Accelerating Problem Solving by Combining Machine Learning and Human Learning

David B. Fogel, Ph.D.
Natural Selection, Inc.
3333 N. Torrey Pines Ct., Suite 200
La Jolla, CA 92037
dfogel@natural-selection.com

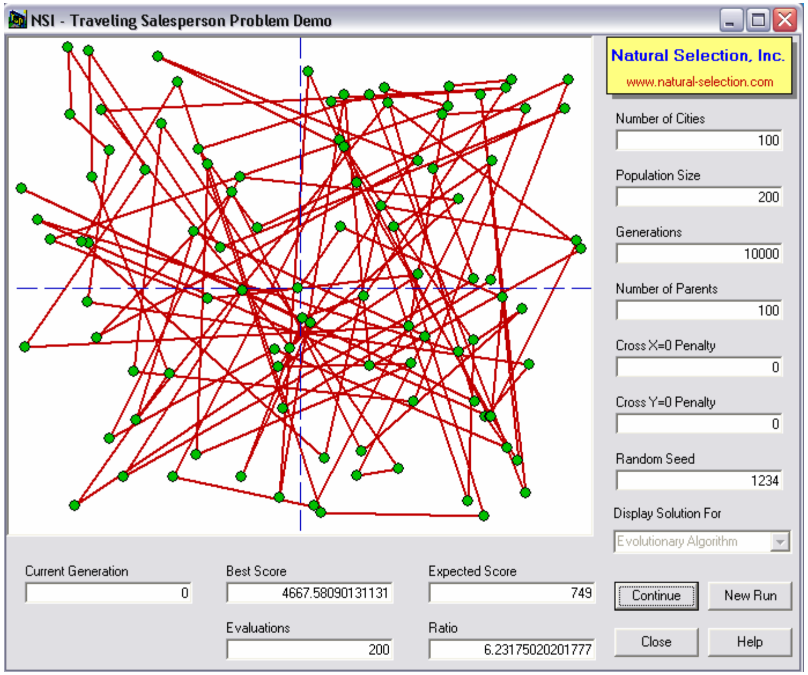
The Future of Problem Solving Combines Machines with Humans

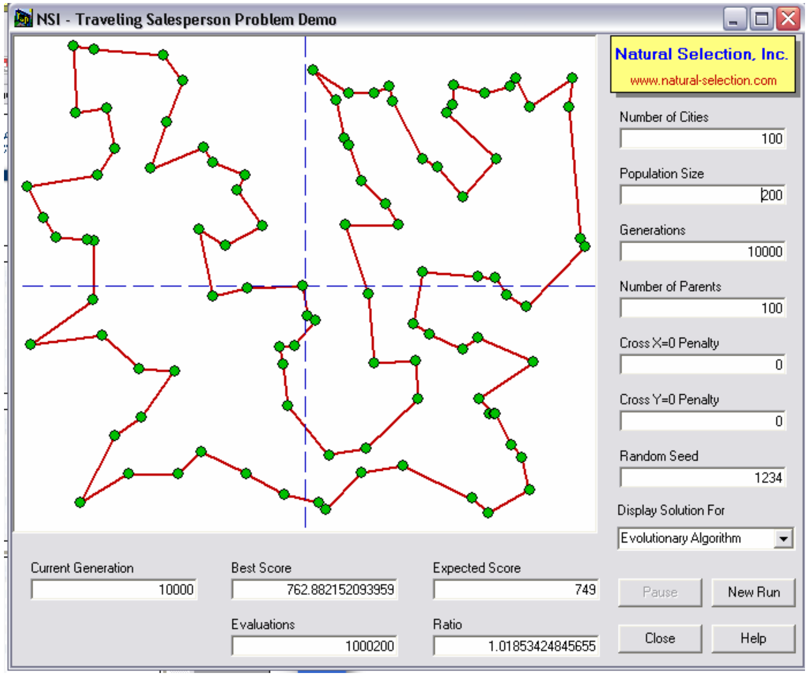
- Computational intelligence methods can solve problems autonomously
- When combined with human expertise, problem solving accelerates
- When combined with hardware advances, we become truly orders of magnitude more able to address the problems that we face
- My focus is on evolutionary machine learning

Combining Machine Intelligence with Human Intelligence

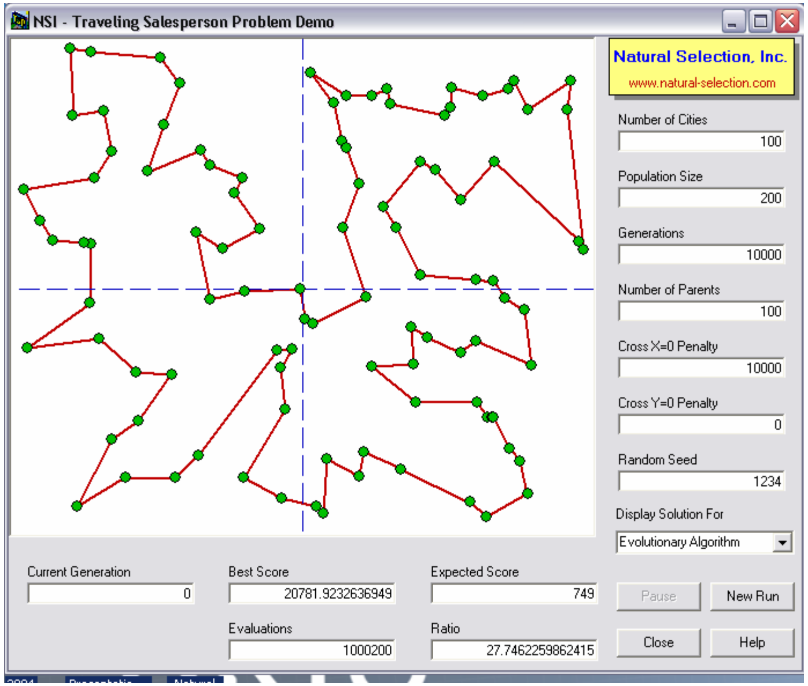
- Evolutionary computation can discover informative patterns using “self-play”
  - Strategies compete against each other
- Each generation reuses the information stored in the previous generation to explore for new strategies
- Can be combined with human intelligence to achieve solutions rapidly
- Projection: Combination of methods leads to significant acceleration of problem solving in the coming decade

Evolutionary Traveling Salesman Demonstration



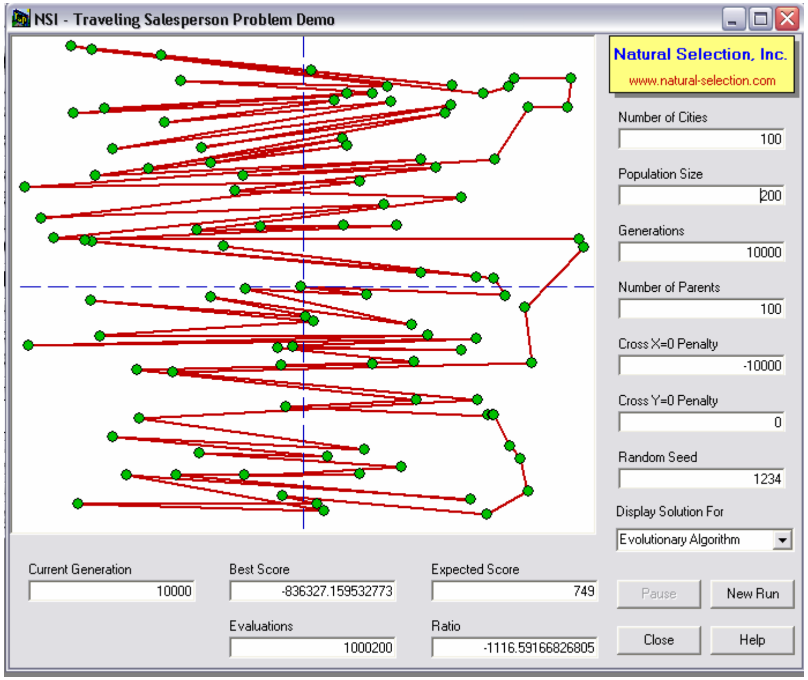


Traveling Salesman Demonstration: Cross the Y-axis as Little as Possible



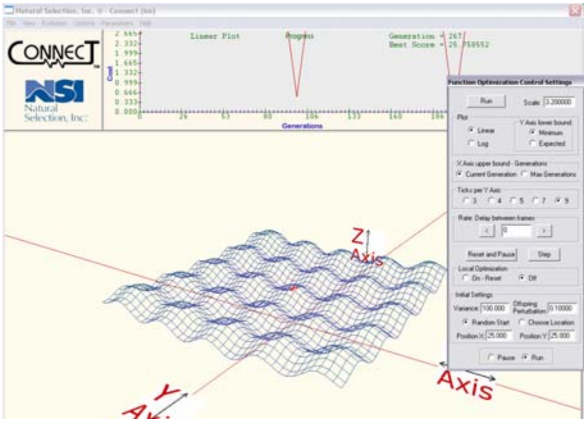
- The traveling salesman must visit every city, so he must cross the y-axis at least twice
- The best evolved solution crosses only twice and still provides an optimized path
- Whether or not the path is perfect is unknown
  - The solution provided is “good enough, fast enough to be useful.”

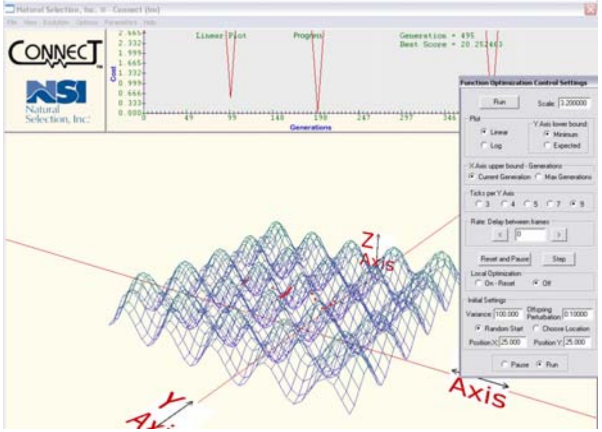
Traveling Diamond Smuggler Example

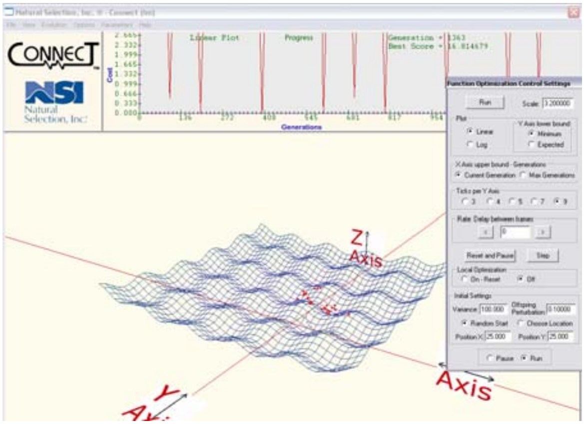


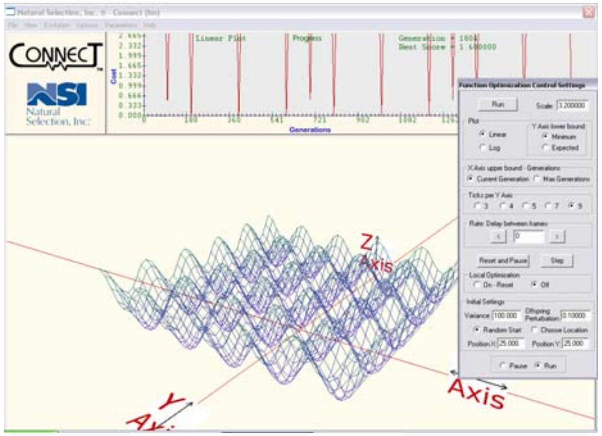
- Traveling diamond smuggler gets paid every time he crosses the y-axis
- Must still minimize path length to avoid detection by authorities

Evolutionary Optimization in Continuous Domains and Changing Environments









Combining Machine Intelligence with Human Intelligence

- Evolutionary computation can discover informative patterns using “self-play”
  - Strategies compete against each other
- Each generation reuses the information stored in the previous generation to explore for new strategies
- Can be combined with human intelligence to achieve solutions rapidly
- Projection: Combination of methods leads to significant acceleration of problem solving in the coming decade

Striving for Computational Intelligence

- Since the advent of the modern digital computer we’ve tried to generate machines as intelligent as humans
- Primary goal of early artificial intelligence
  - General problem solver
- What is intelligence?
  - Central problem: No well-accepted definition of intelligence, let alone “artificial intelligence”
- Surrogate challenges in the form of Turing Tests

Turing Test: The Imitation Game

- Turing (1950) replaced the question of “Can a machine think?” to “Can a machine fool an interrogator as well as a human?”
- The test involves a man trying to convince an interrogator than he is a woman
- Machine “passes the test” if it can fool the interrogator as often as the man

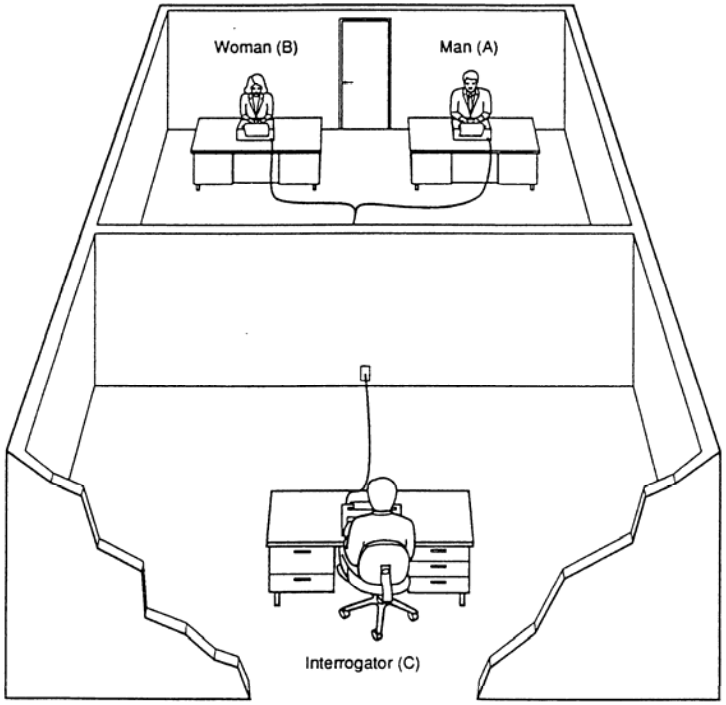


Figure 1-1 The Turing test. An interrogator (C) questions both a man (A) and a woman (B) and attempts to determine which is the woman.

Turing Test → Intelligence?

- Turing (1950) never claimed that a machine that passes the test would be "intelligent"
  – "Too meaningless to deserve discussion"
- The Turing Test is no more of a test for thinking machines than it is a test for femininity
  – If a man can fool an interrogator into believing he is a woman as often as a woman can convince the interrogator, is the man a woman?
- What were the consequences of focusing on this test?

Consequences

- Impossible to envision passing the test in the 1960s based purely on computer speed
- Narrow the focus
  - Try games
- Emphasize applications, emphasize humans
- Ask experts how they do things
- Never mind what intelligence is or what sort of intelligence we are trying to generate
  - Minsky: Intelligence means the ability to solve hard problems
- The Turing Test led to the death of AI

Artificial Intelligence

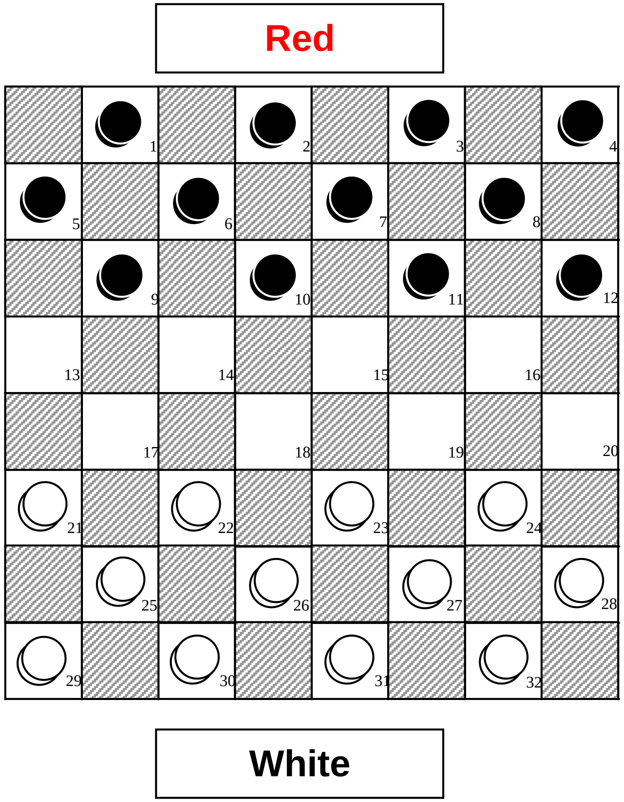
- For an organism (system) to be intelligent, it must make decisions
- A decision arises when available resources must be allocated
  - Must face a range of decisions, otherwise there’s really no decision at all
- Decision making requires a goal
- Intelligence may be defined as “the ability of a system to adapt its behavior to meet its goals in a range of environments”

Adaptive Behavior → Evolutionary Computation

- Adaptation is fundamentally an evolutionary process whether it occurs in phyletic, ontogenetic, or sociogenetic systems
- Unit of mutability and reservoir of stored knowledge
- The mechanisms for change and memory differ but the behavioral effects are notably similar
- If we really want to talk about intelligent machines we have to talk about machines that learn and adapt to meet goals based on experience
  - MACHINES THAT EVOLVE

The Game of Checkers

- 8x8 board with red and black squares
- Two players (Red & White)
- 12 pieces (checkers) for each player
- Diagonal Moves
- Jumps are forced
- Checkers and kings
- Win, lose, and draw

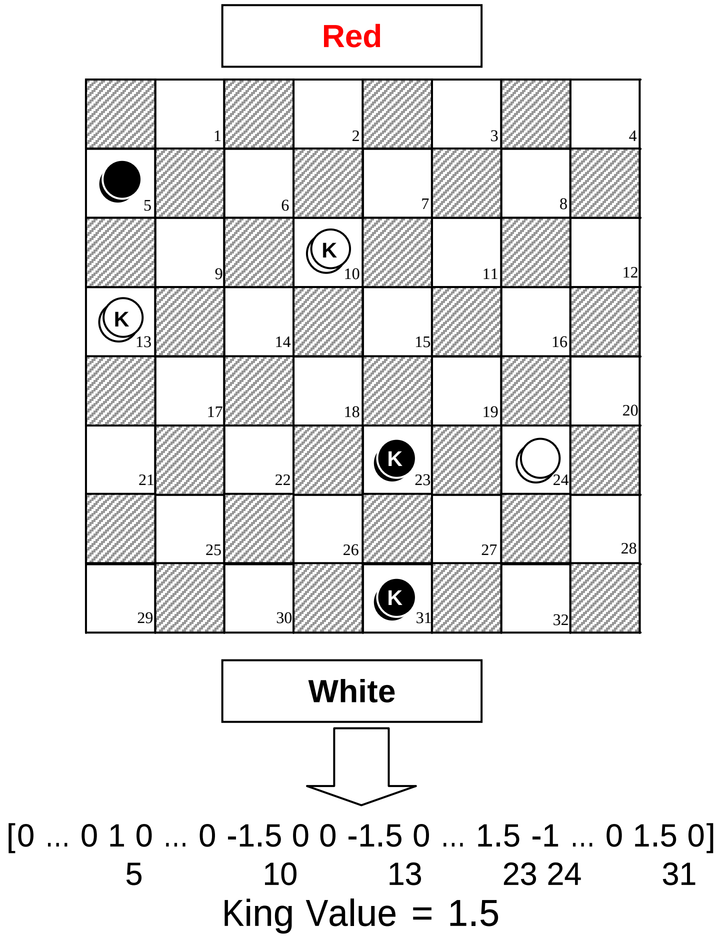


Computer Checkers

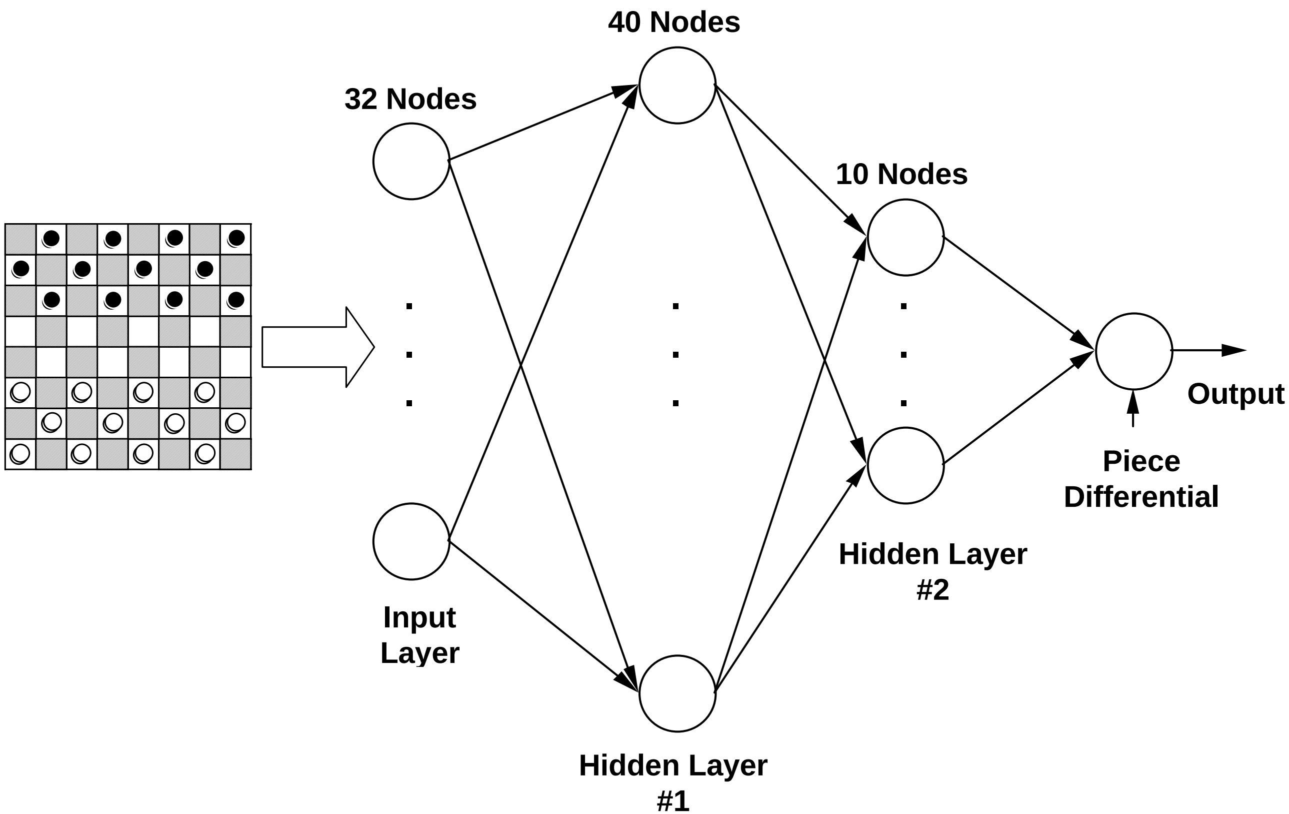
- Samuel’s first checkers program
- World Man-Machine Checkers Championship
  - Chinook defeated Marion Tinsley (human), the world checkers champion and won the championship
- Chinook
  - Incorporated a linear polynomial as a board evaluator
  - All “items” of knowledge were preprogrammed, opening book, and all 8-piece endgame database (440 billion stored positions)
  - Did not use any learning
- Programmed human expertise to beat human expertise

Evolving Strategies for Checkers

- 32x1 board vector
- Entries {$-K$, –1, 0, 1, K}
- Players pieces positive
- Opponents pieces negative
- Each player consisted of
  - A neural network board evaluator
  - A unique king value K
  - The NN and K are evolvable
- Minimax search
  - 4-ply for training and 6-ply for playing against humans



Neural Network Architecture



- The closer the NN output was to 1.0 the better the move
- The pieces changed sign when move alternated between players

Evolving Checkers Players

- Neural network weight update

$$\sigma'_i(j) = \sigma_i(j)\exp\left(\tau N_j(0,1)\right), \qquad j = 1, \ldots, N_w \ \text{ and } \quad \tau = \left(\sqrt{2\sqrt{N_w}}\right)^{-1}$$

$$w'_i(j) = w_i(j) + \sigma'_i(j) N_j(0,1), \qquad j = 1, \ldots, N_w$$

- King value update

$$K' = K + 0.1U, \qquad \text{where } U \in \{-1,0,1\}$$

  - K was limited to [1.0,3.0]
- Tournament
  - Each player (parents and offspring) played one checkers game with five randomly selected opponents from the population
  - Win = 1 points, draw = 0 points, and loss = –2 points
  - Games were limited to a maximum of 100 moves

Evolution

- 0. Initialization
  - 15 parents with NN weights uniformly sampled from [–0.2,0.2]
- 1. Offspring generation
  - Each parent generated one offspring
- 2. Tournament
  - All 30 players competed with 5 randomly selected players from the population
- 3. Selection
  - 15 players with the greatest total points were retained as parents for the next generation
- 4. Loop back to step 1.
- Evolution was conducted for 100 generations

Evaluation Against Human Players

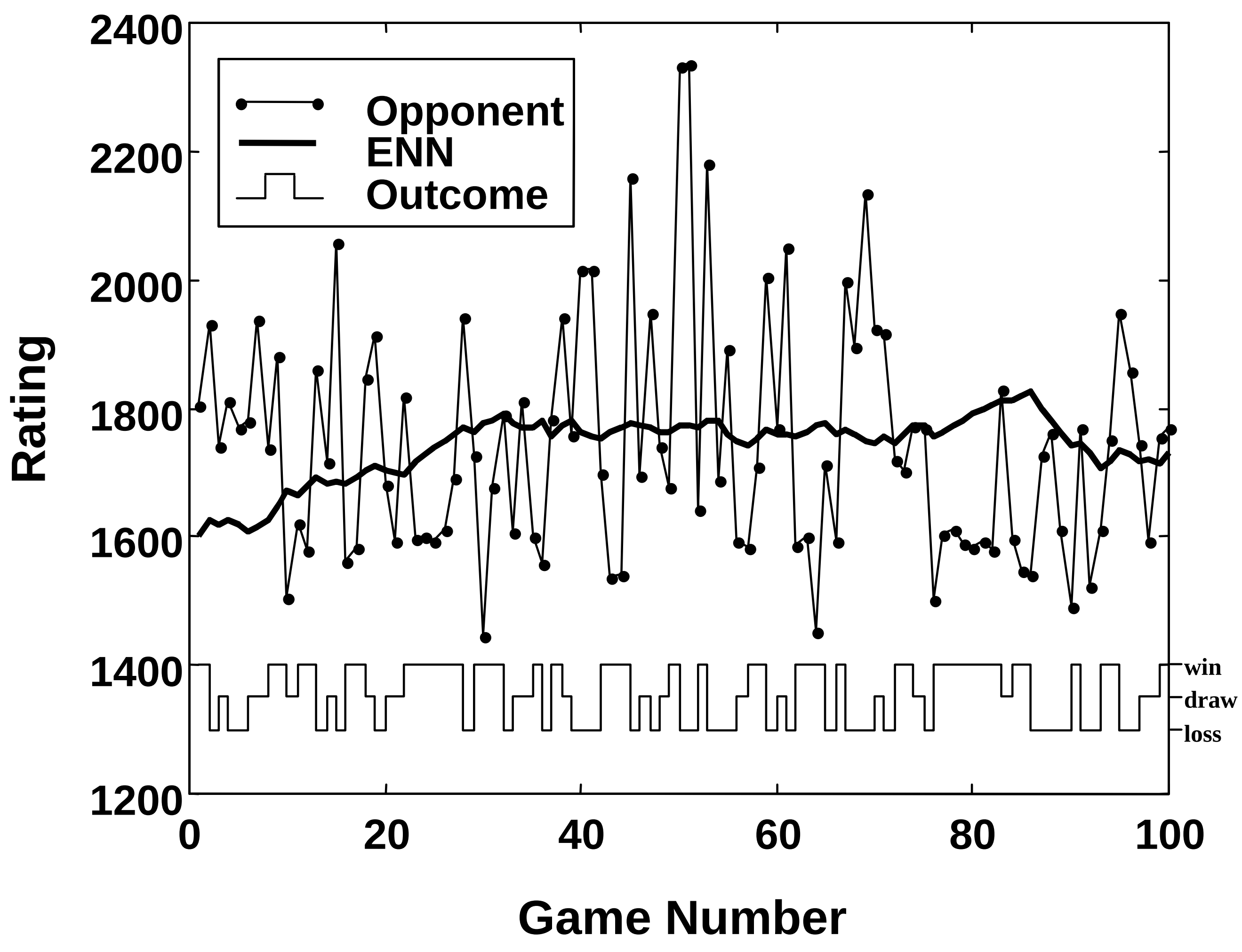
- Best player at generation 10 defeated the authors (novice checkers players)
- Best player at generation 100 was evaluated over 100 games against rated human players at the internet gaming site: www.zone.com
- USCF checkers rating on the zone
  - Starts out at 1600 and follows:

$$\text{Outcome} \in \{\ 1(\text{win}),\ 0.5(\text{draw}),\ 0(\text{loss})\ \}$$

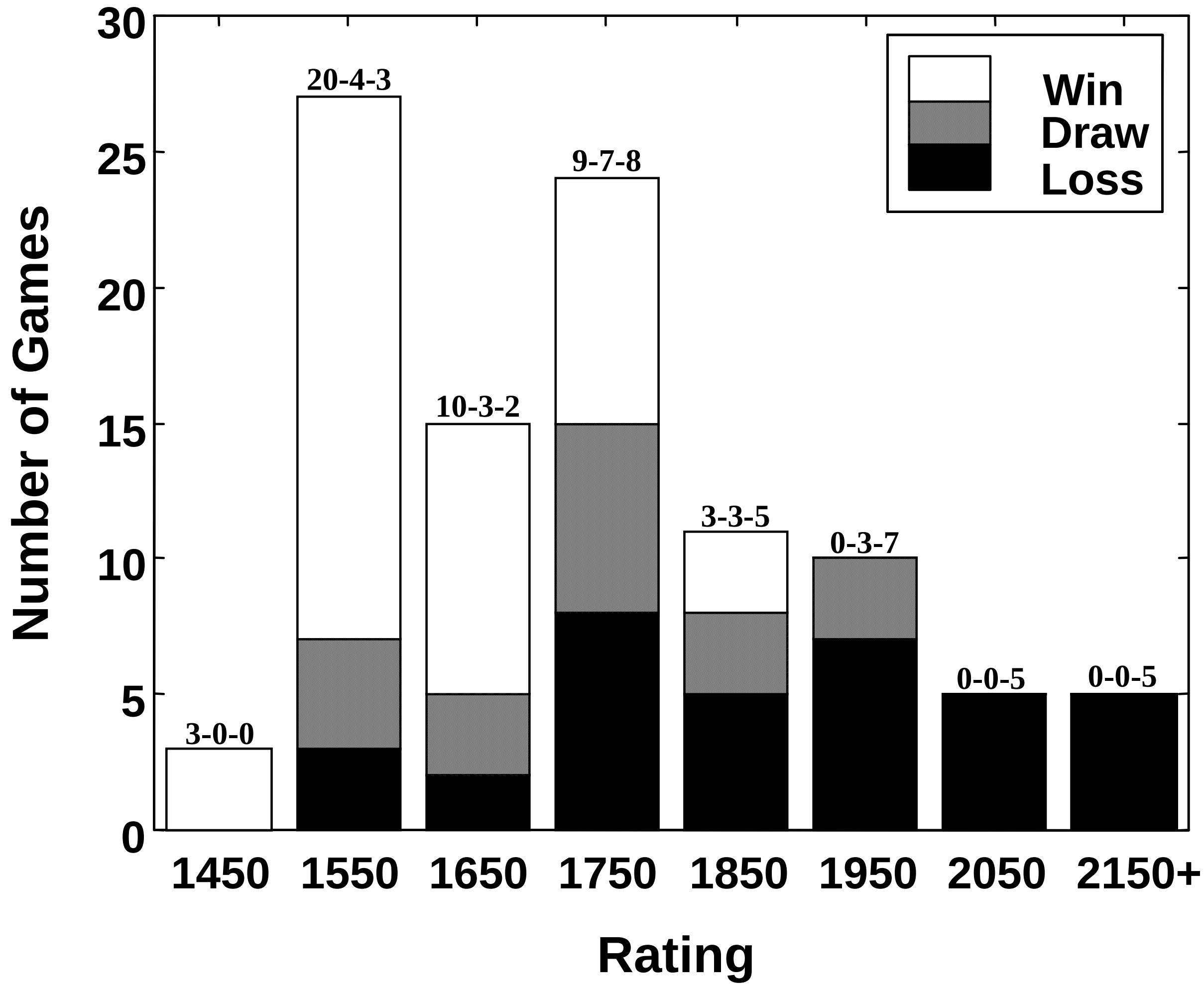
$$W = \frac{1}{1 + 10^{0.0025\left(R_{opp} - R_{Old}\right)}}$$

$$R_{new} = R_{old} + 32(\text{Outcome} - W)$$

Results: Game Outcomes



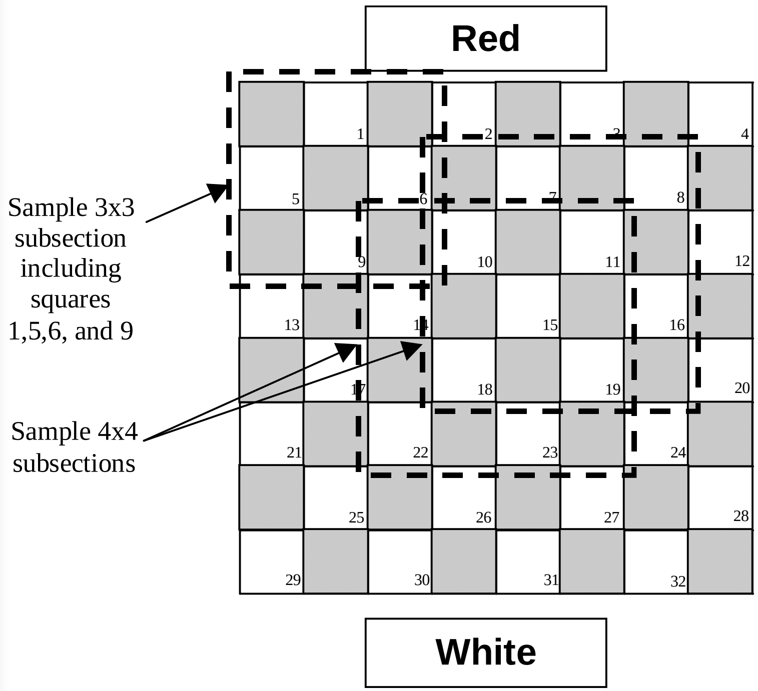
Results: Game Outcomes

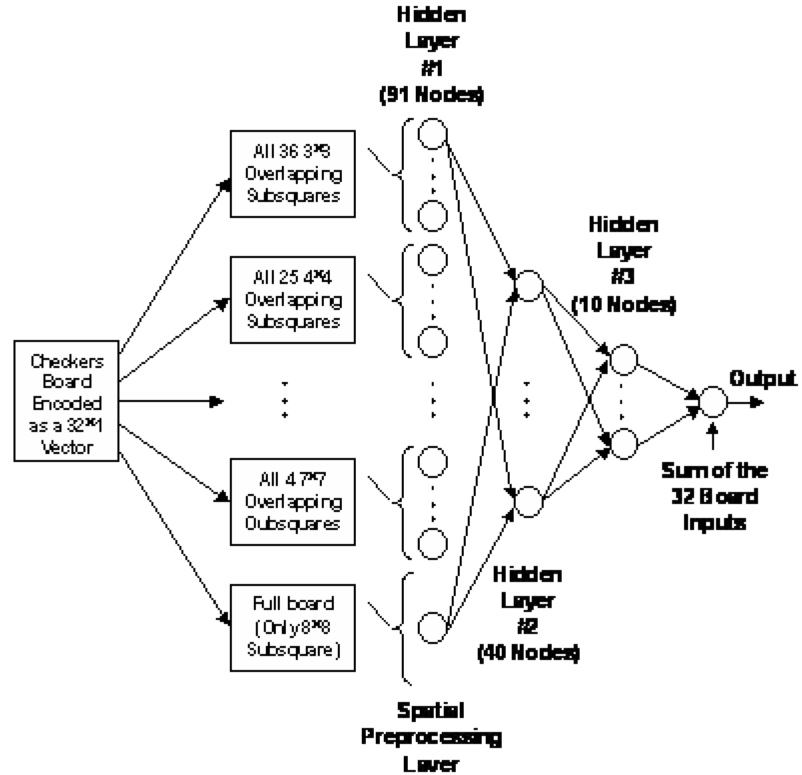


Example of Human Feedback

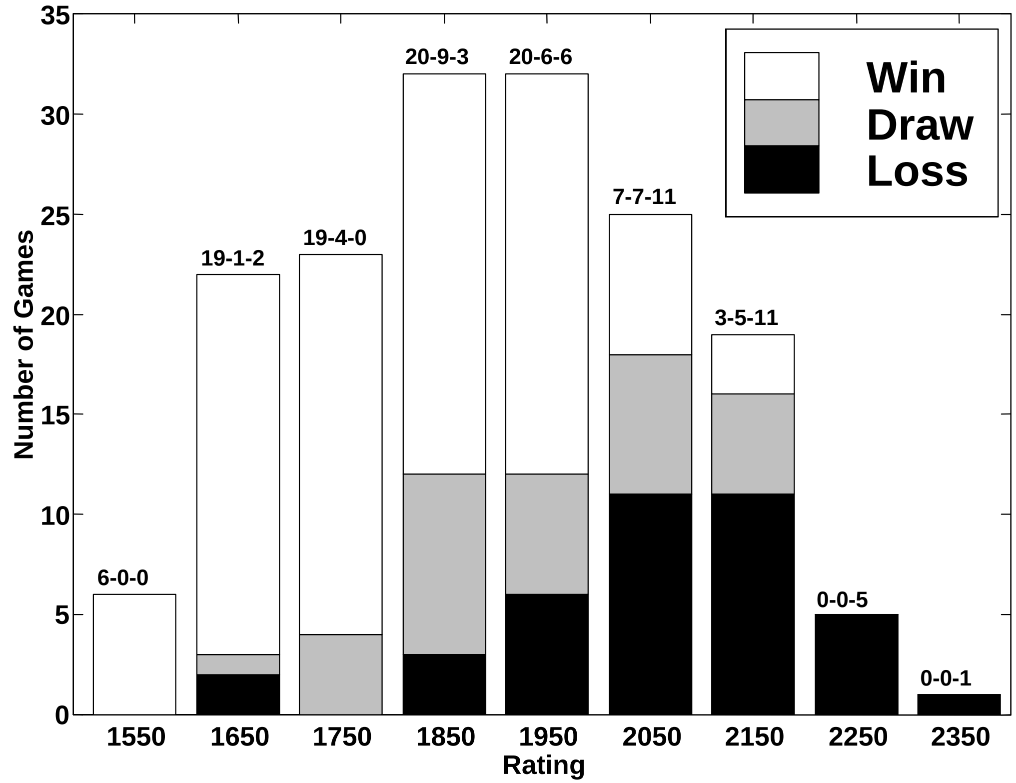
- Chatbox affords possibility for opponents to communicate
- Often received compliments when neural network made good moves

Extension to Object Neural Networks





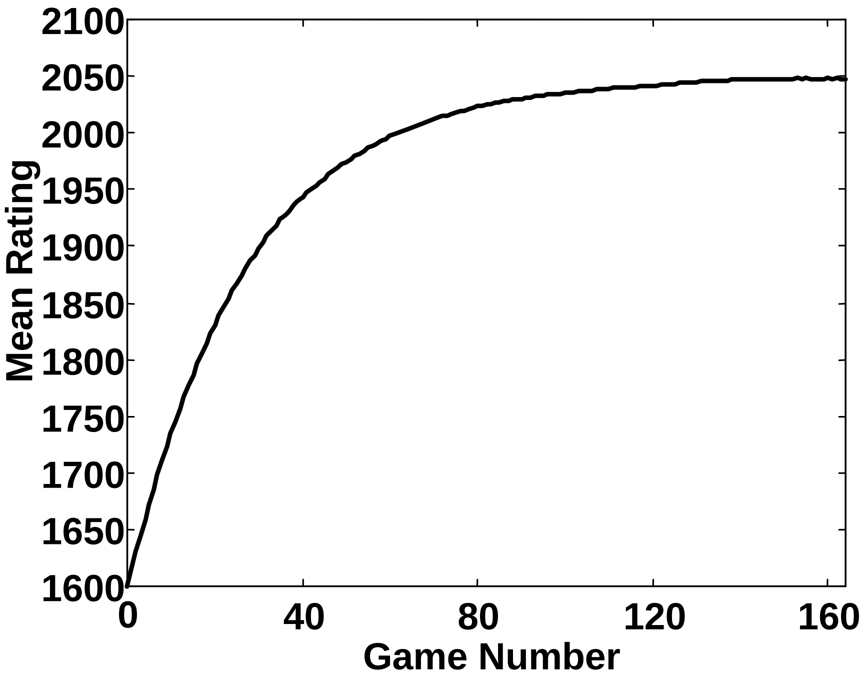
Results with Object Neural Network



- Trained over 840 generations (6 months, P2 450MHz)
- Tested on 165+ games, Blondie24

Summarizing Blondie24 Results

- Rated at 2045, “expert” level
- Often played to restrict mobility of opponent
  - To the extent that the neural network used this feature, it first *had to invent the feature*
  - We named the neural network Anaconda
  - Then brought Blondie24 out of retirement



Checkers Challenges

- ***Chinook***
- 10 games at the novice setting (high-level expert)
- Results:
  - Wins: 2, Draws: 4, Losses: 4
- Verifies expert rating

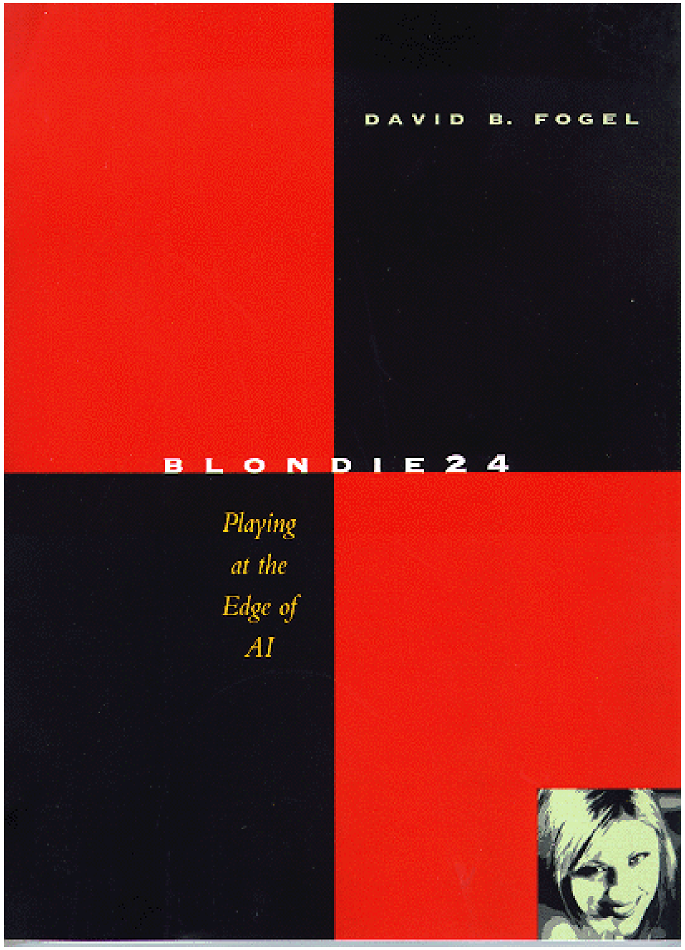
- **Playsite.com Tournament**
- 8 minutes/move
- Blondie24 won the tournament

- ***2000-2001 Congress on Evolutionary Computation***
- Over 100 colleagues have challenged Anaconda for $100-$200
- My money is still safe

- Written about in the *NY Times*

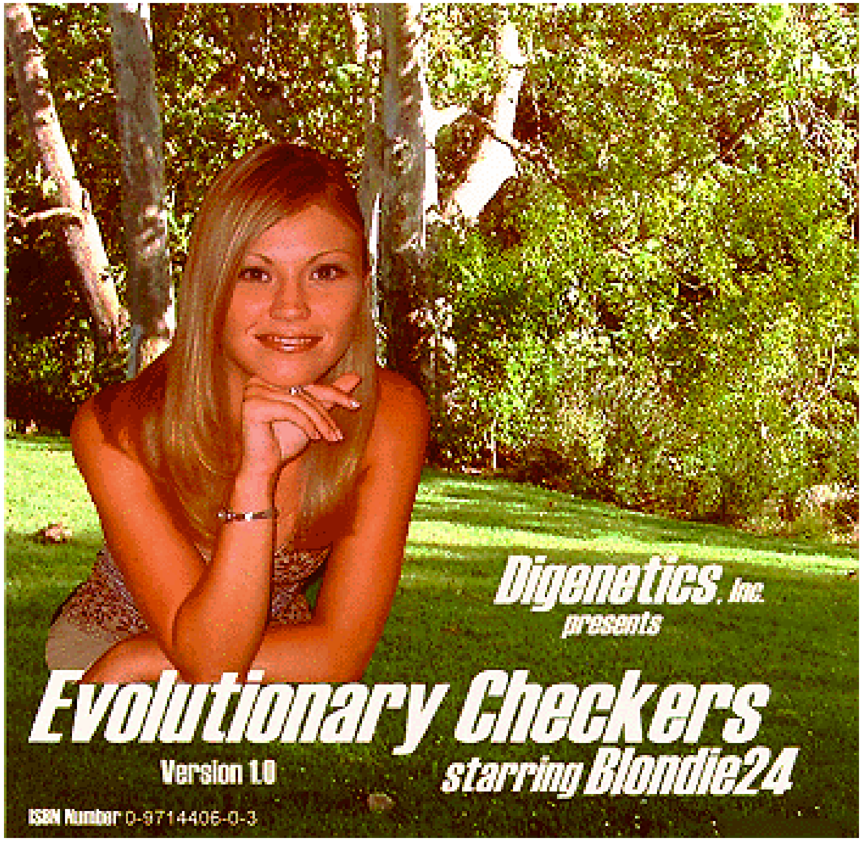
Blondie24:
Playing at the Edge of AI

- Amazon.com
- Written for general science audience
- Published by Morgan Kaufmann Publishers, Inc. (2002)

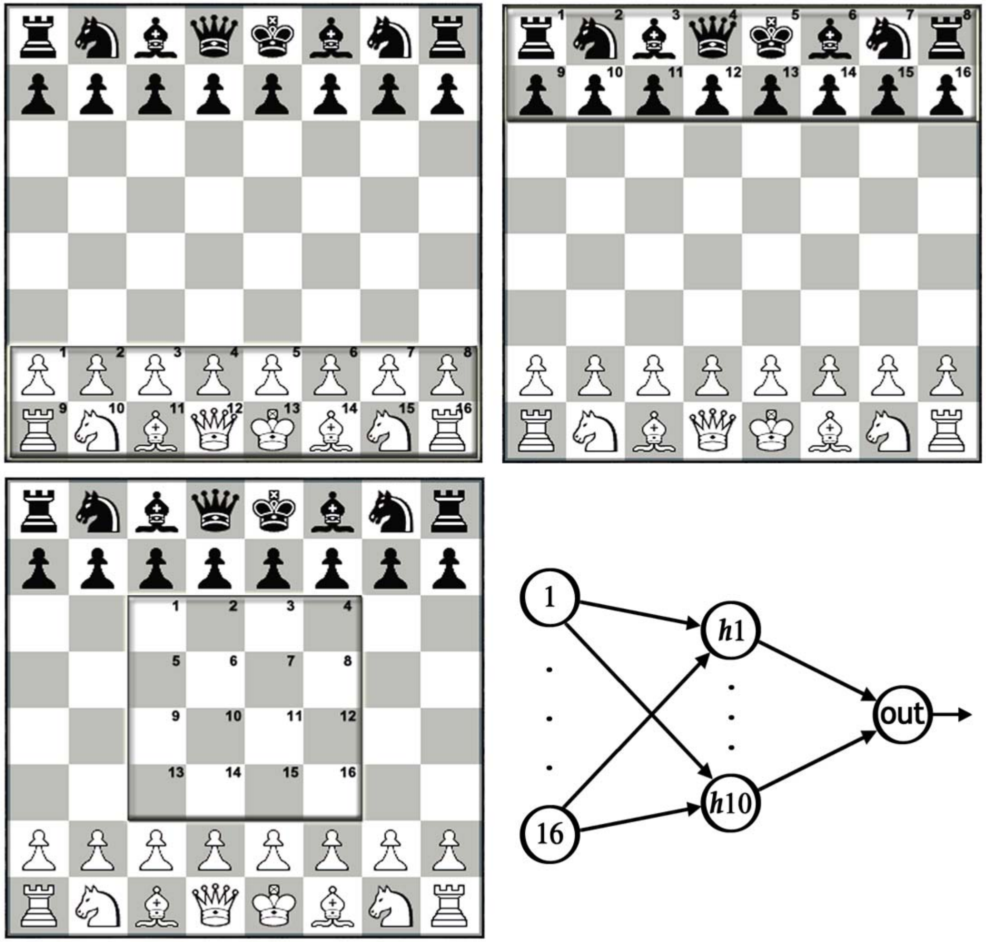


Evolutionary Checkers starring Blondie24

- Digenetics, Inc. has brought Blondie24 to life
- www.digenetics.com
- Version 2.0 includes Blondie's "friends," Shannen and Amber
- At Boston Museum of Science



Evolving Object Neural Networks for Chess



Evolutionary Parameters and Baseline

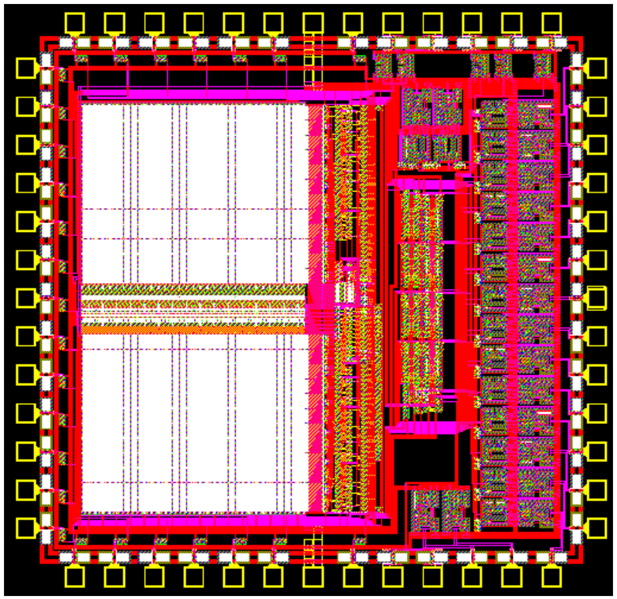
- Material values
- Position value tables per piece type
- Neural networks (3)
- 10 parents, 10 offspring
- Typical mutation and self-adaptation
- 50 generations, 10 independent trials
- Non-evolved material + PVTs yielded rating of 1870 (Class A) when tested against Chessmaster (65 games)

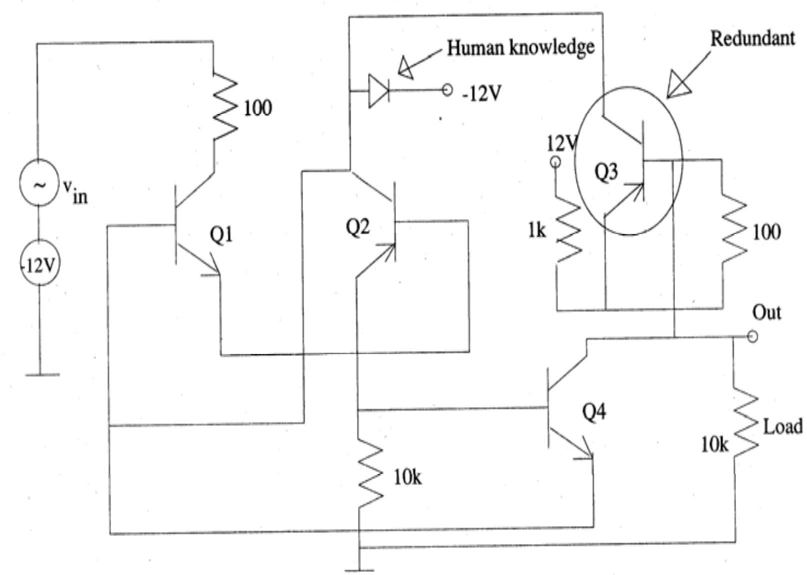
Testing in Tournament Conditions

- Continued evolving Blondie25 this year for 7462 generations
  - All on one P4 2.5GHz/512MB, 7 months
- New tests show 69% win rate against the non-evolved player, up from 61%
- Tests against Pocket Fritz 2.0
  - 13 Wins, 0 Losses, 3 Draws = rating of 2650
- Tests against Fritz 8
  - 3 Wins, 11 Losses, 10 Draws = rating of 2650

Digenetics, Inc.
Chess with an Attitude!

Evolving New Electronic Circuits

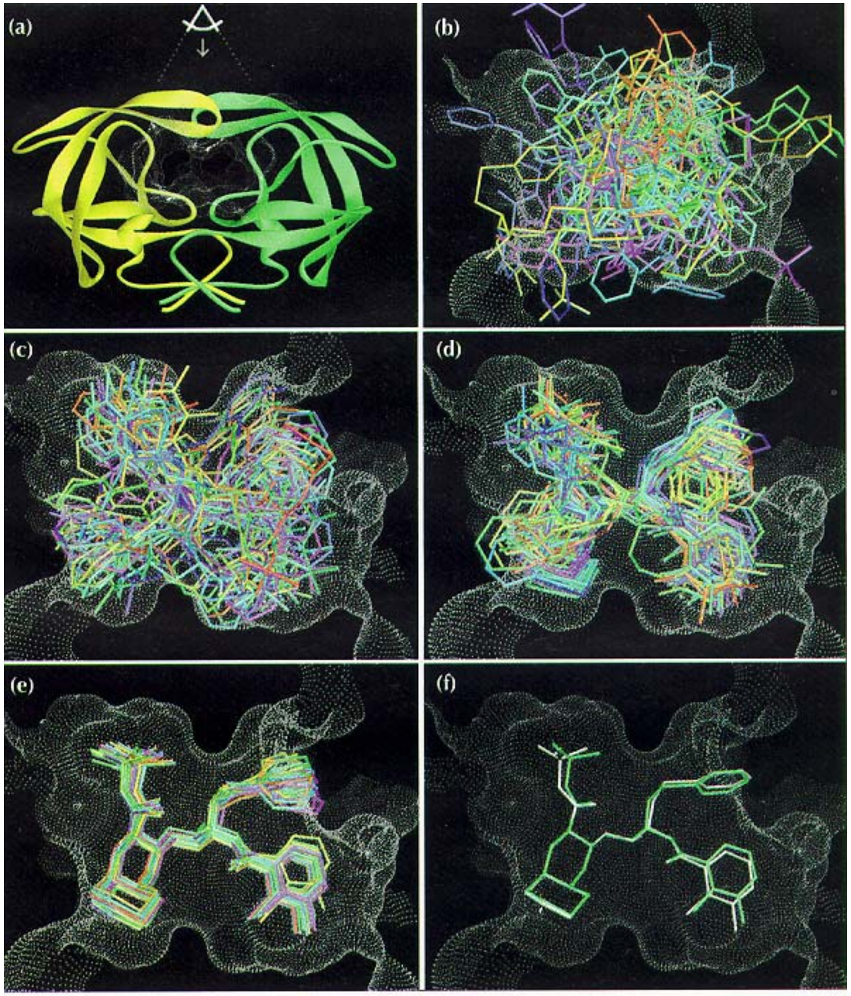




- Complicated circuits can be invented by manipulating the positions of resistors, capacitors, and other electronics
- Hybrid system of domain knowledge, structural representation, variation operators

Evolving New Drugs

- Using accurate models of the target protein, pharma-ceutical companies can screen 50,000-100,000 possible candidate drugs in less than a week by evolving those that appear best
- Hybrid system of domain knowledge, evolution, traditional gradient search



What's Next?

- Combining machine intelligence with human intelligence (and expertise) can already create significant solutions to vexing problems
- Hardware designs in the future will be tailored to make best use of the algorithms, and their synergy with human problem solving
  - Advances in industry, medicine, defense, and finance

Acknowledgments

- Thanks to Tim Hays, Sarah Hahn, James Quon, Kumar Chellapilla, Doug Johnson, Paul Werbos, Garry Kasparov
- Thanks to AC2005 organizers
- Work sponsored in part by NSF SBIR grants DMI-0232124 and DMI-0349604