A common limitation of conventional video games is that players quickly learn the positions and behavior of computer-controlled characters, which usually take the form of monsters. Software developers preprogram these characteristics so, after playing the game several times, the player comes to know exactly how and when the monsters will act. The game eventually becomes boring because the player need only execute a learned script to defeat the monsters and overcome all the hurdles he or she faces.

This pattern has sparked increasing interest in using computational intelligence techniques to control the actions of computer characters, rather than relying on simple heuristics or rule-based systems. In particular, evolutionary computation and neural networks are being adapted to let software characters learn from their own experience, predict what a player might do next, and take appropriate action to meet their own challenges. In this way, a game can remain perpetually novel, posing new tests for the player each time he or she plays.

**TRAINING BLINDFOLDED**

In pursuit of this goal, game developers can combine evolutionary computation and neural networks to let software agents learn appropriate behaviors even in complex strategy games, without resorting to preprogrammed features provided by software engineers a priori. The evolutionary process can even invent its own features for describing situations in a game and learn to weight those features and act based on patterns it recognizes.

To place this capability in context, consider the following thought experiment. Suppose you sit down at a table to play a game with an opponent. You’ve never played a game of any type before. You face a two-dimensional board with eight squares on each side, with alternating colors, and, as Figure 1 shows, pieces occupy every other square. You’re told the rules: The game takes place in turns, red moves first, and pieces move forward one square at a time, diagonally. Further, pieces that reach the back row become kings, a status that lets them move forward or backward one square at a time. Finally, when the possibility of jumping over an opponent’s piece becomes available, you must take that jump move and remove the opponent’s piece from play.

What you aren’t told is the object of the game. Instead, your opponent gives you the first move and responds in kind. After some number of moves, you’re told that the game has ended and you’re challenged to play again. Naturally, you’d like to know the outcome of the first game before playing a second one. This information, however, is withheld from you until you finish a random number of games. Only then do you learn that you earned, say, eight points. Eight points is better than seven and not as good as nine. You aren’t told, however, which games garnered you points and which were less successful.

Given these parameters, consider how many games it would take you to learn to play at the level of human experts—people who know the object of the game, can study and read books about it, converse with other experts, and hone their skills through practice.

**EVOLVING A CHECKERS EXPERT**

The game in this case is checkers, and this is essentially the task that Kumar Chellapilla and I gave to an evolutionary algorithm that uses artificial neural networks as checkerboard evaluators. I describe the experimental design and its implications in depth in my book, *Blondie24: Playing at the Edge of AI* (Morgan Kaufmann, 2002).

We presented only the raw information from the checkerboard as inputs to the neural networks: the position,
type, and number of pieces. The neural networks operated on these data, processing them through two hidden layers of 40 and 10 nodes, respectively, to ultimately derive an output value for any board that would range between −1 and +1. Higher numbers indicated more favorable positions.

In the evolutionary algorithm’s first generation, we set all the neural networks’ connection weights at random values. Each neural network played five games as red against a random collection of other neural networks in the initial population. As a result, the games proceeded at first with random moves, with minimax pruning of nodes and branches to provide a rationale for choosing which move works best in a particular setting when evaluating some number of moves into the future.

Even with random initial moves, some moves proved better than others. The neural networks could earn or lose points in each match: They gained one point for a win, lost two points for a loss, and received no points for a draw. We withheld this specific information from them, however, and used only each neural network’s overall score gained after all neural networks completed their games to evaluate their relative performance.

Culling the weak

Based on their scores, we killed off the lower half of the neural network population and used the upper half as the basis for generating offspring in the next generation. We kept variation to a very simple mutation of each of the weights of every parent neural network, along with a mutation of the neural network’s overall score gained after all neural networks completed their games to evaluate their relative performance.

Next, we invented her personality, making her an attractive, single, 24-year-old UC San Diego math major who surfs and skis. Using the chatbox, we explained that she became really good at checkers by teaching herself how to play during the six months it took her to recover from breaking her leg in a skiing accident.

This information comprised a version of the truth because the program had in fact run for nearly six months on a Pentium II 400-MHz computer as it completed the 840 generations of its evolution. The players on the Internet site greeted this new personality much more favorably. My book contains many detailed stories of these interactions.

Currently, Blondie24 generates moves in the software product Evolutionary Checkers Starring Blondie24, published by Digenetics. Figure 2 provides a screenshot of the game interface, which offers simultaneous two- and three-dimensional views of the board in conjunction with live-motion video of an opponent framed by a realistic background. This background can be animated and the video segments

Many years ago, at an early stage of research into artificial intelligence, such an achievement appeared impossible. The famous computer science pioneer Allen Newell once said, “It is extremely doubtful whether there is enough information in ‘win, lose, or draw’ when referred to the whole play of the game [such as checkers] to permit any learning at all over available time scales” (Marvin Minsky, “Steps Toward Artificial Intelligence,” *Computers and Thought*, McGraw-Hill, 1963).

We present an even higher hurdle here because we provided no specific win, loss, or draw information—only a final point tally to each neural network over a series of games. Nor were the neural networks afforded the luxury of any human expertise beyond the position, number, and type of pieces on the board. The neural networks had to invent any other features they would use on their own, through the process of evolution.

The evolutionary technology we have developed can be used to control characters in intelligent board games and other software entertainment products. Efforts are under way to incorporate self-learning techniques in other games, such as chess, and may provide superior methods for designing continually novel games in the future.

David B. Fogel is the CEO of Natural Selection Inc. Contact him at dfogel@natural-selection.com.

---

blended together, triggered by actions on the board or even player inaction. The effect can be quite striking, and some people have asked, after playing a few moves, “Where is she?”

... and daunting expertise

After completing 165 games, each played by manually entering opponent moves into the neural network and again entering moves suggested by the neural network to the human opponent on the Internet, Blondie24’s rating settled in the expert category at 2,045. This placed the program in the top 500 of 120,000 players on www.zone.com. In a subsequent 10-game match, Blondie24 played against the novice version of Chinook, the world-champion checkers program developed by the University of Alberta team led by Jonathan Schaeffer. This match generated evidence corroborating her www.zone.com rating (D.B. Fogel and Kumar Chellapilla, “Verifying Anaconda’s Expert Rating...