# [The Mystery of Go, the Ancient Game That Computers Still Can’t Win](http://www.wired.com/2014/05/the-world-of-computer-go/" \o "Permanent Link to The Mystery of Go, the Ancient Game That Computers Still Can’t Win)

* By Alan Levinovitz
* 05.12.14

[](http://www.wired.com/wp-content/uploads/2014/05/Go-01.jpg)

Remi Coulom (left) and his computer program, Crazy Stone, take on grandmaster Norimoto Yoda in the game of Go. Photo: Takashi Osato/WIRED

TOKYO, JAPAN — Rémi Coulom is sitting in a rolling desk chair, hunched over a battered Macbook laptop, hoping it will do something no machine has ever done.

That may take another ten years or so, but the long push starts here, at Japan’s University of Electro-Communications. The venue is far from glamorous — a dingy conference room with faux-wood paneling and garish fluorescent lights — but there’s still a buzz about the place. Spectators are gathered in front of an old projector screen in the corner, and a ragged camera crew is preparing to broadcast the tournament via online TV, complete with live analysis from two professional commentators.

Coulom is wearing the same turtleneck sweater and delicate rimless glasses he wore at last year’s competition, and he’s seated next to his latest opponent, an ex-pat named Simon Viennot who’s like a younger version of himself — French, shy, and self-effacing. They aren’t looking at each other. They’re focused on the two computers in front of them. Coulom’s is running a piece of software called Crazy Stone — the work of over seven years — and the other runs Nomitan, coded by Viennot and his Japanese partner, Kokolo Ikeda.

Crazy Stone and Nomitan are locked in a game of Go, the Eastern version of chess. On each screen, you can see a Go board — a grid of 19 lines by 19 lines — filling up with black and white playing pieces, each placed at the intersection of two lines. If Crazy Stone can win and advance to the finals, it will earn the right play one of the best human Go players in Japan. No machine has ever beaten a top human Go player — at least not without a huge head-start. Even if it does advance to the man-machine match, Crazy Stone has no chance of changing this, but Coulom wants to see how far his creation has come.

Computers match or surpass top humans in chess, Othello, Scrabble, backgammon, poker, even Jeopardy. But not Go.

The challenge is daunting. In 1994, machines took the checkers crown, when a program called Chinook beat the top human. Then, three years later, they topped the chess world, IBM’s Deep Blue supercomputer besting world champion Garry Kasparov. Now, computers match or surpass top humans in a wide variety of games: Othello, Scrabble, backgammon, poker, even Jeopardy. But not Go. It’s the one classic game where wetware still dominates hardware.

Invented over 2500 years ago in China, Go is a pastime beloved by emperors and generals, intellectuals and child prodigies. Like chess, it’s a deterministic perfect information game — a game where no information is hidden from either player, and there are no built-in elements of chance, such as dice.1 And like chess, it’s a two-person war game. Play begins with an empty board, where players alternate the placement of black and white stones, attempting to surround territory while avoiding capture by the enemy. That may seem simpler than chess, but it’s not. When Deep Blue was busy beating Kasparov, the best Go programs couldn’t even challenge a decent amateur. And despite huge computing advances in the years since — Kasparov would probably lose to your home computer — the automation of expert-level Go remains one of AI’s greatest unsolved riddles.

Rémi Coulom is part of a small community of computer scientists hoping to solve this riddle. Every March, the world’s most dedicated Go programmers gather at the University of Electro-Communications to compete in the UEC Cup, a computer Go tournament that, uniquely, rewards two finalists with matches against a “Go sage,” the equivalent of a chess grandmaster. Organizers dub these machine-versus-man matches the Densei-sen, or “Electric Sage Battle.”

At this year’s UEC Cup, Coulom’s Crazy Stone is the favorite. On the first day of the competition, the software program went undefeated, which earned it top seed in today’s 16-member single-elimination bracket and a bye in the first round. Now, it’s the second round, and Viennot, a relative newcomer to the computer Go scene, tells me he’ll be happy if his program just puts up a good fight. “Nomitan uses many of Rémi’s tricks, but I don’t think it will be enough,” he says. “Crazy Stone is a much stronger program.”

[](http://www.wired.com/wp-content/uploads/2014/05/Go-02.jpg)

Rémi Coulom and Crazy Stone. Photo: Takashi Osato/WIRED

THe computer screens in front of Coulom and Viennot display statistics that show the relative confidence of each program. Although the match has just begun, Crazy Stone is already 58 percent sure it will prevail. Oddly, Nomitan’s confidence level is about the same. When I point this out to Coulom and Viennot, they both laugh. “You can’t trust these algorithms completely,” explains Viennot. “They are always a little over-confident.”

The official commentary doesn’t start until the final match, but as the second round progresses, a small crowd forms around commentator Michael Redmond to hear his thoughts. The charismatic Redmond, an American, is one of very few non-Asian Go celebrities. He began playing professionally in Japan at the age of 18, and remains the only Westerner to ever reach 9-dan, the game’s highest rank. “I don’t know the black player,” he says, referring to Nomitan, “but it has a flashy style, flashier than Crazy Stone. Very good tesuji. With humans, tesuji are a fairly accurate gauge of strength, and now, I’m seeing computers do them more.”

Tesuji means something like “clever play,” and Nomitan’s tesuji are giving Crazy Stone serious trouble. With the game nearly halfway done, Crazy Stone is only 55 percent confident, which means it’s even money. After a few more turns, another professional named O Meien pronounces Nomitan the leader. As other games in the room finish, the crowd in front of the projector screen grows larger and louder. From the sound of it, Crazy Stone’s prospects are increasingly bleak.

Even for Coulom — a good but not great Go player himself — Crazy Stone’s moves can be incomprehensible.

Most people in the room take the pros like O Meien at their word. We have to, since games of Go are often so complex that only extremely high-level players can understand how they’re progressing. Even for Coulom — a good but not great Go player himself — Crazy Stone’s moves can be incomprehensible. But Coulom identifies as a programmer more than a player, which allows him to remain calm in the face of professional skepticism. He trusts the confidence level Crazy Stone shows him. “Maybe O Meien is thinking about which side looks better,” he says, with a lilting French accent. “But I know Crazy Stone is much stronger than Nomitan. So I just think at some point Nomitan will probably mess up.”

And so it does. Crazy Stone makes a number of moves that prompt murmurs of approval from the crowd. Despite those initial tesuji, Nomitan squanders its advantage. Soon, Crazy Stone’s confidence levels are in the high 80s, and Nomitan resigns.

The other matches leading up to the final are uneventful, with the exception of one semi-final contest. Zen, Crazy Stone’s biggest rival and last year’s runner-up, nearly loses to a program called Aya. The game begins with a complicated local battle in the upper right corner, each side trying to keep their stones alive. At first, Zen plays with excellent kiai, or fighting spirit. The area looks settled. Then, without warning, Zen makes an obvious mistake, eliciting a collective gasp from the room. Zen’s co-programmer, a Japanese man with long graying hair named Hideki Kato, keeps his eyes on the confidence levels streaming across his laptop screen, and eventually, Zen manages to eke out a lead, before Aya resigns. The final is decided, a rematch of last year’s match: Crazy Stone vs. Zen.

### The Mystery of Go

Even in the West, Go has long been a favorite game of mathematicians, physicists, and computer scientists. Einstein played Go during his time at Princeton, as did mathematician John Nash. Seminal computer scientist Alan Turing was a Go aficionado, and while working as a World War II code-breaker, he introduced the game to fellow cryptologist I.J. Good. Now known for contributing the idea of an “intelligence exposition” to [singularity theories](http://en.wikipedia.org/wiki/Technological_singularity) — predictions of how machines will become smarter than people — Good gave the game a huge boost in Europe with a 1965 article for New Scientist entitled “The Mystery of Go.”

[](http://www.wired.com/wp-content/uploads/2014/05/800px-Korean_Game_from_the_Carpenter_Collection_ca._1910-1920.jpg)

A woman and a man play Go in Korea sometime in the early 1900s. Photo: [*Library of Congress*](http://www.loc.gov/pictures/item/2001705592/)

Good opens the article by suggesting that Go is inherently superior to all other strategy games, an opinion shared by pretty much every Go player I’ve met. “There is chess in the western world, but Go is incomparably more subtle and intellectual,” says South Korean Lee Sedol, perhaps the greatest living Go player and one of a handful who make over seven figures a year in prize money. Subtlety, of course, is subjective. But the fact is that of all the world’s deterministic perfect information games — tic-tac-toe, chess, checkers, Othello, xiangqi, shogi — Go is the only one in which computers don’t stand a chance against humans.

‘There is chess in the western world, but Go is incomparably more subtle and intellectual.’

This is not for lack of trying on the part of programmers, who have worked on Go alongside chess for the last fifty years, with substantially less success. The first chess programs were written in the early fifties, one by Turing himself. By the 1970s, they were quite good. But as late as 1962, despite the game’s popularity among programmers, only two people had succeeded at publishing Go programs, neither of which was implemented or tested against humans.

Finally, in 1968, computer game theory genius Alfred Zobrist authored the first Go program capable of beating an absolute beginner. It was a promising first step, but notwithstanding enormous amounts of time, effort, brilliance, and quantum leaps in processing power, programs remained incapable of beating accomplished amateurs for the next four decades.

To understand this, think about Go in relation to chess. At the beginning of a chess game, White has twenty possible moves. After that, Black also has twenty possible moves. Once both sides have played, there are 400 possible board positions. Go, by contrast, begins with an empty board, where Black has 361 possible opening moves, one at every intersection of the 19 by 19 grid. White can follow with 360 moves. That makes for 129,960 possible board positions after just the first round of moves.

The rate at which possible positions increase is directly related to a game’s “branching factor,” or the average number of moves available on any given turn. Chess’s branching factor is 35. Go’s is 250. Games with high branching factors make classic search algorithms like minimax extremely costly. Minimax creates a search tree that evaluates possible moves by simulating all possible games that might follow, and then it chooses the move that minimizes the opponent’s best-case scenario. Improvements on the algorithm — such as alpha-beta search and null-move — can prune the chess game tree, identifying which moves deserve more attention and facilitating faster and deeper searches. But what works for chess — and checkers and Othello — does not work for Go.

‘I’ll see a move and be sure it’s the right one, but won’t be able to tell you exactly how I know. I just see it.’

The trouble is that identifying Go moves that deserve attention is often a mysterious process. “You’ll be looking at the board and just know,” Redmond told me, as we stood in front of the projector screen watching Crazy Stone take back Nomitan’s initial lead. “It’s something subconscious, that you train through years and years of playing. I’ll see a move and be sure it’s the right one, but won’t be able to tell you exactly how I know. I just see it.”

Similarly inscrutable is the process of evaluating a particular board configuration. In chess, there are some obvious rules. If, ten moves down the line, one side is missing a knight and the other isn’t, generally it’s clear who’s ahead. Not so in Go, where there’s no easy way to prove why Black’s moyo is large but vulnerable, and White has bad aji. Such things may be obvious to an expert player, but without a good way to quantify them, they will be invisible to computers. And if there’s no good way to evaluate intermediate game positions, an alpha-beta algorithm that engages in global board searches has no way of deciding which move leads to the best outcome.

Not that it matters: Go’s impossibly high branching factor and state space (the number of possible board configurations) render full-board alpha-beta searches all but useless, even after implementing clever refinements. Factor in the average length of a game — chess is around 40 turns, Go is 200 — and computer Go starts to look like a fool’s errand.

[](http://www.wired.com/wp-content/uploads/2014/05/Go-04.jpg)

A traditional Go gameboard. Photo: Takashi Osato/WIRED

### In Search of the Mental Leap

Nonetheless, after Zobrist, Go programmers persisted in their efforts and managed to make incremental progress. But it wasn’t until 1979 that a five-year project by computer scientist Bruce Wilcox produced a program capable of beating low-level amateurs. As a graduate student at the University of Michigan, Wilcox and his advisor collected detailed protocols from games played against James Kerwin, who soon after would leave for Japan to become the second-ever Western professional Go player.

Unlike successful chess programmers, Wilcox focused almost entirely on modeling expert intelligence, collecting a vast database of stone relationships from Kerwin’s games. His program divided the board into smaller, more manageable zones, and then used the database to generate possible moves, applying a hierarchal function to choose the best among them. Forward-looking searches like alpha-beta, long the cornerstone of AI gaming, were entirely absent from the program’s first incarnation

Then, somewhat abruptly, progress stalled. The programs had encountered an obstacle that also gives human players trouble.

During the development process, Wilcox became a very strong amateur player, an indispensable asset for early Go programmers, given that programs depended so much on a nuanced understanding of the game. Mark Boon (Goliath), David Fotland (Many Faces of Go), Chen Zhixing (Handtalk and Goemate) — the winners of computer Go competitions throughout the 80s and 90s — were all excellent players, and it was their combined prowess as players and programmers that facilitated steady improvements through the 90s. Then, somewhat abruptly, progress stalled. The programs had encountered an obstacle that also gives human players trouble.

“A lot of people peak out at a certain level of amateur and never get any stronger,” David Fotland explains. Fotland, an early computer Go innovator, also worked as chief engineer of Hewlett Packard’s PA-RISC processor in the 70s, and tested the system with his Go program. “There’s some kind of mental leap that has to happen to get you past that block, and the programs ran into the same issue. The issue is being able to look at the whole board, not the just the local fights.”

Fotland and others tried to figure out how to modify their programs to integrate full-board searches. They met with some limited success, but by 2004, progress stalled again, and available options seemed exhausted. Increased processing power was moot. To run searches even one move deeper would require an impossibly fast machine. The most difficult game looked as if it couldn’t be won.

Enter Rémi Coulom, whose Crazy Stone would inaugurate a new era of computer Go. Coulom’s father was a programmer, and in 1983, he gave his son a Videopac computer for Christmas. Coulom was nine, around the time most Go prodigies leave home to begin intensive study at an academy. After less than a year, he had programmed Mastermind. In four years, he had created an AI that could play Connect Four. Othello followed shortly thereafter, and by 18, Coulom had written his first chess program.

Enter Rémi Coulom, whose Crazy Stone would inaugurate a new era of computer Go.

The program, Crazy Bishop, was awful. Without access to the internet, Coulom had to invent everything from scratch. But a year later, he started engineering school, where university computers allowed him to swap algorithms and strategies in online chess programming communities. Crazy Bishop improved quickly. In 1997, the year Deep Blue defeated Kasparov, Coulom attended the world computer chess championship in Paris, where he made a decent showing and met members of his online community in person. The event inspired him to continue graduate study as a programmer, not an engineer. Following a stint in the military and a masters in cognitive science, Coulom earned a PhD for work on how neural networks and reinforcement learning can be used to train simulated robots to swim.

Although he’d encountered Go at the 2002 Computer Olympiad, Coulom didn’t give it much thought until 2005, when, after landing a job at the University of Lille 3, he began advising Guillaume Chaslot, a masters student who wanted to write a computer Go program as his thesis. Chaslot soon left to start his PhD, but Coulom was hooked, and Go became a full-time obsession.

### The Monte Carlo Bet

It wasn’t long before he made his breakthrough. Coulom had exchanged ideas with a fellow academic named Bruno Bouzy, who believed that the secret to computer Go might lie in a search algorithm known as Monte Carlo. Developed in 1950 to model nuclear explosions, Monte Carlo replaces an exhaustive search with a statistical sampling of fewer possibilities. The approach made sense for Go. Rather than having to search every branch of the game tree, Monte Carlo would play out a series of random games from each possible move, and then deduce the value of the move from an analysis of the results.

Rather than having to search every branch of the game tree, Monte Carlo would play out a series of random games from each possible move.

Bouzy couldn’t make it work. But Coulom hit upon a novel way of combining the virtues of tree search with the efficiency of Monte Carlo. He christened the new algorithm Monte Carlo Tree Search, or MCTS, and in January of 2006, Crazy Stone won its first tournament. After he published his findings, other programmers quickly integrated MCTS into their Go programs, and for the next two years, Coulom vied for dominance with another French program, Mogo, that ran a refined version of the algorithm.

Although Crazy Stone ended up winning the UEC Cup in 2007 and 2008, Mogo’s team used man-machine matches to win the publicity war. Coulom felt the lack of attention acutely. When neither the public nor his university gave him the recognition he deserved, he lost motivation and stopped working on Go for nearly two years.

Coulom might have given up forever had it not been for a 2010 email from Ikeda Osamu, the CEO of Unbalance, a Japanese computer game company. Ikeda wanted to know if he’d be willing to license Crazy Stone. Unbalance controlled about a third of the million-dollar global market in computer Go, but Zen’s commercial version had begun to increase its market share. Ikeda needed Coulom to give his company’s software a boost.

The first commercial version of Crazy Stone hit the market in spring of 2011. In March of 2013, Coulom’s creation returned to the UEC Cup, beating Zen in the finals and — given a four-stone head-start — winning the first Densei-sen against Japanese professional Yoshio “The Computer” Ishida. The victories were huge for Coulom, both emotionally and financially. You can see their significance in the gift shop of the Japan Go Association, where a newspaper clipping, taped to the wall behind display copies of Crazy Stone, shows the pro grimly succumbing to Coulom’s creation.

[](http://www.wired.com/wp-content/uploads/2014/05/Go-03.jpg)

Photo: Takashi Osato/WIRED

### Extremely Human

During the break before this year’s UEC final, the TV crew springs into action, setting up cameras and adjusting boom mikes. Redmond, microphone in hand, positions himself at the front of the room next to the magnetic board. On the other side is Narumi Osawa, a pixieish 4-dan professional who, in standard Japanese fashion, will act as an obsequious female foil — “What was that? Oh, wow, I see! Hai! Hai!” — for Redmond’s in-game analysis.

Once everything is in place, Kato and Coulom are called to the front of the room for nigiri, to determine who plays first. Since he is the favorite, Coulom reaches into one of two polished wooden goke and grabs a fistful of white stones. Kato places one black stone on the board, indicating his guess that Coulom holds an odd number of stones. The white stones are counted. Kato guessed correctly. He will be Black, and the game is underway.

The move is utterly bizarre, and even Kato is somewhat baffled.

It takes only three turns before the room explodes with excitement. After claiming two star points in the corners — a standard opening — Zen has placed its third stone right near the center of the board. The move is utterly bizarre, and even Kato is somewhat baffled. “An inhuman decision,” Viennot whispers to me. “But Zen likes to make moyo in the middle of the board, like Takemiya. Maybe this is a new style.”

Kato and Coulom are sitting next to each other, eyes fixed on their laptops, occasionally exchanging confidence levels. An interesting struggle develops in the upper left corner, where Crazy Stone has invaded and Zen is trying to strengthen its position. The crowd mutters when Redmond pronounces one of Zen’s moves “extremely human.” (“Hai! Hai!”) Black and white stones continue to fill the board, beautiful as always, forming what is technically known as a percolated fractal.

Suddenly, Coulom tenses up. Crazy Stone’s confidence levels are rising quickly, too quickly, and soon, they are far too high, up in the sixties. It appears the program has misjudged a semeai, or capturing race, and believes a group of stones in the upper right corner is safe, when in fact it is not. Since Crazy Stone’s move choices depend on an accurate assessment of the overall board position, the misjudged group proves fatal. On its 186th move, Crazy Stone resigns, and Zen becomes the new UEC Cup champion.

Later that evening, at the celebratory banquet, Coulom says he doesn’t feel too bad, but I suspect he’s extremely disappointed. Still, there’s a chance for redemption. As a finalist, Crazy Stone gets to compete in the Densei-sen.

### The Electric Sage Battle

Coulom plays down the Electric Sage Battle. “The real competition is program against program,” he told me during one early phone interview. “When my opponent is a programmer, we are doing the same thing. We can talk to each other. But when I play against a professional and he explains the moves to me, it is too high level. I can’t understand, and he can’t understand what I am doing. The Densei-sen — it is good for publicity. I am not so interested in that.”

But when we meet at the Densei-sen, he seems excited. The building is humming with activity. Last weekend’s conference room is reserved for press and university dignitaries, and a new, private room has been equipped for the matches. Only the referee and timekeepers will be allowed in the room, and cameras have been set up to capture the action for the rest of us. The professional commentators are now in the building’s main auditorium, where at least a hundred people and three TV crews are ready to watch Crazy Stone and Zen take on a real pro.

In 2013, the Electric Sage Battle starred Ishida “The Computer” Yoshio, so-called because of his extraordinary counting and endgame abilities. This year, the pro is Norimoto Yoda, known for leading the Japanese team to a historic victory over Korea in the 2006 Nongshim Cup, and for shattering Go stones when he slams them down on the hardwood goban. After an introductory ceremony, Coulom and Yoda enter the private room, bow, and take their seats. In his typical style, Yoda has come dressed in an olive green kimono. His left hand holds a folded fan. Coulom, in his typical style, is wearing a blue turtleneck sweater. On the wooden goban between them sit two gokes filled with stones — Black for Coulom, White for Yoda.

In his typical style, Yoda has come dressed in an olive green kimono. His left hand holds a folded fan.

This time, there is no nigiri. Crazy Stone receives a massive handicap, starting with four black stones placed advantageously on the corner star points (the 4 by 4 intersections on a Go board’s 19 by 19 grid). Yoda has no choice but to adopt an aggressive style of play, invading Crazy Stone’s territory in hopes of neutralizing his initial disadvantage. But Crazy Stone responds skillfully to every threat, and Yoda’s squarish face starts to harden. The fan snaps open and shut, open and shut.

In the press room, we can’t hear the auditorium commentary. Instead, I watch as Muramatsu Murakasu, a main organizer of the event, plays the game out on his own board with O Meien. The two take turns trying to predict where Yoda and Crazy Stone will move next, and as the game progresses, both agree that Crazy Stone is doing an excellent job maintaining its lead.

Meanwhile, Coulom is looking at the board, his laptop, the timekeepers, anywhere but the increasingly frustrated Yoda. After Coulom places one particular stone, Yoda’s eyes narrow perceptibly. He grunts and fans himself furiously. “That was an excellent move,” says O Meien. “Yoda-san must be upset.”

Crazy Stone continues to play brilliant Go, and all of Yoda’s incursions prove fruitless. It is only as the end approaches that Crazy Stone reveals its true identity. With a lead of eleven points, any decent human in Crazy Stone’s position would play a few obvious moves and then pass, allowing Yoda resign. But Crazy Stone’s algorithm is structured to care only about winning — not by how much. Coulom winces as Crazy Stone makes a wasted move in its own territory, and then another. The game drags on as Crazy Stone sacrifices points, until mercifully it decides to pass, and the machine is finally declared the winner.

Coulom leaves the fuming Yoda as quickly as possible and joins us in the press room. He’s both ecstatic and mortified. “I am proud of Crazy Stone,” he says. “Very proud. But the first thing I will do at home is work on the endgame, so it does not make such embarrassing moves.” Then things get better. Yoda manages to beat Zen in the second Densei-sen match, and just like that, the glory of the Electric Sage Battle belongs to Coulom, whose program has now bested two professionals after a four-stone handicap.

[](http://www.wired.com/wp-content/uploads/2014/05/Go-06.jpg)

Photo: Takashi Osato/WIRED

### When AI Is Not AI

After the match, I ask Coulom when a machine will win without a handicap. “I think maybe ten years,” he says. “But I do not like to make predictions.” His caveat is a wise one. In 2007, Deep Blue’s chief engineer, Feng-Hsiung Hsu, said much the same thing. Hsu also favored alpha-beta search over Monte Carlo techniques in Go programs, speculating that the latter “won’t play a significant role in creating a machine that can top the best human players.”

Even with Monte Carlo, another ten years may prove too optimistic. And while programmers are virtually unanimous in saying computers will eventually top the humans, many in the Go community are skeptical. “The question of whether they’ll get there is an open one,” says Will Lockhart, director of the Go documentary The Surrounding Game. “Those who are familiar with just how strong professionals really are, they’re not so sure.”

According to University of Sydney cognitive scientist and complex systems theorist Michael Harré, professional Go players behave in ways that are incredibly hard to predict. In a recent study, Harré analyzed Go players of various strengths, focusing on the predictability of their moves given a specific local configuration of stones. “The result was totally unexpected,” he says. “Moves became steadily more predictable until players reached near-professional level. But at that point, moves started getting less predictable, and we don’t know why. Our best guess is that information from the rest of the board started influencing decision-making in a unique way.”

This could mean that computer programs will eventually hit another wall. It may turn out that the lack of progress experienced by Go programs in the last year is evidence of yet another qualitative division, the same one that divides amateurs from professionals. Should that be the case, another breakthrough on the level of the Monte Carlo Tree Search could be necessary before programs can challenge pros.

I was surprised to hear from programmers that the eventual success of these programs will have little to do with increased processing power. It is still the case that a Go program’s performance depends almost entirely on the quality of its code. Processing power helps some, but it can only get you so far. Indeed, the UEC lets competitors use any kind of system, and although some opt for 2048-processor-core super-computers, Crazy Stone and Zen work their magic on commercially available 64-core hardware.

Even more surprising was that no programmers think of their creations as “intelligent.” “The game of Go is spectacularly challenging,” says Coulom, “but there is nothing to do with making a human intelligence.” In other words, Watson and Crazy Stone are not beings. They are solutions to specific problems. That’s why its inaccurate to say that IBM Watson will be used to fight cancer, unless playing Jeopardy helps reduce tumors. Developing Watson might have led to insights that help create an artificial diagnostician, but that diagnostician isn’t Watson, just as MCTS programs used in hospital planning are not Crazy Stone.

The public relations folks at IBM paint a different picture, and so does the press. Anthropomorphized algorithms make for a better story. Deep Blue and Watson can be pitted against humans in highly produced man-machine battles, and IBM becomes the gatekeeper of a new era in artificial intelligence. Caught between atheism and a crippling fear of death, Ray Kurzweil and other futurists feed this mischaracterization by trumpeting the impending technological apotheosis of humanity, their breathless idiocy echoing through popular media. “The Brain’s Last Stand,” read the cover of Newsweek after Kasparov’s defeat. But in truth, these machines are nowhere close to mimicking the brain, and their creators admit as much.

Many Go players see the game as the final bastion of human dominance over computers. This view, which tacitly accepts the existence of a battle of intellects between humans and machines, is deeply misguided. In fact, computers can’t “win” at anything, not until they can experience real joy in victory and sadness in defeat, a programming challenge that makes Go look like tic-tac-toe. Computer Go matches aren’t the brain’s last stand. Rather, they help show just how far machines have to go before achieving something akin to true human intelligence. Until that day comes, perhaps it’s best to view the Densei-sen as programmers do. “It is fun for me,” says Coulom, “but that’s all.”

Alan Levinovitz is assistant professor of philosophy and religion at James Madison University. His writing has appeared in Slate, Salon, The Believer, and elsewhere. Follow him: @top\_philosopher.

1Update 00:05 EST 05/13/14: An earlier version of this story referred to Go as a “perfect information game.” It is more accurate to call it a “deterministic perfect information game.”

<http://www.wired.com/2014/05/the-world-of-computer-go/>