Project NERO: An Experiment in Technology Transfer

Digital Media Collaboratory / Innovation Creativity and Capital Institute
Neuroevolution Group / Department of Computer Sciences
The University of Texas at Austin

Background

In August 2003 the *Digital Media Collaboratory* (DMC) of the *Innovation Creativity and Capital Institute* (IC^2) at the University of Texas held its second annual *GameDev* conference. The focus of the 2003 conference was artificial intelligence, and consequently the DMC invited several Ph.D. students from the Neuroevolution Group at the University's *Department of Computer Sciences* (UTCS) to make presentations on state-of-the-art academic AI research with potential game applications.

The *GameDev* conference also held break-out sessions where groups brainstormed ideas for innovative games, and in one of the sessions Ken Stanley proposed an idea for a game based on a real-time variant of his previously published *NEAT* learning algorithm.

On the basis of Ken's proposal the DMC/IC^2 resolved to staff and fund a project to create a professional-quality demo of the game. (See production credits at end.) The resulting *NERO* project started in October 2003 and has continued through the present, generating several spin-off research projects in its wake. (See bibliography at end of poster.)

As a result of the project we have imported the latest in AI research from the UTCS Neuroevolution Group into a commercial game engine, providing the DMC with a case study in technology transfer and a polished demo of an entertaining game.

The *NERO* Game

Our experimental game is a real-time strategy game called *NERO*, Neuro-Evolving Robotic Operatives. It is set in a fictional post-apocalyptic world, where robots struggle over the relics of human civilization.

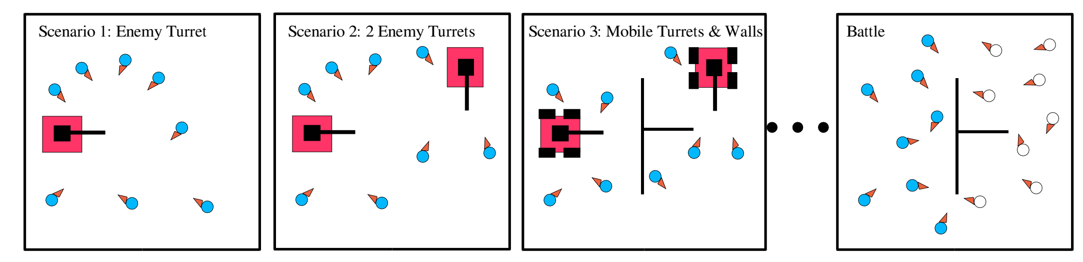
Unlike most RTS games, NERO consists of two distinct phases of play. In the first phase individual players deploy robots in a 'sandbox' and train them to the desired tactical doctrine. Once a collection of robots has been trained, a second phase of play allows players to pit their robots in a battle against robots trained by some other player, to see how well their training regimens prepared their robots for battle. The training phase is the most innovative aspect of game play in NERO, and is also the most interesting from the perspective of AI research. (See screenshot below.)

Screenshot in training mode. Here bipedal *Enforcer*-style NEROs are being trained in our *Mountain Pass* arena. When a brain is installed in an *Enforcer* for training, that robot enters play at a drop point near the crowded area at the right, and the brain is scored on how well the robot performs before the brain's timer expires.

These *Enforcers* have already been trained to approach enemies; several can be seen crowded around the target labeled *E1* to the left, and many more are on their way. Now the player is using the reward controls (lower right) to set up for additional training that will keep this team from approaching the enemy too closely. Many other options are also available on the control panel. (A similar object-placement control panel at the lower left has been retracted to improve visibility.)

The *NERO* Game (continued)

Training for complex tactical behaviors will require a player to think out and implement a shaping plan, leading the robots through a series of sandbox scenarios that guide them stepwise to the desired battlefield doctrine. (See figure below.)



Example training regimen for a team of robots. The player starts the naive robots in a simple environment and progressively adds more powerful enemies and more complex obstacles to the environment until the robots are experienced veterans, ready to be deployed against another player's robotic army.

Real-Time Neuroevolution with *rtNEAT*

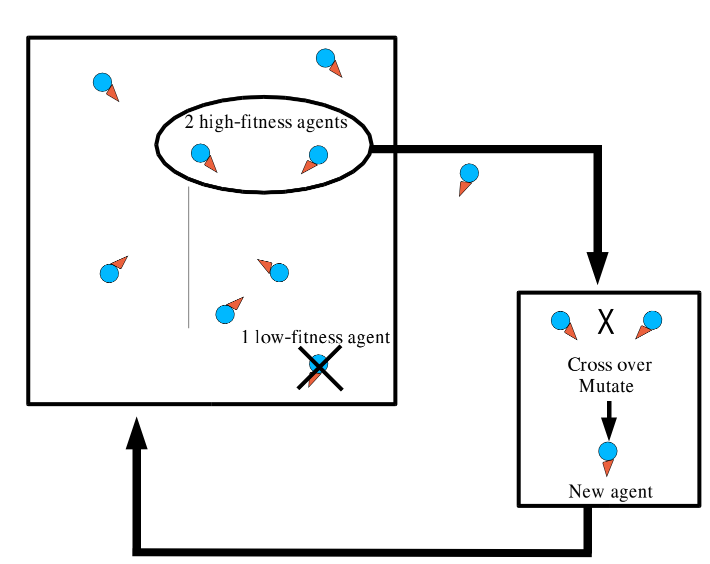
The robots in NERO use artificial neural networks for their "brains", and they learn by means of *neuroevolution*. Neuroevolution is a *genetic algorithm*, a type of *reinforcement learning algorithm* that operates by rewarding the agents in a population that perform the best and punishing those that perform the worst.

In NERO the rewards and punishments are specified by the players, by means of the slider controls shown in the screenshot to the left. (More will be added as development progresses.) The genetic algorithm decides which robots' "brains" are the most and least fit on the basis of the robots' behavior and the current settings of the sliders.

For the NERO project we are using a specific neuroevolutionary algorithm called *NEAT*, *Neuro-Evolution of Augmenting Topologies*. Unlike most neuroevolutionary algorithms, NEAT starts with an artificial neural network of minimal connectivity and adds complexity only when it helps solve a problem. This helps ensure that the algorithm does not produce unnecessarily complex solutions.

In NERO we are introducing a new *real-time* variant of NEAT, called rtNEAT, in which a small population evolves while you watch. (Most genetic algorithms use generation-based off-line processing, and only provide a result at the end of some pre-specified amount of training.)

rtNEAT solves several technical challenges. For example, in order to allow continual adaptation, rtNEAT discards the traditional notion of generations for the genetic algorithm, and instead keeps a small population that is evaluated continuously, with regular replacement of the poorest performers. rtNEAT is powerful enough that we are able to work with a population as small as 50 even for non-trivial learning tasks. This allows the entire population to be replaced quickly enough for human viewers to see the population's behavior adapt while they watch.



Real-time neuroevolution. We break up the regimented schedule of generation-based evolution by imposing an artificial lifetime on each "brain" being evaluated. In this example the brains control individual agents in a simulation. Whenever a brain's evaluation lifetime is up, we compare its fitness (based on the agent's performance while "wearing" that brain) to the fitness of other brains in the population. If it is among the least fit brains it is immediately discarded and replaced by breeding two high-fitness brains together using NEAT. Otherwise the brain is put on the shelf, ready to be inserted into another robot for further evaluation when its turn comes up again.

Current status

To date we have worked through several layers of experiments to validate the feasibility of rtNEAT and the NERO game concept, and to iteratively enhance the learning power and resulting intelligence of the NERO agents. We are currently polishing up the game for a public release, and after that we will start a new layer of AI experiments to enhance the power of rtNEAT and the NERO agents' capabilities, and pursue other research projects based on the NERO game.

Screenshot in battle mode (detail). Two trained teams are engaged in our *Orchard* arena, where a long wall down the center separates the two teams' starting positions but leaves room to go around either end. The Red Team (right) was trained to face the enemy and fire from a safe distance, and the Blue Team (left) was trained to chase down enemies while navigating obstacles such as the wall. Here the Blue Team has formed into single file while negotiating the turn at the end of the wall, and now assaults the Red Team, which has assumed a defensive posture. The smoke is color-coded to indicate which team is firing; the image shows that the Red Team's training has given it the advantage of a concentration of fire in this situation.

Project NERO Production Staff (Spring 2005)

rtNEAT & Game Concept .... Ken Stanley, UTCS
Producer .... Aliza Gold, DMC
AI Leads .... Ken Stanley & Bobby Bryant, UTCS
Design Lead .... Matt Patterson, DMC
Programming Lead .... Michael Chrien, UTCS

Plus a cadre of volunteers, mostly undergraduate programmers from the UT-Austin Department of Computer Sciences and 3D artists from Austin Community College.

For more information visit `nerogame.org`, or write the NERO contacts list at `research@nerogame.org`.

Acknowledgements

Production of the NERO project is funded by the Digital Media Collaboratory and the IC^2 Institute of the University of Texas at Austin. The original NEAT research was supported in part by the National Science Foundation and the Texas Higher Education Coordinating Board. NERO is built on the *Torque* game engine, licensed from GarageGames, Inc.

References

D'Silva, T., Janik, R., Chrien, M., Stanley, K., and Miikkulainen, R. (2005). Retaining learned behavior during real-time neuroevolution. *Proceedings of the Artificial Intelligence and Interactive Digital Entertainment Conference.*

Gold, A. (2005). Academic AI and video games: A case study of incorporating innovative academic research into a video game prototype. *Proceedings of the IEEE Symposium on Computational Intelligence and Games.*

Stanley, K. O., Bryant, B. D., and Miikkulainen, R. (2005a). Evolving neural network agents in the NERO video game. *Proceedings of the IEEE Symposium on Computational Intelligence and Games.*

Stanley, K. O., Bryant, B. D., and Miikkulainen, R. (2005b). Real-time neuroevolution in the NERO video game. *IEEE Transactions on Evolutionary Computation Special Issue on Evolutionary Computation and Games.* In press.

Stanley, K. O., and Miikkulainen, R. (2002). Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10(2):99–127.

Updated May 19, 2005