

Przyrostowe uczenie sieci neuronowych w grze w go

Piotr Ćwiek

opieka prof. nzw. dr hab. Jacek Mańdziuk

Wydział Matematyki i Nauk Informatycznych
Politechnika Warszawska

Seminarium z Metod Inteligencji Obliczeniowej
23 czerwca 2010



Plan prezentacji

① Czym jest go?

Zasady gry

Ranking i system handicapów

Nieco poza zasady

② Komputerowe go

Motywacja

Rozwiązania

Uczenie przyrostowe

③ Skalowalne sieci neuronowe

Przegląd skalowalnych architektur

Modyfikacje podstawowego modelu MDRNN



Dzieje gry

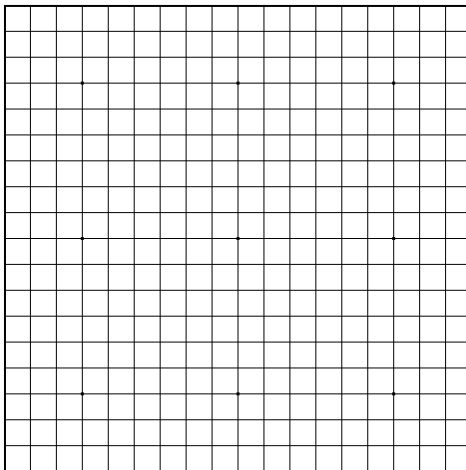
- pochodzenie: Chiny (*weiqi*)
- wzmianka na piśmie ok. IV w.p.n.e.
- rozrywka arystokracji
- + kaligrafia, malarstwo, guqin
- V w. Korea (*baduk*)
- VII w. Japonia (*go*)
- koniec XIX w. Zachód



[1] II w.p.n.e. – I w.n.e (Han)



Plansza, czyli *goban*



Plansza, czyli *goban* (c.d.)

- początki go: 17×17
- nauka: $9 \times 9 \rightarrow 13 \times 13 \rightarrow 19 \times 19$
- teoretycznie dowolny rozmiar...
- mniejsza plansza = prostsze zagadnienia
 - ale większość technik wspólna!

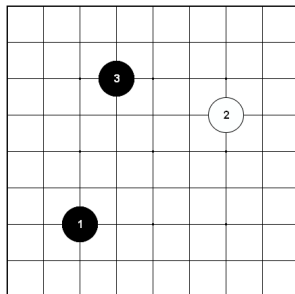


[2] współczesny (drogi) goban



Zasady i cel gry — ruchy

- dwóch graczy
- kamienie czarne i białe
- czarne zaczynają
- ruchy naprzemian
 - położenie kamienia na wolnym punkcie
 - kamienie nie są przesuwane
- można opuścić kolejkę
- wybór punktu niemal nieograniczony
- cel I — otaczanie terytorium

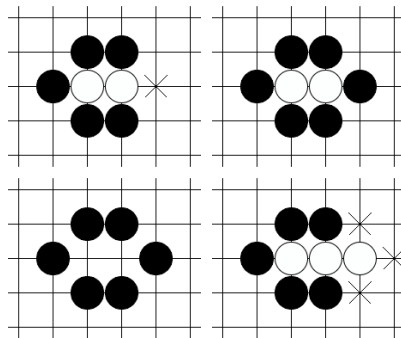


pierwsze trzy ruchy



Zasady i cel gry — „bicia”

- kamienie można usunąć z planszy
 - jeżeli nie mają *oddechów* (są martwe)
 - stają się jeńcami
- ruchy samobójcze zabronione
- cel II — otaczanie kamieni przeciwnika
- koniec gry:
 - brak sensownych ruchów (terytoria ustalone)
 - powierzchnia + jeńcy

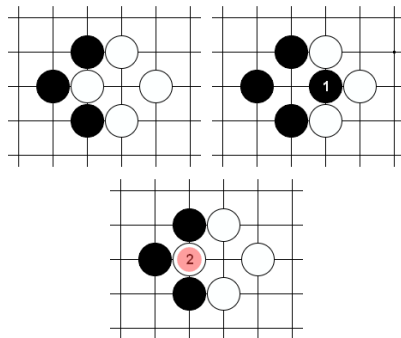


atari: złapanie i zabicie, albo... ucieczka



Zasady i cel gry — zasada ko

- ko — sytuacja cyklicznego bicia
- może to trwać wiecznie...
- zabronione jest natychmiastowe „odbijanie” ko
 - najpierw zagraj w inny punkt
- to wszystkie zasady!
 - to dopiero początek (kłopotów)!



ko: ruch 2 jest niedozwolony



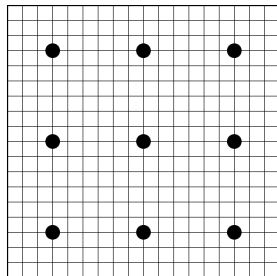
Zasada ko a problem cykli

- dwie z omówionych zasad zapobiegają cyklom
 - zakaz samobójstwa — cykle długości 1
 - zasada ko (ang. *basic ko*) — cykle długości 2
 - możliwe jednak cykle dłuższe
 - tzn. możliwe nieskończone gry
- możliwe zamienniki *basic ko*:
 - *superko* (pozycyjne lub sytuacyjne)
 - przerwanie gry (z wynikiem lub bez) w razie powtórzenia
- argumenty przemawiające za *basic ko*:
 - historia gry nie potrzebna
 - łatwe i szybkie do wykrycia
 - cykle dłuższe niż 2 rzadkie
 - możliwość ograniczenia liczby ruchów



Ranking i system handicapów

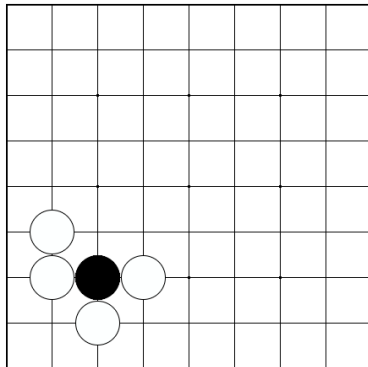
- stopnie *kyu*
 - im mniejszy, tym lepszy
 - 35k — zupełnie początkujący
 - 5k — przeciętny gracz klubowy
 - 1k — kandydat na mistrza (1d)
 - różnica w kyu wskazuje na liczbę kamieni *handicapu*
- stopnie *dan* (mistrzowskie)
 - im większy, tym lepszy
 - różnice coraz subtelniejsze
 - 6d — bardzo silny gracz
- *profesjonalne dan* (7p \approx 1d)



9 kamieni handicapu —
teraz ruch białego



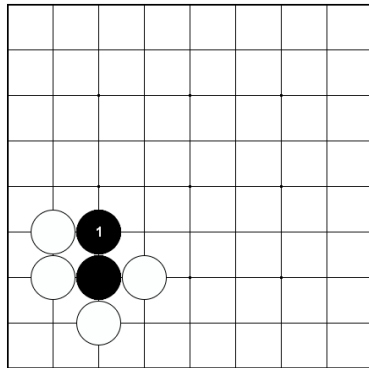
Drabinka



czarny ma tylko jeden oddech — to atari!



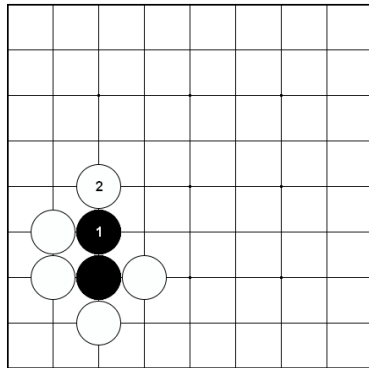
Drabinka



czarny ucieka...



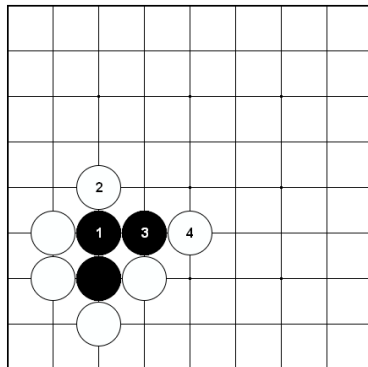
Drabinka



... a biały goni



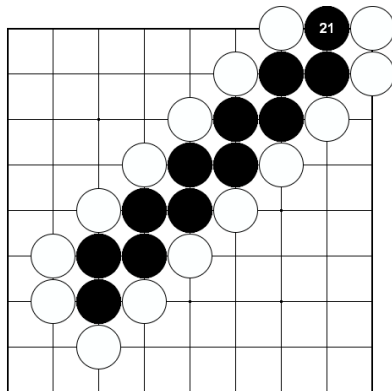
Drabinka



déjà vu? — to właśnie drabinka



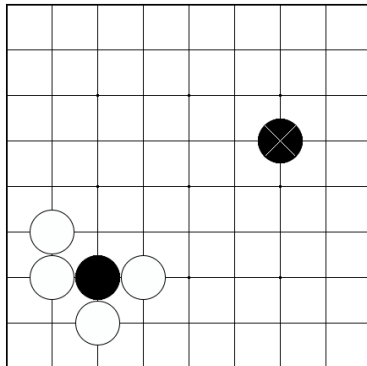
Drabinka



ucieczka czarnego była błędem



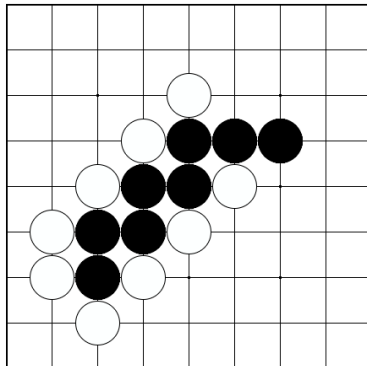
Drabinka



czy tu też nie należy uciekać?



Drabinka

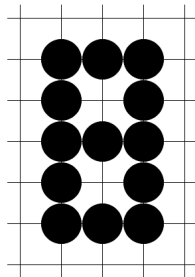


pościg białego był błędem!



Pojęcie „kształtu”

- dobrzy gracze lubią to słowo
 - ale co ono oznacza?
- czy kamienie po prawej mogą zostać zbite?
- nie! — one są żywe
- problem życia i śmierci
 - trudny dla początkujących
 - trudny dla komputerów. . .



kształt „dwa oka” (przykładowy)



Historia i motywacja dla komputerowego go

Zarys sytuacji

- lata 60: teoretyczne rozważania o go
- 1965: spostrzeżenia Dr. I.J. Gooda
 - nadal są aktualne!
- 1970: program A. Zobrista
 - wygrał z początkującym
- 2008: MoGo wygrał z człowiekiem (Kim)
 - oceniony na 2–3 dan
 - 9 handicapów

Motywacja

- rozwiązywanie szachów (1997)
- programy go wciąż bardzo słabe
 - około 10 kyu
- duża złożoność problemu
- pole do popisu dla „prawdziwej” AI
 - jak myśli człowiek?
- pomoc dydaktyczna w nauce go
- *Ing prize* 1,6 mln dolarów



Szachy a go

W go:

- średnio 200 wyborów ruchu (35 w szachach)
- średnio 200 ruchów w grze (60 w szachach)
- około 10^{170} możliwych pozycji (10^{44} w szachach)
- akcja w wielu miejscach („bitwy”)
- słabo zdefiniowana koncepcja „otwarcia”
- zdecydowanie o zakończeniu gry trudne (życie/śmierć)
- ruchy mogą mieć odległy skutek (drabinka!)



Szachy a go (c.d)

W go:

- człowiek często przewiduje dalej (niż w szachach)
 - i wężiej (selektywnie), np. przy drabince
- efekt horyzontu już na poziomie podstawowym
- odmienna percepcja
 - szachy — zhierarchizowane grupy (Chase, Simon)
 - go — przecinające się grupy (Reitman)
- niesprecyzowana koncepcja „kształtu” (wzorce!)
- przyswajanie słabości przeciwnika
 - dzięki handicapom
- większe znaczenie strategii
 - taktyczne zwycięstwa to za mało!
 - problem z oceną stanu planszy



Przykładowe programy

- wczesne rozwiązania (lata 70-te)
 - program Zobrista (1970) i modyfikacje
 - INTERIM.2 (Reitman-Wilcox, od 1972) ~27 kyu
 - ogólnie
 - złożone wielomodułowe systemy
 - duża liczba reguł
 - poszukiwanie wzorców
 - mała skuteczność
- bardziej współczesne
 - The Many Faces of Go (Fotland) ~9 kyu
 - dużo wiedzy eksperckiej (baza danych...)
 - Go4++ (Reiss)
 - porzucenie złożonego systemu regułowego
 - autor nie był zbyt silnym graczem
 - Handtalk (Chen)
 - pisany w asemblerze: mały i szybki



Stosowane techniki (ogólnie)

- *brute-force* + heurystyki
 - bardzo duży rozmiar drzewa gry
 - sensowne dla plansz 9×9
- systemy eksperckie (regułowe, bazy wiedzy)
 - często dostrojone do mistrzowskich technik
 - ale nie mogą przewidzieć wszystkiego
 - obciążone wiedzą ekspercką twórców
- metody Monte-Carlo
 - wiele (prawie) losowych rozgrywek
 - nie wymagają wiedzy eksperckiej
 - ostatnio bardzo popularne
- sieci neuronowe
 - dobre w rozpoznawaniu wzorców
 - mogą się uczyć
 - ograniczona potrzeba wiedzy eksperckiej
- większość systemów to złożone hybrydy



Uczenie przyrostowe

- idea uczenia przyrostowego:
 - naucz się grać na małej planszy
 - przenieś zdobytą wiedzę na dużą planszę
 - uzupełnij braki w wiedzy (doucz się)
- motywacja
 - szybkie uczenie się małych problemów
 - mechanizm wykorzystywany przez ludzi
- skalowalna architektura sieci neuronowej
 - przetwarza dane z gobanu dowolnego rozmiaru
 - wykazuje **korelację** pomiędzy wynikami na różnych rozmiarach



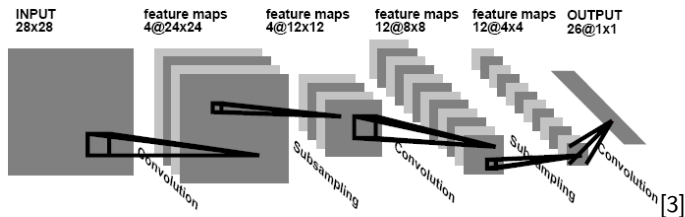
Cel pracy

- 1 projekt sieci neuronowej potrafiącej zastosować nabytą wiedzę do planszy o rozmiarze większym niż treningowy
- 2 implementacja i przetestowanie architektury, w szczególności
 - efektywność nauki przyrostowej
 - zakres koniecznego douczenia sieci



Sieci konwolucyjne

- silne podstawy biologiczne (np. oko)
- jednostka przetwarzająca — sieć neuronowa:
 - wejście: ograniczony obszar czasowy/przestrzenny
 - skanuje całe wejście produkując *mapę cech*
 - kilka różnych jednostek → kilka map cech
- mapa cech jest wejściem dalszych warstw



Sieci konwolucyjne, c.d.

- typowe zastosowanie: analiza mowy, pisma, obrazu
- zalety w stosunku do tradycyjnego MLP:
 - mała liczba parametrów (współdzielenie wag)
 - szybsza nauka
 - mniejsze ryzyko przeuczenia
 - lokalna ekstrakcja cech (ograniczone „okno”)
 - wychwytywanie korelacji czasowych/przestrzennych
 - dowolnie duże wejście
 - prostota implementacji sprzętowej
- problem:
 - dowolnie duże wejście \rightarrow dowolnie duże wyjście
 - konieczność dodatkowego postprocesingu (jakiego?)



„Błądzące oko” (*roving eye*)

- inspiracja systemami *active vision* znanymi z robotyki
 - sensory rejestrują jedynie ograniczony obszar
 - dodatkowy mechanizm kontroluje ruch sensora
 - decyzja podejmowana jest po odpowiedniej liczbie ruchów
 - prawa ruchu mogą być przedmiotem nauki
- błądzące oko to sieć neuronowa posiadająca stan:
 - pozycja (punkt gobanu)
 - orientacja (jeden z czterech obrotów)
- wejście sieci:
 - fragment gobanu 3×3 (informacja o kamieniach)
 - liczba wszystkich kamieni po każdej ze stron oka
 - informacja o bieżącej pozycji na gobanie
 - legalność ruchu w bieżącej pozycji (ko)



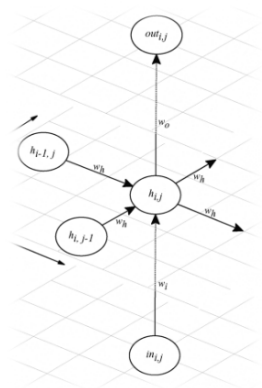
„Błądzące oko” (*roving eye*), c.d.

- wyjście pozwala podjąć decyzję:
 - zagranie w bieżącej pozycji lub spasowanie
 - przesunięcie oka do przodu
 - obrót oka w lewo lub w prawo
 - dodatkowy krok czasowy w tym samym stanie (pauza)
- ograniczenie na czas „myślenia” (100 kroków czasowych)
- połączenia rekurencyjne stanowią pamięć oka
- trenowanie sieci strategią ewolucyjną NEAT
- zalety:
 - możliwość analizy całego gobanu przy małej liczbie parametrów
- problemy:
 - ograniczony obszar widzenia
 - złożone zadanie decyzyjne



Multidimensional Recurrent Neural Network (MDRNN)

- uogólnienie tradycyjnych sieci rekurencyjnych
 - zastąpienie jednego (czasowego) sprzężenia zwrotnego wieloma (np. przestrzennymi)
- szczególny przypadek ogólnych sieci DAG-RNN, w których:
 - zmienne są wierzchołkami skierowanego grafu acyklicznego
 - krawędziami grafu są sieci typu *feed-forward*
 - wagi sieci mogą być współdzielone

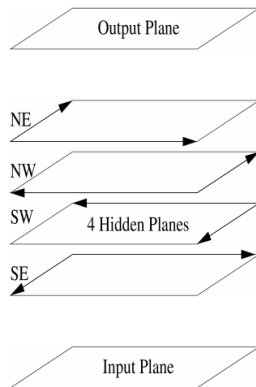


[4]



Sieci MDRNN, c.d.

- element aktywny jest siecią neuronową
 - umieszczony nad określonym punktem gobanu
 - pobiera wejście z tego punktu
 - i wyjście z dwóch poprzednich położen elementu
 - skanuje goban w czterech kierunkach
 - warstwa wyjściowa podsumowuje
- wyjściem jest *mapa preferencji*
- zalety:
 - mała liczba parametrów
 - brak problemu doboru wielkości pola widzenia

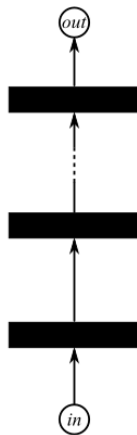


[5]

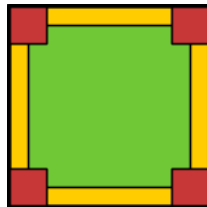
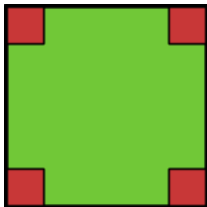
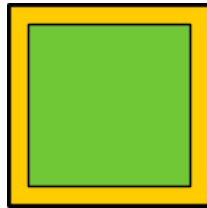


Budowa wielowarstwowa

- wejście to jeden punkt gobanu
- warstwa to sieć typu *feed-forward*
 - nazywana *elementem aktywnym*
- każda warstwa może być inną siecią
- a nawet maksymalnie trzema różnymi sieciami. . .

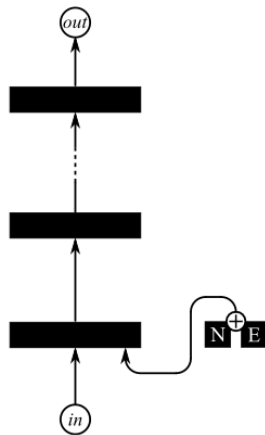


Różnicowanie elementów aktywnych wg obszarów gobanu



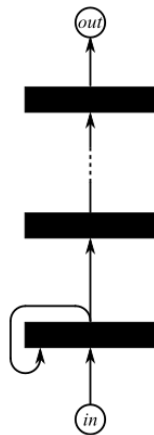
Rekurencja przestrzenna

- oryginalna cecha sieci MDRNN
- warstwa pobiera własne wyjście ze swoich dwóch poprzednich pozycji nad gobanem (np. NE)
- wymagane cztery skany gobanu
- cztery wyjścia warstwy są sumowane (symetria)



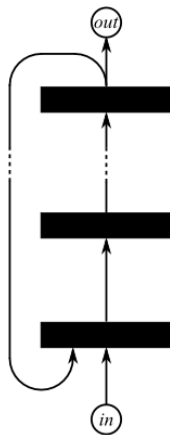
Rekurencja czasowa

- warstwa pobiera własne wyjście z tej samej pozycji nad gobanem, ale z poprzedniego kroku czasowego
- wejściem jest seria gobanów (gra)
 - (nie statyczny obraz gobanu)
- pozwala uchwycić kontekst czasowy



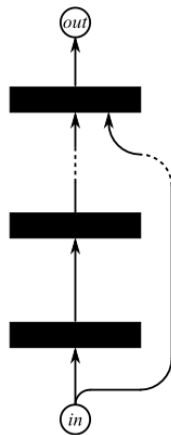
Globalna rekurencja czasowa

- warstwa pobiera wyjście warstwy ostatniej z tej samej pozycji nad planszą, ale z poprzedniego kroku czasowego
- udostępnia sieci jej własną decyzję



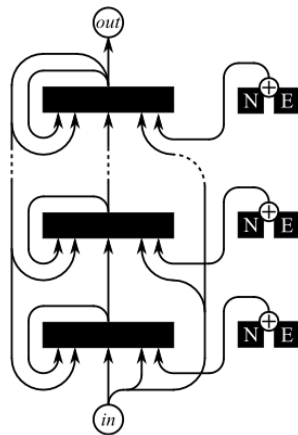
Skrót do wejścia

- warstwa pobiera wejście warstwy pierwszej z tej samej pozycji nad planszą, w tym samym kroku czasowym
- warsta ma bezpośredni dostęp do gobanu



Opcjonalny charakter rozszerzeń

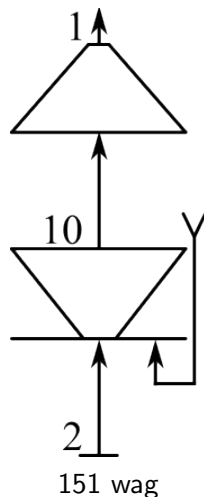
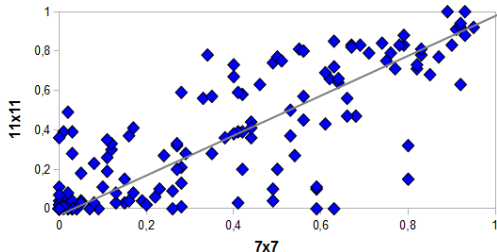
- Każda warstwa *może* skorzystać z dowolnego z przedstawionych mechanizmów niezależnie od pozostałych.
- Jedynie wejście z warstwy poprzedniej jest obowiązkowe.



Wstępne badanie skalowalności

Współczynniki korelacji Pearsona:

- 7×7 vs. $9 \times 9 \rightarrow 0,9$
- 9×9 vs. $11 \times 11 \rightarrow 0,95$
- 7×7 vs. $11 \times 11 \rightarrow 0,86$



Dziękuję za uwagę!



Źródła I

- 1 Arthur, Ch. (2006). Computers just can't seem to get past Go. (The Guardian). Pobrano z <http://www.guardian.co.uk/technology/2006/aug/03/insideit.guardia>
- 2 Baldi, P., Pollastri, G. (2003). The Principled Design of Large-scale Recursive Neural Network Architectures DAG-RNNs and the Protein Structure Prediction Problem W: *Journal of Machine Learning Research* 4, str. 575–602
- 3 Borrell, B. (2006). AI Invades Go Territory. (Wired). Pobrano z <http://www.wired.com/science/discoveries/news/2006/09/71804>
- 4 Burmeister, J., Wiles J. (1995?). An Introduction to the Computer Go Field and Associated Internet Resources. (CS-TR-339 Computer Go Tech Report). Pobrano z <http://www.itee.uq.edu.au/~janetw/ComputerGo/CS-TR-339.html>
- 5 Hsu, F. (2007). Cracking Go. (IEEE Spectrum). Pobrano z <http://spectrum.ieee.org/computing/software/cracking-go>



Źródła II

- 6 Lecun, Y., Bengio, Y. (1995). Convolutional Networks for Image, Speech and Time Series, str. 255–258. MIT Press
- 7 Lubos, J. (2006). *Zagraj ze mną w go*. Ponnuki, Bielsko-Biała
- 8 Myers, B. (2000?). Overview of Computer Go. (Intelligent Go Foundation). Pobrano z [http://www.intelligentgo.org/Home/Overview of Computer Go.html](http://www.intelligentgo.org/Home/Overview%20of%20Computer%20Go.html)
- 9 Schaul, T., Schmidhuber, J. (2008). A Scalable Neural Networks for Board Games W: *Proceedings of the IEEE Symposium Computational Intelligence in Games*. IEEE Press
- 10 Schaul, T., Schmidhuber, J. (2009). Scalable Neural Networks for Board Games W: *Artificial Neural Networks – ICANN 2009*, str. 1005–1014. Springer Berlin / Heidelberg
- 11 Stanley, K.O., Miikkulainen, R. (2004). Evolving a Roving Eye for Go W: Deb, K., i.in. (red.) *GECCO 2004*. LNCS, vol 3103, str. 1226–1238. Springer



Źródła III

- 12 Wu, L., Baldi, P. (2007). A Scalable Machine Learning Approach to Go W: Schölkopf, B., Platt, J., Hoffman, T. (red.) *Advances in Neural Processing Systems 19*, str. 1521–1528. MIT Press
- 13 Computer Beats Pro At US Go Congress. (2008). (American Go Association). Pobrano z http://www.usgo.org/index.php?%23_id=4602
- 14 Computer Go. (Wikipedia, the free encyclopedia). Pobrano z http://en.wikipedia.org/wiki/Computer_Go
- 15 Cycle. (Sensei's Library). Pobrano z <http://senseis.xmp.net/?Cycles>
- 16 Go (game). (Wikipedia, the free encyclopedia). Pobrano z [http://en.wikipedia.org/wiki/Go_\(board_game\)](http://en.wikipedia.org/wiki/Go_(board_game))
- 17 History of Go. (Wikipedia, the free encyclopedia). Pobrano z http://en.wikipedia.org/wiki/History_of_Go
- 18 Superko. (Sensei's Library). Pobrano z <http://senseis.xmp.net/?Superko>



Źródła ilustracji

Numery w poniższej liście odpowiadają liczbom w nawiasach kwadratowych, znajdującym się przy odpowiednich ilustracjach.

- 1 <http://cswnet.blog.sohu.com/39470351.html>
- 2 Internet
- 3 Lecun, Y., Bengio, Y. (1995). Convolutional Networks for Image, Speech and Time Series, str. 255–258. MIT Press
- 4 Schaul, T., Schmidhuber, J. (2008). A Scalable Neural Networks for Board Games W: *Proceedings of the IEEE Symposium Computational Intelligence in Games*. IEEE Press
- 5 Wu, L., Baldi, P. (2007). A Scalable Machine Learning Approach to Go W: Schölkopf, B., Platt, J., Hoffman, T. (red.) *Advances in Neural Processing Systems 19*, str. 1521–1528. MIT Press

