

XXII Seminarium

ZASTOSOWANIE KOMPUTERÓW W NAUCE I TECHNICIE' 2012

Oddział Gdański PTETiS

Referat nr 11

ZASTOSOWANIE KOMPUTERÓW W DZIEDZINIE WYSZUKIWANIA
STRATEGII OPTYMALNYCH W GRACH LOGICZNYCH

Tomasz GOLUCH¹

1. Politechnika Gdańska, Wydział Elektroniki Telekomunikacji i Informatyki, Katedra Algorytmów
i Modelowania Systemów

tel: +48 58 347 22 68

fax: +48 58 347 17 66

e-mail: goluch@eti.pg.gda.pl

Streszczenie: Problem jaki stanowi wyszukiwanie strategii optymalnej w grach logicznych jest bardzo złożony. Można go podzielić na następujące podproblemy: obliczeniowy, pamięciowy oraz operacji wejścia/wyjścia. Jednak rosnąca z roku na rok siła obliczeniowa komputerów, ilość pamięci oraz prędkość transferu danych pomiędzy podzespołami zarówno lokalnymi jak i rozproszonymi, a także wzrost skuteczności wykorzystywanych technik algorytmicznych powoduje, że komputery zaczynają grać doskonale, realizując strategie optymalne. W pierwszej części tej pracy przedstawiono początkowe próby skonstruowania inteligentnego sztucznego gracza. W kolejnym punkcie przedstawiono w układzie chronologicznym większość nietrywialnych, rozwiązanych gier. Trzeci punkt dotyczy aktualnego stanu wiedzy oraz perspektyw co do możliwości rozwiązania najbardziej popularnych gier.

Słowa kluczowe: rozwiązywanie gier, sztuczna inteligencja, gra logiczna.

1. POCZĄTKI SZTUCZNYCH GRACZY

Ślady pierwszej realizacji pragnień ludzkich do stworzenia inteligentnej maszyny możemy odnaleźć już w 1769 roku. Wtedy to W. von Kempelen pierwszy raz zademonstrował słynnego Turka grającego w szachy. Niestety jego kluczowym elementem był ludzki umysł co dyskwalifikuje niniejsze rozwiązanie do nadania mu miana „sztucznego”.

Następnym – tym razem poprawnym – rozwiązaniem (1914 r.) był automat o nazwie El Ajedrecista (hiszp. gracz szachowy) – rozgrywający końcówkę szachową KRK – król i wieża kontra król. Automat zawsze matował chociaż nie zawsze w minimalnej liczbie posunięć.

Podwaliny sztucznej inteligencji do gry w szachy (i nie tylko) stworzył C. Shannon prezentując algorytm min-max wraz z koncepcją funkcji oceniającej. Dokonał również prostej estymacji przestrzeni stanów tej gry oceniając ją na 10^{46} [1].

Rok później A. Turing z braku odpowiedniej maszyny rozegrał partię szachów grając z algorytmem własnego pomysłu zapisanym na kartce – oczywiście wygrał tę partię.

Pierwszy program komputerowy który pretendował do miana sztucznej inteligencji grał w warcaby i został z sukcesem uruchomiony w 1951 roku przez Christophera Strachey'a na pierwszym komercyjnym komputerze

ogólnego przeznaczenia Ferranti Mark 1. Był to komputer o architekturze von Neumanna i wykorzystywał 20-bitowe słowa. Operacje arytmetyczne były wykonywane na podwójnych 40-bitowych słowach. Czas wykonania instrukcji innej niż mnożenie zajmował 1,2 milisekundy (833Hz). Jako pamięć operacyjną służyło osiem tub Williamsa-Kilburna, każda zdolna pomieścić 64 słowa 20 bitowe. Pamięć trwałą stanowiły dwa bębny magnetyczne o pojemności 64 strony po 64 słowa 20 bitowe. Latem 1952 roku program potrafił rozgrywać kompletną partię warcabów w rozsądnym czasie.

Na tym samym komputerze w tym samym roku (1951) Dietrich G. Prinz uruchomił swój pierwszy program rozgrywający końcówki partii szachowych – Mate-in-two.

Pierwszym komputerem zaprojektowanym wyłącznie do gry NIM¹ był NIMROD zaprezentowany w Angli na Wystawie Nauki „The Festival of Britain” w 1951 r. Wykorzystywał prosty algorytm pozwalający na wykonywanie „bezpiecznych posunięć” gwarantujących mu zwycięstwo, warunkiem wygranej była odpowiednia pozycja startowa albo błędne posunięcie przeciwnika podczas prowadzenia rozgrywki. W skład jednostki obliczeniowej komputera wchodziło 350 lamp 12AT7 (podwójne triody). Pobór mocy wynosił 6 kilowatów z czego 4 kilowaty przypadało na zasilanie lamp.

W 1952 roku Alexander S. Douglas napisał program OXO na komputer ESDAC grający w grę kółko i krzyk. ESDAC wykorzystywał 18-bitowe słowa z czego tylko 17 bitów było używanych. W skład procesora wchodziło 3500 lamp elektronowych a średni czas instrukcji zajmował 1,5 ms aczkolwiek mnożenie trwało 4 razy dłużej. Pamięć operacyjna składała się z 32 rzęciowych linii opóźniających z których każda przechowywała 32 18-bitowe słowa co było równoważne 2kB pamięci. Była to pierwsza gra komputerowa wykorzystująca interfejs graficzny do interakcji z użytkownikiem.

Podsumowując osiągnięcia tamtej epoki widać, że wyzwaniem był raczej sam fakt że maszyna robiła coś inteligentnego niżeli poziom tych działań. Jednak były to początki prowadzące do zaskakujących rezultatów

¹ NIMROD potrafił grać zarówno w tradycyjną jak i odwróconą wersję tej gry.

owocujących przejmowaniem tytułów mistrzowskich w wielu liczących się grach logicznych a nawet, co jest tematem tego artykułu, znajdowaniem strategii optymalnych a to z kolei oznacza, że komputer gra doskonale.

2. GRY ROZWIĄZANE

W tablicy 1. w porządku chronologicznym przedstawiono rozwiązane dotychczas nietrywialnie gry deterministyczne dwóch graczy z pełną informacją o sumie zerowej.

Pierwszą z nich, do rozwiązania której potrzebny był komputer, był Qubic. W roku 1980 O. Patashnik wykorzystując 1500 godzin pracy komputera w laboratorium firmy Bell obliczył strategię optymalną. Było to możliwe ponieważ mimo dużej przestrzeni stanów 10^{30} w jej skład wchodziło jedynie 2928 posunięć [2,10].

Drugą była gra Czwórki charakteryzująca się niewielką złożonością stanów 10^{14} . W 1988 roku J. D. Alenowi zabrakło to 300 godzin pracy komputera [3,10].

W 1992 roku została rozwiązana przez V. Alisa gra Go-moku. Obliczenia odbywały się równoległe na 11 stacjach SUN SPARC, a każda z nich była wyposażona od 64 do 128 MB pamięci oraz procesor o mocy obliczeniowej od 16 do 28 MIPS. Rozwiązane zostały dwie wersje tej gry free i standard obliczenia zajęły odpowiednio 11,9 i 15,1 dni [4,10].

Rok później R. Gasser rozwiązał grę młynek. Była to pierwsza gra której strategia optymalna zapewniała jedynie osiągnięcie remisu przy optymalnej grze przeciwnika. Bazy danych wymagane do znalezienia tej strategii obliczane były w przeciągu czterech lat (1989-1993). Do obliczeń wykorzystano takie komputery jak: Macintosh Iix, Macintosh Quadra 800, Cray X-MP/28, DEC VAX 9000-420, IBM RS/6000, 16-Transputer (T805) system, 30-Processorowy MUSIC system (Motorola DSP 96000), DEC 3000 (Alpha). Niestety wiele z nich albo wykonywało obliczenia na niskim priorytecie (wieloużytkownikowe środowisko) albo były zoptymalizowane do przetwarzania operacji zmiennoprzecinkowych które są prawie niewykorzystywane podczas rozwiązywania gier. Weryfikacja wyniku na klastrze złożonym z 30 komputerów Sun Sparcstations zajęła trzy miesiące [5,10].

W 1996 roku H. K. Orman rozwiązał grę Pentomino wykorzystując 64-bitowy komputer 175 MHz DEC Alpha przez dwa tygodnie. Natomiast weryfikacja zajęła pięć dni pracy komputera Sun IPC Sparcstation [6,10].

W 1998 roku L. Goossens wykazał, że gra Quarto w wersji twist zawsze kończy się remisem przy optymalnej grze obydwu graczy [7].

W tym samym roku G. Stelle rozwiązał wszystkie 16 standardów gry Teeko charakteryzującej się jednak bardzo małą złożonością stanów gry [8].

W 2000 roku J. Donkers i inni wykorzystując standardowy komputer PC 300 Mhz Pentium II rozwiązali grę Dakon do wersji 18. Znalezienie otwarcia wygrywającego² zajęło jedynie kilka sekund. Co ciekawe otwarcia wygrywające oryginalnej wersji Dakon-6 znanej pod nazwą Ohvalhu zostały znalezione przez graczy z Malediw bez użycia komputera [9,10].

W tym samym roku J. Donkers i inni rozwiązali grę Domineering rozgrywaną na planszy o rozmiarach 8×8 pól. Obliczenia zajęły 600 godzin pracy komputera [10,11].

Tablica 1. Lista rozwiązanych gier logicznych

Gra	Rok	Złożoność stanów	Złożoność drzewa gry	Wynik gracza rozp.	Ref
Qubic	1980	10^{30}	10^{34}	wygrana	[2,10]
Czwórki	1988	10^{14}	10^{21}	wygrana	[3,10]
Go-Moku	1992	10^{105}	10^{70}	wygrana	[4,10]
Młynek	1993	10^{10}	10^{50}	remis	[5,10]
Pentomino	1996	10^{12}	10^{18}	wygrana	[6,10]
Quarto Twist	1998	10^{27}	b.d.	remis	[7]
Teeko - Standard	1998	10^8	b.d.	remis	[8]
Teeko - Advanced	1998	10^8	b.d.	wygrana	[8]
Dakon-6	2000	10^{15}	10^{33}	wygrana	[9,10]
Domineering 8×8	2000	10^{15}	10^{27}	wygrana	[10,11]
Kalah	2000	10^{13}	10^{18}	wygrana	[10,12]
Renju	2001	10^{105}	10^{70}	wygrana	[10,13]
Avari	2002	10^{12}	10^{32}	remis	[10,14]
Domineering 10×10	2002	10^{21}	10^{42}	wygrana	[15]
Quarto	2003	10^{27}	b.d.	przeigrana	[7]
Warcaby angielskie	2007	10^{21}	10^{31}	remis	[10,16]
Fanorona	2007	10^{22}	10^{46}	remis	[17]
Tygrysy i kozy	2007	10^{10}	10^{31}	remis	[18]
Trzej muszkieterowie	2009	b.d.	b.d.	wygrana	[19]

W 2000 roku G. Irving i inni rozwiązali grę Kalah. Do rozwiązania użyto komputera z procesorem AMD Athlon 800 MHz oraz pamięcią 256 MB. Obliczenia zajęły 4,7 godziny [10,12].

W 2001 roku J. Wagner i I. Virag rozwiązali grę Renju. Obliczenia zostały przeprowadzone na standardowym komputerze PC Pentium 200 MHz i zajęły około 3000 godzin a procedura weryfikacyjna zajęła kolejne 6000 godzin [10,13].

Rok później H. Bal i J. Romein rozwiązali grę Avari używając równoległej jednostki obliczeniowej, składającej się z 72 dwuprocessorowych jednostek produkcji IBM, połączonych szybką siecią Myrinet 2Gb/s (full-duplex). Łącznie wykorzystali 144 procesory Pentium III 1.0 GHz, 72 GB pamięci operacyjnej oraz przestrzeń dyskową o powierzchni 1,4 TB. Rozwiązanie zajęło zaskakująco mało czasu pracy komputera, bo jedynie 51 godzin [10,14].

W tym samym roku N. Bullock ustalił wartość gry³ Domineering 10×10 wykorzystując komputer PC Pentium III 600-900 MHz [15].

Rok później A. Holshouser i H. Reiter bez użycia komputera podali dowód na wartość gry Quarto w wersji standard oraz podali strategię wygrywającą drugiego gracza, która jest w 99.99% skuteczna przeciw naiwnej grze gracza rozpoczynającego [7].

² Otwarcie wygrywające to takie w którym gracz rozpoczynający wygrywa grę w pierwszej turze, nie pozwalając na wykonanie ruchu przeciwnikowi.

³ Wartość gry to skrót od wartości gry w oparciu o teorię gier. Jest to wynik gry jaki osiągną gracze podczas realizowania swoich optymalnych strategii.

Najbardziej spektakularnym osiągnięciem w dziedzinie wyszukiwania strategii optymalnych było rozwiązanie w 2007 roku przez J. Schaeffera Warcabów angielskich. Magazyn Science umieścił to odkrycie na dziesiątym miejscu największych odkryć tego roku. Było to ukoronowaniem 18 lat pracy autora nad tym problemem. W szczytowym momencie obliczeń (1992 rok) ponad 200 procesorów było wykorzystywanych jednocześnie. Obliczenia składały się z dwóch części. Pierwsza to baza końcówek wszystkich pozycji poniżej 10 bierek, która zawierała 39 miliardów pozycji skompresowanych do 237 GB (średnio 154 pozycje na 1 Bajt). Natomiast druga to drzewo dowodzące które zawierało „jedynie” 10^7 pozycji, jego zbudowanie wymagało przejrzenia 10^{14} pozycji. Najdłuższa analizowana ścieżka tego drzewa miała głębokość 154 posunięć. Średnia liczba użytych procesorów do obliczeń to siedem z 1,5 do 4 GB pamięci operacyjnej [16].

W tym samym roku M. Schadd i inni rozwiązali Fanaronę – narodową grę Madagaskaru. Rozwiązanie zajęło miesiąc czasu pracy komputera PC z procesorem Pentium IV 3.0 GHz i 256 MB pamięci operacyjnej [17].

Kolejną grą rozwiązana w tym samym roku była starożytna gra z Nepalu znana pod nazwą Bagh-Chal lub Tygrysy i Kozy. Autorem rozwiązania był Y. L. Lim, który wykorzystał tydzień obliczeń klastra złożonego z 48 komputerów Dual Pentium 3 oraz 2 miesiące pracy 10 komputerów Pentium 4 Xeon [18].

W 2009 roku J. Laire rozwiązał stosunkowo (jak na ten rok) prostą grę – Trzej muszkieterowie [19].

3. PERSPEKTYWY ROZWIĄZYWANIA GIER

Grą która od dłuższego czasu jest na granicy rozwiązania jest Reversi znane również pod nazwą Othello. Już w 1993 roku J. Feinstein rozwiązał tę grę na planszy o wymiarach 6×6 [19].

Kolejnym wyzwaniem są Warcaby międzynarodowe (polskie) są one dużo trudniejsze do rozwiązania od angielskich z powodu większych rozmiarów planszy 10×10 (większa przestrzeń stanów gry) oraz większej mobilności i liczby bierek (większy współczynnik rozwidlenia drzewa gry). Dotychczas udało się opracować bazę końcówek maksymalnie dla ośmiu bierek na warcabnicy [21].

Jedną z ważniejszych gier w punktu widzenia ich rozwiązywania jest Hex. W połowie zeszłego wieku J. Nash podał dowód na istnienie strategii wygrywającej gracza drugiego dla wersji z *swap rule*. Ta gra została rozwiązana dla mniejszych plansz włącznie z planszą o wymiarach 9×9 . Jednak do 2010 roku to ludzie przodowali w wyszukiwaniu optymalnych otwarć.

Dopiero w 2010 roku B. Arenson przedstawił 28 z 41 możliwych otwarć. Rozwiązanie wszystkich ma mu zająć około 870 dni (2013 r.). Sugeruje on również, że około 2020 roku powinien już poznać przynajmniej jedno otwarcie dla oryginalnych rozmiarów tej gry (11×11) co było by jednoznaczne z jej rozwiązaniem [22].

Kolejną grą są Szachy, aby uświadomić sobie jak ciężkie są do rozwiązania należy wspomnieć, że największe bazy końcówek jakie dotychczas policzono zawierają maksymalnie do 6 figur na planszy⁴, a maksymalne wymiary planszy na jakich udało się rozwiązać wszystkie możliwe

układy to 3×4 . Dokonał tego K. Kryukov w 2009 roku a kompletna baza wszystkich możliwych ustawień, od 3 do 12 bierek zajmowała 167 GB. Dla planszy o rozmiarach 4×4 rozwiązał on wszystkie partie do maksymalnie 9 bierek na planszy. Najdłuższa możliwa, optymalna rozgrywka może składać się z 59 posunięć [23].

Najtrudniejszą do rozwiązania, powszechnie znaną grą, jest Go. W 2003 roku E. van der Werf i inni rozwiązali jej mini wersję na planszy o rozmiarach 5×5 . Do obliczeń wykorzystywano komputer PC Pentium IV 2.0 GHz. W zależności od wersji gry obliczenia trwały od 2,7 do 9,7 godzin [24]. W 2009 rozwiązał on wraz z M. H. M. Winandsem wszystkie wersje Go na prostokątnych planszach do 30 pól. Najwięcej czasu, bo miesiąc, zabrało rozwiązanie wersji 3×10 . Do obliczeń używano komputera z procesorem Intel Core 2 3GHz. Rozwiązanie wersji 5×5 zajęło już tylko 14,3 minuty. Według optymistycznych ekstrapolacji plansza o wymiarach 6×6 powinna zostać rozwiązana do 2015 roku [25].

4. WNIOSKI KOŃCOWE

Komputery już dawno przegoniły arcymistrzów prawie we wszystkich grach logicznych, deterministycznych dwóch graczy. W tym artykule przedstawiono większość gier w które rozgrywka z komputerem nie ma sensu, ponieważ albo zawsze będzie on wygrywał, albo (w przypadku kiedy nie posiada strategii wygrywającej) będzie czyhał na nasz jeden błąd w celu przejścia inicjatywy. Jeżeli postęp w mocy obliczeniowej komputerów oraz wydajność wykorzystywanych algorytmów będzie nadal rosnąć to możemy być pewni, że kolejne gry będą tracić swoje strategiczne tajemnice i jedynym sposobem na umożliwienie czerpania człowiekowi z nich przyjemności podczas gry z komputerem będzie celowe obniżenie jego umiejętności.

5. BIBLIOGRAFIA

1. Shannon C. E., Programming a computer for playing chess. Philosophical Magazine, 7th series, 41(314) s. 256-75, 1950.
2. Patashnik O., Qubic: 4x4x4 tic-tac-toe, Mathematics Magazine, (53) s. 202-216, 1980.
3. Allen J., A note on the computer solution of Connect-Four, D.N.L. Levy, D.F. Beal (Eds.), Heuristic Programming in Artificial Intelligence: the First Computer Olympiad, Ellis Horwood, Chichester, s. 134-135, 1989.
4. Allis L. V., Herik, H.J. van den, Huntjens M. P. H., Go-Moku and Threat-Space Search, Rap. tech. CS 93-02, Department of Computer Science, Faculty of General Sciences, Rijksuniversiteit Limburg, Maastricht, Netherlands 1993.
5. Gasser R., Solving Nine Men's Morris, Games of No Chance (ed. R. Nowakowski), MSRI Publications (29), s. 101-113. Cambridge University Press, Cambridge, 1996.
6. Orman H. K., The Game of Pentominoes: A First Player Win, Games of No Chance, 29, s. 339-344, Cambridge University Press, 1994.
7. Holshouser A., Reiter H., Mathematics and Informatics Quarterly, Quarto without the Twist, 16(2) s. 50-55, 2005.

⁴ Sześć figur z dwoma królami włącznie.

8. Weisstein, E. W., "Teeko." MathWorld-A Wolfram Web Resource. URL <http://mathworld.wolfram.com/Teeko.html>.
9. Donkers, H. H. L. M., Voogt, A. J. de, Uiterwijk, J. W. H. M. Human versus Machine Problem-Solving: Winning Openings in Dakon, Board Games Studies; 3: 79-88, 2000.
10. Herik, H.J. van den, Uiterwijk, J.W.H.M., Rijswijk, J. van. Games Solved: Now and in the Future. Artificial Intelligence, Vol. 134, Nos. 1-2, s. 277-311, 2002, ISSN 0304-3975.
11. Breuker D., Uiterwijk J., Herik J. van den. Solving 8×8 Domineering. Theoretical Computer Science, (230), p. 195-206, 2000, ISSN 0304-3975.
12. Irving, G., Donkers, H. H. L. M., Uiterwijk, J. W. H. M., Solving Kalah. ICGA Journal, Vol. 23, (3), s. 139-148, 2000.
13. Wagner, J., Virag, I.: Solving Renju, ICGA Journal, 24 (1), 30-34, 2001.
14. Romein, J.W. and Bal, H.E., Awari is Solved. ICGA Journal, 25(3), s. 162-165, 2002.
15. Yew J. L., On Forward Pruning in Game-Tree Search. Praca magisterska, National University of Singapore, 2007.
16. Schaeffer J., Checkers Is Solved, Science, 317(5844) s. 1518-1522 2007.
17. Schadd M. P. D., Winands M. H. M., Uiterwijk J. W. H. M., Herik H. J. van den, Bergsma M. H. J., Best Play In Fanorona Leads To Draw, New Mathematics and Natural Computation, 4(03) s. 369-387, 2008.
18. Jin L. Y., Nievergelt J., Computing Tigers and Goats, ICGA Journal, 27(3), s. 131-141. ISSN 1389-6911, 2004.
19. Sackson S., A Gamut of Games, Hutchinson, London, 1982.
20. Sakuta M., Iida H., The Performance Of Pn*, Pds, And Pn Search On 6x6 Othello And Tsume-Shogi, Advances in Computer Games, 9 s. 203-222 2001.
21. Grimminick M. URL <http://www.xs4all.nl/~mdgsoft/draughts/stats/index.html>.
22. Arneson B., Hayward, R.B., Henderson, P., Solving Hex: Beyond Humans. Computers and Games, s. 1-10, 2010.
23. Kryukov K., 4x4 Chess, URL <http://kirill-kryukov.com/chess/4x4-chess/about.html>
24. Werf E. van der, Herik J. van den, Uiterwijk J., Solving Go on small boards. ICGA Journal, 26(2) s.92-107, 2003.
25. Werf E. van der, Winands M. H. M., Solving Go for Rectangular Boards. Journal of The International Computer Games Association. 32(2), s. 77-88, 2009.

COMPUTER USE IN FINDING THE OPTIMAL STRATEGY IN LOGIC GAMES

Key-words: solving games, artificial intelligence, logic game.

Summary: The issue of search an optimal strategy in logic games is a complex one. It can be divided into three sub-issues: computational, mnemonic and input/output operational. However, the ever-expanding computational and mnemonic powers of computers, as well as the speed in which local and distributed subdomains can transfer data, and the increased effectiveness help computers play better and better. In the first part of the thesis the early attempts of creating a AI player have been presented. In the following chapter non-trivial game solutions have been presented in chronological order. The third chapter describes the current knowledge and perspectives for solving the most popular games.