

Samenvatting

Dit proefschrift onderzoekt hoe selectieve zoekmethoden de prestaties van een spelprogramma kunnen verbeteren voor een bepaald domein. Selectieve zoekmethoden hebben als doel om alleen de winstgevendste delen van de zoekruimte te verkennen, maar ze nemen het risico om de beste zet te overzien. Wij stellen een aantal selectieve zoekmethoden voor en testen ze in een groot aantal speldomeinen.

In hoofdstuk 1 geven we een korte inleiding in onderzoek in spelen en bespreken selectieve zoekmethoden voor spelen. De volgende probleemstelling stuurt ons onderzoek.

Probleemstelling: *Hoe kunnen we selectieve zoekmethoden op een zodanige wijze verbeteren dat programma's hun prestaties in domeinen van verschillende complexiteit verhogen?*

In plaats van selectieve zoekmethoden te testen op één klasse van spelen, kozen we verschillende klassen van spelen, die allemaal anders aangepakt moeten worden. Elke spelklasse vertegenwoordigt een niveau van complexiteit. Tussen elk niveau bestaat er een *complexiteitsprong*. Met een complexiteitsprong wordt de complexiteit van het spel aanzienlijk verhoogd, omdat het mechanisme van het spel is veranderd. We hebben gekozen voor vijf verschillende niveaus van spelen, wat resulteert in vier complexiteitsprongen. (1) Éénspeler spelen, of puzzels, hebben geen tegenstander en zijn een testdomein voor planningalgoritmes. (2) Tweespeler spelen zijn het klassieke testdomein voor zoekmethoden. We gebruiken ze voor zoekmethoden die de speltheoretische waarde kunnen bewijzen. Voor zoeken met kansknopen kunnen (3) niet-deterministische en (4) imperfecte-informatie spelen worden gebruikt. (5) Meerspeler spelen zijn een testdomein voor het omgaan met coalitievorming. We formuleren vier onderzoeksvragen die ons onderzoek sturen. Elke vraag heeft als thema het zoeken in een andere klasse van spelen en een andere selectieve zoekmethode. De vier onderzoeksvragen gaan over (1) Monte-Carlo Tree Search (MCTS), (2) Proof-Number (PN) search, (3) expectimax, en (4) meerspeler zoekmethoden.

Hoofdstuk 2 is een algemene inleiding in zoekmethoden voor spelen. Het beschrijft het minimax algoritme en de bekende $\alpha\beta$ zoekmethode. Standaardtechnieken voor het verbeteren van de $\alpha\beta$ zoekmethode worden ook besproken. Verder leggen wij MCTS en de bijbehorende verbeteringen uit.

De traditionele benadering voor deterministische éénpeler spelen met perfecte informatie is het toepassen van A* of IDA*. Deze methoden zijn zeer succesvol in het omgaan met deze klasse van spelen. Het nadeel van deze methoden is dat ze een onderschattende evaluatiefunctie vereisen. Het maken van een dergelijke functie kan moeilijk zijn. Omdat de selectieve zoekmethode MCTS geen evaluatiefunctie nodig heeft, is het een mogelijk interessant alternatief. Dit heeft geleid tot de eerste onderzoeksvraag.

Onderzoeksvraag 1: *Hoe kunnen we Monte-Carlo Tree Search aanpassen voor een éénpeler spel?*

Hoofdstuk 3 geeft antwoord op de eerste onderzoeksvraag door een nieuwe MCTS variant voor te stellen, genaamd Single-Player Monte-Carlo Tree Search (SP-MCTS). Twee wijzigingen in MCTS resulteren in SP-MCTS. De wijzigingen hebben betrekking op (1) de selectiestrategie en (2) de propagatiestrategie. De selectiestrategie is aangepast door rekening te houden met de standaarddeviatie van de scores van een zet en door de gemiddelde score te combineren met de hoogste score. De propagatiestrategie is aangepast om de selectiestrategie van alle benodigde informatie te voorzien. Voor de evaluatie van SP-MCTS, is de puzzel SameGame gekozen als testdomein. Tot dusver bestaat er geen goede onderschattende heuristische evaluatiefunctie voor dit spel.

Voor 20 gestandaardiseerde SameGame testposities scoorde de handmatig ingestelde SP-MCTS methode, waarbij alle zoektijd geïnvesteerd werd aan het begin van het spel, 73.998 punten. Dit was de hoogste score op de testposities in 2008. De belangrijkste bijdrage is dan ook dat wij met succes MCTS aangepast hebben voor een éénpeler spel. Geïnspireerd door onze aanpak, hebben twee andere Monte-Carlo gebaseerde methoden, Nested Monte-Carlo Search en Heuristically Guided Swarm Tree Search, ons record later verbroken. Op het moment van publicatie van dit proefschrift heeft SP-MCTS, met parameters afgesteld door de Cross-Entropy Method en met de tijd gelijkmatig verdeeld over alle zetten, 78.012 punten gescoord op de testposities, wat momenteel de derde hoogste score is (2010). Dus, om onderzoeksvraag 1 te beantwoorden, we hebben aangetoond dat MCTS toepasbaar is op een deterministisch éénpeler spel met perfecte informatie. Onze variant, SP-MCTS, is in staat om goede resultaten te behalen in SameGame. SP-MCTS is een waardig alternatief voor puzzels waar geen goede onderschattende evaluatiefunctie gevonden kan worden.

Idealiter is een zoekmethode in staat om te bewijzen dat een zet optimaal is voor een bepaald spel. Het spel is opgelost als dit wordt bereikt. Een selectieve zoekmethode speciaal ontworpen voor het oplossen van (eind)spelen is Proof-Number (PN) search. PN search is efficiënt in het doorzoeken van spelbomen met een niet-uniforme vertakkingsgraad. Daarnaast hebben voor veel spelen eindspeldatabases een rol gespeeld bij het oplossen. Dit heeft geleid tot de tweede onderzoeksvraag.

Onderzoeksvraag 2: *Hoe kunnen we een tweespeler spel oplossen met behulp van Proof-Number search in combinatie met eindspeldatabases?*

Hoofdstuk 4 geeft antwoord op de tweede onderzoeksvraag door het onderzoeken van de verhouding tussen de tijd besteed aan PN search en de tijd besteed aan het creëren van de eindspeldatabases bij het oplossen van het spel Fanorona. Dit spel is qua complexiteit van de toestandsruimte vergelijkbaar met Engels dammen. Uit statistieken van de eindspeldatabases blijkt dat (1) de speler aan zet een voordeel heeft en (2) dat ook met minder stukken op het bord dan de tegenstander een remise vaak kan worden gerealiseerd. De optimale grootten van de eindspeldatabase voor de 3×9 , 5×5 , en 7×5 Fanorona varianten zijn respectievelijk 3, 4, en 5 stukken. We concluderen dat de optimale grootte van de database op het punt ligt waar de tijd die nodig is om de database uit te rekenen en de tijd die nodig is voor het oplossen met PN search van dezelfde orde zijn. Ons belangrijkste resultaat is dat standaard Fanorona (5×9) is opgelost. Het resultaat is een remise als beide spelers optimaal spelen. Fanorona is dus toegevoegd aan de lijst van opgeloste spelen. Dit resultaat werd bereikt door het combineren van de PN-search variant PN^2 met alle eindspeldatabases tot 7 stukken.

Een andere vorm van selectieve zoekmethoden in deterministische tweespeler spelen met perfecte informatie is variable-depth search. Takken kunnen worden gesnoeid als ze weinig belovend lijken (voorwaarts snoeien), of verlengd indien de takken veelbelovend zijn (zoekextensies). Er bestaan een aantal succesvolle voorwaarts snoeitechnieken voor de $\alpha\beta$ zoekmethode. Voor tweespeler spelen die niet deterministisch zijn of imperfecte informatie hebben kan expectimax worden gebruikt. Expectimax voegt kansknopen toe aan de zoekboom. Er zijn echter geen voorwaarts snoeitechnieken beschikbaar voor de kansknopen. Dit heeft geleid tot de derde onderzoeksvraag.

Onderzoeksvraag 3: *Hoe kunnen we voorwaarts snoeien in de kansknopen van expectimax?*

Hoofdstuk 5 geeft antwoord op de derde onderzoeksvraag door de voorwaarts snoeitechniek ChanceProbCut te beschrijven voor expectimax. Deze techniek is de eerste in zijn soort om voorwaarts te snoeien in kansknopen. ChanceProbCut is geïnspireerd door de voorwaarts snoeitechniek ProbCut voor $\alpha\beta$. ChanceProbCut schat waarden van kansgebeurtenissen gebaseerd op ondiepe zoekopdrachten. Gebaseerd op een correlatie tussen evaluaties van zoekopdrachten met verschillende dieptes, kan ChanceProbCut kansknopen voorwaarts snoeien als het resultaat van de kansknoop waarschijnlijk buiten het zoekvenster valt. Twee niet-deterministische spelen (Dice en ChanceBreakthrough) en één imperfecte-informatie spel (Stratego) dienden als testdomeinen. De experimenten laten zien dat ChanceProbCut in staat is om grote delen van de zoekboom zonder kwaliteitsverlies te snoeien. Een veilige reductie in knopen van tussen de 30% en 85% kan worden bereikt voor alle spelen. Dus ChanceProbCut vindt de beste zet sneller in expectimax zonder de speelsterkte te beïnvloeden. De uitgespaarde tijd kan worden geïnvesteerd in een diepere zoekboom. Experimenteel is gebleken dat er een kleine, maar relevante verbetering in speelsterkte is in Stratego en Dice. In ChanceBreakthrough werd een significante toename van de speelsterkte gemeten. ChanceProbCut wint 54,4% van 4.000 partijen.

In deterministische tweespeler spelen met perfecte informatie is het merendeel van het onderzoek gericht op de $\alpha\beta$ zoekmethode. Voor deterministische meerspeler spelen met perfecte informatie is de keuze van zoekmethode niet zo eenvoudig. De twee belangrijkste zoekmethoden zijn \max^n en paranoid. Ze benaderen beide het probleem vanuit een andere hoek. \max^n gaat ervan uit dat elke speler probeert de eigen score te maximaliseren, terwijl paranoid ervan uitgaat dat alle tegenstanders een coalitie gesloten hebben. Deze aannames hebben echter een nadeel. Vanwege het ontbreken van veilige snoeitechnieken kan \max^n slechts een beperkte zoekdiepte bereiken. Bovendien kan de onderliggende aanname van \max^n onrealistisch zijn. Dit kan resulteren in een te optimistisch spel. Bij het diepe zoeken met de paranoid zoekmethode kan de coalitie van andere spelers te dominant worden. Dit kan resulteren in een te pessimistisch spel. Dit heeft geleid tot de vierde onderzoeksvraag.

Onderzoeksvraag 4: *Hoe kunnen we het zoeken in meerspeler spelen verbeteren?*

Hoofdstuk 6 geeft antwoord op de vierde onderzoeksvraag door een nieuwe zoekmethode voor te stellen, genaamd Best-Reply Search (BRS), voor deterministische niet-coöperatieve meerspeler spelen met perfecte informatie. Dit algoritme laat slechts één tegenstander een zet spelen. Deze tegenstander is degene met de sterkste tegenzet. De andere spelers moeten passen. Met behulp van deze aanpak kunnen meer beurten van de wortelspeler worden doorzocht. Dit resulteert in een lange-termijnplanning. Tegelijkertijd wordt een soort van voorzichtigheid bewaard door rekening te houden met de sterkste tegenzet.

We hebben gekozen voor drie deterministische meerspeler spelen met perfecte informatie, Chinese Checkers, Focus, en Rolit. BRS is in staat om vergeleken met \max^n significant beter te presteren in deze spelen, en wint tussen de 65% en 95% van alle partijen. Tegen paranoid is BRS aanzienlijk sterker in Chinese Checkers en Focus, en wint tussen de 57% en 71% van alle partijen. In Rolit zijn BRS en paranoid gelijkwaardig. Bij het spelen tegen verschillende soorten tegenstanders tegelijkertijd is BRS de sterkste zoekmethode in Chinese Checkers en Focus. In Rolit is BRS enigszins zwakker dan paranoid. In het algemeen heeft het verhogen van de zoektijd geen negatief effect op de prestaties van BRS. Dit impliceert dat het zoeken van illegale posities, die worden gegenereerd door tegenstanders geforceerd te laten passen, geen groot invloed heeft. Een mogelijk negatief effect kan worden goedge maakt door een grotere zoekdiepte.

Hoofdstuk 7 geeft de conclusies van het proefschrift en geeft een vooruitblik op open vragen en aanwijzingen voor toekomstig onderzoek. Rekening houdend met de antwoorden op de bovenstaande onderzoeksvragen zien we dat er vier manieren zijn om selectieve zoekmethoden te verbeteren. Ten eerste, Single-Player Monte-Carlo Tree Search balanceert de exploitatie en exploratie zodanig dat het een waardig alternatief voor éénspeler spelen is waar geen goede evaluatiefunctie kan worden gevonden. Ten tweede, PN search met eindspeldatabases is in staat te bewijzen dat de speltheoretische waarde van het tweespeler spel Fanorona een remise is. Ten derde, ChanceProbCut maakt het mogelijk om takken van kansknopen voorwaarts

te snoeien in tweespeler spelen met niet-determinisme of imperfecte informatie. Ten vierde, in niet-coöperatieve deterministische meerspeler spelen met perfecte informatie realiseert Best-Reply Search langetermijnplanning door te veronderstellen dat het slechts één tegenstander toegestaan is om een tegenzet te spelen.

Hoewel dit proefschrift aantoont dat selectieve zoekmethoden succesvol zijn in veel domeinen, kunnen de door ons voorgestelde verbeteringen en zoekmethoden ook in andere domeinen worden getest. Deze domeinen omvatten klassen van spelen met hetzelfde niveau van complexiteit, maar ook klassen van spelen die niet zijn opgenomen in dit onderzoek. We noemen éenspeler spelen met niet-determinisme, tweespeler spelen met niet-determinisme en imperfecte informatie, en meerspeler spelen met niet-determinisme en/of imperfecte informatie.