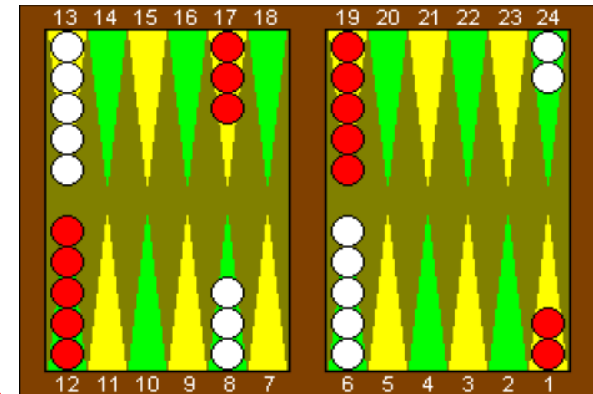
- Another effect of random dice is that they contribute to the terminal states of the system being attractors (i.e., for all playing strategies including random strategies, the sequences of moves will eventually terminate in either a won or lost state).
- In backgammon this comes about partly due to the dice rolls, and partly due to the fact that one can only move one's pieces in the forward direction.

Stochastic Environment

- Finally, non-deterministic games have the advantage that the target function one is trying to learn, the true expected outcome of a position given perfect play on both sides, is a real-valued function with a great deal of smoothness and continuity, that is, small changes in the position produce small changes in the probability of winning.
- In contrast, the true game-theoretic value function for deterministic games like chess is discrete (win, lose, draw) and presumably more discontinuous and harder to learn.






Opening positions

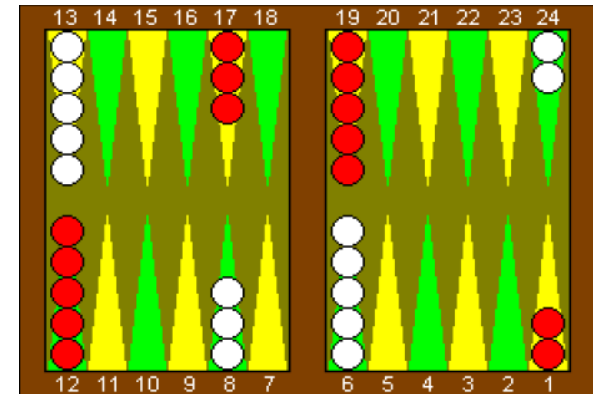
	Spieler dachten, dass (24/23, 13/11) die beste Lösung wäre, aber umfangreiche Computerberechnungen haben gezeigt, dass (13/11, 6/5) Ihnen die besten Chancen zum Sieg ermöglicht. Dieser Zug ist etwas riskanter und ist abhängig von der Punktzahl, die beim ersten Spiel gespielt wird.
	Nur eine Möglichkeit: (8/5, 6/5).
	(24/23, 13/9) ist hier der beste Zug. Sie müssen keinen offenen Stein am 5. Point hinterlassen, da der Spielstein des 9. Points Ihnen bereits genug Möglichkeiten bietet einen wichtigen Point beim nächsten Wurf zu spielen.
	Die beste Option hier beläuft sich auf: (24/23, 13/8). Es hat einen Vorteil gegenüber (13/8, 6/5) was eine gute Möglichkeit ist, wenn Sie mehr Risiko eingehen wollen um ein "gammon" zu spielen.
	(13/7, 8/7), kein Zweifel.
	Die beste Möglichkeit: (24/21, 13/11) aber eine gute Alternative ist: (13/11, 13/10).
	(8/4, 6/4), Sie können keine Chance verpassen einen 4.Point zu spielen.
	Computerberechnungen sagen, dass (24/22, 13/8) Ihnen die besten Möglichkeiten geben zu gewinnen. Experten dachten über Jahre, dass (13/11, 13/8) am besten ist.
	(24/18, 13/11) ist am besten. Mit ihrem letzten Spielstein zu laufen (24/16) bringt Sie nicht weit genug.




Computer

Opening positions

	Sie können auswählen zwischen: (13/10, 13/9) (24/20, 13/10) (24/21, 13/9). Wählen Sie das aus, was am besten zu Ihrem Spielstil passt.
	Machen Sie ihren 3-Point (8/3, 6/3).
	Aufteilen (24/18, 13/10) ist vorteilhafter als: (24/15).
	Zwei Möglichkeiten, entweder: (24/20, 13/8) oder (13/9, 13/8).
	Das lässt Ihnen drei Möglichkeiten. Sie können laufen (24/14) oder aufteilen (24/18, 13/9). Diese hier (8/2, 6/2) ist laut Computerberechnungen auch nicht so schlecht. Professionelle Spieler jedenfalls, können sich nicht damit abfinden, dass man diese Spielsteine nicht mehr strategisch nutzen können.



	Leicht, laufen Sie einfach mit ihrem letzten Stein (24/13).
---	---

Another games

- Reversi
- Poker
- Siedler von Catan