



MICK WEST

» THE INNER PRODUCT

INTELLIGENT MISTAKES

How to incorporate stupidity into your AI code without actually looking stupid

TWENTY YEARS AGO, I WAS WORKING ON my first commercial game: STEVE DAVIS WORLD SNOOKER, one of the first snooker/pool games to have an AI opponent. The AI I created was very simple. The computer just picked the highest value ball that could be potted, and then potted it. Since it knew the precise positions of all the balls, it was very easy for it to pot the ball every time. This was fine for the highest level of difficulty, but for easy mode I simply gave the AI a random angular deviation to the shot.

Toward the end of the project, we got some feedback from the client that the AI was "too good." I was puzzled by this and assumed the person wanted the expert mode to be slightly less accurate. So I changed that. But then I heard complaints about the decreased accuracy, and again that the AI was still too good.

Eventually the clients paid a visit to our offices and tried to demonstrate in person what they meant. It gradually came out that they thought the problem was actually with the "easy" mode. They liked that the computer missed a lot of shots, but they thought that the positional play was too good. The computer always seemed to be leaving the white ball in a

convenient position after its shot, either playing for safety or lining up another ball. They wanted that changed.

The problem was, there was no positional play! The eventual position of the white ball was actually completely random. The AI only calculated where the cue ball should hit the object ball in order to make that object ball go into a pocket. It then blindly shot the cue ball toward that point with a speed proportional to the distance needed to travel, scaled by

“AI needs to be more intelligent in order to appear less intelligent.”

the angle, plus some fudge factor. Where the white ball went afterward was never calculated, and it quite often ended up in a pocket.

So why was it a problem? Why did they think the AI was "too good" when it was actually random?

Humans have a tendency to anthropomorphize AI opponents. We think the computer is going through a thought process just like a human would do in a similar situation. When we see the ball end up in an advantageous position, we think the computer must have intended that to happen.

The effect is magnified here by the computer's ability to pot a ball from any position, so for the computer, all positions are equally advantageous. Hence, it can pot ball after ball, without having to worry about positional play. Because sinking a ball on every single shot would be impossible for a human, the player assumes that the computer is using positional play.

DESIGN OR CODE?

Is this a design problem or a code problem? To a certain extent it depends on the type of game, and to what extent the AI-controlled opponents are intended to directly represent a human in the same situation as the player.

In a head-to-head game such as pool, chess, or poker, the AI decisions are very much determined at a pure code level. In a one-versus-many game, such as an FPS, there is some expectation that your

opponents are generally weaker than you are. After all, you are generally placed in a situation of being one person against countless hordes of bad guys. Other game genres, particularly

racing games, pit you against a field of equal opponents. Here the expectation of realistic AI is somewhere between that of chess and the FPS examples.

The more the computer AI has to mimic the idiosyncrasies of a human player, the more the task falls to the programmer. The vast majority of the AI work in a chess game is handled by programmers. Game designers would focus more on the presentation. In an FPS, the underlying code is generally vastly simpler than chess AI. There is path finding, some state transitions, some goals, and some basic behaviors. The majority of the behavioral content is supplied via the game designers, generally with some form of scripting. The designers will also be responsible for coding in actions, goals, and responses that emulate the idiosyncrasies of human behavior.

HEADS UP!

In some heads-up games, such as chess and pool, the computer has a huge advantage over the player. Modern

MICK WEST was a co-founder of Neversoft Entertainment. He's been in the game industry for 19 years and currently works as a technical consultant. Email him at mwest@gdmag.com.

chess programs such as FRITZ are vastly stronger than nearly all human players. In pool and snooker games, the computer can be programmed to never miss a shot. However, people want to play against an opponent that is well matched to their skills, and so there are generally levels of AI in the game that the player can choose from.

The simplest way to introduce stupidity into AI is to reduce the amount of computation that it's allowed to perform. Chess AI generally performs billions of calculations when deciding what move to make. The more calculations that are made (and the more time taken), then (generally) the better the computer will play. If you reduce the amount of calculations performed, the computer will be a worse player.

The problem with this approach is that it decreases the realism of the AI player. When you reduce the amount of computation, the AI will begin to make incredibly stupid mistakes—mistakes that are so stupid, no human would ever make them. The artificial nature of the game will then become apparent, which destroys the illusion of playing against a real opponent.

Remember what we are trying to accomplish: We want people to have an enjoyable experience. No matter what the game, we want the players to feel challenged so that when they win, they feel a sense of accomplishment. We want them to feel that they were playing against an opponent who was really trying to beat them.

By reducing the amount of computation, we create an AI opponent that is trying to win, but has been crippled in a way that leads to unrealistic gameplay. But does the player actually care about what is going on under the hood? What if we don't cripple our AI, but instead let it play at full strength, but have the AI deliberately throw the game?

THROWING THE GAME

In sports, "throwing the game" means one side makes a series of intentional

mistakes that look natural, but result in losing the game. This behavior is rightly vilified by players and fans, as the agreement is that there be a contest between two equal opponents, or at least, two opponents who are trying equally hard to win.

But in computer games, it's impossible to have an equal match. It's humans versus machines. One side has an advantage of being able to perform a billion calculations per second, and the other has the massively parallel human brain. Any parity here is an illusion, and it's that illusion that we seek to improve and maintain via the introduction of intelligent mistakes and artificial stupidity.

The computer has to throw the game in order to make it fun. When you beat the computer, it's an illusion. The computer let you win. We just want it to let you win in a way that feels good.

AI programmers need to get used to this idea. We are manipulating the game, creating artificial stupidity, fake stupidity. But we are not predetermining the outcome of the game. We don't set our AI with the intent to lose the game, but rather to give the human player a reasonable chance of winning. If the human plays poorly, the AI will still win, but the player will at least feel like she came close to beating a strong opponent, and thus feel like playing one more game.

HIDDEN HANDICAPPING

Computer chess expert Steven Lopez (see Resources) describes how in human versus human chess, it's acceptable for a high-ranking player to give a much lower ranking player an advantage at the start of the game by removing some of his pieces from the board before the game begins. When the game starts, the master player and the novice player are still playing to the height of their abilities, and yet the game is more evenly balanced. The master player does not have to play "stupid" in order to give the novice player a chance.

However, humans playing against a computer do not like to be given an

advantage in this way, and prefer to play the full board against an AI opponent of approximately their skill level.

The programmers of FRITZ hit upon a solution that involved the AI deliberately setting up situations that the human player could exploit (with some thought) that would allow the human to gain a



FRITZ 11 from ChessBase.

positional or piece advantage. Once the human player gained the advantage, the AI would resume trying to win.

At no point here is the AI actually dumbed down. If anything, there is actually quite a bit more computation going on, and certainly more complexity. The goal of the AI has shifted from "win the game" to "act like you are trying to win the game, but allow the human to gain a one-pawn advantage, and then try to win." The AI needs to be more intelligent in order to appear less intelligent.

POKER AI

When I programmed the AI for Left Field's WORLD SERIES OF POKER, the AI computation was basically the same for each difficulty level. The computer would calculate the odds of winning based on the known cards, and an estimate of the opponent's hand strength based on betting history. The odds would then be used to calculate a rate of return, which

would be used to decide if they would fold, call, or raise.

There were many special case rules and exceptions, but that's the basics. The AI players would all make the same extensive computations, running tens of thousands of simulated hands through an evaluator to calculate the rate of return. After these calculations were performed, only then would the differentiation be performed. At that point, the best players would play their best move, and the weak AI players would make intelligent mistakes.

For weak poker AI, an intelligent mistake consists of figuring out what you should do, and then not doing it, so long as not doing it does not make you look stupid. For example, if the human player just put in a big raise, yet you know there's a 75 percent chance your hand is the best, then an intelligent mistake would be to fold. The odds are the AI would win, yet we are simulating a weak human player, and weak human players often fold to a large raise when they are unclear on their odds. Conversely, weak human players often call when their chances are weak. It's a natural thing to do and allows us to reduce the strength of the AI player, without it looking artificially stupid.

These intelligent mistakes were implemented in a probabilistic manner. The fake-stupid AI would not always fold when the human player seemed to be bluffing—it was just more likely to. This worked very well in the highly random game of poker, because the player could never tell in any individual situation if the AI was actually making a mistake. Since the AI was still performing its full set of millions of calculations, it never made mistakes that were inhumanly stupid, but the layer of artificial stupidity brought on by increased recklessness was enough to even the playing field

and give the weak and average human players an enjoyable game.

ARTIFICIAL INACCURACY

In pool and in shooters, the computer AI is blessed with an omniscient accuracy. The shooter AI knows down to the billionth of an inch exactly where you are, and could shoot your hat off your head from five miles away. Similarly in pool, the AI knows the position of every ball and can calculate where every ball will end up before it takes a shot.

When I implemented my snooker AI, it could perfectly pot any ball off two cushions, and would almost always get a perfect break of 147 every time it

“To provide an exciting and dynamic game, the AI needs to manipulate the gameplay to create situations that the player can exploit.”

played (except when it potted the white due to its lack of positional play). It was obviously not a fun opponent to play against, so even at the highest levels, the accuracy had to be reduced, and the cushion shots had to be restricted to getting out of snookers.

Simply reducing the accuracy of the AI is not always the best way to improve gameplay. As I found with the “positional play” in snooker, random outcomes that happen to favor the computer are perceived as being intentional. If the ball ends up in a good place, or the poker AI makes a lucky call and wins on the river, it can be perceived as unfair or even cheating.

So instead of reducing the accuracy, I'd suggest, as in chess, we increase the accuracy. In order to provide an exciting and dynamic game, the AI needs to manipulate the gameplay to create situations that the player can exploit. In pool this could mean, instead of blindly taking a shot and not caring

where the cue ball ends up, the AI should deliberately fail to pot the ball and ensure that the cue ball ends up in a place where the player can make a good shot. In a shooter, the enemy aliens should not simply randomly break from cover—they should sometimes break from cover when the player is close to them and panning toward them. They should “accidentally” throw themselves into the line of fire to make the game more interesting.

LUCK OF THE DRAW

Playing against a perfect opponent is no fun. But playing against a crippled opponent is no fun either. To create more interesting gameplay, we have

to introduce the concepts of artificial stupidity and intelligent mistakes. Intelligent mistakes seem like failings on the part of the AI, but are actually carefully calculated ways of throwing the game that make it more entertaining

for the player. This does not remove the challenge, as the player still has to have a certain level of skill.

For the programmer, adding intelligent mistakes is much more complex than simply reducing the accuracy of the AI, but provides a much more rewarding experience for the player. ❖

RESOURCES

Lidén, Lars. “Artificial Stupidity: The Art of Intentional Mistakes,” in *AI Game Programming Wisdom 2*, Charles River Media, 2004. http://lars.liden.cc/Publications/Downloads/2003_AIWisdom.pdf

Lopez, Steven. “Intelligent Mistakes,” Chessbase News, 2005. www.chessbase.com/newsdetail.asp?newsid=2579