

# ORMS TODAY

October 2016

Volume 43 • Number 5  
ormstoday.informs.org

## Also Inside:

- **AlphaGo:**  
man vs. 'intelligent' machines
- **Analytics education:**  
leaders weigh in on present & future



# Decisions, decisions

So many questions, so many answers.

- 2016 survey of decision analysis software
- Decision definition: irrevocably allocating resources
- Election analytics: predicting the next president



# Contents

October 2016 | Volume 43, No. 5 | [ormstoday.informs.org](http://ormstoday.informs.org)



## On the Cover

### The road not taken

From corporate decision-making to citizens voting, the world is full of decision analysis problems.

Image © alphaspirit | 123rf.com

## Features

### 20 Using big data in cybersecurity

By Douglas A. Samuelson

Operations research analysts search for “cyber subs”: defending U.S. cyber assets is mission critical for U.S. military.

### 24 Google DeepMind’s AlphaGo

By Hyeon Soo Chang, Michael C. Fu, Jiaqiao Hu and Steven I. Marcus

O.R.’s unheralded role in path-breaking achievement signals major advancement of truly “intelligent” machines.

### 30 Election Analytics

By Wenda Zhang, Jason J. Sauppe and Sheldon H. Jacobson

Student-driven STEM learning lab’s election forecasting website predicts presidency and Senate races.

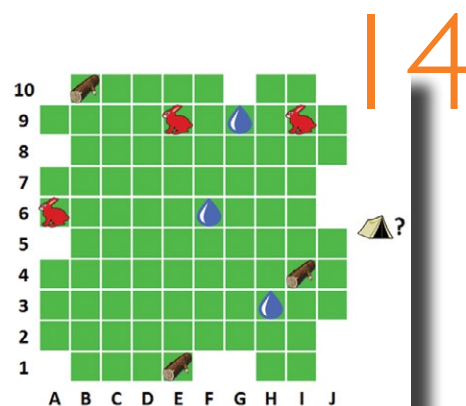
### 32 Present and future of analytics education

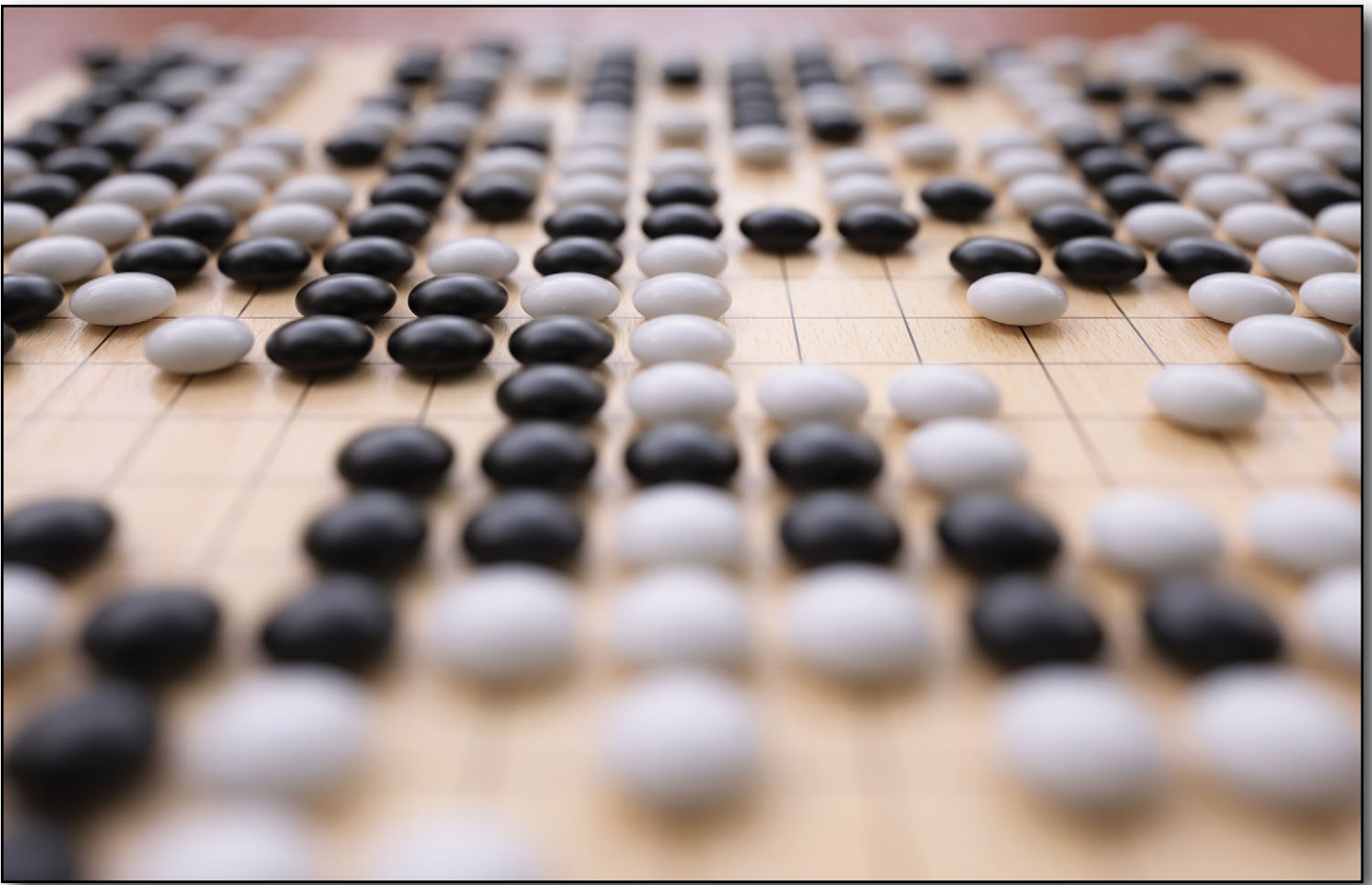
By Kaibo Liu, Diego Klabjan, David Shmoys and Joel Sokol

Panel discussion: Directors from four of the nation’s top university analytics programs share their insights and expertise.

## departments

- 6 Inside Story
- 8 President’s Desk
- 10 INFORMS in the News
- 12 Issues in Education
- 14 PuzzlOR
- 16 INFORMS Initiatives
- 18 Viewpoint
- 54 Industry News
- 55 Literature Files
- 56 Classifieds
- 64 ORacle





The game of Go originated in China more than 2,500 years ago.

Image © Sergey Soldatov | 123rf.com

# Google DeepMind's AlphaGo

**Operations research's  
unheralded role  
in the path-breaking  
achievement.**

**By Hyeong Soo Chang,  
Michael C. Fu, Jiaqiao Hu  
and Steven I. Marcus**

**AlphaGo's shockingly dominant victory over the reigning world Go champion Lee Sedol in Seoul, Korea,** in March 2016 signaled another great leap in the seemingly relentless advancement of machines becoming truly “intelligent” in the sense of being able to learn and outsmart humans. When IBM's Deep Blue defeated world chess champion Garry Kasparov in 1997, it was thought at the time to have reached the ultimate pinnacle in computer game-playing abilities, since chess had been considered (at least in the Western world) to be the game requiring the most human brainpower, so the fact that a computer had finally surpassed the abilities of humankind's best would seem to have indicated that some aspects only found in science fiction might be finally approaching reality.

However, the “primitive brute force-based” [1] Deep Blue appeared to be more of a showcase of hardware than software, as it relied more on its computational firepower rather than any intuition or actual learning in its approach to the game. On the other hand, because of the mind-boggling

number of possible configurations due to a much larger board size (19x19 vs. 8x8 for chess), such an approach is just not feasible in any foreseeable future for the game of Go. Thus, a paradigm shift was required, and at the heart of all successful Go-playing computer programs over the past decade has been the use of intelligent sampling (Monte Carlo simulation) rather than “intelligent” enumeration, which basically means immense storage of games and doing more clever enumeration (e.g., by pruning). However, all of these game-playing programs do have one thing in common: the concept of “learning,” using algorithms under the general umbrella of machine-learning techniques (where it appears that concepts such as branching, bounding, and pruning now have been absorbed under this catchall).

According to a Google blog posting [2]:

*“The game of Go originated in China more than 2,500 years ago. The rules of the game are simple: Players take turns to place black or white stones on a board, trying to capture the opponent’s stones or surround empty space to make points of territory. As simple as the rules are, Go is a game of profound complexity. There are more possible positions in Go than there are atoms in the universe. That makes Go a googol times more complex than chess.*

*“The objective of the game – as the translation of its name (weiqi in Mandarin) implies – is to have surrounded a larger total area of the board with one’s stones than the opponent by the end of the game.”*

This article covers several aspects regarding AlphaGo’s success. We begin with an introduction to DeepMind and a brief description at a higher level of the main parts of the AlphaGo program – specifically the two “deep” neural networks and the technique of Monte Carlo tree search with upper confidence bounds (UCBs) that is used in all modern computer Go-playing (as well as other computer game-playing) programs. The main ideas of the latter can be found in a paper that appeared in the INFORMS journal *Operations Research* in 2005 [3], but is not well known in the computer science (CS) and artificial intelligence (AI) game-playing community. In describing this apparent disconnect, we will segue into comparing and contrasting the different academic cultures of the operations research (O.R.) and CS/AI communities, and raising questions as to how these two communities could possibly interact in a mutually beneficial manner to build upon the research strengths of both.

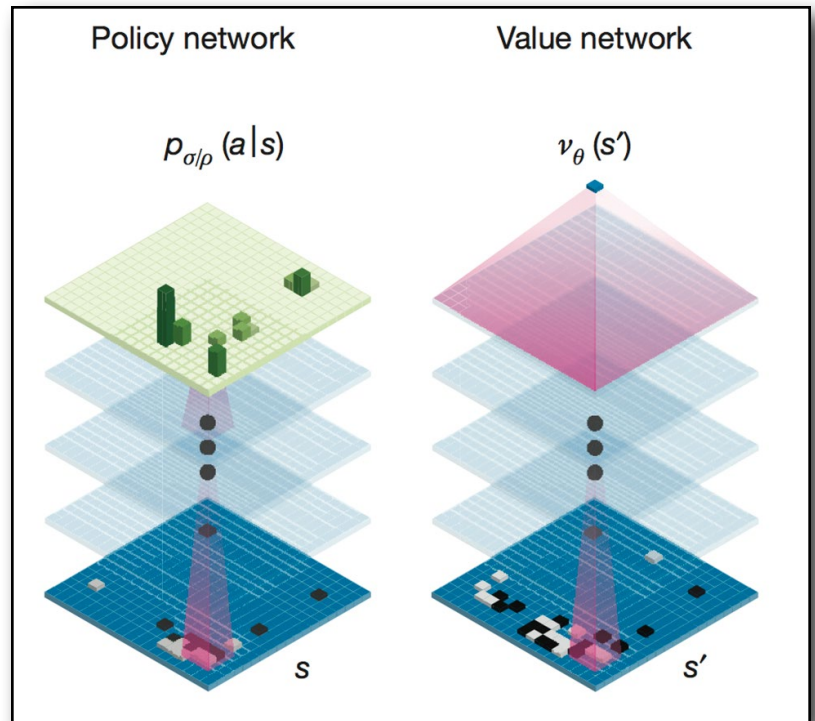


Figure 1: AlphaGo’s two-deep neural networks.

Source: Nature, 2016 [6]

### Google DeepMind

Google DeepMind’s web page on AlphaGo screams out in all-caps: “THE FIRST COMPUTER PROGRAM TO EVER BEAT A PROFESSIONAL PLAYER AT THE GAME OF GO” [4]. AlphaGo accomplished this incredible feat, assumed by AI experts to be many decades away, in October 2015, when it defeated the reigning European champion, Fan Hui, by a stunning margin of 5-0.

The creator of AlphaGo, Google DeepMind, is a London-based artificial intelligence company founded in 2010 by Demis Hassabis, Shane Legg and Mustafa Suleyman. As promoted on their web page [5]:

*“DeepMind was supported by some of the most iconic tech entrepreneurs and investors of the past decade, prior to being acquired by Google in early 2014 in their largest European acquisition to date. ... The algorithms we build are capable of learning for themselves directly from raw experience or data, and are general in that they can perform well across a wide variety of tasks straight out of the box. Our world-class team consists of many renowned experts in their respective fields, including but not limited to deep neural networks, reinforcement learning and systems neuroscience-inspired models. ... Our Nature paper ... describes the technical details behind a new approach to computer Go that combines Monte-Carlo tree search with*

Because of  
**a much  
larger board  
size**  
(19x19 vs. 8x8),  
the game of  
**Go offers**  
a  
**mind-  
boggling  
number**  
of  
possible  
configurations  
**compared  
to chess.**



**AlphaGo**  
was first  
**trained**  
**using**  
past human  
**games.**  
Then it  
**played**  
**against itself**  
thousands of  
times  
**to further**  
**adjust**  
the  
**neural**  
**network.**

*deep neural networks that have been trained by supervised learning, from human expert games and by reinforcement learning from games of self-play."*

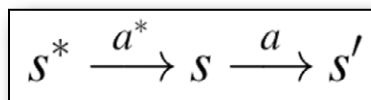
## Core of AlphaGo: Deep Learning

The following two-deep neural networks comprise AlphaGo's core: value network – estimates the "value" (probability of winning) of a given board configuration; and policy network – provides a probability distribution over (opponent's) actions for a given board configuration.

Each network has 12 layers and millions of connections, and as depicted in Figure 1 (from the 2016 *Nature* article [6]) takes as input a representation of the board configuration. The subscripts  $\sigma$  and  $\rho$  in the policy network probability correspond to two different networks used, based on supervised learning and reinforcement learning, respectively, whereas the subscript  $\theta$  in the value network represents the parameterization of the function.

Because the number of "states" (board configurations) is way too huge to be enumerated, the probability distribution provides a "weighting" of which moves are more preferable by the opponent. If the optimal move in a given board configuration is known, the distribution becomes deterministic, but for a large proportion of the state space, the opponent's move must be sampled using Monte Carlo simulation to generate sample paths of possible games. If simulated to the end, a win or loss could be determined, and then propagated backwards to update the value function. Examples depicting the process using tic-tac-toe can be found in Fu [7].

AlphaGo was first trained using past human games, considering more than 30 million moves (supervised learning). Then it played against itself thousands of times to further adjust the neural network parameters (reinforcement learning) using Monte Carlo tree search with upper confidence bounds (UCBs), which directs which actions to take. In terms of Figure 1, the latter determines which move  $a^*$  to select in a given board configuration  $s^*$ , which leads to board configuration  $s$ , for which an opponent move  $a$  is sampled (simulated) from  $p(a|s)$ , which leads to state  $s'$ ,



at which point the value network could be used or the previous process could be repeated until a

satisfactory state (in terms of the value network) or the end of game is reached.

## Monte Carlo Tree Search

Monte Carlo tree search (MCTS) was coined by Rémi Coulom in his 2006 paper [8], where he also refers to the adaptive multi-stage sampling algorithm of Chang et al. [3] – to be described shortly – as a Monte Carlo tree search algorithm. The name Monte Carlo tree search captures the essence of the approach far better than "adaptive multi-stage sampling" and thus stuck in the AI community. The abstract of the survey article, "A Survey of Monte Carlo Tree Search Methods" (Browne et al., 2012 [9]), provides an overview of the approach:

*"Monte Carlo Tree Search (MCTS) is a recently proposed search method that combines the precision of tree search with the generality of random sampling. It has received considerable interest due to its spectacular success in the difficult problem of computer Go, but has also proved beneficial in a range of other domains."*

Browne et al. go on to provide the following summary description of MCTS:

*"The basic MCTS process is conceptually very simple. ... A tree is built in an incremental and asymmetric manner. For each iteration of the algorithm, a tree policy is used to find the most urgent node of the current tree. The tree policy attempts to balance considerations of exploration (look in areas that have not been well sampled yet) and exploitation (look in areas which appear to be promising). A simulation is then run from the selected node and the search tree updated according to the result. This involves the addition of a child node corresponding to the action taken from the selected node, and an update of the statistics of its ancestors. Moves are made during this simulation according to some default policy, which in the simplest case is to make uniform random moves. A great benefit of MCTS is that the values of intermediate states do not have to be evaluated, as for depth-limited minimax search, which greatly reduces the amount of domain knowledge required. Only the value of the terminal state at the end of each simulation is required."*

In practice, most implementations of Monte Carlo tree search, including all of those in the best Go-playing computer programs, use an algorithm called UCT (upper confidence bound 1 applied to trees) introduced in Kocsis and Szepesvári (2006, [10]), based on the UCB1 formula of Auer et al. (2002, [11]) and the provably convergent algorithm

first applied to multi-stage decision-making models (specifically, Markov decision processes) by Chang et al. (2005, [3]), a paper published in *Operations Research*.

### Connection to Operations Research

The *Operations Research* paper (Chang et al. 2005) cited above was first submitted in May 2002 and presented just prior to that at the Cornell ORIE Colloquium on April 30. The adaptive multi-stage sampling (AMS) algorithm of Chang et al. chooses to sample the action that maximizes the upper confidence bound (UCB). As described in Chang et al., AMS “approximates the optimal value to break the well-known curse of dimensionality in solving finite horizon Markov decision processes (MDPs). The algorithm is aimed at solving MDPs with large state spaces and relatively smaller action spaces. The approximate value computed by the algorithm not only converges to the true optimal value, but it does so in an ‘efficient’ way. The algorithm adaptively chooses which action to sample as the sampling process proceeds, and the estimate produced by the algorithm is asymptotically unbiased.”

Kocsis and Szepesvári (2006) cited the work as follows:

*“Recently, Chang et al. also considered the problem of selective sampling in finite horizon undiscounted MDPs. ... At each node they sample (recursively) a sufficient number of samples to compute a good approximation of the value of the node. The subroutine returns with an approximate evaluation of the value of the node. ... Similar to our proposal, they suggest to propagate the average values upwards in the tree and sampling is controlled by upper-confidence bounds.”*

Coulom (2006) writes:

*“Our approach is similar to the algorithm of Chang, Fu and Marcus [sic] ... In order to avoid the dangers of completely pruning a move, it is possible to design schemes for the allocation of simulations that reduce the probability of exploring a bad move, without ever letting this probability go to zero. Ideas for this kind of algorithm can be found in ...n-armed bandit problems, ... (which) are the basis for the Monte-Carlo tree search algorithm of Chang, Fu and Marcus [sic].”*

## INFORMS CAREER FAIR



The INFORMS Career Center offers employers expanded opportunities to connect to qualified O.R. and analytics professionals, as well as a complete line of services to be used alone or in conjunction with the Career Fair at the 2016 Annual Meeting. Both give applicants and employers a convenient venue to connect. The Career Center is free to INFORMS members.

### SEARCHING? HIRING?

Access INFORMS CAREER CENTER

- More analytics jobs
- More O.R. jobs
- Preferred jobs
- Featured jobs
- Job alerts
- Anonymous career profiles
- Powerful searching capabilities



Learn more about the 2016 INFORMS Annual Meeting Career Fair,  
<http://meetings.informs.org/nashville2016/career-fair.html>

**AlphaGo**  
**represents**  
a formidable  
**AI**  
**achievement**  
stemming  
from  
**several**  
**research**  
**streams,**  
for which  
the  
**OR/MS**  
**community**  
has played  
**a role.**

In other words, as mentioned earlier, Coulom himself refers to the AMS algorithm as a Monte Carlo tree search. The AMS algorithm was the first work to explore the idea of UCB-based exploration and exploitation in constructing sampled/simulated (Monte Carlo) trees and is clearly the main seed for UCT.

## A Tale of Two Cultures in Academia

As described, a core element in AlphaGo is MCTS with UCB, which came from the 2005 *Operations Research* journal paper, but the two 2006 AI conference papers (which as noted already did cite the 2005 paper) receive all the publicity (and citations; well over 2,000 in Google Scholar versus 75 for the *Operations Research* paper as of Sept. 11, 2016), because much of the AI community is unaware of O.R. advances.

Much of the CS community publishes primarily in conference proceedings, whereas, in general, the OR/MS community doesn't value conference proceedings papers for promotion and tenure (especially in business schools, where they may even be viewed negatively), and journal papers take years to get published. Moreover, once a work is published in a conference, it becomes more difficult to publish it in a journal, which is probably the main reason the main INFORMS conferences have no proceedings.

In contrast, the CS (and AI community within it) culture values conference proceedings the most, which puts the research agenda in a very different mode, always seeking to meet the next deadline (e.g., NIPS, AAAI, ICML). There's nothing like a deadline for increasing productivity, and CS/AI conference papers generally get much higher citation counts than our journal papers; however, this modus operandi is not without its costs. For example, due to the tight reviewing deadlines, mathematical proofs are rarely if ever closely checked, and many unsubstantiated claims end up being published. Many CS communities, such as the AI community, have tighter links with industry, at least the research-oriented groups within such giants as Google, Microsoft, Facebook and Yahoo, and this is evidenced by the active participation in the previously mentioned conferences.

It should be noted that many IEEE societies (of which both the CS/AI and OR/MS communities participate to some degree) have a hybrid model. For example, the IEEE Conference on Decision and Control, for which INFORMS is one of the sponsoring organizations, has a system whereby work submitted for the conference and published in the proceedings can also be considered (in lengthier form) for a corresponding journal, the *IEEE Transactions on Automatic Control*. Perhaps this is a model that could be considered by some INFORMS-sponsored conferences.

One attempt to bridge the gap between the O.R. and AI communities was a National Science Foundation (NSF)-sponsored workshop entitled, "A Conversation Between Computer Science and Operations Research on Stochastic Optimization." As described in Fu and Barton (2012, p.36, [12]), "This workshop is co-funded and co-sponsored by the Robust Intelligence (the new name for artificial intelligence (AI) at NSF) Program in CISE, and is motivated by a mutual feeling by the Robust Intelligence Program Director Sven Koenig and the OR Program director that the AI and OR communities carry out research on essentially the same topics oftentimes unaware of the contributions from the other community due to differences in terminology and notation. Such differences prevent research ideas from being understood and shared, so the workshop aims to break down such barriers, focusing on one broad area of interest to both communities, encompassing well-known topics such as AI planning and Markov decision processes." The workshop, led by Warren Powell (Princeton) and Satinder Singh (Michigan), was held May 31-June 1, 2012 on the campus of Rutgers University. It was there that one of us first heard about the UCB Monte Carlo tree search algorithm, which sounded so similar to the AMS algorithm that an O.R. participant (not one of us) noted it.

At the time, a co-author of the paper that is credited with "inventing" UCT turned to one of us to say, "We cited your paper." Sure enough, this was confirmed, and it was at that time that it was first revealed to the O.R. community that this algorithm was being used successfully in Go-playing computer programs. Most of the participants likely do not even remember this seemingly trivial exchange, as confirmed when one of us checked with the O.R. participant who had noted the algorithmic resemblance at the time. To find out more about the workshop, the web page for the workshop is still available at <http://castlelab.princeton.edu/nsfcsor.html>.

## Conclusion

AlphaGo represents a formidable AI achievement stemming from several research streams, for which the OR/MS community has played a role such as neural networks, learning algorithms, Monte Carlo tree search and multi-armed bandit models. What makes the work perhaps more impressive, at least on the surface, is that unlike IBM's DeepBlue and Watson, which were basically tuned to a single application – playing chess or playing Jeopardy, the machinery behind AlphaGo can be easily adapted to other games and applications. In other words, the general neural network/Monte Carlo tree search architecture can be viewed as a very general-purpose



tool for games or sequential decision-making under uncertainty, much like regression is used for the social sciences. However, the devil's in the details, in this case the selection of the appropriate "features" in the neural networks, so that there is still as much art as science at this point.

Finally, a broader theme that we touched upon is raising conscientiousness in the OR/MS community as to how better to promulgate our research to other communities such as CS/AI. Clearly, INFORMS is aware of this general challenge, as evidenced by its seizing the initiative in claiming a well-deserved piece of the analytics pie. **ORMS**

**Hyeong Soo Chang** ([hschang@sogang.ac.kr](mailto:hschang@sogang.ac.kr)) is a professor in the Department of Computer Science and Engineering at Sogang University in Seoul, South Korea.

**Michael C. Fu** ([mfu@umd.edu](mailto:mfu@umd.edu)) is the Ralph J. Tyser Professor of Management Science in the Robert H. Smith School of Business (with a joint appointment in the Institute for Systems Research, A. James Clark School of Engineering) at the University of Maryland, College Park. **Jiaqiao Hu** ([jqhu@ams.sunysb.edu](mailto:jqhu@ams.sunysb.edu)) is an associate professor in the Applied Mathematics and Statistics Department at Stony Brook University. **Steven I. Marcus** ([marcus@umd.edu](mailto:marcus@umd.edu)) is a professor in the Department of Electrical Engineering (with a joint appointment in the Institute for Systems Research) in the A. James Clark School of Engineering at the University of Maryland, College Park.

## REFERENCES

1. Wikipedia, "Deep Blue versus Garry Kasparov," accessed April 28, 2016.
2. "AlphaGo: Using Machine Learning to Master the Ancient Game of Go," Jan. 27, 2016; <https://googleblog.blogspot.nl/2016/01/alphago-machine-learning-game-go.html>.
3. Chang, H. S., Fu, M. C., Hu, J. and Marcus, S. I., 2005, "An Adaptive Sampling Algorithm for Solving Markov Decision Processes," *Operations Research*, Vol. 53, pp. 126-139.
4. <https://deepmind.com/alpha-go.html>, accessed May 11, 2016.
5. [deepmind.com](https://deepmind.com), accessed May 11, 2016.
6. Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D., 2016, "Mastering the game of Go with deep neural networks and tree search," *Nature*, Vol. 529 (Jan. 28), pp. 484-503.
7. Fu, M.C., 2016, "AlphaGo and Monte Carlo Tree Search: The Simulation Optimization Perspective," *Proceedings of the 2016 Winter Simulation Conference*, to appear (downloadable at <http://informs-sim.org> after mid-December).
8. Coulom, R., 2006, "Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search," *Computers and Games, 5th International Conference, CG 2006, Turin, Italy, May 29-31*.
9. Browne, C., Powley, E., Whitehouse, D., Lucas, S., Cowling, P.I., Rohlfshagen, P., Tavener, S., Perez, D., Samothrakis, S., and Colton, S., 2012, "A Survey of Monte Carlo Tree Search Methods," *IEEE Transactions on Computational Intelligence and AI in Games*, Vol. 4, No. 1, pp. 1-49.
10. Kocsis, L. and Szepesvári, C., 2006, "Bandit based Monte-Carlo planning," *Proceedings of the 17th European Conference on Machine Learning*, Berlin, Germany: Springer, 282-293.
11. Auer, P., Cesa-Bianchi, N., and Fisher, P., 2002, "Finite-time Analysis of the Multiarmed Bandit Problem," *Machine Learning*, Vol. 47, pp. 235-256.
12. Fu, M.C. and Barton, R.R., 2012, "O.R. and the National Science Foundation (part 2)," *OR/MS Today*, June, pp. 34-38.

## CALL FOR APPLICATIONS 2017

DEADLINE FOR APPLICATIONS IS MONDAY, OCTOBER 31, 2016.



Are you the best in OR/MS/Analytics education?

The UPS George D. Smith Prize is created in the spirit of strengthening ties between industry and the schools of higher education that graduate young practitioners of operations research. INFORMS, with the help of CPMS, will award the prize for the effective and innovative preparation of students to be good practitioners of operations research, management science, or analytics to an academic department or program.

The prize will include a trophy and \$10,000 award. The UPS George D. Smith Prize will be announced at the 2017 Edelman Gala at the INFORMS Conference on Business Analytics and Operations Research, in Las Vegas, NV.

For more information, questions can be sent to Robin Lougee, 2017 Smith Prize Chair at [rlougee@us.ibm.com](mailto:rlougee@us.ibm.com).



**2016 UPS Smith Prize Winners**  
Carnegie Mellon University, H. John Heinz III College,  
School of Information Systems and Management,  
and School of Public Policy and Management

academia **strengthening ties** industry

[www.informs.org/smithprize](http://www.informs.org/smithprize)

**ups george d smith prize**