CONTRIBUTION TO SYMPOSIUM

WILEY

Insightful artificial intelligence

Marta Halina (ORCID)

Department of History and Philosophy of Science, University of Cambridge, Cambridge, UK

Correspondence
Marta Halina, Department of History and Philosophy of Science, University of Cambridge Free School Lane, Cambridge, CB2 3RH UK.
Email: mh801@cam.ac.uk

In March 2016, DeepMind's computer programme AlphaGo surprised the world by defeating the world-champion Go player, Lee Sedol. AlphaGo exhibits a novel, surprising and valuable style of play and has been recognised as "creative" by the artificial intelligence (AI) and Go communities. This article examines whether AlphaGo engages in creative problem solving according to the standards of comparative psychology. I argue that AlphaGo displays one important aspect of creative problem solving (namely mental scenario building in the form of Monte Carlo tree search), while lacking another (domain generality). This analysis has consequences for how we think about creativity in humans and AI.

KEYWORDS

artificial intelligence, comparative psychology, creativity, deep learning, insight

1 | INTRODUCTION

In the 1970s science fiction thriller, *Colossus*, a super-intelligent machine quickly discovers mathematical knowledge unknown to humankind. Although this is science fiction, it mirrors current aspirations in artificial intelligence (AI) research. One of the goals of AI research today is to develop "the kind of AI we need for science" (Hassabis, 2017a). This goal is beginning to look achievable. Computer programmes like AlphaFold can now predict the three-dimensional structure of proteins from amino acid sequences alone (Senior et al., 2020).

In 2015, the AI company DeepMind developed AlphaGo, a computer programme capable of playing the ancient Chinese game of Go. Go playing has long been viewed as a form of art

requiring human intelligence and creativity to master. In 2016, AlphaGo not only became the first computer programme to defeat a world-champion Go player but also introduced innovative and valuable strategies and tactics to the Go community. As professional Go player, Ke Jie, observed, AlphaGo "can see the whole universe of Go. I can only see a small area around me" (CGTN, 2017). If something like AlphaGo could be applied to science and technology, then *Colossus* might not remain fiction for long.

Is AlphaGo an example of the kind of creative intelligence required for scientific insight and innovation? AlphaGo fulfils the criteria for creativity widely adopted in the philosophical litera-ture: It is capable of producing novel, surprising and valuable solutions to problems (Boden, 2014). The Korean Go Association awarded it the highest rank (9 *dan*) in Go in part due to its creative play and its developers describe it as creative (Baker & Hui, 2017; Hassabis, 2017b). Thus, AlphaGo appears to be a significant milestone for AI. But the chess-playing programme Deep Blue, when it defeated Gary Kasparov in 1997, was also thought to represent a major milestone on the path to artificial general intelligence. Deep Blue's reliance on the specialised knowledge of chess grandmasters, however, meant that it was not ultimately a significant advance in technology (Ensmenger, 2012). What, if anything, makes AlphaGo different?

This article examines whether AlphaGo is capable of creative problem solving according to the standards currently set in comparative psychology. Comparative psychologists have long been developing accounts of the psychological capacities underpinning creative behaviour and how to identify such capacities in non-human animals (Köhler, 1976/1925). Their methods are particularly useful for examining artificial systems because they are designed to accommodate a wide range of structures, functions and behaviours. Applying these standards to AlphaGo pro-vides common ground for evaluating whether this programme is capable of creative problem solving and, if not, what more is required.

I begin by providing some brief background regarding the importance of Go to AI research (Section 2) and why members of the AI and Go communities have identified AlphaGo as crea-tive (Section 3). I then introduce the criteria and methods for identifying insightful problem solving in nonhuman animals currently used in comparative psychology (Section 4). As we will see, central to these accounts are the capacities for mental scenario building and domain-general understanding. Lastly, I introduce how AlphaGo works (Section 5) and evaluate its capacities according to the criteria developed in Section 4. I argue that while AlphaGo has capacities that resemble mental scenario building (particularly, its use of Monte Carlo tree sea-rch), the domain specificity of its world model means that its capacity for "insight" is signifi-cantly different from what we find in human and nonhuman animals. My conclusion, however, is not that AlphaGo fails to be creative *tout court*, but that the process by which it produces novel, surprising and valuable outcomes is unlike that found in animals. While AlphaGo lacks some of the virtues of animal insight, it exhibits other unique and surprising strengths, such as the capacity to transform a conceptual space in ways that do not appear available to human minds.

A few caveats: This article focuses on whether AlphaGo is capable of creative problem solv-ing, rather than the more general question of whether computer programmes can generate cre-ative products like music and art. I also leave open the precise connection between creative problem solving and scientific discovery. Much has been written on the role creativity plays in the sciences (Dunbar, 1997; Simonton, 2004). My analysis here, however, focuses on the specific question of how programmes like AlphaGo compare in their creative problem-solving abilities to animals. Finally, throughout this article, I use the terms "creativity" and "insight"

interchangeably. These terms are used variably in the philosophical and psychological literature. My objective is not to defend one definition over another, but rather to use existing accounts to probe what artificial and biological systems can and cannot do. In this article, I adopt Margaret Boden's definition of creative *products* as ideas or artefacts that are novel, surprising and valuable (Boden, 2014), while drawing on comparative psychology for an account of those cognitive *processes* responsible for creative products in the domain of problem solving (Section 4).

2 | GO: THE BETTER DROSOPHILA

Computer scientists have described chess as the "Drosophila of AI" since the 1960s (McCarthy, 1990). In the same way that Drosophila is a fruitful model organism, leading to major advances in our understanding of molecular biology and genetics, the successful development of a chess-playing AI, the argument goes, would represent a major leap in our understanding of human-like intelligence. The idea that good chess playing requires human-like general intelligence dates back to the 18th century (Ensmenger, 2012). If playing chess requires general intelligence, then engineering a machine that can play chess well is a suitable target for researchers aiming to develop artificial general intelligence. As it turned out, chess was ultimately not a good "model organism" of choice for AI, however, as computer scientists found ways to develop programmes that could defeat humans in chess while relying on brute-force computational techniques (Kasparov, 2017).

A widely recognised alternative to chess as the appropriate model organism for AI is the game of Go (McCarthy, 1990). Go is a strategy game in which two players take turns placing white and black stones on a 19×19 grid. Players place only one stone at a time and the objective of the game is to capture territory by surrounding areas of the board with one's stones (and preventing one's opponent from capturing territory). Although the rules of Go are simple, the game is complex. The branching factor (i.e., the possible moves from a given board position) of Go is 250, which is almost an order of magnitude more than chess (Silver et al., 2016). There are 10^{170} legal positions in Go, while the number of atoms in the known universe is 10^{80}. Thus, making decisions about where to place a stone requires vastly reducing the search space.

The possibility space of Go is vast, yet humans have learned to play Go well. This is achieved through a combination of theory and practise. Go professionals also emphasis their use of creative intuition (Hassabis, 2017b). Indeed, the capacity to play Go has long been viewed as a form of art in that people "play it without knowing how they [are] able to play so well" (Zobrist, 1969). Given the complexity of Go, and the fact that intelligent human play cannot be distilled into explicit rules programmable into a computer, it is not possible to engineer a programme that can play Go by either brute-force search or hand-coded rules. In this way, Go seems like a better Drosophila for AI than chess. If building a program that can play Go at human levels is at all possible, then such a programme might represent a milestone on the path to artificial creative intuition and judgement.

3 | ALGORITHMIC INSPIRATION

Although building a computer system that can play Go has been an objective in AI research for over half a century, the first significant achievement towards this goal took place in October

2015 when AlphaGo defeated the European Go champion Fan Hui. No machine could beat a professional Go player before this date (Silver et al., 2016). AlphaGo went on to achieve international recognition in 2016 when it defeated Lee Sedol, a South Korean professional Go player with 18 world titles. Subsequent programmes developed using similar deep-learning architectures proved even stronger. AlphaGo Zero, for example, beat AlphaGo 100-0 (Silver et al., 2017).

The incredible capacities of these programmes deserve philosophical attention. I focus here on the ability of these programmes to make moves that are surprising; novel and valuable; and described as "creative" by AI researchers and Go professionals. As software engineers Lucas Baker and Fan Hui write, "AlphaGo's strategy embodies a spirit of flexibility and open-mindedness: a lack of preconceptions that allows it to find the most effective line of play" (Baker & Hui, 2017). AlphaGo has introduced many valuable and innovative styles of play to the Go community. To give two examples: Firstly, AlphaGo regularly plays *tenuki* before the end of a *joseki* to its advantage. A "joseki" is a sequence of moves that is considered balanced for both players. "Tenuki" means ignoring a local exchange in order to play elsewhere. Joseki are intensely studied sequences of play; the fact that AlphaGo regularly plays tenuki one or more moves before what professional Go players would consider the end of a joseki suggests that the accumulated wisdom concerning these local exchanges may be mistaken. Secondly, a common proverb in Go is "high move for influence, low move for territory" where "high" and "low" refer to the fourth and third lines of the board, respectively. Another widely accepted proverb is "the second line is the route to defeat". AlphaGo, however, plays contrary to these conventions. The famous "move 37" was a play on the fifth line and a new joseki has now been attributed to AlphaGo, the "giant crawl" which consists of playing repeatedly on the second line. AlphaGo also favours "shoulder hits", "three-three invasions" and other moves that are unconventional among contemporary Go professionals. Exactly how such strategies contribute to AlphaGo's extraordinary playing strength is currently a topic of much discussion in the community.[1]

To put the above in perspective, humans have been playing Go for over 3,000 years and the consensus in the professional Go community before AlphaGo was that humans were converging on perfect play (Hassabis, 2017b). In contrast, after losing the tournament against AlphaGo, Lee Sedol observed, "What surprised me the most was that AlphaGo showed us that moves humans may have thought are creative, were actually conventional" (Kohs, 2017). As co-founder and CEO of DeepMind, Demis Hassabis writes, "[t]hese moments of algorithmic inspiration give us a glimpse of why AI could be so beneficial for science: The possibility of machine-aided scientific discovery" (Hassabis, 2017a).

4 | ANIMAL INSIGHT

AlphaGo has radically changed the way humans think of Go strategy, but does this programme play Go creatively in a way that resembles creative problem solving in cognitive creatures like

---

[1]See Sensei's Library for an online resource of lists and discussions of Go proverbs and joseki, including new joseki introduced by AlphaGo (https://senseis.xmp.net/). For more examples and discussion of AlphaGo's innovative play, see Baker and Hui (2017) and professional commentary on the Google DeepMind Challenge Match and *The future of Go summit*.

humans and other animals? To answer this question, we need to draw on what is known about creative problem solving in psychology.

Psychological studies on creative problem solving in human and nonhuman animals typically begin by presenting participants with a puzzle or problem. The puzzle is designed to ensure that the solution is not easy to find and often requires overcoming a cognitive bias of "functional fixedness". Functional fixedness is the tendency to view objects as serving particular functions and having difficulty imagining them being put to alternative uses (Shettleworth, 2012). This bias impedes the ability to think flexibly about how an object might be used in novel ways. In one classic human study on insight, participants are given a candle, a book of matches and a box of thumbtacks, and asked to attach the lit candle to a wall (Duncker, 1945). An effective, although rarely discovered, solution is to attach the tack box to the wall with some tacks, using the box as a shelf or ledge for the candle to stand on. Finding this solution requires thinking flexibly about how the box can be used, given its known physical properties, such as rigidity.

Animal cognition researchers have been interested in insightful problem solving since the early 20th century when the German Gestalt psychologist Wolfgang Köhler performed his classic studies on chimpanzees (Köhler, 1976/1925).[2] Köhler observed that chimpanzees, when faced with a puzzle (such as how to obtain an out-of-reach banana suspended from a high rope), would examine the situation and then seemingly become aware of the full solution all at once. Psychologists working in this area have since defined insight as "the sudden emergence of a complete solution without trial and error" (Seed & Boogert, 2013; see also Thorpe, 1956). In one contemporary insight task, corvids – a family of birds, including crows and ravens – are faced with food tied to the end of a string suspended from a perch (Figure 1). Like Köhler's apes, these birds are unable to reach the food by extending their bodies towards it. Simply pulling once on the string is equally ineffective, as the string is too long and the food remains out of reach. One solution to this problem is to grab a segment of the string with one's beak, pull it up, hold the slack part of the string underfoot while grabbing the next bit of string, and continuing in this way until the food is hauled up. Remarkably, some individuals from the corvid and parrot families execute this solution from the first trial, suggesting that they have solved the problem through a flash of insight (Heinrich, 1995; Heinrich & Bugnyar, 2005; see Taylor et al., 2010 for an alternative interpretation of these results).

What cognitive capacities underpin insightful problem solving? Research is ongoing, but two capacities are regularly highlighted as critical. The first is the ability to mentally plan or simulate (Taylor et al., 2010; but see Call, 2013). This capacity to plan accounts for several features of insightful problem solving, such as reaching a solution, and knowing that the solution will work, before engaging in action. When researchers explain a bird's success on the string-pulling task as a result of insight, they mean that the bird has imagined or mentally simulated (consciously or not) the sequence of actions of pulling and stepping on the string needed to reach the food (Taylor et al., 2010). After engaging in such a simulation, the bird knows how the problem can be solved and proceeds to conduct the sequence of actions required to do so (Seed & Boogert, 2013). This ability to think through problems in one's head is evolutionarily advantageous: It removes the risks of tinkering or intervening in an unknown situation in the real world and allows one to act appropriately in situations that have not been encountered before. Daniel Dennett refers to organisms that are capable of generating and testing hypotheses internally in this way as "Popperian creatures" (Dennett, 1996; Godfrey-Smith, 2018).

---

[2]Indeed, Köhler's work on chimpanzees inspired early research on creative problem solving in humans (Sobel, 1939).
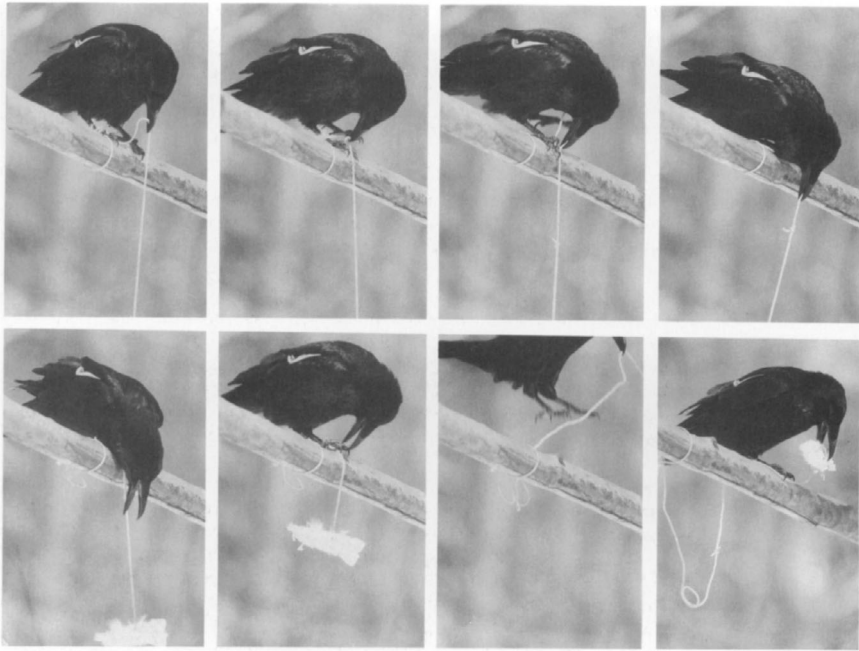
FIGURE 1 A Common Raven (*Corvus corax*, Bird 4 in Heinrich, 1995) solves the string-pulling task with no apparent trial-and-error learning. In the first trial, after a few initial pecks at the string, this participant suddenly performed a sequence of six steps (reach, grab, pull up, set down, step on, let go, reach down) and then proceeded to repeat these steps at least five more times (for a total of 30 steps) without error until obtaining the reward at the end of the string. Reprinted by permission from Oxford University Press (Heinrich, 1995)

In order to effectively generate and test hypotheses about the world, one must have some understanding of how the world works. This leads to the second capacity highlighted as critical for creative problem solving: The ability to learn and reason about objects and situations in terms of their domain-general properties (Brown, 1989). In the physical domain, such properties include the weight, rigidity, solidity, malleability, fluidity, and so forth, of objects, as well as capacities such as impact, displacement, and connection. Knowledge of these properties and capacities enables an organism to overcome cognitive biases like functional fixedness and reason flexibly about a problem. If an individual can reason about an object's affordances independently of the object's typical use, then he or she might conceive of new functions for that object, functions that can solve the problem at hand. Such an inferential strategy is hugely powerful as it enables an organism to flexibly deploy objects in ways that are not limited to the domain or function in which they were originally used or experienced (Godfrey-Smith, 2018). Humans seem to rely on such inferences from an early age, showing "rapid insightful transfer if they are familiar with the mechanism of causality that underlies the deep structural similarity between problems" (Brown, 1989).

In order to ensure that nonhuman animals are engaging in insightful problem solving, comparative psychologists control for the possibility that individuals might be relying on alternative cognitive and behavioural abilities. Firstly, as mentioned above, the problems presented in insight tasks are designed to be novel to the participants. Participants are not given access to the problem beforehand, or tested over many trials, and researchers choose problems that participants are unlikely to find in their everyday environment. This helps ensure that successful

performance is not due to past trial-and-error learning. Care is also taken to ensure that participants cannot draw on their natural behavioural repertoire or fixed action patterns to solve the problem in a routine or reflexive way (Heinrich & Bugnyar, 2005). Lastly, researchers probe individuals' understanding of the problem in order to determine the extent to which they understand the underlying properties and affordances of the objects involved (Taylor et al., 2010).

One recent example from the animal cognition literature illustrates the above experimental strategy. Betty the crow achieved international fame for her performance in a study at the University of Oxford (Weir, Chappell & Kacelnik, 2002). The study required that Betty choose between two tools – a straight wire and a hooked wire – to retrieve a small bucket containing food from a vertical, transparent tube. Another crow, however, interfered with the experiment and flew away with the hooked wire. Betty nevertheless proceeded to pick up the straight wire, bend it, and retrieve the food with the hooked end. Betty had not experienced wire before and had very little experience with wire-like material (Weir et al., 2002). Various considerations and follow-up tests led researchers to conclude that it was unlikely Betty had solved this task through the application of either instrumental learning or an evolutionarily acquired fixed action pattern (Weir et al., 2002; Weir & Kacelnik, 2006). This led researchers like Alex Kacelnik to conclude that Betty might have "a level of competence and understanding of the function of hooks unknown as yet outside our own species" (Graham, 2002) and was possibly using this knowledge to plan a solution to the problem (Weir & Kacelnik, 2006). If Betty were in fact deploying knowledge about the dispositions of objects (a malleable wire's capacity to be fashioned into a hook and a hook's capacity to retrieve food) to mentally solve this problem, then this would be a paradigm case of insightful problem solving.

Although Betty's performance suggested insightful problem solving, various aspects of her behaviour were at odds with this interpretation. She would sometimes modify a tool and then attempt to use the unmodified non-functional end of the tool to retrieve food, for example (Weir & Kacelnik, 2006). More recent studies have revealed that fashioning hooked tools is in fact part of the natural behavioural repertoire of New Caledonian crows (Rutz, Sugasawa, Van der Wal, Klump, & St Clair, 2016) and that crows are faster at retrieving food with hooked over straight tools regardless of the context, tool material, or food type (St Clair et al., 2018). The latter suggests that there are strong selection pressures for the capacity to produce hooked tools in crows. Whether this capacity has evolved in the form of a fixed tool-manipulation routine or something more flexible is to be determined.

The above case illustrates how comparative psychologists identify creative problem solving in nonhuman animals. They probe an individual's understanding of a problem and control for a range of ontogenetic and evolutionary factors that might provide alternative explanations for successful performance on novel problem-solving tasks. In what follows, I examine how AlphaGo works before proceeding to evaluate its capacities from the perspective of animal cognition research.

5 | HOW AlphaGo WORKS

As we have seen, AlphaGo produces results that are surprising, novel and valuable. How does it accomplish this? The version of AlphaGo that defeated Lee Sedol ("AlphaGo Lee") learned to play Go through a combination of training on human expert play and self-play (Silver et al., 2017). The basic principles behind the programme can be understood in terms of three

components: Two neural networks (the policy network and the value network) and a Monte Carlo tree search.[3]

A Monte Carlo tree search (MCTS) works by building a search tree according to the outcomes of simulated games. A "search tree" is an abstract data structure of linked nodes, which represents in this case legal board positions (the nodes) and moves (the edges) within the game of Go. Go is a game of perfect information. Each player is fully informed of the prior moves and board positions of the game. Given this, one can, in principle, solve the game by calculating an optimal value function, which maximises rewards, given perfect play by both players. Solving Go is not possible in practise, however, given the vast number of possible board positions and legal moves. Thus, AlphaGo employs what Claude Shannon (1950) calls a "type B" search strategy or one that is heuristically guided in its explorations (in contrast to a "type A" strategy, where all branches of the search tree are methodically explored). In the case of AlphaGo, two neural networks guide the construction of the search tree: The policy and value networks.

AlphaGo's policy network is a neural network with 12 convolutional layers trained through supervised and reinforcement learning. Supervised learning involves training a network on labelled data. In this case, the network was trained on 28.4 million board positions and moves from 160,000 unique games played by professional (6–9 dan) human players until it could accurately predict new human expert moves 56% of the time (Silver et al., 2016). The policy network was then further trained through reinforcement learning by playing 30 million games against a randomly selected previous version of itself. Here, the network was not told which moves a human expert would make, but instead the weights of the network were updated in the direction of winning games. The full version of AlphaGo Lee was iteratively trained in this way on a total of 100 million or more games (Lake, Ullman, Tenenbaum & Gershman, 2017). It is worth noting here that the policy network on its own is a strong Go programme. When competing against the advanced amateur Go programme, Pachi, it won 85% of the games (Silver et al., 2016, p. 485). Learning from millions of games then is a powerful strategy for playing Go well. As we will see below, however, MCTS allows AlphaGo to go beyond what it has learned from past experience, encouraging exploration of parts of the search tree that it has not encountered in its training.

The millions of self-played games used to train AlphaGo resulted in a huge data set of games. A subset of 30 million board positions from these games was used to train the value network. This network was trained using reinforcement learning to sort board positions according to whether they led to a win or loss. By doing so, the network learned the probability of a particular board state leading to a win. It is worth highlighting that each of the above 30 million board positions was sampled from different games in order to avoid overfitting. Training a network on board positions from the same game resulted in the network "memorising" which positions led to a win, since the board positions in any particular game are highly correlated (in a game, only one stone is added to the board each turn, and stones that have been placed on the board stay in one position throughout the game unless captured and removed). In these cases of memorisation, the programme performed well on the training set but poorly on new test sets (Silver et al., 2016, p. 486).

The main components of AlphaGo then are the above-trained deep neural networks combined with Monte Carlo tree search. Faced with a situation in an actual game (against Lee Sedol, e.g.), the neural networks guide the construction of the search tree: The policy network

---

[3]See Silver et al. (2016) for details regarding the training of AlphaGo Fan (the program that played against Fan Hui in October 2015 and Silver et al. (2017) for the differences between this program and AlphaGo Lee.
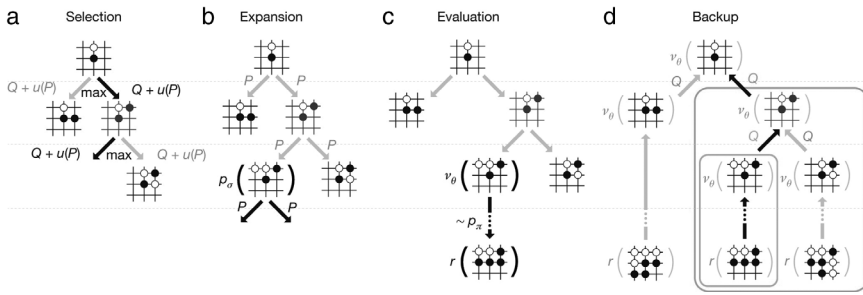
FIGURE 2 The main steps of the Monte Carlo Tree Search simulation in AlphaGo. *Source*: Reprinted by permission from Nature (Silver et al., 2016)

narrows the breadth of the search by choosing high-probability moves, while the value network together with Monte Carlo rollouts (discussed below) reduce the depth of the search by providing information about the value of a position (Silver et al., 2016, 2017).

A central motivation behind the creation of AlphaGo successor programmes like AlphaGo Zero and AlphaZero has been the desire to develop systems that can play complex games without relying on human expert knowledge (Silver et al., 2017, 2018). The "zero" in these programme names refers to "tabula rasa" learning; starting with nothing besides the rules of the game. Go and chess professionals focus their search of the game tree on those parts they think will lead to positive outcomes. AlphaGo draws on this human knowledge to guide its search. Without any such knowledge, AlphaGo Zero and AlphaZero begin by playing at random. Some of these randomly chosen options are bound to eventually lead to a win, which the Monte Carlo tree search then uses to guide its search of the possibility tree. Programmes like AlphaZero differ from AlphaGo in several other respects: They rely on a single neural network (instead of separate policy and value networks) and include MCTS in the course of training, for example. Although it is beyond the scope of this paper to consider the individual implications of these changes, together, they enable AlphaZero to not only achieve superhuman performance in chess, Go, and shogi, but defeat previous programmes, such as AlphaGo Lee, 100-0 (Silver et al., 2017).

6 | IS AlphaGo CREATIVE?

6.1 | Monte Carlo tree search as mental scenario building

Does AlphaGo meet the criteria for creative problem solving as understood in cognitive psychology? With respect to mental planning or simulation, the answer I believe is "yes". The Monte Carlo tree search used in AlphaGo simulates games in order to determine the value of a move, given a particular board position (Figure 2). The simulation proceeds by first selecting a particular path (the selection phase) and adding one or more valid moves to that path (expansion). One of these moves is then selected and, if it itself does not end the game in a win or a loss, a "playout" or "rollout" is carried out. Rollouts involve sampling a sequence of actions for both players and are guided by a "fast rollout policy". This rollout policy is similar to the policy network in that it is trained on human expert play, but it is faster and less accurate than the policy network. The outcome of the simulated game (who won) is combined with a value-network

evaluation to provide a general evaluation of the simulated node (evaluation). Once the simulation is complete, this information, along with information about how many times each node has been visited is propagated up the tree (backup).

AlphaGo's simulation phase is distinct from and prior to the phase in which it determines which move to make against its real-world opponent. When simulating possible moves, the programme is guided by the prior probability that a particular move will lead to a win (as indicated by the policy network) and a preference for those nodes that have been visited infrequently to encourage exploration. It is the exploration parameter that allows AlphaGo to go beyond its training, encouraging it to simulate moves outside of those recommended by the policy network. As the search tree is constructed, the system starts choosing moves with the highest "action value" to simulate, where the action value indicates how good a move is based on the outcome of rollouts and value-network evaluations. The more simulations there are, the larger the search tree, and the more likely that a high-action-value move will in fact be good. Nevertheless, when it comes time to decide which move to make in the real-world game at hand, AlphaGo does not rely on action values, but rather visit counts: The node that has been visited the most during the simulation phase is chosen. Visit counts are less susceptible to outliers than action values and are thus a less risky indicator of what move to play in an actual game (Silver et al., 2016).

Why should we view MCTS as analogous to mental planning or scenario building? MCTS allows an agent to explore a problem space by advancing various hypotheses about which move might be effective given the situation (i.e., the real-world board position that it currently faces). "Considering" a move does not commit AlphaGo to playing it, however. Game simulations may reveal that the move being considered is likely to lead to a loss. Indeed, the programme considers multiple avenues of play simultaneously. And, if at the end of a search, AlphaGo finds that the move with the highest visit count does not also have a high action value, it will continue searching. Finally, given that the programme is encouraged to explore moves beyond those recommended by the policy network, the hypotheses generated are not simply those that have proven effective in the past. All of this grants the programme some degree of flexibility: It can evaluate the potential outcome of a move without having experienced the effects of making that move in a real-world game (either during training or against another opponent). This search strategy fits Dennett's conception of a system capable of generating and evaluating actions within a simulated world before trying them out in the physical world (Dennett, 1996).

6.2 | Domain-specific learning

How does AlphaGo compare with respect to the second criterion of insightful problem solving – the capacity to draw on one's domain-general knowledge to generate novel solutions to problems? Here, the analogy to insightful problem solving in animals starts to break down. The input that AlphaGo receives is restricted to a 19×19 game of Go under standard Chinese rules with a *komi* of 7.5.[4] The policy network receives a $19 \times 19 \times 48$ image stack of 48 feature planes and the value network receives the same input with one additional feature plane representing the current colour to play.[5] In a standard game of Go, a stone can be placed on any

---

[4]In standard Go, the player with black stones has the advantage of playing first. *Komi* is the number of points given to the player with white stones to compensate them for playing second.

[5]These feature planes are 19×19 planes of binary values, where a value represents the presence or absence of a particular feature.

intersection of the 19 × 19 board (presuming the move is legal). Accordingly, each feature plane represents the status of these 361 intersections with respect to a particular feature, such as stone colour (black, white, empty), liberties (how many empty spaces there are adjacent to one's stones), self-atari size (how many of one's own stones would be captured with this move), and others (see Silver et al., 2016). This relatively basic set of inputs provides AlphaGo with what it needs to analyse and classify board states at the scale of both local battles and whole-board strategy for a standard 19 × 19 game of Go. Any aspect of the world outside of this domain, however, does not exist, so to speak, for AlphaGo.

We can understand AlphaGo as constructing and employing a "world model" of its environment.[6] AlphaGo's input and training allows it to construct a model of board positions, Go rules, promising moves (given human expert play), and valuable positions (given the known outcomes of games). Critically, however, this world model represents a very specific part of the world – that of a standard 19 × 19 game of Go. As Silver et al. (2017) write, "the neural network architecture is matched to the grid-structure of the [19 × 19] board" (p. 360). AlphaGo has been trained to identify valuable moves and board positions within this context, according to Chinese rules with a *komi* of 7.5 points. Of course, as we have seen, AlphaGo is not limited to actions that have been reinforced in the past. Faced with a particular board position in a real-world game, it can explore and assess new avenues of play through simulation. Nevertheless, these simulations themselves are constrained by the system's world model. The policy network guides the moves to be considered and the value network provides information on the value of board positions. Also, the sequences of actions sampled by rollouts depend on the fast rollout policy and the rules of a standard 19 × 19 game.

The implication of this is that although AlphaGo is capable of planning flexibly within the context of its world model, it lacks the input to know which actions might be rewarding in contexts other than a standard game of Go. To see this, consider that there are many variants of Go that introduce subtle changes to the rules and structure of the game, which human Go players can typically understand and adapt to. Tibetan Go, for instance, awards extra points to the player that occupies all four corner points on the board, and the Korean Go variant, bangneki, includes a fixed wager for every 10 points by which a player is defeated. Although these and other variants of Go differ by only a few elements from the version of Go on which AlphaGo was trained, the programme is unable to successfully play them (Lake et al., 2017). AlphaGo's input and training are crucial for its success: These factors enable the programme to construct an incredibly effective world model for navigating the vast space that is a 19 × 19 game of Go. AlphaGo, however, loses traction on environments that require an alternative world model for success.

The above analysis applies to programmes like AlphaGo Zero and AlphaZero insofar as their successful performance depends on domain-specific input (Silver et al., 2017, 2018). Depending on the game they are being trained to play, these programmes receive as input an image of a game board (8 × 8 for chess, 9 × 9 for shogi, 19 × 19 for Go) with feature planes representing possible board positions. They are also given knowledge of the game rules, which are used during MCTS. Like for AlphaGo, the result is a world model that fits a particular part of the world. Although AlphaZero is described as a "general" algorithm, what is meant by this is that the same algorithm can be retrained to learn how to play different two-player games of perfect information (such as shogi, chess and Go), not that it can be trained on one game and transfer what it has learned to other contexts (Silver et al., 2018). Doing so would mean

---

[6]I thank an anonymous reviewer for suggesting this framing.

applying a model of one game, such as shogi, to another, such as Go, which would be an unsuccessful strategy, given that the board positions and rules from one game do not transfer to the other. This is not to minimise the power, value or even creative potential (see below) of these systems, but to identify an important difference between them and the capacities believed to be operating in animals when they engage in insightful problem solving (Shevlin & Halina, 2019).

7 | CREATIVITY IN ANIMALS AND AI

Computer programmes like AlphaGo are not creative in the sense of having the capacity to solve novel problems through a domain-general understanding of the world. They cannot learn about the properties and affordances of objects in one domain and proceed to abstract away from the contingencies and idiosyncrasies of that domain in order to solve problems in a new context. AlphaGo's world model is instead tightly coupled to its input, which is in turn coupled to a standard 19×19 game of Go. Faced with a situation in a real-world game, AlphaGo has the remarkable ability to generate and test hypotheses concerning which moves might be valuable before choosing which move to play. But this ability to plan depends on AlphaGo's model of its environment. Insofar as this model fails to apply to the situation (i.e., insofar as the conditions for success differ from those of a standard 19×19 game of Go), this ability to flexibly plan will fail to yield valuable results.

Although AlphaGo does not engage in insightful problem solving as found in humans and other animals, this does not mean it completely lacks creative power. Indeed, from what we have seen here, there is a compelling case that AlphaGo is responsible for what Margaret Boden calls "transformational creativity" with respect to Go. Transformational creativity involves not just combining old things in new ways or originating something new within an existing conceptual space, but transforming the space in which a problem is conceived (Boden, 2014). This form of creativity is traditionally believed to be out of reach of AI (Ibid.). Boden defines a conceptual space as "any disciplined way of thinking that's familiar to (and valued by) a certain social group" (Boden, 2004, p. 4). Under this definition, AlphaGo is responsible for radically transforming the conceptual space of Go – a game that humans have devoted vast amounts of time and effort to understand. In this way, AlphaGo demonstrates that there are forms of creativity – indeed transformational creativity – that can be achieved without domain-general understanding.

What is the significance of AlphaGo's creativity? In addition to providing humans with a new conceptual understanding of Go, AlphaGo might give us insight into constraints on human knowledge. Although studies on the psychology of Go playing are limited, there is some evidence that learning and playing Go relies on culturally acquired concepts and templates (Gobet, de Voogt & Retschitzki, 2004). As we saw above, players learn Go through the application of proverbs, such as "the second line is the route to defeat". Players at all levels also rely on *joseki* or sequences of moves that are believed to lead to a balanced outcome for both sides. In chess, there is evidence that professionals rely on a "system of playing methods" where individuals draw on a specific blend of tactical and strategic methods, allowing them to rely more on routine knowledge than look-ahead (Gobet et al., 2004, p. 120). Go professionals similarly have distinctive styles of play and a recurring theme of professional commentators on AlphaGo is that it diverges from humans in this respect. As 9-dan professional Kim Sung Yong observed in response to AlphaGo's performance at *The future of Go summit*: "When human artists start drawing landscapes, they keep drawing landscape no matter what happens during the process.

However, AlphaGo can quickly switch from landscape to portrait".[7] Kim Sung Yong's point here is that while human Go players tend to have a fixed style of play, AlphaGo does not.

The templates, concepts, patterns and other aids that humans use to teach, learn and play Go might explain in part AlphaGo's success relative to humans. Conceptual and linguistic tools such as these facilitate human learning, memory, and reasoning but can also lead to functional fixedness. It might be difficult to view the second line as something other than a "route to defeat" if one has relied on this belief successfully in the past and seen a community of experts similarly do so. In contrast, through exploration and playing tens of millions of games against itself, AlphaGo has discovered value functions that radically diverge from these conceptual guides. Although AlphaGo's remarkable abilities are specific to the domain of Go, they provide humans with a powerful tool for overcoming functional fixedness in Go play and in this sense have great creative potential.

Upon discovering that fashioning hooked tools is part of the natural behavioural repertoire of New Caledonian crows, the biologist Christian Rutz remarked that Betty "might have been a little robot … just following a natural, behavioral routine" (Morell, 2016). The account provided here expands our understanding of what machines can do and how these capacities relate to the abilities of animals. Artificial systems do not act only according to preprogrammed rules hand-coded by engineers. Moreover, current deep-learning methods are capable of producing systems that are superhuman in their abilities to discover novel and valuable solutions to problems within specific domains. However, if AI researchers are seeking to build systems with animal-like insight, then they will likely have to move beyond the methods used to construct AlphaGo. Domain-general insightful problem solving is powerful because it allows one to reason and intervene on the world effectively without having encountered that particular part of the world before. Given this, there has been a large push in AI research recently to imbue machines with "intuitive physics", "theory of mind", and other domain-general forms of reasoning (Buckner, 2018; Lake et al., 2017). The tools developed by comparative psychologists for studying animal minds are an incredibly valuable resource for probing the capacities of machines and rigorously comparing them to the diverse forms of intelligences we find in the biological world (Crosby, Beyret & Halina, 2019). Perhaps the new Drosophila of AI is the fruit fly itself.

ORCID

Marta Halina https://orcid.org/0000-0002-3482-4281

---

[7]For this and other comments on AlphaGo's performance at *The future of Go summit*, see https://deepmind.com/alphago-china.

REFERENCES

Baker, L., & Hui, F. (2017, April 10). Innovations of AlphaGo. DeepMind. https://deepmind.com/blog/innovations-alphago/

Boden, M. A. (2004). *The creative mind: Myths and mechanisms*. New York, NY: Routledge.

Boden, M. A. (2014). Creativity and artificial intelligence: A contradiction in terms. In E. S. Paul & S. B. Kaufman (Eds.), *The philosophy of creativity: New essays* (pp. 224–246). Oxford: Oxford University Press.

Brown, A. (1989). Analogical learning and transfer: What develops? In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning* (pp. 369–412). Cambridge: Cambridge University Press.

Buckner, C. (2018). Empiricism without magic: Transformational abstraction in deep convolutional neural networks. *Synthese*, *195*(12), 5339–5372.

Call, J. (2013). Three ingredients for becoming. In C. M. Sanz, C. Boesch & J. Call (Eds.), *Tool use in animals: Cognition and ecology* (pp. 3–20). Cambridge: Cambridge University Press.

CGTN. (2017). Exclusive: Ke Jie wants you to know weiqi is fun and easy to learn [video]. Youtube. Retrieved from https://youtu.be/OCevCII1zo0

Crosby, M., Beyret, B. & Halina, M. (2019). The animal-AI olympics. *Nature Machine Intelligence*, *1*(5), 257.

Dennett, D. (1996). *Kinds of minds: Toward an understanding of consciousness*. New York, NY: Basic Books.

Dunbar, K. (1997). How scientists think: On-line creativity and conceptual change in science. In T. B. Ward, S. M. Smith & J. Vaid (Eds.), *Conceptual structures and processes: Emergence, discovery, and change* (pp. 461–493). Washington, DC: American Psychological Association Press.

Duncker, K. (1945). On problem-solving (Lees, L. S., Trans.). *Psychological Monographs*, *58*(5), i–113.

Ensmenger, N. (2012). Is chess the drosophila of artificial intelligence? A social history of an algorithm. *Social Studies of Science*, *42*(1), 5–30.

Gobet, F., de Voogt, A. & Retschitzki, J. (2004). *Moves in mind: The psychology of board games*. New York, NY: Psychology Press.

Godfrey-Smith, P. (2018). Towers and trees in cognitive evolution. In B. Huebner (Ed.), *The philosophy of Daniel Dennett* (pp. 226–253). Oxford: Oxford University Press.

Graham, K. (2002, August 9) Stone the cover crows. ABC Science. Retrieved from https://www.abc.net.au/science/articles/2002/08/09/643534.htm

Hassabis, D. (2017a, April 22). The mind in the machine: Demis Hassabis on artificial intelligence. *The Financial Times*, 12.

Hassabis, D. (2017b). *Explorations in optimality*. Paper presented at the Beneficial AI conference, Asilomar, CA.

Heinrich, B. (1995). An experimental investigation of insight in common ravens (*Corvus corax*). *The Auk*, *112*(4), 994–1003.

Heinrich, B. & Bugnyar, T. (2005). Testing problem solving in ravens: String-pulling to reach food. *Ethology*, *111* (10), 962–976.

Kasparov, G. (2017). *Deep thinking: Where machine intelligence ends and human creativity begins*. London: John Murray.

Köhler, W. (1976/1925). *The Mentality of Apes (Winter, E., Trans.)*. New York, NY: Liveright.

Kohs, G. (2017). *Alphago* [Motion picture]. USA: Reel as Dirt.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B. & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences*, *40*, 1–72.

McCarthy, J. (1990). Chess as the drosophila of AI. In T. A. Marsland & J. Schaeffer (Eds.), *Computers, chess, and cognition* (pp. 227–237). New York, NY: Springer.

Morell, V. (2016). Was Betty the crow a genius—Or a robot? *Science* Retrieved from. https://www-sciencemag-org.ezp.lib.cam.ac.uk/news/2016/08/was-betty-crow-genius-or-robot

Rutz, C., Sugasawa, S., Van der Wal, J. E., Klump, B. C., & St Clair, J. J. (2016). Tool bending in New Caledonian crows. *Royal Society Open Science*, *3*(8), 160439.

Seed, A. M. & Boogert, N. J. (2013). Animal cognition: An end to insight? *Current Biology*, *23*(2), R67–R69.

Senior, A. W., Evans, R., Jumper, J., Kirkpatrick, J., Sifre, L., Green, T., … Penedones, H. (2020). Improved protein structure prediction using potentials from deep learning. *Nature*, *577*, 1–5.

Shannon, C. (1950). Programming a computer for playing chess. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, *41*(314), 256–275.

Shettleworth, S. J. (2012). Do animals have insight, and what is insight anyway? *Canadian Journal of Experimental Psychology*, *66*(4), 217–226.

Shevlin, H. & Halina, M. (2019). Apply rich psychological terms in AI with care. *Nature Machine Intelligence*, *1*(4), 165–167.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., … Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, *529*, 484–489.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., … Hassabis, D. (2018). A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, *362*(6419), 1140–1144.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., … Hassabis, D. (2017). Mastering the game of Go without human knowledge. *Nature*, *550*(7676), 354–359.

Simonton, D. K. (2004). *Creativity in science: Chance, logic, genius, and zeitgeist*. Cambridge: Cambridge University Press.

Sobel, B. (1939). A study of the development of insight in preschool children. *The Pedagogical Seminary and Journal of Genetic Psychology*, *55*(2), 381–388.

St Clair, J. J., Klump, B. C., Sugasawa, S., Higgott, C. G., Colegrave, N., & Rutz, C. (2018). Hook innovation boosts foraging efficiency in tool-using crows. *Nature Ecology & Evolution*, *2*(3), 441–444.

Taylor, A. H., Medina, F. S., Holzhaider, J. C., Hearne, L. J., Hunt, G. R. & Gray, R. D. (2010). An investigation into the cognition behind spontaneous string pulling in New Caledonian crows. *PLoS One*, *5*(2), e9345.

Thorpe, W. H. (1956). *Learning and instinct in animals*. London: Methuen & Co.

Weir, A. A., Chappell, J. & Kacelnik, A. (2002). Shaping of hooks in New Caledonian crows. *Science*, *297*(5583), 981–981.

Weir, A. A. & Kacelnik, A. (2006). A New Caledonian crow (*Corvus moneduloides*) creatively re-designs tools by bending or unbending aluminium strips. *Animal Cognition*, *9*(4), 317–334.

Zobrist, A. L. (1969). A model of visual organization for the game of Go. *Proceedings of the May 14–16, 1969, Spring Joint Computer Conference*, Association for Computing Machinery, New York, NY.