

Nosce hostem : searching with opponent models

Citation for published version (APA):

Donkers, H. H. L. M. (2003). *Nosce hostem : searching with opponent models*. Datawyse / Universitaire Pers Maastricht. <https://doi.org/10.26481/dis.20031205hd>

Document status and date:

Published: 01/01/2003

DOI:

[10.26481/dis.20031205hd](https://doi.org/10.26481/dis.20031205hd)

Document Version:

Publisher's PDF, also known as Version of record

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.umlib.nl/taverne-license

Take down policy

If you believe that this document breaches copyright please contact us at:

repository@maastrichtuniversity.nl

providing details and we will investigate your claim.

Summary

The thesis deals with the question how opponent models can be used in computer game-playing. The search algorithms normally used in programs that play games like chess are algorithms equivalent with Minimax. They do not use explicit knowledge of the opponent. Yet it seems obvious that knowledge of the opponent can improve game playing. In 1993 two research teams, one in Israel and one in the Netherlands, simultaneously and independently invented a game-tree search method that involved knowledge of the opponent. It is called Opponent-Model search (OM search). This search method is our first main subject. The second main subject is a new search method that also uses an opponent model, but in contrast to the previous model incorporates uncertainty. Therefore, it is called Probabilistic Opponent-Model search (PrOM search).

In the first chapter we provide a short introduction to the research domain, which is the playing of board games by computers, then some relevant notions of mathematical game theory are given together with the basic notions of computer game-playing. The introduction is completed by the following problem statement: *under what conditions can OM search and PrOM search improve computer game-playing?* To answer this statement we formulate four research questions: (1) what are the important theoretical properties of OM search and PrOM search? (2) how can these search methods be implemented efficiently? (3) what is the nature of the practical circumstances in which OM search and PrOM search perform better than the current implementations of Minimax search? and (4) can PrOM search be used in practice to learn an opponent model?

The second chapter is dedicated to related research. We start with a comparison of five foregoing search methods that use some form of opponent model. Then we treat the simultaneous invention of OM search by Carmel and Markovitch in Israel and by Iida, Uiterwijk, Van den Herik, and Herschberg in the Netherlands. The core idea of OM search is that when the strategy of the opponent is known to the player, and the opponent uses Minimax search, then this knowledge can be used to lure the opponent into positions that are profitable for the player. The chapter presents the basic formulation of the search method and discuss the work that has been performed by both research teams. Each of them has developed specialisations and generalisations of OM search. The chapter ends with a discussion of some topics from mathematical game theory that are related to opponent models, such as special equilibria and nonzero-sum games.

The topic of the third chapter is the efficient implementation of OM search. We discuss two basic approaches for implementation: a one-pass approach and an approach that uses α - β probes. An important factor of efficiency is the number of game positions that can be disregarded when searching for the best move. Therefore efficient *pruning* of the search tree is a main research issue. We discuss several pruning algorithms for OM search for both the one-pass version and the probing version. For one-pass β -pruning OM search and β -pruning OM search with α - β probes, we provide a best-case analysis, showing that the former is the most efficient in the best case. The number of position evaluations that both algorithms need appear to be an almost linear function of the number of evaluations that α - β search needs. The last part of the chapter discusses the application of a range of search enhancements that are widely used in computer game-playing. Not surprisingly many enhancements are applicable, especially in the versions of OM search with α - β probes.

In chapter four we discuss the role of evaluation functions in OM search, and especially of errors in these functions. Since it is impossible to investigate all positions of a game, evaluation functions are used in computer game-playing to assess positions at the point where the search is exhausted. The inventors of OM search assume that the own evaluation function should be better than the opponent's function. However, evaluation functions can be interpreted in different ways. The chapter compares three interpretations of evaluation functions and lists eight possible orderings of functions. One of the orderings leads to the definition of estimation errors in evaluation functions. The second part of the chapter shows that these estimation errors can cause serious problems in OM search. We define four types of errors in OM search that follow from estimation errors; two of them are beneficial but the other two are harmful. Especially the type-I error can be dangerous. They appear when the own evaluation function overestimates a position that is assessed correctly by the opponent's evaluation function. In such circumstances OM search tries to lure the opponent into this position, because it seems profitable. The opponent follows willingly, causing the player to be caught in his or her own trap. Therefore, we define a condition, called *admissibility* condition, on the pair of evaluation functions that should prevent such awful circumstances.

After these theoretical chapters, we present in the fifth chapter the results of experiments with OM search in four different game domains. The first domain is random game trees. It is used to measure the average-case behaviour of the two β -pruning OM-search algorithms mentioned above. The version with α - β probes appears to be more efficient on the random game trees. We used the same domain to show that transposition tables have a positive effect on the efficiency of β -pruning OM search with α - β probes. The second domain is the game lines of action (LOA). The experiments performed show a poor performance of OM search although perfect knowledge of the opponent is available. An explanation is the absence of admissibility of the evaluation functions used. The third game domain is the King-Queen-King-Rook endgame of chess. In the experiments, admissibility is guaranteed since the player has access to an endgame database and the opponent not. The experiments are designed to show whether a player in a lost position could reach a draw or a win with the use of OM search. The results are not conclusive although in some conditions, OM search plays better than α - β search. The fourth

game domain is the mancala game bao. In these experiments, we attempt to discover what exactly is needed to bring OM search to success. It appears that a combination of good opponent prediction and extended search depth is needed for good results. Both factors cause OM search to use more resources. Of these two factors, an extended search depth seems to be more important than a good prediction. For the effect of OM search, the quality of the evaluation functions appears to be important, too.

Chapter six describes PrOM search. The chapter begins with an explanation of the probabilistic opponent model that underlies PrOM search. The model is related to the game-theoretic notion of a mixed strategy. It consists of a series of opponent types, characterized by evaluation functions, and a probability distribution. One of the opponent types is the player self, a mechanism that should prevent the negative effects of overestimations. Next we formulate the PrOM-search method and some of its theoretical properties. We present two different implementations of PrOM search: a one-pass version and a version with α - β probing. Similar to OM search, pruning is studied in PrOM search, too. It appears that pruning is even more restricted in PrOM search than in OM search. A best-case study reveals that the computational complexity of PrOM search is much higher than that of OM search in the version with α - β probing and even more in the one-pass version. This is mainly caused by the impossibility of deep pruning in PrOM search. The chapter ends with a discussion on how a probabilistic opponent model can be obtained. Opponent models for PrOM search can in theory be learned from observing the opponent. We concentrate on the learning of the opponent-type probabilities in an off-line and on-line setting.

In the seventh chapter we present experiments with PrOM search, again in four game domains. Experiments in the first domain (random game trees) show that also in the average case, the version of PrOM search with α - β probing is again more efficient than the one-pass version, but both versions use more resources than OM search. The experiments show that transposition tables can increase the efficiency of the version with α - β probes. The second game domain is LOA. The experiments are performed alongside the OM-search experiments above. PrOM search clearly outperforms OM search and α - β search in these experiments. It indicates that admissibility might be a less strong demand for PrOM search than it is for OM search. The third game domain is a simulated game that we use to test the learning of opponent-type probabilities. The experiments not only confirm some of the expectations from the theoretical analysis, but also give some insight into the dynamics of the learning. It appears that dependencies between opponent types can slow down the learning. The last game domain is bao. We first perform some experiments to learn opponent-type probabilities for a number of probabilistic opponent models. Next we study the performance of PrOM search with different opponent models. The experiments show that PrOM search can perform better in bao than α - β search even when no perfect knowledge of the opponent is available. However, the gain from using PrOM search is not very large. Furthermore, the search depth was kept constant to six plies for both sides, which means a considerable advantage for PrOM search. The opponent-type probabilities that are learned off-line, appear not to be the best probabilities to use in PrOM search: the probability on the ‘self’ should be larger than the learned value. Hence, the opponent model

should put more weight on the player's own evaluation function than is needed for the prediction of the opponent's moves. A possible explanation for this is that the additional weight balances the negative effects of type-I errors.

The last chapter of the thesis returns to the research questions and the problem statement as formulated in chapter one. From the experiments we learned factors that influence the performance of OM search and PrOM search: the quality of the knowledge and prediction of the opponent, the quality of the evaluation functions, admissibility and the size of the game tree. Taking the answers to the research questions into consideration, the answer to the problem statement must be that OM search or PrOM search can improve computer game-playing, but only in sufficiently profitable circumstances. Both search methods need (far) more resources than the current search methods that are based on α - β search. Yet, especially OM search suffers from many severe risks. If both methods are applied without serious precautions, the performance of computer game-playing is bound to decrease instead of to improve.

Samenvatting

Het proefschrift behandelt de vraag hoe opponentmodellen door computerprogramma's kunnen worden toegepast in bordspelen. De zoekmethoden die meestal in programma's voor schaken en andere spelen worden toegepast, gebruiken geen expliciete kennis over de tegenstander (opponent). Het lijkt echter vanzelfsprekend dat kennis over de tegenstander het spel kan verbeteren. In 1993 ontdekten twee onderzoeksgroepen, een in Israël en een in Nederland, tegelijkertijd en onafhankelijk van elkaar een zoekmethode die kennis over de opponent gebruikt. De methode wordt Opponent-Model search (OM search) genoemd. Deze zoekmethode is het eerste hoofdonderwerp. Het tweede hoofdonderwerp is een nieuwe zoekmethode die eveneens een model van de tegenstander gebruikt. In tegenstelling tot OM search wordt in dit model ook onzekerheid over de opponent gemodelleerd. De methode wordt Probabilistic Opponent-Model search (PrOM search) genoemd.

In het eerste hoofdstuk wordt een korte inleiding van het onderzoeksgebied gegeven, namelijk het spelen van bordspelen door computerprogramma's. Dan volgen enige relevante begrippen uit de wiskundige speltheorie en de grondbeginselen van het computerspelen. Na deze introductie komen we tot de volgende probleemstelling: *onder welke condities kunnen OM search en PrOM search het spelen door computers verbeteren?* Teneinde deze probleemstelling te beantwoorden zijn vier onderzoeksvragen opgesteld: (1) wat zijn de belangrijkste theoretische eigenschappen van OM search en PrOM search? (2) hoe kunnen deze methoden efficiënt worden geïmplementeerd? (3) wat is de aard van de omstandigheden waaronder OM search en PrOM search beter presteren dan de huidige implementaties van Minimax search? en (4) kan PrOM search in de praktijk worden gebruikt om een opponent model te leren?

Het tweede hoofdstuk is gewijd aan verwant onderzoek. We beginnen met een vergelijking van vijf voorafgaande zoekmethoden die een of andere vorm van een opponentmodel gebruiken. Daarna behandelen we de gelijktijdige uitvinding van OM search door Carmel en Markovitch in Israël en door Iida, Uiterwijk, Van den Herik en Herschberg in Nederland. De kerngedachte van OM search is dat wanneer de strategie van de tegenstander bij de speler bekend is en de tegenstander gebruik maakt van Minimax, de speler deze kennis kan gebruiken om de tegenstander in posities te lokken die gunstig voor de speler zijn. In het hoofdstuk presenteren we de basisformulering van de zoekmethode en bespreken het werk dat beide onderzoeksgroepen verder aan dit onderwerp hebben gewijd. Elk van de twee groepen heeft specialisaties en generalisaties van de zoekmethode ontwikkeld. Het hoofdstuk eindigt met de

bespreking van enkele thema's uit de wiskundige speltheorie die te maken hebben met opponentmodellen, zoals bijzondere evenwichten en niet-nulsomspelen.

Het onderwerp van het derde hoofdstuk is een efficiënte implementatie van OM search. We bespreken twee elementaire wijzen van implementatie: een eengangs-aanpak (one-pass) en een aanpak met α - β -peilingen (α - β -probing). Een bepalende factor voor de efficiëntie is het aantal spelposities dat kan worden overgeslagen tijdens het zoeken naar de beste zet: het efficiënt snoeien (pruning) van de zoekboom is een belangrijk onderzoeksonderwerp. We bespreken verschillende snoeimethoden voor OM search voor zowel de eengangsversie als de versie met peilingen. Voor de belangrijkste twee algoritmen (one-pass β -pruning OM search en β -pruning OM search met α - β -probing) geven we een best-case analyse. De analyse toont aan dat het eerste algoritme het efficiëntst is in de beste-case. Het aantal positie-evaluaties dat beide algoritmen nodig hebben blijkt een nagenoeg lineaire functie te zijn van het aantal evaluaties dat α - β search nodig heeft. Het laatste gedeelte van het hoofdstuk bespreekt de toepasbaarheid van een reeks zoekverbeteringen die algemeen gebruikelijk zijn in computerspelen. Menig verbetering is toepasbaar, in het bijzonder in OM search met α - β -probing.

In hoofdstuk vier bespreken we de rol van evaluatiefuncties in OM search en met name de fouten in die evaluatiefuncties. Aangezien het onmogelijk is om alle posities van een spel te onderzoeken gebruiken computerprogramma's evaluatiefuncties om posities te beoordelen op het moment dat het zoeken is uitgeput. De uitvinders van OM search namen aan dat de eigen evaluatiefunctie beter moest zijn dan die van de opponent. Er zijn echter verschillende interpretaties van evaluatiefuncties mogelijk die leiden tot verschillende definities van 'beter'. Het hoofdstuk behandelt drie interpretaties en acht ordeningen van evaluatiefuncties die daaruit voortkomen. Een van de ordeningen leidt tot de definitie van inschattingfouten in evaluatiefuncties. Het tweede deel van dit hoofdstuk toont aan dat deze inschattingfouten grote problemen kunnen veroorzaken in OM search. We definiëren vier foutsituaties in OM search; twee van deze zijn gunstig, de andere zijn ongunstig. Met name de type-I fouten zijn gevaarlijk. Zij treden op wanneer de eigen evaluatiefunctie een positie overschat die door de tegenstander correct wordt ingeschat. OM search zal proberen om de tegenstander in deze positie te lokken en de tegenstander zal zonder weerstand volgen - hetgeen ertoe leidt dat de speler in zijn of haar eigen valkuil stapt. We formuleren een voorwaarde die deze situatie moet vermijden: *admissibility*.

Na de theoretische hoofdstukken geven we in hoofdstuk vijf de resultaten van experimenten met OM search in vier domeinen. Het eerste domein is random spelbomen. Het wordt ingezet om het average-case gedrag van de twee bovengenoemde β -pruning varianten van OM search te meten. Het blijkt dat de versie met α - β -probing efficiënter is met deze random spelbomen. We benutten dit domein tevens om aan te tonen dat transpositietabellen een positief effect hebben op β -pruning OM search met α - β -probing. Het tweede domein is het spel lines of action (LOA). De experimenten in dit domein tonen een slechte prestatie van OM search ondanks de beschikbaarheid van perfecte kennis over de tegenstander. Een mogelijke verklaring is het ontbreken van *admissibility* in de betrokken evaluatiefuncties. Het volgende domein is het Koning-Dame-Koning-Toren eindspel in schaken. In deze experimenten kan *admissibility* gegarandeerd worden omdat de speler een eindspel-database

ter beschikking heeft in tegenstelling tot de tegenstander. De experimenten zijn ontworpen om aan te tonen of een speler in een verloren positie remise of winst kan bereiken met OM search. De resultaten zijn niet eenduidig alhoewel in sommige omstandigheden OM search beter speelt dan α - β search. Het vierde domein is het mancalaspel bao. In deze experimenten proberen we te ontdekken wat precies nodig is om OM search succesvol te maken. Het blijkt dat een combinatie van een goede voorspelling van de tegenstander en grotere zoekdiepte noodzakelijk is voor goede resultaten. Beide factoren veroorzaken dat OM search meer middelen (tijd) nodig heeft. Van deze twee factoren blijkt extra zoekdiepte belangrijker dan een goede voorspelling. Daarnaast is ook de kwaliteit van de gebruikte evaluatiefuncties belangrijk.

Hoofdstuk zes beschrijft PrOM search. Het hoofdstuk begint met een exposé over het probabilistisch opponentmodel dat aan PrOM search ten grondslag ligt. Het model is verwant aan het speltheoretisch begrip van gemengde strategie. Het model bestaat uit een reeks opponenttypes die ieder door een evaluatiefunctie worden gekenmerkt, en een kansverdeling. Een van de opponenttypes is de speler zelf, een voorziening die de negatieve effecten van overschattingen moet opheffen. Vervolgens formuleren we de PrOM search zoekmethode en een aantal theoretische eigenschappen. We presenteren twee verschillende implementaties van PrOM search: een one-pass aanpak en een aanpak met α - β -probing. Net zoals bij OM search bestuderen we ook pruning in PrOM search. Het blijkt dat pruning in PrOM search nog beperkter is dan in OM search. Een best-case analyse toont aan dat de computationele kosten van PrOM search veel hoger zijn dan die van OM search, met name in de one-pass versie. Dit wordt voornamelijk veroorzaakt door de onmogelijkheid van zogenaamd diep snoeien in PrOM search. Het hoofdstuk eindigt met de bespreking hoe een probabilistisch opponentmodel verkregen kan worden. We bestuderen met name het leren van kansverdelingen in een off-line en een on-line opstelling.

In het zevende hoofdstuk presenteren we experimenten met PrOM search, opnieuw in vier domeinen. De experimenten in het eerste domein, random spelbomen, laten zien dat ook in average case de versie van PrOM search met α - β -probing efficiënter is dan de one-pass versie, maar beide versies benutten veel meer middelen dan OM search. De experimenten tonen ook aan dat transpositietabellen de efficiëntie van de versie met α - β -probing kan vergroten. Het tweede domein is LOA. Deze experimenten zijn tegelijk met de experimenten hierboven voor OM search uitgevoerd. PrOM search speelt duidelijk beter dan OM search en α - β search. Dit duidt erop dat admissibility wellicht een minder sterke voorwaarde voor PrOM search is dan voor OM search. Het derde domein is een gesimuleerd spel dat wordt benut om het leren van kansen op opponenttypes te bestuderen. Niet alleen bevestigen de experimenten enkele van de verwachtingen van de theoretische analyse, maar ze geven ook inzicht in de dynamiek van het leren. Het blijkt dat afhankelijkheden tussen opponenttypes het leren vertraagt. Het laatste domein is bao. In dit domein voeren we eerst een reeks experimenten uit om kansen op opponenttypes te bepalen voor een aantal probabilistische opponentmodellen. Daarna bestuderen we de prestaties van PrOM search in een aantal toernooien. De experimenten tonen aan dat PrOM search soms beter presteert dan α - β search, zelfs wanneer geen perfecte kennis over de opponent aanwezig is. De winst van PrOM search is echter niet groot. Bovendien

wordt de zoekdiepte constant gehouden, wat een enorm voordeel voor PrOM search betekent. De kansverdeling over opponenttypes die we hadden geleerd blijkt niet de optimale kansverdelingen te zijn in de toernooien: de kans op het eigen opponenttype blijkt steeds hoger te zijn bij de optimale prestaties. Dit suggereert dat het opponentmodel meer kans moet toekennen aan het eigen opponenttype dan op basis van de zetten van de opponent wordt voorspeld. Een mogelijke verklaring hiervoor is dat de extra kans een tegenwicht is voor de negatieve effecten van type-I fouten.

Het laatste hoofdstuk van dit proefschrift keert terug naar de onderzoeksvragen en de probleemstelling zoals opgesteld in het eerste hoofdstuk. Uit de experimenten leren we de factoren die van invloed zijn op de prestaties van OM search en PrOM search: de kwaliteit van de kennis over de tegenstander en de voorspellingen, de kwaliteit van de evaluatiefuncties, aanwezigheid van admissibility en tenslotte de omvang van de zoekboom. Wanneer we de antwoorden op de onderzoeksvragen in overweging nemen, moet het antwoord op onze probleemstelling luiden dat zowel OM search als PrOM search het spelen door computers kan verbeteren, maar slechts onder zeer gunstige omstandigheden. Beide zoekmethoden benutten veel meer middelen dan het gebruikelijke α - β search, desalniettemin lijdt met name OM search onder ernstige risico's. Wanneer een van beide methoden zonder ernstige voorzorgsmaatregelen wordt toegepast is het spelen door computers eerder gedoemd te verslechteren dan te verbeteren.