Theory of Computer Games

電腦對局理論

Tsan-sheng Hsu

徐讚昇

tshsu@iis.sinica.edu.tw

http://www.iis.sinica.edu.tw/~tshsu

Goal

- Course name: Theory of Computer Games
  - 電腦對局理論
- 20週年!!!
- Prerequisite: Computer Programming, Linux/Unix, C/C++, and Data Structure and Algorithms.
  - Enjoy playing classical board games!
  - Heavy programming projects!
- Goal: This course introduces techniques for computers to play various classical board games which includes Chinese chess and Go. Through the course, links between AI and games are hinted which come from system building via perfect balancing of good programming, algorithm design, and mathematics.

More course info

- Disclaimers:
  - NOT yet a course on game theory.
  - NOT yet a course on video games.
  - NOT yet a course on war game simulations.
- Web page:
  http://www.iis.sinica.edu.tw/~tshsu/tcg/2026

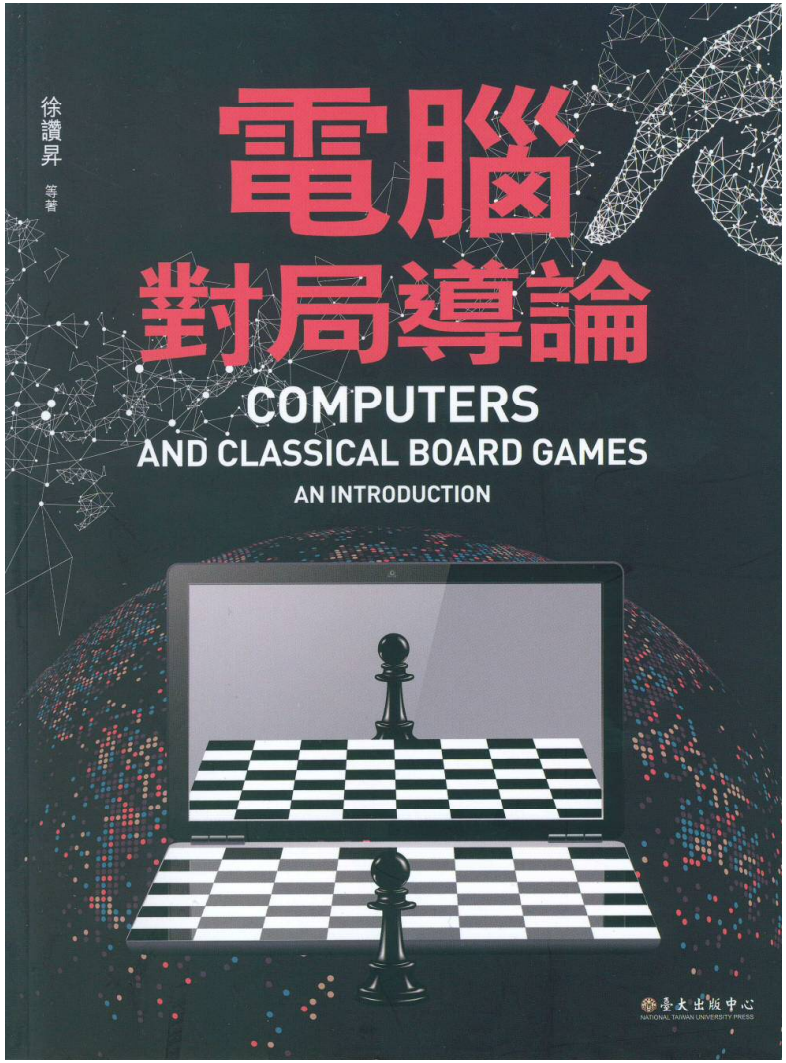
About this course

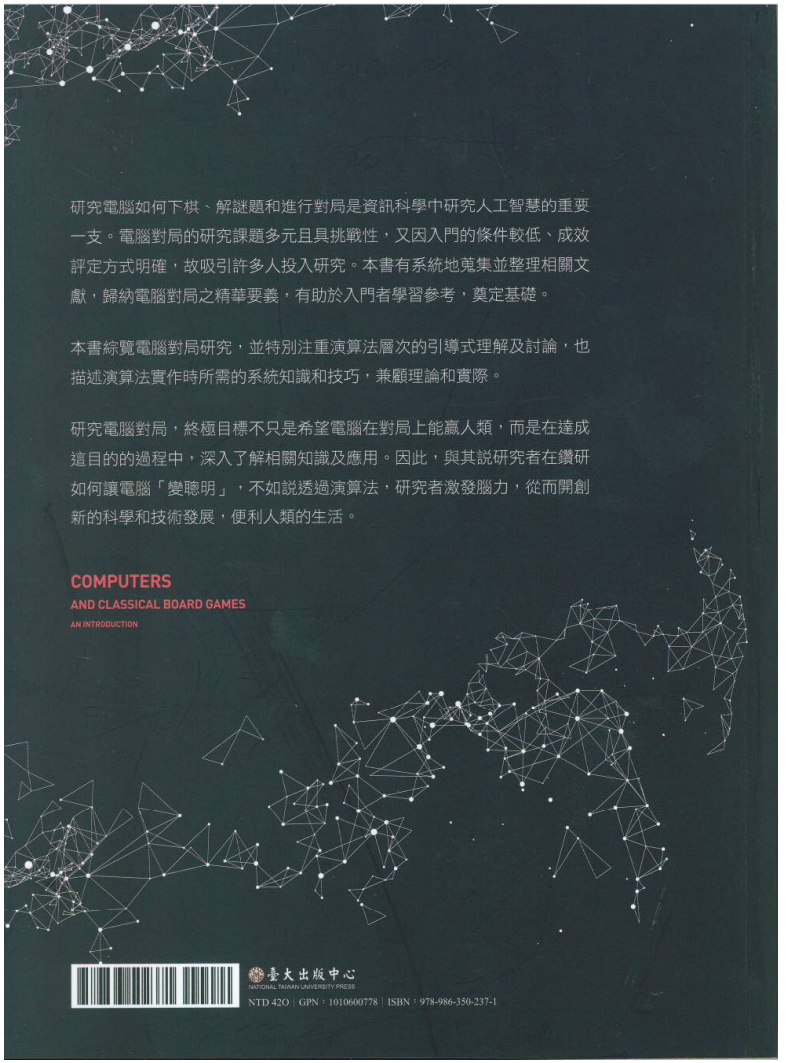
- Time and Place: Every Thursday from 2:20pm to 5:20pm at Room 111 (NTU CSIE building)
- Dates:

Sep		10	17	24	
Oct	1	8	15	22	29
Nov	5	12	19	26	
Dec	3	10	17	24	31
Jan	7				

- Format:
  - Lectures.
  - Presentations for homework projects (optional).
  - Invited lectures: TBA.
- Class materials
  - textbook: 電腦對局導論, 臺大出版中心, June 2017 ; ISBN: 978-986-350-237-1; required!!
  - Class notes
  - Collection of papers

Textbook





勘誤表: `http://www.iis.sinica.edu.tw/~tshsu/tcg/errata20210824.pdf`

電腦對局導論

Computers and classical board games: An Introduction

2017年 6月一版

勘誤表

August 23, 2021

頁碼/位置	內容	
	修改前	修改後
p.I 序一的第二段	預官退伍之後，讚昇出國到美國德州奧斯汀大學深造，專攻演算法研究。	預官退伍之後，讚昇出國到美國德州大學奧斯汀校區深造，專攻演算法研究。
p.I 序一的第三段	2005年8月，讚昇與我共同主辦第十屆國際電腦奧林匹亞大賽和CG2005電腦對局國際會議，開啓ICGA國際電腦對局學會在亞洲地區舉辦活動的新頁。	2005年8月，讚昇與我共同主辦第十屆國際電腦奧林匹亞大賽和CG2005電腦對局國際會議，開啓ICGA國際電腦對局學會在亞洲地區舉辦活動的新頁。
圖目錄之 2.4	混合雙佇列實作佇列之示意	混合雙佇列實作佇列之示意圖
圖目錄之 4.4	六貫棋性質證明：連接黑方棋子	六貫棋性質證明：連接各行中的黑方棋子
圖目錄之 5.7	位在中央的騎士	騎士的影響
圖目錄之 5.14	栓鏈的範例	栓鏈
圖目錄之 5.19	欠行局例	欠行
圖目錄之 7.4	斥候演算法搜尋的節點數比 Alpha-Beta 切捨演算法拜訪的節點數多的例子	斥候搜尋時 TEST 拜訪的節點數比 Alpha-Beta 切捨多的例子
圖目錄之 7.8	斥候演算法拜訪最少的節點數的例子	斥候演算法拜訪最少節點數的例子
演算法目錄之15	15 F′(position p)	15 F′(position p, integer $depth$)
演算法目錄之16	16 G′(position p)	16 G′(position p, integer $depth$)
演算法目錄之17	17 F(position p)	17 F(position p, integer $depth$)
演算法目錄之18	18 F2′(position p, value $alpha$, value $beta$)	18 F1′(position p, value $alpha$, value $beta$)
演算法目錄之19	19 G2′(position p, value $alpha$, value $beta$)	19 G1′(position p, value $alpha$, value $beta$)
演算法目錄之20	20 F2(position p, value $alpha$, value $beta$)	20 F2(position p, value $alpha$, value $beta$, integer $depth$)
演算法目錄之21	21 F2(position p, value $alpha$, value $beta$)	21 F2(position p, value $alpha$, value $beta$, integer $depth$)
演算法目錄之29	29 IDAS(position p, integer $limit$, integer $threshold$)	29 IDAS(position p, integer $limit$, value $threshold$)
演算法目錄之30	30 IDAS′(position p, integer $limit$, integer $threshold$)	30 IDAS′(position p, integer $limit$, value $threshold$)
演算法目錄之36	36 F4.4′(position p, value $alpha$, value $beta$, integer $depth$, Boolean do_null)	36 F4.4′(position p, value $alpha$, value $beta$, integer $depth$, Boolean in_null)
演算法目錄之37	37 F4.5′(position p, value $alpha$, value $beta$, integer $depth$, Boolean do_lmr)	37 F4.5′(position p, value $alpha$, value $beta$, integer $depth$, Boolean in_lmr)

Acknowledgements

- Thanks to TA's and the students of this course for providing constructive feedbacks on the slides.
  - Classes of 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024 and 2025.
- Special thanks to the following students.
  - Yuh-Jie Chen (class of 2008)
  - Jennya Chang (class of 2011)
  - Jessica Lin (class of 2011)

TA of the previous years

- 陳柏年
- 陳冠伶
- 許祐程
- 張紘睿
- 樂正
- 鍾詠先
- 陳約廷
- 郭麗莎
- 許嘉銘
- 詹凱傑
- 杜宗頡
- 薛韋霖
- 蔡數真
- 陳泓瑋
- 陳建宏

Classroom rules

- No drinking and eating.
- Maintain a proper social distance.
- If you do not feel well, rest at home.
  - Wear a mask effectively.

Evaluation (1/3)

- Homework (34%)
  - A project about single-player search using C programming (17%)
    - About A* search, single-person project, C/C++;Linux/Unix knowledge is required.
    - A one-player version of the Chinese Dark Chess (CDC) game with non-reachable cells on the board and additions types of pieces called One-CDC 3. The goal is to find a shortest number of moves for a player to capture all opponent's pieces using only cannons.
    - Your score is compared with the score of a good heuristic program playing on the same set of test data plus some early-bird bonus.
    - Implement a set of required techniques learned from the class and submit a written report.
    - Possible early bird bonus.
  - A project about Monte-Carlo simulation (17%)
    - A single-person project, C/C++;Linux/Unix knowledge is required.
    - CDC Kari 3: CDC with pieces either revealed or removed.
    - Your program plays against a baseline program and other opponents. Your score is compared with the score of a baseline CDC program playing on the same set of test data.
    - Implement a set of required techniques learned from the class and submit a written report.
    - Possible early bird bonus.

Evaluation (2/3)

- Written mid-term exam (33%)
- Final coding project (33%)
  - An alpha-beta based computer game program for the original version of CDC.
    - ▷ A sample code with GUI will be provided.
    - ▷ The usage of this sample code is restricted for anything related to this course only.
  - The 20th NTU-TCG Cup.
  - The competition will be held on-line or physically during the 17th (practice round) and 18th weeks (tournament) of the semester.
  - Submitted package: Code + documents.
  - Final written report: during the 18th week of the semester.
    - ▷ Document for the final project
    - ▷ Study notes.
    - ▷ . . .
- Class participation
  - Bonus for good participation
  - Students being recorded as not attending the classes will be penalized
  - More rules will be announced during the lectures

Evaluation: Backup plans (3/3)

- In case of unexpected circumstances due to COVID-19 or others.
- Possible options:
  - A written final exam.
  - Off-line competition.
    - Code submitted and then executed by TA.
    - Game logs are provided.
    - Multiple runs.
- For students that are falling behind during the semester, we will invite them to do extra work to make up the score.

Lecturing format

- Lecturing is entirely done in Mandarin
- For each topic
  - The first and most influential papers are introduced.
  - A list of recent and latest papers is provided for further readings and/or topics for presentations.

Course at a glance: tentative (1/3)

- Introduction (chapter 1): an A.I. oriented overview
- Single-player games (chapter 2): lectures for chapter 2.2.1 – 2.2.4 and chapter 3 are skipped
  - Brute-force search
  - A* search
- Programming tips (chapter 14.2.2)
- Homework I: announce at the 3rd week and is due at the 6th week.
- Two-player perfect information games
  - Survey (chapter 4)
  - Introduction from Chess's point of view (chapter 5)
  - Alpha-beta search (chapter 6)
  - Basic Monte-Carlo search (chapter 9)
- Mid-term exam: 2:20pm–5:20pm Oct 29 (Thursday; 8th week)

Course at a glance: tentative (2/3)

- Two-player perfect information games
  - Advanced Monte-Carlo method (chapter 10)
  - Nega-scout search (chapter 7)
- Homework II: announce at the 9th week and due at the 12th week.
- Practical considerations
  - Transposition tables (ch. 8.1–8.3)
  - Move ordering (ch 8.4–8.5)
- Searching chance nodes (ch 13.4)
- Advanced topics
  - Parallelization (ch. 11) (?)
  - Open-game and Endgame (ch. 12)
  - The graph-history interaction (GHI) problem (ch 13.2) (?)
  - Opponent model(ch 13.3) (?)
- Concluding remarks (ch 14)
  - Timing control
  - Software and hardware enhancements
  - Conclusion

Course at a glance: tentative (3/3)

- Final project: announce at the 12th week and due at the 18th week.
  - On-line competition of a practice round and a tournament round, respectively, are held during Thursdays of the 17th and 18th weeks.
- Final report: 18th week.

Introduction and an A.I. oriented overview

- Relations between computer games and Artificial Intelligence.
  - Why we study computer games?
  - Why we play or study games?
- History [SvdH02] [Sha50a]
  - The Turk, a chess playing "machine" at 1780's [LN82]
  - The endgame playing machine at 1910's [McC04]
  - C. E. Shannon (1950) [Sha50b] and A. Samuel (1960) [Sam60]
- Games that machines have beaten human champions [SvdH02] [Sch00]
  - Chess [CHH02]
  - Othello [Bur97]
  - Checker [SLLB96]
  - Go [SHM+16]
  - ...

Single-player games

- Games that can be played by one person [DH09]
  - combinatorial games such as 15-puzzle or Sukodu
  - other solitaire
- Classical approaches [Kor85] [KF02] [CS98]
  - Brute-force, BFS, DFS and its variations including DFID
  - Bi-directional search
  - A*
  - IDA*
  - IDA* with databases
- Disk-based approach [KS05]

Two-player perfect information games (1/2)

- A survey of current status [vdHUvR02]
- The original Computer Chess paper by C.E. Shannon [Sha50a] in 1950.
- Classical approaches
  - Alpha-beta search and its analysis [KM75]
  - Scout and Negascout [Pea80] [Rei83] [Fis83]
  - MTD(f): Best-first fixed-depth search [PSPdB96] [Pea80] if time allowed
- Enhancements to the classical approaches
  - Aspiration search
  - Quiescence search [Bea90]
  - Move ordering and other techniques [Sch89] [AN77] [Hsu91]
  - Further pruning techniques [SP96] including null move pruning and late move reduction
  - Proof-number search [AvdMvdH94] if time allowed

Two-player perfect information games (2/2)

- Monte-Carlo game tree search [BPW+12]
  - Original ideas [Bru93]
  - Best first game tree growing
  - UCT
  - Pruning techniques
    - ▷ Online knowledge [BH04] [YYK+06]
    - ▷ Offline knowledge [ST09] [HCL10a]
    - ▷ Deep learning [SHM+16]
- Searching chance nodes
- Case study:
  - Computer Chinese chess [YCYH04]
  - Computer Chinese dark chess [CSH10] if time allowed

Practical considerations (1/2)

- Transposition tables
  - Recording prior-search results to avoid researching
  - Design of a good hash function
    - Zobrist's hash function [Zob70]
- Open-game [Hya99] [Bur99] and endgame databases [Tho86] [Tho96] [WLH06]
  - Off-line collecting of knowledge
  - Computation done in advance
- Parallelization
  - Parallel alpha-beta based game tree search [Bro96] [FMM94] [HM02] [HSN89] [Hya97] [Man01]
  - Parallel Monte-Carlo game tree search [CJ08] [CWvdH08]
- The graph-history interaction (GHI) problem [Cam85] [BvdHU98] [WHH05]
  - The value of a position depends on the path leading to it.
    - Position value is dynamic and static.

Practical considerations (2/2)

- Bit board
- Opponent model [CM96]
  - How to take advantage of knowing the playing style of your opponent.
- Timing and resource usage control [Hya84] [HGN85] [MS93]
  - Using time wisely
    - ▷ Use too little time in the opening may be fatal.
    - ▷ Use too much time in opening may be fatal, too.
    - ▷ Knowledge from real tournament environments [vV09].
    - ▷ For Monte-Carlo type of search [HCL10b].
- Hardware enhancements [DL04]

Others – if time allowed

- Games with imperfect information and stochastic behaviors [FBM98]
  - Backgammon
  - Bridge
- Multi-player game tree search and pruning [Stu06]
  - Poker
  - Majon

Concluding remarks

- How to put everything together?
- How to test your implementation?
- How to measure the strength?

Collection of papers

References

[ACBF02] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. Machine Learning, 47:235–256, 2002.

[AHH11] B. Arneson, R. Hayward, and P. Henderson. Solving Hex: Beyond humans. In H. Jaap van den Herik, H. Iida, and A. Plaat, editors, Lecture Notes in Computer Science 6515: Proceedings of the 7th International Conference on Computers and Games, pages 1–10. Springer-Verlag, New York, NY, 2011.

[AN77] Selim G. Akl and Monroe M. Newborn. The principal continuation and the killer heuristic. In ACM '77: Proceedings of the 1977 annual conference, pages 466–473, New York, NY, USA, 1977. ACM Press.

[AvdHH91] L. V. Allis, H. J. van den Herik, and I.S. Herschberg. Which games will survive? In D.N.L. Levy and D.F. Beal, editors, Heuristic Programming in Artificial Intelligence 2: The Second Computer Olympiad, volume 2, pages 232–243. Ellis Horwood, Chichester, England, 1991.

[AvdMvdH94] L. V. Allis, M. van der Meulen, and H. J. van den Herik. Proof-number search. Artificial Intelligence, 66(1):91–124, 1994.

[Bal83] Bruce W. Ballard. The *-minimax search procedure for trees containing chance nodes. Artificial Intelligence, 21(3):327 – 350, 1983.

[Bea90] D. F. Beal. A generalised quiescence search algorithm. Artificial Intelligence, 43:85–98, 1990.

[BH04] B. Bouzy and B. Helmstetter. Monte-Carlo Go developments. In H. Jaap van den Herik, Hiroyuki Iida, and Ernst A. Heinz, editors, Advances in Computer Games, Many Games, Many Challenges, 10th International Conference, ACG 2003, Graz, Austria,

November 24-27, 2003, Revised Papers, volume 263 of IFIP, pages 159–174. Kluwer, 2004.

[Bou04] Bruno Bouzy. Associating shallow and selective global tree search with Monte Carlo for 9x9 Go. In Lecture Notes in Computer Science 3846: Proceedings of the 4th International Conference on Computers and Games, pages 67–80, 2004.

[BPW+12] Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, Simon Colton, et al. A survey of monte carlo tree search methods. Computational Intelligence and AI in Games, IEEE Transactions on, 4(1):1–43, 2012.

[Bro96] M.G. Brockington. A taxonomy of parallel game-tree searching algorithms. ICCA Journal, 19(3):162–174, 1996.

[Bru93] B. Bruegmann. Monte Carlo Go. unpublished manuscript, 1993.

[Bur97] Michael Buro. The othello match of the year: Takeshi murakami vs. logistello. Icca Journal, 20(3):189–193, 1997.

[Bur99] M. Buro. Toward opening book learning. International Computer Game Association (ICGA) Journal, 22(2):98–102, 1999.

[BvdHU98] D. M. Breuker, H. J. van dan Herik, and J. W. H. M. Uiterwijk. A solution to the GHI problem for best-first search. In H.J. van den Herik and H. Iida, editors, Lecture Notes in Computer Science 1558: Proceedings of the 1st International Conference on Computers and Games, pages 25–49. Springer-Verlag, New York, NY, 1998.

[Cam85] M. Campbell. The graph-history interaction: on ignoring position history. In Proceedings of the 1985 ACM annual conference on the range of computing : mid-80's perspective, pages 278–280. ACM Press, 1985.

[Che00] K. Chen. Some practical techniques for global search in Go. International Computer Game Association (ICGA) Journal, 23(2):67–74, 2000.

[CHH02] Murray Campbell, A Joseph Hoane, and Feng-hsiung Hsu. Deep blue. Artificial intelligence, 134(1):57–83, 2002.

[CHP+09] G. Chaslot, J.-B. Hoock, J. Perez, A. Rimmel, O. Teytaud, and M. Winands. Meta monte-carlo tree search for automatic opening book generation. In The IJCAI-09 Workshop on General Game Playing General Intelligence in Game-Playing Agents (GIGA'09), 2009.

[CJ08] T. Cazenave and N. Jouandeau. A parallel Monte-Carlo tree search algorithm. In H. Jaap van den Herik, X. Xu, Z. Ma, and M. H.M. Winands, editors, Lecture Notes in Computer Science 5131: Proceedings of the 6th International Conference on Computers and Games, pages 72–80. Springer-Verlag, New York, NY, 2008.

[CLHH06] B.-N. Chen, P.F. Liu, S.C. Hsu, and T.-s. Hsu. Abstracting knowledge from annotated chinese-chess game records. In H. Jaap van den Herik, P. Ciancarini, and H.H.L.M. Donkers, editors, Lecture Notes in Computer Science 4630: Proceedings of the 5th

International Conference on Computers and Games, pages 100–111. Springer-Verlag, New York, NY, 2006.

[CLHH08] B.-N. Chen, P.F. Liu, S.C. Hsu, and T.-s. Hsu. Knowledge inferencing on Chinese chess endgames. In H. Jaap van den Herik, X. Xu, Z. Ma, and M. H.M. Winands, editors, Lecture Notes in Computer Science 5131: Proceedings of the 6th International Conference on Computers and Games, pages 180–191. Springer-Verlag, New York, NY, 2008.

[CLHH10] B.-N. Chen, P.F. Liu, S.C. Hsu, and T.-s. Hsu. Conflict resolution of Chinese chess endgame knowledge base. In H. Jaap van den Herik and P. Spronck, editors, Lecture Notes in Computer Science 6048: Proceedings of the 12th Advances in Computer Games Conference, pages 146–157. Springer-Verlag, New York, NY, 2010.

[CLHH11] B.-N. Chen, P.F. Liu, S.C. Hsu, and T.-s. Hsu. Knowledge abstraction in Chinese chess endgame databases. In H. Jaap van den Herik, H. Iida, and A. Plaat, editors, Lecture Notes in Computer

Science 6515: Proceedings of the 7th International Conference on Computers and Games, pages 176–187. Springer-Verlag, New York, NY, 2011.

[CLHH12] B.-N. Chen, B.-F. Liu, S.-C. Hsu, and T.-s. Hsu. Aggregating consistent endgame knowledge in Chinese chess. Knowledge-Based Systems, 34:34–42, 2012.

[CM96] David Carmel and Shaul Markovitch. Learning and using opponent models in adversary search. Technical Report CIS9609, Technion, 1996.

[Cou06] Rémi Coulom. Efficient selectivity and backup operators in Monte-Carlo tree search. In Lecture Notes in Computer Science 4630: Proceedings of the 5th International Conference on Computers and Games, pages 72–83. Springer-Verlag, 2006.

[CS98] J. Culberson and J. Schaeffer. Pattern databases. Computational Intelligence, 14(3):318–334, 1998.

[CS11] T. Cazenave and A. Saffidine. Score bounded Monte-Carlo tree search. In H. Jaap van den Herik, H. Iida, and A. Plaat, editors, Lecture Notes in Computer Science 6515: Proceedings of the 7th International Conference on Computers and Games, pages 93–104. Springer-Verlag, New York, NY, 2011.

[CSH10] B.-N. Chen, B.-J. Shen, and T.-s. Hsu. Chinese drak chess. International Computer Game Association (ICGA) Journal, 33(2):93–106, 2010.

[CTH12] H.-J. Chang, M.-T. Tsai, and T.-s. Hsu. Game tree search with adaptive resolution. In H. Jaap van den Herik and A. Plaat, editors, Lecture Notes in Computer Science 7168: Proceedings of the 13th Advances in Computer Games Conference, pages 306–319. Springer-Verlag, New York, NY, 2012.

[CtSU+06] Guillaume Chaslot, Jahn takeshi Saito, Jos W. H. M. Uiterwijk, Bruno Bouzy, and H. Jaap Herik. Monte-Carlo strategies for computer Go. In Proceedings of the 18th BeNeLux Conference on Artificial Intelligence, pages 83–91, Namur, Belgium, 2006.

[CWvdH08] G. M.J.-B. Chaslot, M. H.M. Winands, and H. J. van den Herik. Parallel Monte-Carlo tree search. In H. Jaap van den Herik, X. Xu, Z. Ma, and M. H.M. Winands, editors, Lecture Notes in Computer Science 5131: Proceedings of the 6th International Conference on Computers and Games, pages 60–71. Springer-Verlag, New York, NY, 2008.

[DH01] E. Demaine and R. A. Hearn. Playing games with algorithms: Algorithmic combinatorial game theory. Technical report, Massachusetts Institute of Technology, USA, 2001. http://arxiv.org/abs/cs.CC/0106019, last revised 22 April 2008.

[DH09] E Demaine and B Hearn. Games, puzzles, and computation. AK Peters: I-IX, pages 1–237, 2009.

[DL04] C. Donninger and U. Lorenz. The chess monster Hydra. In Jürgen Becker, Marco Platzner, and Serge Vernalde, editors, Field Programmable Logic and Application, 14th International Conference , FPL 2004, Leuven, Belgium, August 30-September 1, 2004, Proceedings, volume 3203 of Lecture Notes in Computer Science, pages 927–932. Springer, 2004.

[DL05] C. Donninger and U. Lorenz. Innovative opening-book handling. In H. Jaap van den Herik, Shun-Chin Hsu, Tsan-sheng Hsu, and H.H.L.M. Donkers, editors, Lecture Notes in Computer Science 4250: Proceedings of the 11th Advances in Computer Games Conference, pages 1–10, New York, NY, 2005. Springer-Verlag.

[EM10] Markus Enzenberger and Martin Müller. A lock-free multithreaded Monte-Carlo tree search. In H. Jaap van den Herik and P. Spronck, editors, Lecture Notes in Computer Science 6048: Proceedings of the 12th Advances in Computer Games Conference, pages 14–20. Springer-Verlag, New York, NY, 2010.

[FBM98] I. Frank, D. A. Basin, and H. Matsubara. Finding optimal strategies for imperfect information games. In AAAI/IAAI, pages 500–507, 1998.

[Fis83] John P. Fishburn. Another optimization of alpha-beta search. SIGART Bull., (84):37–38, 1983.

[FMM94] Rainer Feldmann, Peter Mysliwietz, and Burkhard Monien.

Studying overheads in massively parallel min/max-tree evaluation. In SPAA, pages 94–103, 1994.

[Gin99] Matthew L. Ginsberg. Gib: Steps toward an expert-level bridge-playing program. In In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI-99, pages 584–589, 1999.

[GS07] Sylvain Gelly and David Silver. Combining online and offline knowledge in UCT. In Proceedings of the 24th international conference on Machine learning, ICML '07, pages 273–280, New York, NY, USA, 2007. ACM.

[HAH09] P. Henderson, B. Arneson, and R. B. Hayward. Solving 8x8 Hex. In Proceedings of IJCAI, pages 505–510, 2009.

[HCL10a] S. C. Huang, R. Coulom, and S. S. Lin. Monte-Carlo simulation balancing applied to 9x9 Go. International Computer Game Association (ICGA) Journal, 33(4):191–201, 2010.

[HCL10b] S. C. Huang, R. Coulom, and S. S. Lin. Time management for Monte-Carlo tree search applied to the game of Go. In International Workshop on Computer Games (IWCG). 2010. Hsinchu, Taiwan, Nov 18–20, 2010.

[Hei00] E. A. Heinz. Scalable Search in Computer Chess. Vieweg, 2000. ISBN: 3-528-05732-7.

[HGN85] R. M. Hyatt, A. E. Gower, and H. L. Nelson. Using time wisely, revisited (extended abstract). In Proceedings of the 1985 ACM annual conference on the range of computing : mid-80's perspective, pages 271–271. ACM Press, 1985.

[HL02] T.-s. Hsu and P.-Y. Liu. Verification of endgame databases. International Computer Game Association (ICGA) Journal, 25(3):132–144, 2002.

[HM02] R. M. Hyatt and T. Mann. A lockless transposition-table implementation for parallel search. International Computer Game Association (ICGA) Journal, 25(1):36–39, 2002.

[HSN89] Robert M. Hyatt, Bruce W. Suter, and Harry L. Nelson. A parallel alpha/beta tree searching algorithm. Parallel Computing, 10(3):299–308, 1989.

[Hsu91] S.-C. Hsu. Searching techniques of computer game playing. Bulletin of the College of Engineering, National Taiwan University, 51:17–31, 1991.

[Hya84] R. M. Hyatt. Using time wisely. International Computer Game Association (ICGA) Journal, pages 4–9, 1984.

[Hya97] R. M. Hyatt. The dynamic tree-splitting parallel search algorithm. ICCA Journal, 20(1):3–19, 1997.

[Hya99] R. M. Hyatt. Book learning — a methodology to tune an opening book automatically. International Computer Game Association (ICGA) Journal, 22(1):3–12, 1999.

[JS79] Wm. Woolsey Johnson and William E. Story. Notes on the "15" puzzle. American Journal of Mathematics, 2(4):pp. 397–404, 1879.

[Jui99] Hugues Juille. Methods for Statistical Inference: Extending the Evolutionary Computation Paradigm. PhD thesis, Department of Computer Science, Brandeis University, May 1999.

[KF02] R. E. Korf and A. Felner. Disjoint pattern database heuristics. Artificial Intelligence, 134:9–22, 2002.

[KM75] D. E. Knuth and R. W. Moore. An analysis of alpha-beta pruning. Artificial Intelligence, 6:293–326, 1975.

[KM04] A. Kishimoto and M. Müller. A general solution to the graph history interaction problem. In Proceedings of Nineteenth National Conference on Artificial Intelligence, pages 644–649, 2004.

[KMN02] Michael Kearns, Yishay Mansour, and Andrew Y Ng. A sparse sampling algorithm for near-optimal planning in large markov decision processes. Machine Learning, 49(2-3):193–208, 2002.

[Kor85] R. E. Korf. Depth-first iterative-deepening: An optimal admissible tree search. Artificial Intelligence, 27:97–109, 1985.

[KPS08] G. Kendall, A. Parkes, and K. Spoerer. A survey of NP-complete puzzles. International Computer Game Association (ICGA) Journal, 31(1):13–34, 2008.

[KS05] R. E. Korf and P. Schultze. Large-scale, para;;e; breadth-first search. In Proceedings of AAAI, pages 1380–1385, 2005.

[KT08] Hideki Kato and Ikuo Takeuchi. Parallel Monte-Carlo tree search with simulation servers. In 13th Game Programming Workshop (GPW-08), November 2008.

[LN82] David Levy and Monroe Newborn. Chess machines. In All About Chess and Computers, pages 1–23. Springer Berlin Heidelberg, 1982.

[Man01] Valavan Manohararajah. Parallel alpha-beta search on shared memory multiprocessors. Master's thesis, Graduate Department of Electrical and Computer Engineering, University of Toronto, Canada, 2001.

[McC04] Pamela McCorduck. Machines who think: A personal inquiry into the history and prospects of artificial intelligence, ak peters. Natick, Mass, 2004.

[MS93] Shaul Markovitch and Yaron Sella. Learning of resource allocation strategies for game playing. In R. Bajcsy, editor, Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI-93), pages 974–979, 1993.

[Neu28] J v Neumann. Zur theorie der gesellschaftsspiele. Mathematische Annalen, 100(1):295–320, 1928.

[Pea80] J. Pearl. Asymptotic properties of minimax trees and game-searching procedures. Artificial Intelligence, 14(2):113–138, 1980.

[Pea82] J. Pearl. The solution for the branching factor of the alpha-beta pruning algorithm and its optimality. Communications of ACM, 25(8):559–564, 1982.

[Pea84] J. Pearl. Heuristics: intelligent search strategies for computer problem solving. Addison-Wesley, 1984.

[PSPdB96] Aske Plaat, Jonathan Schaeffer, Wim Pijls, and Arie de Bruin. Best-first fixed-depth minimax algorithms. Artifical Intelligence, pages 255–293, 1996.

[Rei83] A. Reinefeld. An improvement of the scout tree search algorithm. ICCA Journal, 6(4):4–14, 1983.

[Ric83] Elaine Rich. Artificial Intelligence. McGraw-Hill, Inc., New York, NY, USA, 1983.

[RTT11] A. Rimmel, F. Teytaud, and O. Teytaud. Biasing Monte-Carlo simulations through RAVE values. In H. Jaap van den Herik, H. Iida, and A. Plaat, editors, Lecture Notes in Computer Science 6515: Proceedings of the 7th International Conference on Computers and Games, pages 59–68. Springer-Verlag, New York, NY, 2011.

[SA83] David J Slate and Lawrence R Atkin. Chess 4.5-the northwestern university chess program. In Chess skill in Man and Machine, pages 82–118. Springer, 1983.

[Sam60] A. Samuel. Programming computers to play games. Advances in Computers, 1:165–192, 1960.

[Sam67] A. Samuel. Some studies in machine learning using the game of checkers. IBM J. Res. Develop., 11:601–617, 1967.

[SBB+07] Jonathan Schaeffer, Neil Burch, Yngvi Bjornsson, Akihiro Kishimoto, Martin Muller, Robert Lake, Paul Lu, and Steve Sutphen. Checkers Is Solved. Science, 317(5844):1518–1522, 2007.

[SCA03] AysePinar Saygin, Ilyas Cicekli, and Varol Akman. Turing test: 50 years later. In JamesH. Moor, editor, The Turing Test, volume 30 of Studies in Cognitive Systems, pages 23–78. Springer Netherlands, 2003.

[Sch89] J. Schaeffer. The history heuristic and alpha-beta search enhancements in practice. IEEE Transactions on Pattern Analysis and Machine Intelligence, 11(11):1203–1212, 1989.

[Sch00] Jonathan Schaeffer. The games computers (and people) play. Advances in Computers, 52:190–268, 2000.

[Sha50a] C. E. Shannon. Programming a computer for playing chess. Philosophical Magazine, 41(314):256–275, 1950.

[Sha50b] Claude E Shannon. Xxii. programming a computer for playing chess. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 41(314):256–275, 1950.

[SHM+16] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. Nature, 529(7587):484–489, 2016.

[SLLB96] Jonathan Schaeffer, Robert Lake, Paul Lu, and Martin Bryant. Chinook the world man-machine checkers champion. AI Magazine, 17(1):21, 1996.

[SP96] J. Schaeffer and A. Plaat. New advances in alpha-beta searching. In Proceedings of ACM Conference on Computer Science, pages 124–130, 1996.

[ST09] David Silver and Gerald Tesauro. Monte-carlo simulation balancing. In Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09, pages 945–952, New York, NY, USA, 2009. ACM.

[Sta07] T. Stam. Solving Mahjong solitaire positions, 2007. BSc thesis.

[Sti89] L. Stiller. Parallel analysis of certain endgames. ICCA Journal, 12(2):55–64, 1989.

[Sti91] L. Stiller. Some results from a massively parallel retrograde analysis. ICCA Journal, 14(3):91–93, 1991.

[Stu06] N. Sturtevant. Current challenges in multi-player game search. In H. Jaap van den Herik, Y. Björnsson, and N. S. Netanyahu, editors, Lecture Notes in Computer Science 3846: Proceedings of the 4th International Conference on Computers and Games, pages 285–300. Springer-Verlag, New York, NY, 2006.

[SvdH02] J. Schaeffer and H. J. van den Herik. Games, computers, and artificial intelligence. Artificial Intelligence, 134:1–7, 2002.

[SWvdH+08] M. P.D. Schadd, M. H.M. Winands, H. J. van den Herik, G. N.J.-B. Chaslot, and J. W.H.M. Uiterwijk. Single-player Monte-Carlo tree search. In H. Jaap van den Herik, X. Xu, Z. Ma, and M. H.M. Winands, editors, Lecture Notes in Computer Science 5131: Proceedings of the 6th International Conference on Computers and Games, pages 1–12. Springer-Verlag, New York, NY, 2008.

[TBBS53] Alan M Turing, MA Bates, BV Bowden, and C Strachey. Digital computers applied to games. Faster than thought, 101, 1953.

[Tho86] K. Thompson. Retrograde analysis of certain endgames. ICCA Journal, 9(3):131–139, 1986.

[Tho96] K. Thompson. 6-piece endgames. ICCA Journal, 19(4):215–226, 1996.

[vdHUvR02] H. J. van den Herik, J. W. H. M. Uiterwijk, and J. van Rijswijck. Games solved: Now and in the future. Artificial Intelligence, 134:277–311, 2002.

[vV09] R. Šolak and R. Vučković. Time management during a chess game. International Computer Game Association (ICGA) Journal, 32(4):206–220, 2009.

[WH05] I.-C. Wu and D.-Y. Huang. A new family of k-in-a-row games. In H. Jaap van den Herik, Shun-Chin Hsu, Tsan sheng Hsu, and H.H.L.M. Donkers, editors, Lecture Notes in Computer Science 4250: Proceedings of the 11th Advances in Computer Games Conference, pages 180–194, New York, NY, 2005. Springer-Verlag.

[WHH05] K.-c. Wu, S.-C. Hsu, and T.-s. Hsu. The graph history interaction problem in Chinese chess. In H. Jaap van den Herik, Shun-Chin Hsu, Tsan-sheng Hsu, and H.H.L.M. Donkers, editors, Lecture Notes in Computer Science 4250: Proceedings of the 11th Advances in Computer Games Conference, pages 165–179, New York, NY, 2005. Springer-Verlag.

[Win84] Patrick Henry Winston. Artificial Intelligence (2Nd Ed.). Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1984.

[WLH06] P.-s. Wu, P.-Y. Liu, and T.-s. Hsu. An external-memory retrograde analysis algorithm. In H. Jaap van den Herik, Y. Björnsson, and N. S. Netanyahu, editors, Lecture Notes in Computer Science 3846: Proceedings of the 4th International Conference on Computers and Games, pages 145–160. Springer-Verlag, New York, NY, 2006.

[YCYH04] S.-J. Yen, J.-C. Chen, T.-N. Yang, and S.-C. Hsu. Computer Chinese chess. International Computer Game Association (ICGA) Journal, 27(1):3–18, 2004.

[YHM+11] Takayuki Yajima, Tsuyoshi Hashimoto, Toshiki Matsui, Junichi Hashimoto, and Kristian Spoerer. Node-expansion operators for the UCT algorithm. In H. Jaap van den Herik, H. Iida, and A. Plaat, editors, Lecture Notes in Computer Science 6515: Proceedings of the 7th International Conference on Computers and Games, pages 116–123. Springer-Verlag, New York, NY, 2011.

[YLP01] J. Yang, S. Liao, and M. Pawlak. A decomposition method for finding solution in game Hex 7x7. In Proceedings of International

Conference on Application nd Development of Computer games in the 21st century, pages 93–112, November 2001.

[YYK+06] Haruhiro Yoshimoto, Kazuki Yoshizoe, Tomoyuki Kaneko, Akihiro Kishimoto, and Kenjiro Taura. Monte Carlo Go has a way to go. In AAAI, 2006.

[Zob70] A. L. Zobrist. A new hashing method with applications for game playing. Technical Report 88, Department of Computer Science, University of Wisconsin, Madison, USA, 1970. Also in ICCA journal, vol. 13, No. 2, pp. 69–73, 1990.