

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Aleš Tavčar

**PATOLOGIJA MINIMIN
PREISKOVANJA**

Diplomsko delo
na univerzitetnem študiju

Mentor: akad. prof. dr. Ivan Bratko

Ljubljana, 2009



Št. naloge: 01550/2009

Datum: 15.03.2009

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **ALEŠ TAVČAR**


Naslov: **PATOLOGIJA MINIMIN PREISKOVANJA**
PATHOLOGY IN MINIMIN SEARCH

Vrsta naloge: Diplomsko delo univerzitetnega študija

Tematika naloge:

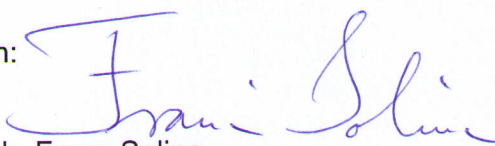
Pri reševanju problemov s hevrističnim preiskovanjem problemskega prostora je zaradi kompleksnosti preiskovanje pogosto omejeno do določene globine. V praksi navadno velja, da so rezultati preiskovanja tem boljši, čim večja je globina preiskovanja. Vendar lahko pod določenimi pogoji pride do t.i. patologije, ko iskanje do večje globine daje v povprečju celo slabše rezultate kot pri manjši globini. Patologija je bila najprej odkrita na področju igranja iger, torej pri preiskovanju dreves iger med dvema nasprotnikoma po načelu minimaks. Minimaks patologija je dobro raziskana, medtem ko to ne velja za patologijo pri enoagentnem preiskovanju, imenovanem tudi minimin preiskovanje. Naloga je eksperimentalno raziskati na sintetičnih domenah nekatere dejavnike, ki vplivajo na minimin patologijo. Med temi dejavniki so: stopnja vejanja drevesa, zrnatost ocenitvene funkcije in stopnja odvisnosti med bližnjimi vozlišči v drevesu.

Mentor:


akad. prof. dr. Ivan Bratko



Dekan:


prof. dr. Franc Solina

Rezultati diplomskega dela so intelektualna lastnina Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .

Namesto te strani **vstavite** original izdane teme diplomskega dela s podpisom mentorja in dekana ter žigom fakultete, ki ga diplomant dvigne v študentskem referatu, preden odda izdelek v vezavo!

Zahvala

Za pomoč in koristne nasvete pri izdelavi diplomske naloge se zahvaljujem mentorju Ivanu Bratku.

Prav tako bi se rad zahvalil Matjažu Gamsu in Mitji Luštreku za vse nasvete in uporabne ideje, ki so mi bile v veliko pomoč pri izdelavi diplomske naloge.

Posebna zahvala gre moji družini, ki me je v času študija vselej podpirala, ter bližnjim, ki so me potrpežljivo prenašali v času pisanja.

Kazalo

Povzetek	1
Abstract	3
1 Uvod	5
1.1 Minimaks in njegova patologija	7
1.2 Enoagentno preiskovanje in njegova patologija	8
1.3 Namen naloge in rezultati	9
1.4 Organizacija naloge	9
2 Pregled sorodnega dela	11
2.1 Patologija minimaksa	11
2.2 Patologija enoagentnega preiskovanja	12
3 Neodvisni model	15
3.1 Opis modela	15
3.2 Rezultati poskusov	20
4 Odvisni modeli	23
4.1 Popolnoma odvisna drevesa	23
4.2 Delno odvisna drevesa	24
4.3 Merjenje podobnosti	30
4.3.1 Faktor grozdenja	30
4.3.2 Korelacija	32
4.4 Primerjava z minimaks modelom	34
4.5 Primerjava z igro osmih ploščic	37
5 Sklepne ugotovitve	39
A Tabele	41

Seznam slik	43
Seznam tabel	45
Seznam algoritmov	46
Literatura	47

Seznam uporabljenih kratic in simbolov

Oznake

b	vejitev drevesa igre
c_b	verjetnost poraza za igralca na potezi pri vejitvi drevesa b
$c_{b,s}$	verjetnost poraza za igralca na potezi pri vejitvi drevesa b in stopnji odvisnosti s
d	izbrana globina preiskovanja drevesa
d_{max}	največja globina drevesa
cf	faktor grozdenja
h	poljubna hevristična funkcija
p	stopnja patološkosti
r	korelacija
v_i	prava vrednost i -tega vozlišča
v_{ij}	prava vrednost j -tega sina i -tega vozlišča
M	število vseh vozlišč
N	število vseh notranjih vozlišč
$E(d, h)$	verjetnost napačne izbire poteze pri izbrani globini d in hevristični funkciji h
$H_s(i)$	gladka porazdelitev, ki določa verjetnost zamenjave lista z indeksom i

Povzetek

Igranje iger lahko smatramo kot eno prvih področij umetne inteligence, s katerim so se začeli ukvarjati raziskovalci. Razvoj ustreznih algoritmov in heuristik je privedel do tega, da so se računalniški programi sposobni kosati z najboljšimi človeškimi igralci. Programom to uspeva predvsem s pomočjo primernih algoritmov, ki jim omogočajo omejeno pregledovanje velikega prostora stanj in izbiro dobrih potez.

Pri igranju iger običajno pregledamo drevo igre od trenutnega položaja do neke globine, stanja tam heuristično ocenimo in te ocene prenesemo nazaj do korena drevesa, ter se na njihovi podlagi odločimo za ustrezno potezo. Opis daje slutiti, da globlje preiskovanje daje boljše rezultate, kar se tudi pokaže v praksi. Matematične analize modelov pa so pokazale ravno nasprotno: v nekaterih primerih globlje preiskovanje daje slabše odločitve. Ta pojav se imenuje *preiskovalna patologija*.

V tej nalogi se ukvarjamo s patološkimi modeli enoagentnega preiskovanja, ter obravnavamo dejavnike, ki vplivajo na patologijo. Enoagentno preiskovanje se uporablja v primeru, ko en agent išče rešitev problema v okolju, ki mu dejavno ne nasprotuje. Na umetnih preiskovalnih drevesih smo preučili dejavnike, ki vplivajo na pojav in stopnjo patologije. Med najpomembnejše dejavnike spadajo vejitev, število različnih možnih vrednosti vozlišča in stopnja odvisnosti med vrednostmi bratskih vozlišč. Glavni cilj je ugotoviti vpliv, ki ga posamezen dejavnik prispeva k nastanku patologije. V ta namen zgradimo neodvisna umetna drevesa, ter jih kombiniramo z odvisnimi modeli in tako pridobimo delno urejena drevesa. Izkaže se, da imajo podobne lastnosti. Z naraščanjem vejitve se patološkost okrepi, naraščanje odvisnosti pa ta pojav oslabi.

Ključne besede:

minimin, patologija minimin preiskovanja, drevo igre, odvisna drevesa

Abstract

Game playing is one of the first areas of artificial intelligence studied by AI researchers. The evolution of adequate algorithms and heuristics have brought us to a point where computer players are able to compete with the best human players. This is achieved above all by algorithms that are able to efficiently search large game trees when choosing proper moves.

In game playing it is common to examine the game tree from the current position to some depth. The states at that depth are heuristically evaluated and backed-up back to the root node, where they are used to choose a move. From this description we may surmise that deeper search gives better results than shallow, which can also be perceived in practice. However, mathematical analyses have shown that under certain conditions the opposite happens: deeper search gives worse decisions. This phenomenon was termed *search pathology*.

In this thesis we deal with single-agent pathological models and we examine factors influencing the pathology. We have examined the properties that affect the occurrence of the pathology in synthetic search trees. The most important factors are the branching of the game tree, the dependence between nearby nodes and the number of different node values. Our main goal was to explicate their influence on the occurrence of the pathology. Next, we combined independent and dependent trees. That way we can construct partially dependent trees and analyze the pathology in such trees. Finally, we concluded that all constructed models behave similarly: the pathology is strengthened by increasing tree branching and is weakened by increasing the dependence between nearby nodes.

Key words:

minimin, minimin pathology, tree search, game tree, dependent game tree

Poglavje 1

Uvod

Številne probleme lahko predstavimo s prostorom stanj, katerega predstavimo z usmerjenim grafom. Vozlišča ustrezajo možnim stanjem problema, povezave med njimi pa možnim prehodom med stanji [1]. K definiciji problema sodijo tudi posebno začetno stanje, ki je navadno samo eno, ter množica končnih stanj. Tak graf je lahko podan v naprej (npr. podan je zemljevid prostora) ali pa so podana pravila za generiranje vozlišč v grafu, ki predstavljajo pravila igre. Pri reševanju problema poskušamo najti najboljše končno stanje ali najboljšo rešitev problema (na zemljevidu prostora skušamo najti čim krajšo pot od izhodišča do cilja).

V nadaljevanju se bomo omejili na preiskovalna drevesa, ki so poseben primer usmerjenih grafov. Vrh drevesa, ki ga poimenujemo koren, predstavlja začetno stanje. Notranja vozlišča drevesa predstavljajo stanja problema, povezave med njimi pa možne prehode med stanji. V listih drevesa so vrednosti, ki ustrezajo rezultatom v končnih položajih. Vrednosti v listih sta lahko le dve, npr. zmaga in poraz, lahko pa jih je poljubno mnogo.

Zgradbo preiskovalnega drevesa bomo lažje razumeli, če si najprej ogledamo drevo igre, torej primer igre dveh igralcev. Igro začnemo v korenu drevesa, kjer prvi igralec želi izbrati najboljšo potezo, torej takšno, ki ga bo vodila k zmagi. Prvi igralec torej izbere maksimalno vrednost sinov. Te vrednosti pa še niso definirane, zato se premaknemo za en nivo navzdol v drevesu. Tu poteze izbira nasprotni igralec, ki izbira poteze slabe za prvega igralca, zato vozliščem priredimo minimum sinov. Te nivoje imenujemo maks in min. Nivoji se navadno izmenjujejo do listov, kjer se nahajajo prave vrednosti končnih položajev. Te vrednosti se nato prenesejo v koren drevesa, kjer prvi igralec izbere potezo, ki vodi k sinu z največjo povratno vrednostjo.

Probleme enoagentnega iskanja lahko predstavimo s preiskovalnim dreve-

som, ki je definirano enako kot drevo igre, le da se vrednosti od listov do korena namesto z minimaksom prenašajo z miniminom. Načeloma bi namesto minimuma lahko iskali tudi maksimum, vendar je minimum bolj običajen.

Psevdokoda minimin algoritma za določanje prave vrednosti položaja je formalno zapisana v algoritmu 1.

Algoritem 1 *minimin*(*node*, *depth*)

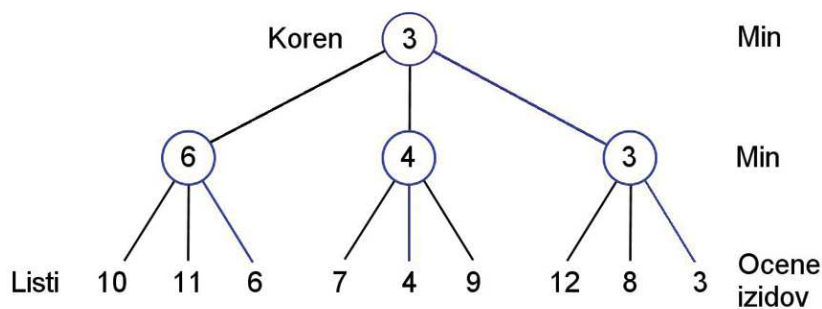
Require: *node*, *depth*

```

if node is a leaf node or depth = 0 then
  return value of node
else
   $\alpha \leftarrow \infty$ 
  for all child of node do
     $\alpha \leftarrow \min(\alpha, \text{minimin}(\text{child}, \text{depth} - 1))$ 
  end for
  return  $\alpha$ 
end if

```

Na sliki 1.1 je prikazano drevo, na katerem bo potekalo preiskovanje. Vozišča min so obarvana modro, prav tako je na vsakem nivoju obarvana izbrana povezava. Igralec bo v korenu izbral potezo, ki bo vodila do vozišča z vrednostjo 3.



Slika 1.1: Primer preiskovanja z miniminom.

1.1 Minimaks in njegova patologija

Večina iger z dvema igralcema uporablja algoritem minimaks, ki si ga je, kot postopek za izbiro poteze pri šahu, izmislil John von Neumann [15]. Osnovna ideja je dokaj enostavna. Igro predstavimo s pomočjo drevesa igre, v katerem vozlišča predstavljajo položaje, povezave pa možne poteze med njimi. Koren drevesa predstavlja naš trenutni položaj, povezave, ki vodijo iz njega, pa možne poteze, med katerimi lahko izbiramo. Listi drevesa ustrezajo položajem, do katerih pridemo z odigranjem zadnje poteze v igri, torej listi navadno ponazarjajo zmago ali poraz. Igralec, ki je na vrsti, želi izbrati najboljšo potezo, ki bo vodila k zmagi. V ta namen izbere potezo, ki vodi do sina z maksimalno vrednostjo. Teh vrednosti pa še ne poznamo, zato moramo ovrednotiti vozlišča na nižjem nivoju. Na tem nivoju pa nasprotni igralec izbira slabe poteze za prvega igralca, zato temu vozlišču priredimo minimum sinov. Opisani min in maks nivoji se tako izmenjujejo vse do listov drevesa, kjer se nahajajo prave vrednosti položajev. Te vrednosti se po istem principu prenesejo do korena drevesa. Zdaj ima prvi igralec na razpolago vrednosti sinov in se lahko odloči za najboljšo potezo.

Če bi v realnih igrah lahko igrali na zgoraj opisan način, bi preiskali celotno drevo igre in z algoritmom minimaks vedno izbrali potezo, ki bi vodila do najboljšega rezultata. Na žalost pa so drevesa igre navadno prevelika, zato si moramo pomagati s heuristikami. Kot primer kompleksnosti realnih iger si oglejmo oceno števila stanj za igro šah, ki jo je izračunal Claude Shannon [34]. Njegovi izračuni kažejo, da je kompleksnost drevesa igre vsaj 10^{120} , medtem ko ocena števila atomov v vidnem vesolju znaša 10^{80} . Iz tega lahko vidimo, da je pregledovanje celotnega drevesa neuresničljivo opravilo. Drevo zato razvijemo do izbrane globine in tam položaje ocenimo s heuristično funkcijo. Ta funkcija nam vrne približno oceno težavnosti ali ugodnosti položaja. Tako dobljene ocene imenujemo statične heuristične vrednosti.

Pričakujemo lahko, da globlje preiskovanje drevesa privede do boljšega igranja, saj igralec pregleda več potez v naprej. Izkušnje iz prakse to domnevo tudi potrjuje. Ko pa so raziskovalci poskušali razložiti, zakaj minimaks napako heurističnih ocen zmanjša in so povratne heuristične ocene zanesljivejše, so prišli do nenavadnega odkritja, saj se zgodi ravno nasprotno: minimaks napako heurističnih ocen poveča. Ta pojav so imenovali *patologija minimaksa*.

1.2 Enoagentno preiskovanje in njegova patologija

Z enoagentnim preiskovanjem se srečujemo v primerih, ko agentu pri iskanju najboljše rešitve nihče ne nasprotuje, tako kot je to pri igrah za dva igralca. Pri iskanju najkrajše poti se igralec neovirano premika npr. po zemljevidu in išče poteze, ki ga bodo privedle do iskanega cilja. Poleg omenjenega primera spada v to kategorijo še reševanje uganke osmih (ali več) ploščic [30], reševanje Rubikove kocke, računalniško dokazovanje izrekov [32], problem trgovskega potnika [33] itd.

Podobno kot pri igrah z dvema igralcema, lahko tudi pri tej vrsti preiskovanja uporabimo predstavitev z drevesom igre, kjer vozlišča predstavljajo položaje, povezave pa možne poteze. V listih pa se nahajajo vrednosti, ki označujejo kakovost rešitve in ne le zmago ali poraz. Bistvena razlika med enoagentnim preiskovanjem in igrami za dva igralca pa je način prenosa vrednosti v listih do korena drevesa. Te vrednosti prenašamo z miniminom, torej vrednost nekega vozlišča določimo tako, da izberemo najmanjšo vrednost v sinovih tega vozlišča. Ekvivalentno bi lahko za določitev povratnih vrednosti uporabili maksimum sinov, vendar je uporaba minimuma bolj običajna, saj v večini primerov iščemo najkrajšo ali najcenejšo pot pri preiskovanju prostora. Poznamo več algoritmov za enoagentno preiskovanje, med katerimi je najbolj znan A^* [?]. Ta spada med polne preiskovalne algoritme, ki delujejo po metodi najprej najboljši. Z uporabo hevristične funkcije se algoritem odloča, katero vozlišče bo razvil naslednje. V prostoru stanj, kot je definiran v [1], velja, da če je izbrana hevristična funkcija popolna, kar pomeni, da ne sme precenjevati razdalje do konca, je prva rešitev, ki jo algoritem vrne, tudi optimalna. Zaradi preiskovanja celotnega drevesa pojav patologije pri A^* ni mogoč, saj se le ta pojavi pri različnih globinah preiskovanja.

Uporaba algoritma A^* pa ni vedno mogoča, saj bi v primeru zelo velikih preiskovalnih dreves izvajanje terjalo preveč časa, zato pogosto pregledamo le del takega drevesa, vozlišča na določeni globini pa hevristično ocenimo. Hevristična funkcija $f(x)$ je navadno sestavljena iz dveh delov: iz dolžine poti od začetnega stanja do trenutnega vozlišča $g(x)$, ter oceno poti od trenutnega položaja do konca $h(x)$. To skupaj zapišemo kot $f(x) = g(x) + h(x)$. Algoritmi, ki so primerni za pregledovanje velikih dreves po zgoraj opisanem postopku, spadajo v skupino delnih algoritmov ali algoritmov s takojšnjim odzivom. Najbolj znan med njimi je algoritem LRTA* [2]. Podobno kot minimaks, LRTA* pregleda preiskovalno drevo do določene globine, tam vozlišča hevristično oceni, ter vrednosti prenese do korena. Na podlagi teh ocen izbere ustrezno potezo. Ta postopek se ponavlja, dokler ne doseže končnega stanja.

Delni algoritmi najdejo rešitev veliko hitreje kot polni, vendar pa taka rešitev ni nujno optimalna.

1.3 Namen naloge in rezultati

Začetne raziskave enoagentnega preiskovanja so le potrdile možnost patologije [4] in na umetnih drevesih podale razloge za njen nastanek [7]. Piltaver je v diplomski nalogi [24] proučeval patologijo v igri osmih ploščic. V igro je dodal nove poteze in tako omogočil spreminjanje vejitve na navzgor omejeno vrednost, spreminjal je tudi zrnatost hevrstične funkcije. Podobnosti bratskih vozlišč pa ni bilo mogoče neposredno spreminjati, saj je podobnost vozlišč lastnost problema. Uporabil je le različne podobnosti med večkratnim poganjem igre z enakimi parametri.

Naša naloga je zgraditi bolj splošen model, ki bi omogočal poljubno spreminjanje vseh treh dejavnikov: vejitve, zrnatosti in podobnosti. V ta namen zgradimo več dreves, ki omogočajo uravnavanje podobnosti med vozlišči, določanje vejitve in spreminjanje števila različnih vrednosti v listih drevesa. S spreminjanjem teh količin pregledamo celoten prostor vrednosti parametrov in iz rezultatov ugotovimo iskane značilnosti modela. Na neodvisnem modelu tudi preverimo, kako vpliva na patologijo, če lahko hevrstične ocene zavzamejo vrednosti izven zgornje meje intervala $[0, 1]$. Več o tem je opisano v podpoglavju 3.1.

S pomočjo računalniških simulacij pokažemo, da se s povečevanjem odvisnosti dreves stopnja patološkosti zmanjša. Podoben učinek dosežemo tudi s povečevanjem zrnatosti v listih. Ugotovimo, da se z večanjem vejitve povečuje tudi število različnih vrednosti v listih, ki so potrebne za odpravo patologije. Vejitev torej ojača pojav patologije, odvisnost med drevesi pa jo omili.

Pričujoče delo je nastalo kot plod raziskave na Odseku za inteligentne sisteme na Institutu "Jožef Stefan" in predstavlja enoten pregled celotne raziskave.

1.4 Organizacija naloge

Diplomsko delo je razdeljeno na pet poglavij. Začnemo z uvodom, kjer smo opisali metode enoagentnega preiskovanja in preiskovanje z minimaksom. Dotaknemo se problemov, ki jih patologija preiskovanja prinaša, ter omenimo kdaj do patologije ne more priti.

V 2. poglavju opravimo krajši pregled sorodnega dela. Na začetku si ogledamo dela avtorjev, ki sta patologijo minimaksa odkrila, predstavimo pa

tudi avtorje, ki so s svojimi prispevki pomagali pri boljšem razumevanju patologije. Nadaljujemo s pregledom literature, ki je obravnavala patologijo enoagentnega preiskovanja.

V 3. poglavju se ukvarjamo z večvrednostnim modelom preiskovanja, pri katerem so vozlišča drevesa med seboj neodvisna. Najprej model predstavimo, podamo ključne lastnosti modela in omenimo nekaj problemov na katere smo naleteli med samo gradnjo modela. Na koncu poglavja predstavimo rezultate poskusov in pregledamo vpliv posameznih dejavnikov na patologijo v našem modelu.

V 4. poglavju se posvetimo odvisnim modelom. Na začetku opišemo možne načine gradnje popolnoma odvisnih dreves in predstavimo postopek, ki ga bomo uporabili v nadaljevanju. V naslednjem razdelku se ukvarjamo z zgradbo postopnih modelov odvisnih dreves, kjer opišemo več načinov kombiniranja odvisnih in neodvisnih dreves. Predstavimo težave, ki jih prinaša posamezen način in opravimo krajšo primerjavo z minimaks modelom. Rezultate meritev patološkosti grafično prikažemo in analiziramo. Na koncu poglavja opišemo dve meri za računanje podobnosti med vozlišči drevesa in izračunane vrednosti za zgrajeni model grafično prikažemo.

Diplomsko delo v petem poglavju zaključim s povzetkom dela in pregledom rezultatov in ugotovitev, ki smo jih pridobili z izvajanjem poskusov. Na koncu podamo še nekaj zamisli za nadaljne preučevanje patologije.

Poglavje 2

Pregled sorodnega dela

V tem poglavju je povzet kratek pregled sorodnega dela na področju pojasnjevanje patologije minimaksa in enoagentnega preiskovanja.

2.1 Patologija minimaksa

Odkritje patologije povezujemo s prizadevanji raziskovalcev o poizkusu formalne razlage, zakaj globlje preiskovanje z minimaksom v realnih igrah daje boljše rezultate. Pri tem sta Dana S. Nau leta 1979 [12] in Donald F. Beal leta 1980 [3] neodvisno odkrila zanimiv pojav, ki so ga poimenovali patologija minimaksa.

Beala je poskušal podati formalno razlago, zakaj so z minimaksom dobljene povratne hevristične vrednosti zaneslivejše od statičnih. V ta namen je sestavil preprost matematični model preiskovanja z minimaksom. V nasprotju s pričakovanji je prišel do odkritja, da so povratne hevristične vrednosti manj zanesljive od statičnih. Ugotovil je tudi, da napaka v korenu drevesa igre narašča s povečevanjem globine preiskovanja.

Tudi Nau je, tako kot Beal, poskusil pokazati, zakaj je preiskovanje z minimaksom koristno. Tudi on je sestavil matematični model, ki je kompleksnejši od Bealovega in posledično bolj podoben programom za igranje iger. Razlik je precej, ena bolj pomembnih je število hevrističnih vrednosti. V Bealovem modelu se uporabljata le zmaga in poraz, Nau pa dopušča končno mnogo različnih hevrističnih vrednosti. Modela tudi različno obravnavata napako. Nau je upošteval napako poteze, Beal pa napako položaja. Kljub razlikam in večji realističnosti se je Nauov model tudi izkazal za patološki.

Razhajanja med teoretičnimi modeli in uporabo minimaksa v praksi je povzročilo veliko zanimanje med raziskovalci. Začeli so iskati lastnosti re-

alnih iger, ki prispevajo k uspešnosti minimaksa. Močan vpliv vejitve na patološkost so opazili vsi razikovalci, saj povečanje vejitve ojača stopnjo patologije. Pomemben dejavnik, ki prispeva k odpravi patologije je odvisnost bratskih vozlišč. Raziskovci so uporabljali različne načine vpeljave odvisnosti v modele. Nau [13] in Luštrek [19] so uporabili postopno spreminjanje vrednosti vozlišč od korena do listov. Bratko in Gams [11] sta uporabila delež zanesljivo ocenjenih vozlišč, ki stabilizirajo preiskovanje. Beal [10] je v osnovni model uvedel faktor grozdnja, ki pomeni delež vozlišč, katerih sinovi imajo vsi enako vrednost. Meritve so pokazale, da je grozdenje v šahovskih končnicah dovolj veliko, da je minimaks koristen.

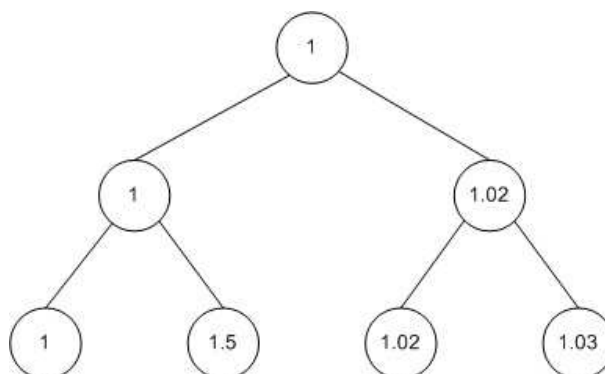
Dognanja o vplivu števila vrednosti na patologijo niso bila enotna. Prvi modeli so uporabljali le dve vrednosti, Bratko in Gams [11] sta naletela na patologijo tudi pri uporabi več vrednosti. Luštrek [19, 21] je pokazal, da večje število vrednosti oslabi patološkost.

Kaluža idr. [17, 20] so sestavili minimaks model, ki upošteva vse naštete parametre in preveri njihov vpliv na pojav patologije. Vpliv vejitve in števila pravih vrednosti na stopnjo patologije so najprej opazovali na neodvisnem modelu. Izkaže se, da vejitev ojača patološkost, število vrednosti v vozliščih (zrnatost) pa jo omili. Osredotočili so se tudi na gradnjo različnih modelov, ki upoštevajo odvisnost bratskih vozlišč in izkaže se, da so različni modeli kvalitativno podobni. S povečevanjem odvisnosti dreves in števila vrednosti v vozliščih pripomoremo k odpravi patologije.

V tej nalogi bomo naredil analizo, ali podobne relacije veljajo tudi v modelu minimin.

2.2 Patologija enoagentnega preiskovanja

Patologijo enoagentnega preiskovanja so odkrili Bulitko idr. [4] šele leta 2003, zato je tudi precej slabše raziskana od patologije minimaksa. Bulitko idr. [4] je patologijo pokazal na preiskovalnem drevesu na sliki 2.1. Prave vrednosti so v drevesu izbrane tako, da je preiskovalno drevo patološko. Hevristična funkcija pa ima obe željeni lastnosti: je popolna in monoton nepadajoča. Hevristična funkcija je monoton nepadajoča, če so vse ocene hevristične funkcije na trenutno pregledani poti večje ali enake hevrističnim ocenam pred njimi, $f(N_j) \leq f(N_{j+1})$. To pa avtor doseže tako, da so hevristične vrednosti enakomerno porazdeljene med statično hevristično oceno starša in pravo vrednostjo vozlišča. Avtor pa ne ponudi razlage, zakaj se patologija pojavi. Podrobnejšo razlago pojava poda Luštrek [7, 8, 19], kjer razloge razdeli v dve skupini:



Slika 2.1: Drevo, na katerem je Bulitko pokazal patologijo enoagentnega preiskovanja.

- domena problema, na katero težko vplivamo
- uporabljena hevristična funkcija, na katero imamo vpliv.

Znotraj domene avtor določi še dve lastnosti, ki vplivata na patologijo. Prva lastnost, ki povečuje patološkost je čimvečja razlika med cenami naslednikov prvonivojskega vozlišča z najmanjšo ceno, v primerjavi z razliko naslednikov ostalih vozlišč na prvem nivoju. Druga lastnost pa je razlika med pravimi vrednostimi najcenejšega vozlišča na nivoju 1 in ostalimi brati tega vozlišča. Manjša kot je ta razlika, bolj se patologija okrepi in obratno, večja kot je ta razlika, bolj se patologija oslabi. Obe lastnosti lahko opazimo na preiskovalnem drevesu na sliki 2.1. V večini primerov na te dve lastnosti ne moremo vplivati, kar pa ne velja za izbiro hevristične funkcije. Luštrek [7, 8, 19] predlaga, da je potrebno izbrati monotono nepadajočo hevristično funkcijo. Optimističnost pa v večini primerov nima znatnega vpliva.

Bulitko je kasneje [5] ugotovil, da se patologija pojavi tudi izven sintetičnih preiskovalnih dreves, saj je tudi pri proučevanju igre osmih ploščic z uporabo hevristične funkcije predstavljene z umetno nevronske mrežo naletel na patologijo. Ločil je tri vrste patologije:

- pri večji globini preiskovanja najdemo slabše rešitve,
- pri večji globini preiskovanja pregledamo manj vozlišč,
- pri večji globini preiskovanja pogosteje izberemo neoptimalno prvo potezo.

Piltaver idr. [24, 25] so proučeval patologijo v igri osmih ploščic. V igro so dodali nove poteze in tako omogočil spreminjanje vejitve. Pri različicah igre

so opazili 13 skupin iger s faktorji vejitve med 1,56 in 4,44. Omenjeni faktorji vejitve so povprečni, saj nobena izmed iger nima uniformnega faktorja vejitve. Do jasnih zaključkov glede vpliva faktorja vejitve avtor ni prišel. Možno je zaznati, da večja vejitev povzroči večjo patologijo. Naslednji parameter, ki ga je moč spreminjati, je zrnatost hevristične funkcije. Izkaže se, da večja zrnatost hevristične funkcije zavira patologijo. Odvisnost bratskih vozlišč pa ni bilo mogoče neposredno spreminjati, saj je odvisnost vozlišč lastnost problema. Vpliv odvisnosti bratskih vozlišč so obravnavali na vseh različicah igre hkrati in tako dobil več izmerjenih vrednosti faktorja grozdenja za posamezne igre. Dobljene vrednosti so uredili po naraščajočem faktorju grozdenja in iz grafa razbrali, da večja odvisnost vozlišč povzroči manjšo patologijo preiskovanja.

S patologijo v igri osmih ploščic sta se ukvarjala tudi Sadikov in Bratko [9], kjer sta ugotovila, da je pesimistična hevristična funkcije učinkovitejša od optimistične, saj patologijo oslabi.

Zgrajeni minimin model je v določenih delih podoben modelu igre osmih ploščic [24, 25], z nekaj pomembnimi razlikami. Minimin model je v osnovi bolj splošen, saj je faktor vejitve uniformni za celotno drevo in ni navzgor omejen, torej ga je mogoče poljubno spreminjati. V model smo morali vpeljati odvisnost dreves, ki je v igri osmih ploščic že dana. Zgrajeni model omogoča poljubno spreminjanje stopnje odvisnosti dreves, ne glede na izbrano vejitev in zrnatost. V model vključimo tudi spreminjanje števila vrednosti v listih dreves.

Poglavje 3

Neodvisni model

V tem poglavju bomo začeli z obravnavo neodvisnega modela preiskovanja, torej modela s takimi lastnostmi, ki so raziskovalce pripeljali do odkritja patologije minimaksa. Na začetku poglavja so razložene osnovne predpostavke modela in način gradnje dreves. Predstavljeni so rezultati meritev in interpretacija teh rezultatov.

3.1 Opis modela

Gradnja neodvisnega modela se zgleduje po modelih opisanih v [17, 18, 21], s to bistveno razliko, da je prirejen minimin preiskovanju in nekaterim dodatnim omejitvam domene. Model omogoča uporabo različnega števila pravih in hevrističnih vrednosti v vozliščih drevesa. Glavne predpostavke našega neodvisnega modela lahko opišemo s šestimi glavnimi točkami:

1. Drevo igre ima konstanto vejitev b in globino d_{max} .
2. Število pravih in hevrističnih vrednosti vozlišč drevesa podamo s parametrom $g \in [2, \infty]$. Prave vrednosti v vozliščih lahko zavzamejo katerokoli realno vrednost z intervala $[0, 1]$, hevristične pa realne vrednosti z intervala $[0, 1.5]$. Parameter g poimenujemo *zrnatost* (angl. *granularity*).
3. Vrednosti vozlišč znotraj enega nivoja so med seboj neodvisne.
4. Statično napako hevristične funkcije modeliramo z Gaussovimi šumom s standardnim odklonom $\sigma = 0.1$, ter je neodvisna od prave vrednosti vozlišča in globine preiskovanja.

5. V vozlišča na višjem nivoju, med izvajanjem algoritma, ne prenašamo minimum sinov, temveč maksimum. S tem se izognemo nepraktičnim negativnim vrednostim, ki bi jih lahko vrnila hevristična funkcija. To je ekvivalentno minimin preiskovanju, le da je manj pogosto.
6. Z realno vrednostjo $p \in [0, 1]$, ki predstavlja verjetnost pojavitve maksimalne vrednosti v sinovih korena, določimo delež maksimalnih vrednosti v listih preiskovalnega drevesa.

V tem delu se bomo osredotočili na napako poteze. Zanimal nas bo delež izbranih potez v korenu, ki ne vodijo v sina z najboljšo pravo vrednostjo. Izbrana poteza je napačna, če obstaja druga poteza, ki vodi v vozlišče z večjo pravo vrednostjo kot je vrednost vozlišča v katerega pridemo z izbrano potezo. Kar lahko matematično zapišemo $\exists a^* \neq a [f^*(\delta(s, a^*)) < f^*(\delta(s, a))]$, pri čemer je δ funkcija, ki nas iz nekega vozlišča (s) z uporabo določene akcije (a) privede v drugo vozlišče (s). $f^*(s)$ je ocenitvena funkcija, ki uporablja pravo ceno najcenejše poti od vozlišča s do cilja ($h(s)$). Verjetnost take napake, pri določeni globini preiskovanja d in hevristični funkciji h , označimo z $E(h, d)$. Sedaj lahko definiramo patološkost p z naslednjim izrazom:

$$p = \frac{E(h, d_2)}{E(h, d_1)}, 1 \leq d_1 < d_2 \leq d_{max} \quad (3.1)$$

Do patološkega obnašanja pride, kadar velja:

$$\exists d_1 < d_2 : [E(h, d_1) < E(h, d_2)] \quad (3.2)$$

Iz 3.2 lahko sklepamo, da vrednost $p > 1$ pomeni prisotnost patologije. Nasprotno pa je patologija odsotna, ko je $p < 1$, saj je v tem primeru napaka pri globljem preiskovanju manjša kot pri plitvejšem. Izraz 3.1 bomo uporabili pri računanju patologije v našem modelu.

Ugotavljanje patologije v modelu začnemo z gradnjo sintetičnega preiskovalnega drevesa. Zgrajeno drevo je podane vejitve b in globine d_{max} . Patološkost smo opazovali pri izbranih globinah preiskovanja $d_1 = 1$ in $d_2 = 5$. Globina preiskovanja je izbrana tako, da omogoča čimbolj učinkovito izvajanje simulacije. Povečanje globine bi namreč znatno povečalo čas izvajanja, na sam pojav patologije pa ne bi imelo večjega učinka. S prirejenim algoritmom minimin, ki na vsakem nivoju vrne maksimalno vrednost, se sprehodimo čez drevo in ko pridemo do končne globine d_{max} , listom dodelimo eno izmed g naključnih realnih vrednosti z intervala $[0, 1]$. Ob vračanju algoritma pa se izračunajo tudi prave vrednosti notranjih vozlišč. Pri tem si zapomnemo b pravih vrednosti na

prvem nivoju drevesa v_r , s katerimi bomo v nadaljevanju ugotavljali napako poteze. Statično napako hevristične funkcije simuliramo tako, da pri preiskovanju do določene globine d pravim vrednostim vozlišč na nivoju d dodamo šum in s tem dobimo hevristične vrednosti. Za dodajanje šuma uporabimo Gaussov šum [22] s standardnim odklonom $\sigma = 0,1$. Izračunane hevristične vrednosti ponovno prenesemo do korena drevesa in jih shranimo v tabelo v_n . S pomočjo prej prenesenih pravih vrednosti v_r izračunamo $E(h, d_1)$, $E(h, d_2)$ in z uporabo izraza 3.1 še patologijo p . Postopek računanja $E(h, d_1)$ je formalno napisan v algoritmu 2 – *error_level*(o, n).

Algoritem 2 *error_level*(V_r, V_n)

Require: real backed-up values from level 1 V_r , heuristic backed-up values from level 1 V_n

Ensure: error e

origMax \leftarrow maximum value in set V_r

maxInd \leftarrow find indexes of max values in set V_n

count \leftarrow 0

for all *elem* **in** *maxInd* **do**

if $V_r[\textit{elem}] == \textit{origMax}$ **then**

count \leftarrow *count* + 1

end if

end for

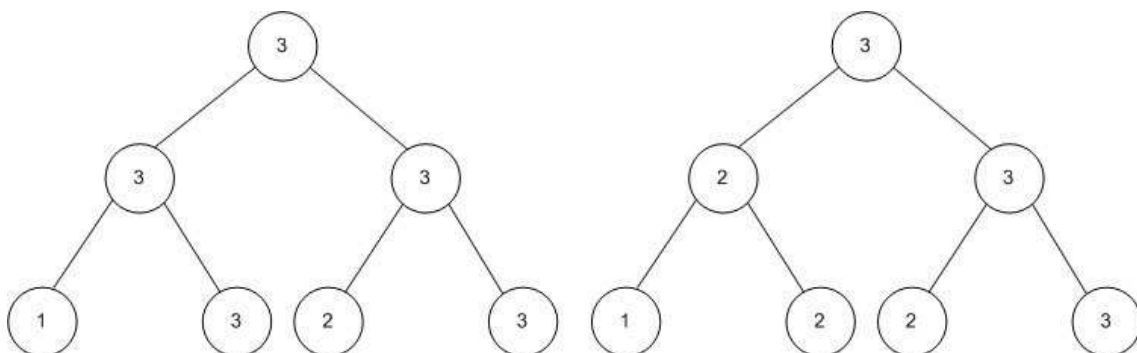
$e \leftarrow 1 - (\textit{count}/\textit{size}(\textit{maxInd}))$

Naj podamo primer za zgornji algoritem. Pri vejitvi drevesa $b = 3$ in zrnatosti $g = 2$ so se na prvi nivo prenesle prave vrednosti $[0, 1, 1]$. Prenesene hevristične vrednosti pa so $[1, 1, 0]$. Vidimo, da je maksimalna vrednost 1. Ker se prenesena hevristična vrednost ujema le v eni od dveh možnih pozicij, je napaka $\frac{1}{2}$.

Opisani postopek gradnje sintetičnih drevesa smo v sklopu simulacije večkrat ponovili, običajno tridesettisočkrat. Vsakič smo izračunali količine, ki nas zanimajo, in rezultate povprečili. Tako smo dobili povprečene vrednosti za raven patologije in se izognili nihanjem, ki jih vnaša gradnja naključnih sintetičnih dreves.

Naš namen proučevanja parametrov, ki vplivajo na pojav patologije, nas privede tudi do zrnatosti vozlišč preiskovalnega drevesa. Ta lastnost se je izkazala za zelo pomembno pri analizi minimaks patologije, zato smo jo na podoben način vpeljali tudi v našo simulacijo. Število pravih vrednosti v modelu uravnavamo s parameterom g . Pri prvem modelu smo interval $[0, 1]$

razdelili kar na g enakih podintervalov. To pomeni, da je bil delež maksimalnih vrednosti v listih drevesa $\frac{1}{g}$. Tak model je bil za proučevanje patologije neprimeren, saj je bilo vseeno, katero potezo v korenu izberemo. Predpogoj za patologijo je namreč napaka poteze na prvem nivoju drevesa. V naslednjem koraku smo sledili razmisleku, da delež maksimalnih vrednosti v listih drevesa vpliva na pojav patologije na sledeč način: če je vrednosti preveč, do patologije sploh ne more priti, saj algoritem prenese do korena drevesa samo maksimalne vrednosti. Na sliki 3.1 si lahko bolj nazorno ogledamo zgornjo ugotovitev. Drevo na levi sliki je nepatološko, saj tudi napake pri hevristični oceni ne morejo pokvariti odločitve za ustrezno potezo. Pri desnem drevesu pa se lahko zgodi napaka poteze. V primeru, da se na podlagi hevristične ocene odločimo za levo poddrevo, smo naredili napako poteze, saj je število maksimalnih vrednosti manjše. Takrat lahko hevristična funkcija pokvari pravo vrednost v maksimalno in patologija je bolj verjetna.



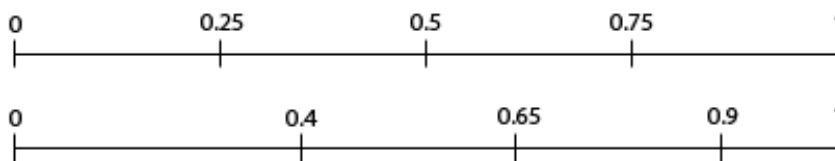
Slika 3.1: Levo preiskovalno drevo je vedno nepatološko. Desno pa je v določenih primerih lahko patološko.

V drugem modelu smo z namenom preiskusa zgornje ugotovitve dodali parameter, ki uravnava število maksimalnih vrednosti v listih preiskovalnega drevesa. Ta poteza se je izkazala za učinkovito, saj se je v modelu ob primerni izbiri vrednosti dodanega parametra pojavila patologija. Taka rešitev pa se je izkazala za nepraktično, saj je bilo potrebno najti primerno število maksimalnih vrednosti za vsako vejitev posebej. Pri iskanju ustrezne rešitve smo se zatekli k analizi patologije minimaksa. Tam namreč parameter c_b uravnava verjetnost poraza za igralca na potezi in posledično pojav patologije. Podobno mero p_{leaves} smo uvedli tudi v našem modelu. Tudi ta namreč predstavlja verjetnost poraza igralca na potezi. Definiramo pa jo z izrazom 3.3:

$$p_{leaves} = \sqrt[N]{1 - p_1}, N = b^{d_{max}-1} \quad (3.3)$$

Poglejmo še razmislek, ki je privedel do tega izraza. Preveč maksimalnih vrednosti v sinovih korena prepreči pojav patologije, zato definiramo mero p_1 , ki predstavlja delež maksimalnih vrednosti na prvem nivoju drevesa. Iz tega podatka želimo izračunati kolikšen mora biti delež nemaksimalnih vrednosti v listih (p_{leaves}), da dosežemo v korenu delež p_1 maksimalnih vrednosti. Ravno to pa izračunamo z izrazom 3.3.

Z uporabo 3.3 izračunamo p_{leaves} za vsak b posebej, razdelimo interval $[0, 1]$ na g enako velikih podintervalov, ter popravimo meje podintervalov glede na izračunano vrednost. Na vrednost p_{leaves} premaknemo mejo zadnjega intervala, vendar le v primeru, ko je trenutna meja manjša kot p_{leaves} . V primeru $b = 2$, $p_1 = 0.5$ znaša parameter $p_{leaves} = 0.9576$. Podintervala, ki jih tako dobimo, sta $[0, 0.9576)$ in $[0.9576, 1]$. Na sliki 3.2 lahko zgoraj vidimo razdelitev intervala $[0, 1]$ na štiri enako velikih podintervalov, spodaj pa poravnavo mej na p_{leaves} .



Slika 3.2: Poravnava mej podintervalov.

Vsakemu podintervalu pa določimo predstavnika, ki je kar srednja vrednost intervala. Vsem vrednostim, ki padejo znotraj izbranega podintervala priredimo vrednost predstavnika. To storimo zato, da vozlišča lahko zavzamejo natanko g vrednosti.

Naj na koncu omenimo še problem, na katerega smo naleteli pri preizkušanju modela. Pri dodajanju šuma pravim vrednostim vozlišč se lahko zgodi, da postane hevristična vrednost večja od ena. V tem primeru smo tako vrednost popravili kar na vrednost skrajno desnega intervala. Tako ravnanje pa se je izkazalo za problematično, saj povečuje število maksimalnih hevrističnih vrednosti in s tem stopnjo patologije. Z namenom odprave tega problema smo uvedli še $\lceil \frac{g}{2} \rceil$ dodatnih podintervalov. Tudi tem podintervalom popravimo meje in določimo predstavnike podobno kot zgoraj. Vse hevristične vrednosti, ki zavzamejo vrednosti večje od ena, uvrstimo v ustrezne podintervale. Ta problem je specifičen za maksmaks model, saj zaradi maksmaks prenašanja ohranjamo

večje vrednosti, medtem ko pri minimaks algoritemu, prenesene vrednosti po drevesu težijo k srednji vrednosti intervala $[0, 1]$. V naslednjem poglavju bomo prikazali učinke dodajanja dodatnih intervalov na stopnjo patologije.

3.2 Rezultati poskusov

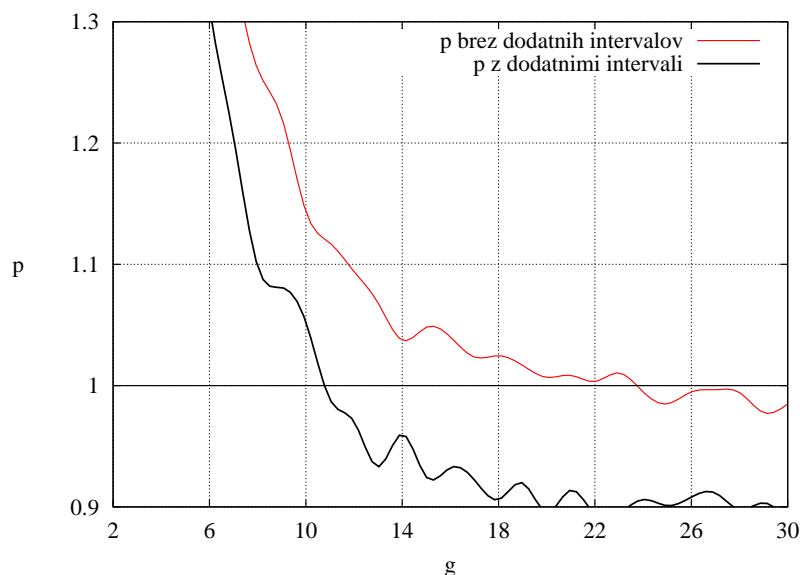
Patološkost neodvisnih preiskovalnih dreves smo analizirali s simulacijo Monte Carlo. Za vsak nabor parametrov b iz množice $\{2, 3, \dots, 10\}$ in g iz množice $\{2, 3, \dots, 40\}$ smo tvorili 30.000 dreves globine $d_{max} = 5$. Patološkost smo ugotavljali tako, da pri globini preiskovanja d_{max} in 1 iz povratnih pravih in hevrističnih vrednosti izračunamo napako v korenu ter jo povprečimo čez vseh 30.000 dreves. Do patološkega obnašanja pride, ko je ta napaka pri globljem pregledovanju večja kot pri plitvejšem. Čas računanja pri povečevanju vejitve narašča eksponentno zato smo parameter vejitve omejili na zgornji interval. Število dreves pri vsaki meritvi pa smo pustili enako, saj da večje število dreves boljše zanesljivost rezultatov. Izbrani nabor parametrov se je izkazal kot primeren, saj smo že s tako omejenimi parametri našli v dobljenih podatkih lastnosti, ki so nas zanimale.

Najprej si pogledjmo vpliv dodatnih $\lceil \frac{g}{2} \rceil$ podintervalov na pojav patologije. Za nabor parametrov $b = 2$, $g \in \{2, 3, \dots, 40\}$ smo dvakrat pognali simulacijo. Prvič smo hevristične ocene, ki so zavzele vrednosti večje od ena priredili zadnjemu podintervalu, drugič pa smo jih uvrstili v enega od ustreznih dodatnih podintervalov. Na sliki 3.3 lahko vidimo krivulji stopnje patološkosti za oba primera.

Opazimo, da patologija vztraja dlje časa v primeru, ko dodatnih intervalov nimamo. V vseh naslednjih simulacijah bomo imeli dodatne intervale omogočene.

Sedaj si pogledjmo rezultate simulacije s celotnim naborom parametrov. Na sliki 3.4 vidimo grafični prikaz izračunane stopnje patološkosti neodvisnih dreves ($s = 0.0$). Ta je na grafu prikazana v odvisnosti od zrnatosti g in vejitve b . Navpično os grafa smo omejili na vrednost $p = 10$, saj tako lahko natančneje ugotovimo smernice pri spreminjanju posameznih parametrov.

Pri konstantni zrnatosti se s povečevanjem vejitve povečuje stopnja patološkosti. To lahko razberemo iz naraščajočih krivulj, ki so prečne na g os. Ta značilnost je bolj izrazita pri manjši zrnatosti. Pri večjih vejitvah je intenzivnost p ponazorjena z živo rdečo barvo, kar nakazuje visoko patološkost. Pri večjih zrnatostih pa je prehod bolj umirjen. Podoben pojav je mogoče zaznati pri konstantni vejitvi in povečani zrnatosti. Pri vrednosti

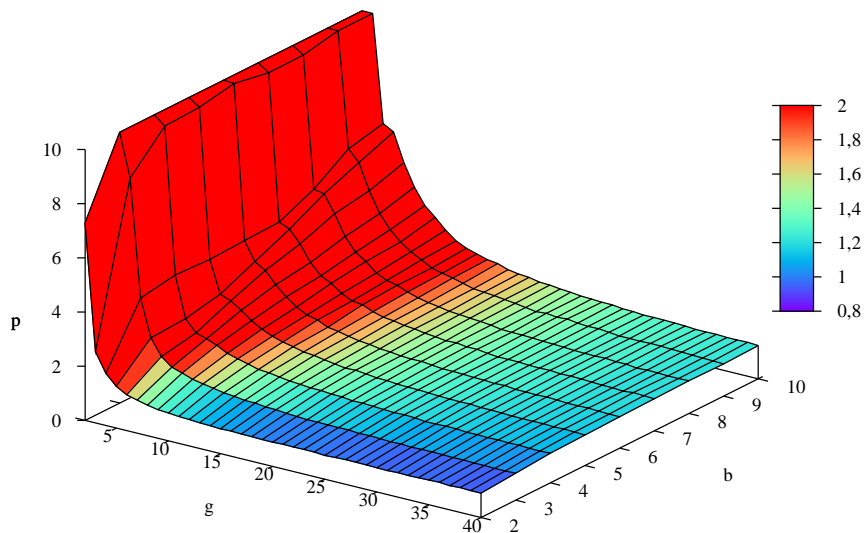


Slika 3.3: Prikaz stopnje patološkosti p v primeru, ko se hevristične ocene lahko porazdelijo v dodatne podintervale in takrat, ko jih umestimo v zadnji podinterval.

$g = 2$ je model namreč zelo patološki ($p > 10$), s povečevanjem zrnatosti patološkost relativno hitro pada in se približuje proti $p = 1$. S povečevanjem vejitve se število pravih vrednosti, potrebnih za odpravo patologije, eksponentno povečuje. Ta pojav je nazorno prikazan v tabeli 3.1. Vrednosti v tabeli smo pridobili z večkratnim izvajanjem simulacije. Najprej smo za posamezno vejitev tvorili manjše število dreves (2.000) in za vsak nabor parametra g iz množice $\{2, 3, \dots, 1000\}$ izračunali patološkost. Tako smo določili okvirno območje mejnih vrednosti, ki smo ga nato natančneje pregledali z večjim številom dreves. Vrednosti so izračunane le do vejitve $b = 5$, saj zrnatost zelo hitro narašča.

p/b	2	3	4	5
0,98	12	42	171	x
1,00	11	28	77	184
1,02	10	21	64	114

Tabela 3.1: Število pravih vrednosti g , pri katerih patološkost doseže vrednosti $p = 0,98$, $p = 1$, $p = 1,02$.



Slika 3.4: Prikaz stopnje patološkosti neodvisnih dreves. Prikazana je v odvisnosti od zrnatosti g in vejitve b

Dobljeni rezultati so kvalitativno primerljivi z minimaks modelom [17], saj se s povečevanjem zrnatosti patološkost zmanjša, povečevanje vejitve pa patološkost okrepi. Opazimo pa pomembno razliko med modeloma. Število pravih vrednosti g , potrebnih za odpravo patologije, se pri minimin modelu z vejitvijo veliko hitreje povečuje.

Poglavje 4

Odvisni modeli

V prejšnjem poglavju smo opisali gradnjo naključnih preiskovalnih dreves. Kot je bilo pričakovano so taka drevesa patološka. Naslednja naloga je najti primeren način gradnje odvisnih ter delno odvisnih dreves. Taka drevesa so v praksi namreč bolj pogosta, saj so pri realnih igrah sosednja vozlišča precej podobna. Z eno odigrano potezo se namreč položaji bistveno ne spremenijo.

4.1 Popolnoma odvisna drevesa

Raziskovalci so že pred časom ugotovili, da so realne igre večinoma nepatološke ravno zaradi podobnosti med sosednjimi vozlišči v drevesu igre [10, 11]. Vpliv podobnosti na stopnjo patološkosti pa se je pretežno ugotavljalo pri iskanju po minimaks drevesih. V nadaljevanju bomo predstavili nekaj načinov gradnje odvisnih dreves in na izbranem postopku preverili vpliv podobnosti na pojav patologije.

Eden izmed možnih načinov gradnje odvisnih dreves je s porazdeljevanjem pomožnih vrednosti. Tak način je uporabil Luštrek [19] pri proučevanju odvisnih minimaks dreves. Gradnjo popolnoma odvisnega drevesa začnemo pri korenu. Temu priredimo neko pomožno vrednost, ki se uporablja kot seme (angl. *seed*) za določitev pomožnih vrednosti sinov korena in je porazdejena okrog starša z normalno porazdelitvijo. V naslednjem koraku zgornji postopek rekurzivno ponavljamo na vseh naslednikih korena, dokler ne dosežemo nivoja d_{max} . Tako dobljene pomožne vrednosti v listih na nivoju d_{max} postanejo prave vrednosti, prave vrednosti notranjih vozlišč pa izračunamo z maksmaksom (ali miniminom) iz pravih vrednosti listov. Na ta način dosežemo, da so si bratska vozlišča najbolj podobna, bolj oddaljena vozlišča pa manj.

Naslednji način vpeljave odvisnosti je model z normalizirano vsoto [14].

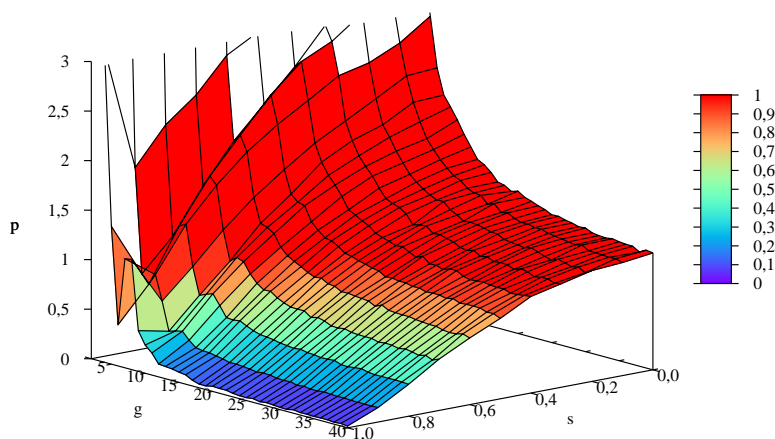
Osnovna ideja izvira iz Nauovega načina vpeljave odvisnosti v Pearlovo igro. Povezavam v drevesu igre je naključno pripisal 1 ali -1 in za vsak list je seštel vrednosti povezav do korena. Tistim listom, pri katerih je vsota povezav pozitivna, je pripisal 1, ostalim pa 0. Vpeljava take odvisnosti je odpravila patologijo v Pearlovi igri. Tak princip pa ni mogoče uporabiti v večvrednostnem modelu, zato je Kaluža [20] Nauov pristop prilagodil. V korenu drevesa se definira pomožna vrednost ζ_0 . Sinovom v_{ij} vozlišča v_i se dodeli pomožna vrednost tako, da starševi vrednosti prištejemo normalno porazdeljeno spremenljivko s srednjo vrednostjo 0. To ponavljamo, dokler ne izračunamo pomožnih vrednosti listom drevesa. Nato poiščemo ekstremne vrednosti v listih in normaliziramo vse liste na interval $[0,1]$.

Opisana postopna modela sta si po načinu gradnje precej podobna, saj gradnjo začnemo od korena navzdol in na koncu postopka pripišemo listom prave vrednosti. Postopek, ki smo ga izbrali za gradnjo odvisnih dreves, uporablja povsem drugačen pristop in temelji na urejanju naključnih vrednosti. Tak model se je uporabilo tudi pri analizi patologije minimaksa [17, 18], kjer se je izkazalo, da so rezultati kvalitativno podobni postopnima modeloma. Gradnja popolnoma odvisnih dreves se pri izbranem modelu začne od spodaj, torej pri listih drevesa. Vsem listom dodelimo pomožne naključne vrednosti, ki jih nato sortiramo po velikosti. Sedaj imajo listi od leve proti desni naraščajoče pomožne vrednosti. Dobljene vrednosti nato še preslikamo v ustrezne razrede zrnatosti na enak način, kot smo to storili v poglavju 3.1. Tako smo zgradili popolnoma odvisno drevo, kateremu pripišemo podobnost $s = 1$.

4.2 Delno odvisna drevesa

Sedaj ko imamo zgrajena neodvisna in popolnoma odvisna drevesa, nam ostane le še kombiniranje le-teh v delno odvisna drevesa. Prva možnost je vsavljanje listov odvisnega drevesa v neodvisno glede na verjetnost $s = s_0$. Na predzadnjem nivoju drevesa obiščemo vsa vozlišča in jim z verjetnostjo s_0 dodelimo sinove vozlišča odvisnega drevesa. Z verjetnostjo $1 - s_0$ pa pustimo vozlišču liste z naključnimi pravimi vrednostmi. Na ta način zamenjamo delež s_0 listov drevesa. Izkaže se, da tak način gradnje delno odvisnih dreves v minimaks modelu ni ravno posrečen [17]. Stopnja patologije namreč s podobnostjo narašča in doseže višek pri $s = 0,3$. Pri tej vrednosti pa ne izgine niti s povečevanjem zrnatosti. Presenetljivo pa do tega pojava pri minimin modelu ne pride. Na sliki 4.1 lahko namreč opazimo da patološkost vzdolž osi s monotono pada. Razlika je najverjetneje posledica same narave modelov. Pri

minimaksu namreč model teži k srednjim vrednostim v drevesu, minimin pa k maksimalnim. Zato tudi vrednosti, ki jih dodajamo na sredino, nimajo takega vpliva, kot pri minimaksu.



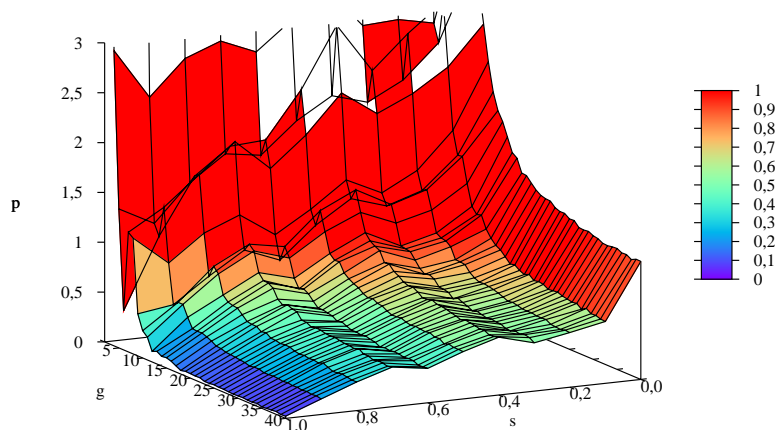
Slika 4.1: Stopnja patologije pri vejitvi $b = 6$ in naključnem vstavljanju listov odvisnega drevesa.

Z izbiranjem ustreznega načina gradnje delno odvisnih dreves bi lahko kar prenehali, saj smo že našli ustrezen način. Vendar smo se zaradi ugotovljenih razlik med minimaks in minimin modeloma odločili, da pregledamo vse načine gradnje in jih primerjamo med sabo.

Vzroke opisanemu problemu vztrajanja patologije pri $s = 0,3$ so pri minimaks modelu pripisali premalo dodanim vrednostim na robove neodvisnega drevesa. V izogib tej težavi je potrebno zamenjati več robnih vrednosti in manj srednjih. V prvem poskusu se zamenja delež $\frac{s}{2}$ robnih vrednosti neodvisnega drevesa z istoležečimi listi odvisnega drevesa. Tako zgrajeno drevo ima večjo odvisnost na robovih drevesa, s povečevanjem parametra s pa se odvisnost povečuje tudi v srednjem delu. Pri tem pa moramo biti pozorni na dejstvo, da so listi odvisnega drevesa urejeni naraščajoče. Z zgoraj opisanim dodajanjem listov pa povečamo verjetnost izbire zadnje veje, saj vsebuje večje vrednosti. Problem nastane, ko zadnja veja ne vsebuje prave odločitve in tako favoriziramo napačno vejo, kar še okrepi patološko obnašanje. Zato pred gradnjo delno odvisnega drevesa uredimo veje neodvisnega drevesa na prvem

nivoju. Dejanskih sprememb v drevesu ne delamo, le veje obravnavamo v drugačnem vrstnem redu.

Tak način gradnje pri minimaks modelu odpravi težavo s stalno patološkostjo, vendar je odvisnost zgrajenih dreves premočna, saj patologija strmo pade z večanjem podobnosti in pri $s = 0,3$ že popolnoma izgine. Na sliki 4.2 lahko vidimo obnašanje patologije na minimin modelu. Opazimo lahko podobno obnašanje, kot pri minimaks modelu, a z nekaj razlikami. Prvo kar opazimo, je strm padec patološkosti že pri majhni stopnji odvisnosti. Pri $s = 0,1$ z naraščanjem zrnatosti zaznamo hiter padec stopnje patološkosti krepko pod $p = 1$, ki se kasneje ustali nekje okoli $p = 0,6$. Patološkost se pri $s = 0,4$ rahlo poveča in nato počasi pada. Pri $s = 1$ in višji zrnatosti pa skoraj popolnoma izgine.



Slika 4.2: Stopnja patologije pri vejitvi $b = 6$ in dodajanju odvisnih listov na robove neodvisnega drevesa.

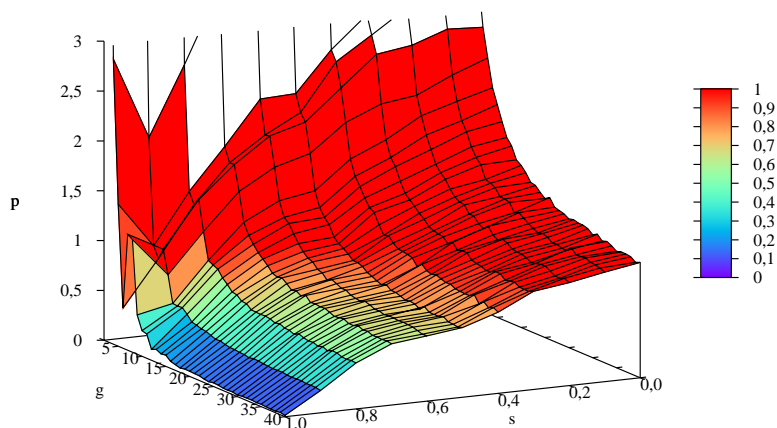
Iz opisanega lahko vidimo, da tudi pri tem načinu gradnje prihaja do razlik med minimaks in minimin modeloma. Patološkost pri minimin modelu sicer krepko pade že pri majhni odvisnosti dreves, vendar pa z dodajanjem podobnosti počasi pada in ne popolnoma izgine.

Zamenjava deleža $\frac{s}{2}$ skrajnih listov na obeh straneh torej povzroči preveliko odvisnost zgrajenih dreves. Za ublažitev tega pojava bi potrebovali drugačno, bolj postopno zamenjavo listov. Naslednji način predvideva zamenjavo deleža

s vseh listov tako, da je verjetnost zamenjave listov na robu drevesa večja kot za liste na sredini. Tako bi predvidoma dosegli zmernejši prehod med odvisnimi in neodvisnimi drevesi.

Definirati je potrebno še porazdelitveno funkcijo, ki bi imela zgoraj opisano lastnost. Definirajmo funkcijo $H_s(i)$, ki predstavlja verjetnost zamenjave i -tega lista v neodvisnem drevesu z istoležečim listom v odvisnem drevesu. Najbolj enostavna funkcija je trikotniška porazdelitev [17] in je definirana za vsak s posebej.

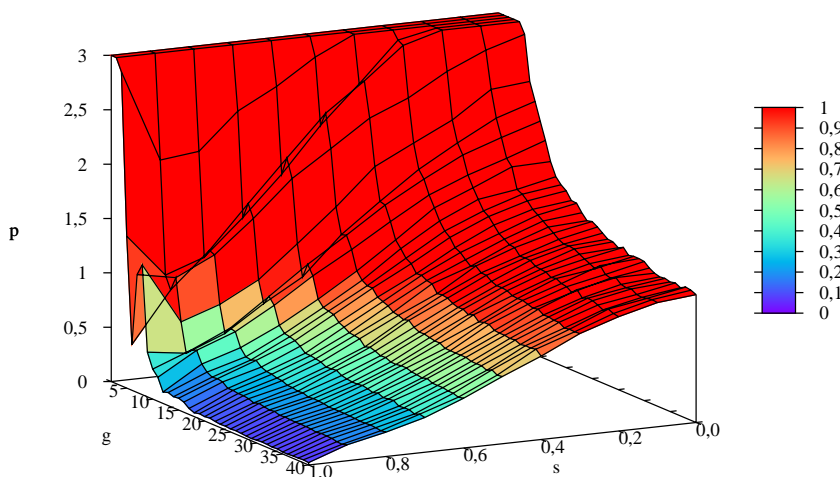
Meritve na minimaks modelu so pokazale, da tak način gradnje dreves res zmanjša odvisnost in omogoča zmernejši prehod, vendar se pri vrednosti $s = 0,5$ prelomi in tam lahko zaznamo močno spremembo v naklonu stopnje patološkosti. Na sliki 4.3 grobega preloma sicer ne opazimo, vendar se minimin model pri $s = 0,5$ obnaša čudno, saj stopnja patologije pade, kar je na sliki razvidno kot vboklina na ravnini.



Slika 4.3: Stopnja patologije pri vejitvi $b=6$ in dodajanju odvisnih listov z uporabo trikotniške porazdelitve $H_s(i)$.

Do odstopanja pri odvisnosti $s = 0,5$ pride zaradi izbrane porazdelitvene funkcije. Pri tej vrednosti se namreč začne dvigovati srednja točka in funkcija postane nezvezna. Se pa problem porazdelitvene funkcije odraža drugače na minimaks in minimin modelu. Pri prvem se stopnja patologije rahlo poveča, pri drugem pa rahlo zmanjša. V izogib temu problemu moramo izbrati novo

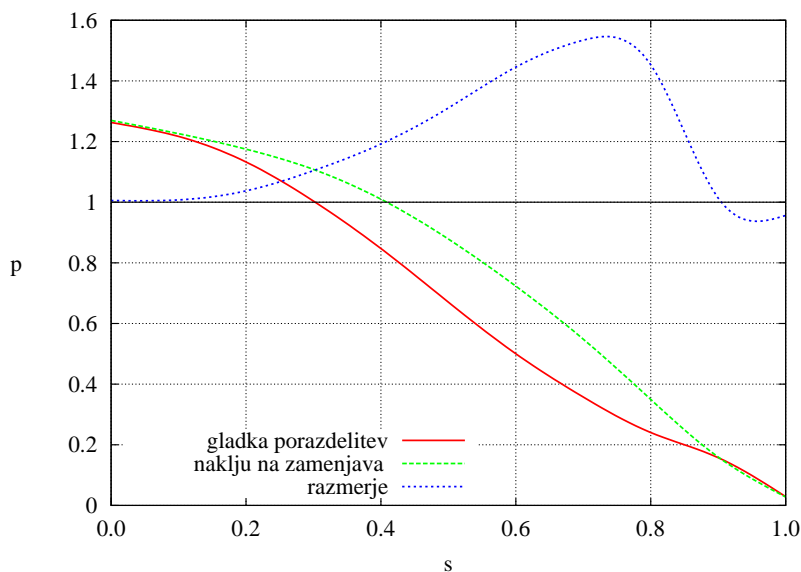
porazdelitveno funkcijo, s katero se izognemo dvigovanju srednje točke porazdelitve $H_s(i)$. Izpeljemo jo iz trikotne porazdelitve tako, da list i odvisnega drevesa dodamo na istoležno mesto v neodvisno drevo z verjetnostjo $H_{0,5}(i)$ [17]. To ponavljamo, dokler drevo ne vsebuje s odvisnih vozlišč. S tem dosežemo, da se več vrednosti doda na robove drevesa. Vrednosti nove porazdelitve $Hg_s(i)$ smo pri posamezni stopnji odvisnosti izračunali v ločeni simulaciji in med gradnjo delno odvisnih dreves uporabili izračunane vrednosti. Tako dosežemo hitrejšo simulacijo računanja stopnje patologije, saj se izognemo sprotnemu računanju vrednosti porazdelitve. Na sliki 4.4 vidimo potek stopnje patologije pri uporabi gladke porazdelitvene funkcije. Graf sedaj poteka po pričakovanjih, saj stopnja patološkosti enakomerno pada s povečevanjem odvisnosti. Graf je zelo podoben tistemu na sliki 4.1, le z manjšim odstopanjem pri srednjih vrednostih odvisnosti.



Slika 4.4: Stopnja patologije pri vejitvi $b = 6$ in dodajanju odvisnih listov z uporabo gladke porazdelitvene funkcije.

Kot zanimivost si pogledjmo primerjavo med modeloma na sliki 4.1 in 4.4 z razmerje med stopnjo patološkosti obeh modelov. Za 20 vrednosti g iz množice $\{21, 22, \dots, 40\}$ smo izračunali patologijo in jo povprečili. To smo ponovili za vse stopnje odvisnosti s . Tako dobljene vrednosti so prikazane na sliki 4.5. Na graf smo dodali še krivuljo, ki prikazuje razmerje med obema modeloma. Ta

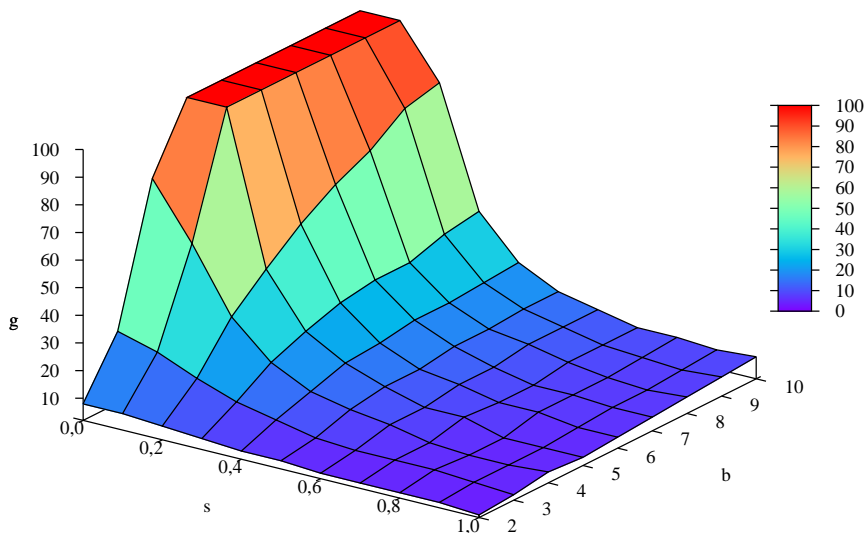
krivulja nam pokaže, da je stopnja patološkosti pri modelu z uporabo gladke porazdelitve manjša, kot pa pri modelu z naključnimi vrednostimi. To je najbolj opazno pri odvisnosti $s = 0.7$, saj je stopnja patologije pri modelu z uporabo naključne porazdelitve, večja za več kot 50%.



Slika 4.5: Povprečne vrednosti stopnje patološkosti za model z naključnim vstavljanjem in gladko porazdelitvijo, ter razmerje med obema krivuljama. Krivulje so prikazane v odvisnosti od stopnje s .

Sedaj imamo ustrezen postopek gradnje delno urejenih dreves. Ostane nam le še poganjanje simulacije za vse naborne vrednosti parametrov b iz množice $\{2, 3, \dots, 10\}$, s iz množice $\{0.0, 0.1, \dots, 1.0\}$ in ugotavljanje vrednosti g , pri kateri patološkost v modelu pade pod mejo $p = 1$. Iz slike 4.6 lahko razberemo podobne značilnosti, kot smo jih zaznali pri grafih postopnih modelov: s povečevanjem podobnosti meja patologije pada. Opazimo tudi, da je za odpravo patologije pri večjih vejitvah potrebna večja zrnatost. Model je kvalitativno primerljiv z minimaks modelom, le da v našem primeru meja patologije pri majhni stopnji podobnosti ne pade tako hitro, kot pri minimaks modelu. Pri stopnji podobnosti $s = 0.2$ je namreč potrebna relativno visoka zrnatost za odpravo patologije.

Na sliki 4.6 je prikazan prostor parametrov vejitve, odvisnosti in zrnatosti ter izmerjena stopnja patologije za posamezne vrednosti. Na sliki je viden padeč patološkosti pri povečevanju odvisnosti. Pojav je najbolj izrazit pri



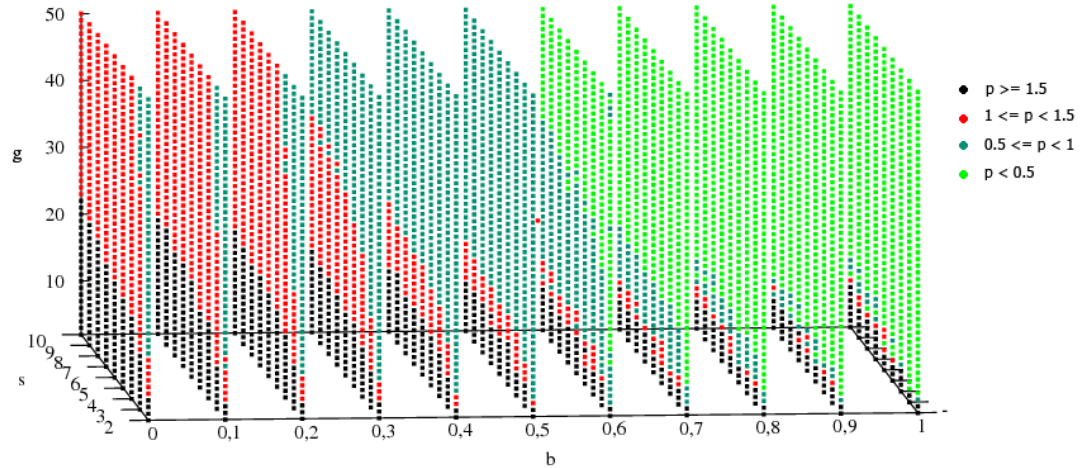
Slika 4.6: Vrednost zrnatosti g , ki je potrebna, da meja patologije pade pod $p = 1$ v modelu z urejanjem naključnih vrednosti.

večjih vejitvah, kjer je jasno viden prehod iz rdeče v svetlo zeleno barvo. Naslednja značilnost je povečevanje stopnje patologije s povečevanjem vejitve, saj se z večevanjem le-te črno obarvana območja širijo proti večjim zrnatostim. Če opazujemo le en stolpec na grafu opazimo še eno značilnost: s povečevanjem zrnatosti stopnja patologije pada.

4.3 Merjenje podobnosti

4.3.1 Faktor grozdenja

Pri gradnji našega modela smo določili, da ima popolnoma podobno drevo odvisnost $s = 1$, naključno zgrajeno drevo iz prejšnjega poglavja pa odvisnost $s = 0$. Glede na delež odvisnega drevesa v neodvisnem pa dobimo vmesne vrednosti podobnosti. Taka poenostavitev bi bila primerna, če se osredotočimo le na proučevanje našega modela. Če pa hočemo primerljivost s prejšnjimi raziskavami in umestitev realnih iger v naš model moramo stopnjo podobnosti izmeriti z eno od standardnih mer. V tem delu smo uporabili faktor grozdenja f (angl. clustering factor). To mero je Beal [10] uvedel za analiziranje stopnje



Slika 4.7: Stopnja patologije v minimin modelu preiskovanja.

odvisnosti sosednjih vozlišč pri razlagi patologije minimaksa. Po definiciji so vsa vozlišča z enako vrednostjo združena v grozde. Mera f je definirana kot delež takih grozdov v neki domeni. Beal je ugotovil, da se pri majhnem f patologija okrepi, pri velikem f pa patologija oslabi. Tako definirane mere pa ne moremo uporabljati na našem večvrednostnem modelu, ker je Bealov model uporabljal le dve vrednosti: zmago in poraz.

To pomanjkljivost je odpravil Sadikov [26], saj je mero prilagodil za uporabo na večvrednostnih modelih. Definicija prilagojene mere predstavlja razmerje med varianco bratov in varianco celotnega drevesa. Formula za izračun faktorja grozdenja je podana z izrazom 4.1.

$$f = \frac{\sqrt{\frac{1}{bN} \sum_{i=1}^N \sum_{j=1}^b (v_{ij} - \bar{v}_i)^2}}{\sqrt{\frac{1}{N} \sum_{i=1}^N (v_i - \bar{v})^2}} \quad (4.1)$$

$$\bar{v} = \frac{1}{N} \sum_{i=1}^N v_i, \quad \bar{v}_i = \frac{1}{b_i} \sum_{j=1}^b v_{ij}$$

N število vseh vozlišč, v_i je prava vrednost i -tega vozlišča, v_{ij} je prava vrednost j -tega naslednika i -tega vozlišča, \bar{v} je povprečna vrednost vseh vozlišč, \bar{v}_i je

povprečje vrednosti naslednikov vozlišča i .

Faktor f zavzema vrednosti med 0 in 1. Manjši f pomeni večjo odvisnost med sosednjimi vozlišči, torej večji delež grozdov v drevesu. Potrebno je poudariti, da definicija faktorja grozdenja podana z izrazom 4.1 ni primerljiva z definicijo, ki jo je podal Beal.

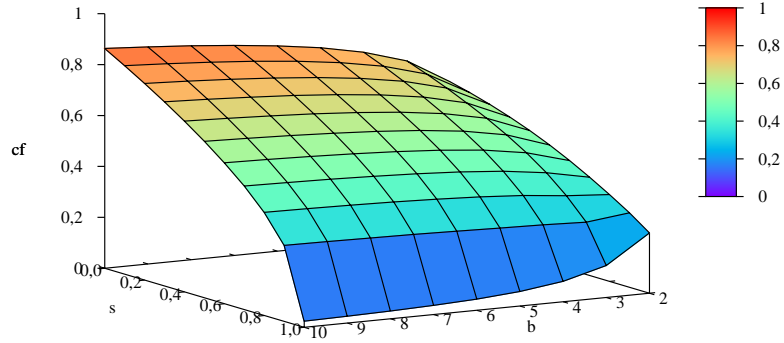
Obstajajo pa tudi druge mere, med katerimi je tudi korelacija med očetom in sinovi. Koreliranost je višja, ko so vrednosti očeta in sinov odvisne, torej blizu skupaj. Obratno pa je koreliranost manjša pri neodvisnih, naključnih vrednostih očeta in sinov. Več o tem pa v naslednjem podpoglavju.

V nadaljevanju bomo pod pojmom faktorja grozdenja razumeli definicijo, ki jo je podal Sadikov.

Na sliki 4.8 je grafično prikazana stopnja odvisnosti, izmerjena s faktorjem grozdenja. Prikazana je v odvisnosti od parametra s in vejitve b . Manjša vrednost cf predstavlja močnejšo odvisnost med vozlišči, saj je tako definiran faktor grozdenja. Na grafu lahko vidimo, kako se z večanjem stopnje s manjša izmerjena vrednost faktorja grozdenja, kar smo tudi pričakovali. Opazimo lahko tudi, da imajo drevesa z večjo vejitvijo močnejšo odvisnost med vozlišči. Pri vejitvi $b = 10$ je namreč faktor grozdenja blizu vrednosti 0. Večji faktor grozdenja pri manjših vejitvah pa je najverjetneje posledica manjšega števila vozlišč, s katerimi računamo standardni odklon. S pomočjo spodnjega grafa lahko primerjamo skladnost našega modela s kakšnim drugim realnim problemom (npr. igro osmih ploščic). Na drevesu igre izmerimo faktor grozdenja, ga primerjamo z izmerjenimi vrednostmi modela in tako dobimo stopnjo s , ki najbolj ustreza odvisnosti realnega problema. Izmerjene vrednosti cf so zbrane v tabeli A.1 v prilogi A.

4.3.2 Korelacija

Naslednji način ugotavljanja stopnje odvisnosti med vozlišči je z izračunom korelacije med očetom in sinovi. Korelacija je ena najbolj pogostih in uporabnih mer v statistiki, saj nam podaja stopnjo sorodnosti med dvema spremenljivkama, ali v našem primeru med vozlišči drevesa igre. Vrednosti spremenljivk, ki so odvisne in posledično blizu skupaj, imajo večjo koreliranost. Pri neodvisnih, naključnih vrednostih pa je koreliranost manjša. Vrednost korelacije lahko zavzame vrednosti na intervalu $[-1, 1]$, pri čemer vrednost 1 predstavlja največjo možno odvisnost vozlišč, neodvisna vozlišča imajo vrednost korelacije blizu 0, vrednost -1 pa predstavlja negativno koreliranost vozlišč. Taka definicija pa je ravno nasprotna faktorju grozdenja, kjer vrednosti blizu 0 označujejo močnejšo odvisnost.

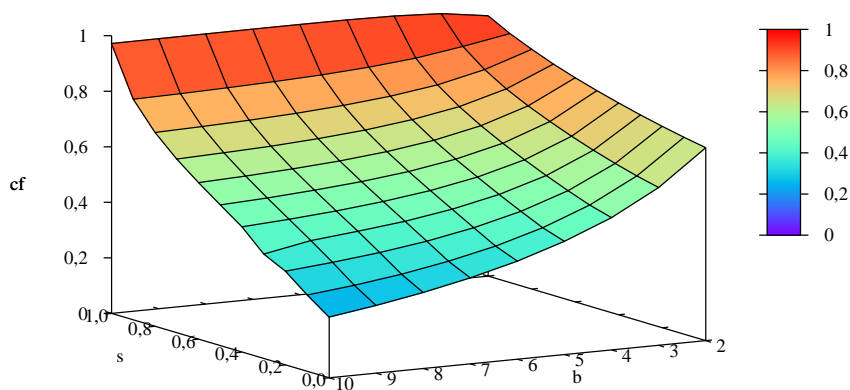


Slika 4.8: Prikaz stopnje odvisnosti v modelu z urejanjem naključnih vrednosti, izmerjene s faktorjem grozdenja.

Korelacijo računamo z izrazom 4.2. Spremenljivka v_i v spodnjem izrazu predstavlja notranje vozlišče, v_{ij} pa sinove tega vozlišča. Za vsako notranje vozlišče drevesa igre dobimo torej b parov (v_i, v_{ij}) . Te vstavimo v izraz 4.2 in izračunamo korelacijo med vrednostmi očetov in sinov.

$$r = \frac{Nb \sum_{i=1}^N \sum_{j=1}^b v_i v_{ij} - b \sum_{i=1}^N v_i \left(\sum_{i=1}^N \sum_{j=1}^b v_{ij} \right)}{\sqrt{Nb^2 \sum_{i=1}^N v_i^2 - \left(b \sum_{i=1}^N v_i \right)^2} \sqrt{Nb \sum_{i=1}^N \sum_{j=1}^b v_{ij}^2 - \left(\sum_{i=1}^N \sum_{j=1}^b v_{ij} \right)^2}} \quad (4.2)$$

Na sliki 4.9 je prikazana stopnja odvisnosti, izmerjena s korelacijo. Kot smo pričakovali, se s povečevanjem parametra s korelacija povečuje. Opazimo lahko tudi podobnost med grafoma 4.8 in 4.9, saj se pri obeh stopnja podobnosti pri $s = 1.0$ z vejitvijo povečuje. Pri $s = 0$ pa stopnja podobnosti z vejitvijo pada. Izmerjene vrednosti korelacije so zbrane v tabeli A.2 v prilogi A.



Slika 4.9: Prikaz stopnje odvisnosti v modelu z urejanjem naključnih vrednosti, izmerjene s korelacijo.

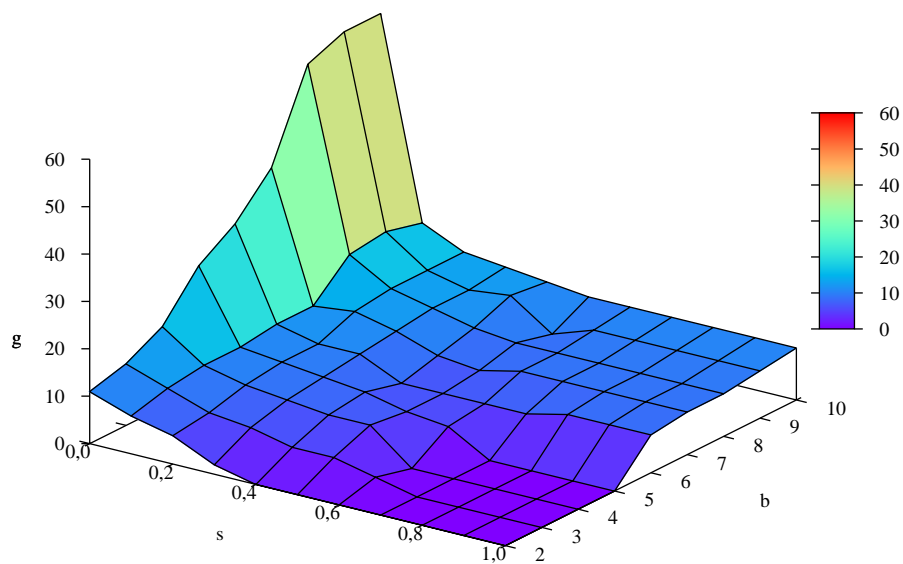
4.4 Primerjava z minimaks modelom

Kot smo že opisali v podpoglavju 2.1, so se Kaluža idr. [17, 20] ukvarjali z izgradnjo minimaks modela, ki je analiziral vpliv parametrov vejitve, odvisnosti in števila vrednosti v vozliščih drevesa na stopnjo patološkosti. V tej nalogi smo na podoben način ugotavljali vpliv teh parametrov na patološkost v minimin modelu, zato je primerjava z izsledki analize minimaksa na tem mestu primerna.

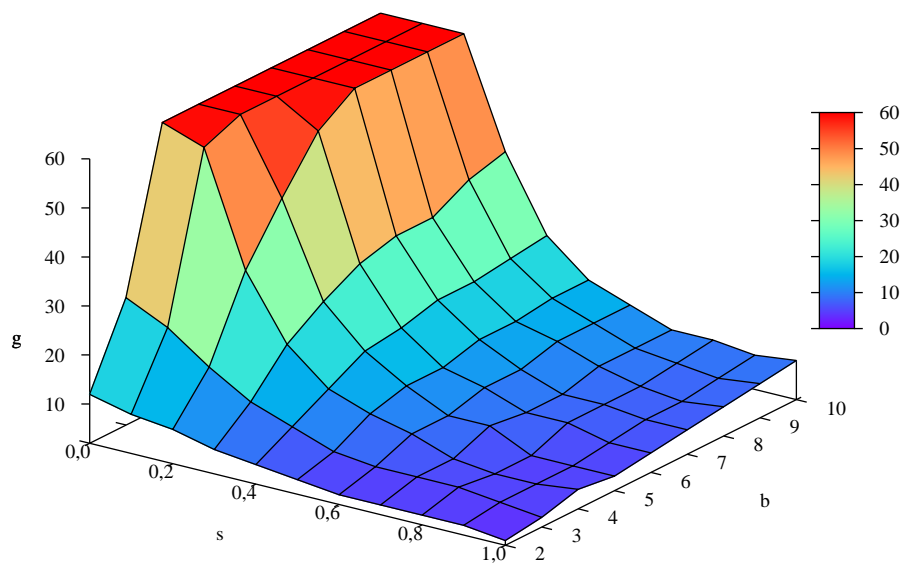
Slika 4.10a prikazuje isti pojav kot slika 4.10b, torej stopnjo patologije v odvisnosti od vejitve in podobnosti le da so mejne vrednosti g , ki so potrebne za odpravo patologije, izmerjene z minimaks modelom z urejanjem naključnih vrednosti.

Na obeh slikah opazimo podobne značilnosti modela.

1. meja zrnatosti g , pri kateri patološkost doseže $p = 1$ z vejitvijo narašča, s povečevanjem podobnosti pa pada,
2. patološkost je v modelu minimin bolj izrazita, saj je pri manjši odvisnosti meja zrnatosti, kjer patologija izgine, precej višja kot v minimaks modelu.



(a) Minimaks model z urejanjem naključnih vrednosti

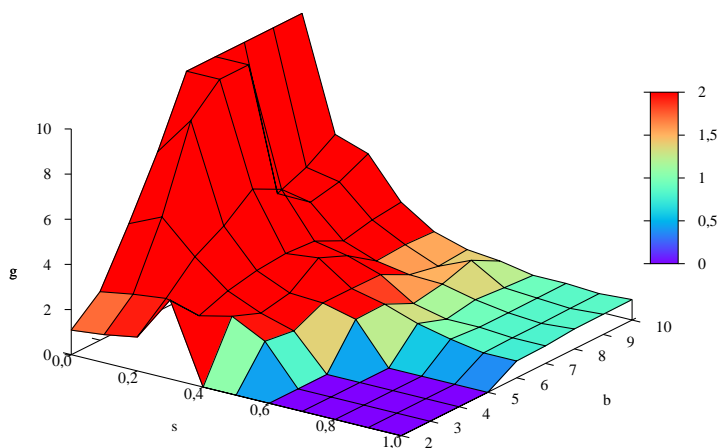


(b) Minimin model

Slika 4.10: Mejne vrednosti g , pri katerih je $p = 1$

Zadnjo značilnost smo želeli preveriti tudi z različnimi merami.

Najprej si pogledajmo, kako se obnaša kvocient točk obeh grafov. Vsako točko v grafu na sliki 4.10b smo delili z istoležno točko v grafu 4.10a. Slika 4.11 nam prikaže, za kolikšen delež je model minimina bolj ali manj patološki od modela minimaksa. Opazimo lahko, da je minimin model povečini bolj patološki, do razlik pride le pri večji vejitvi in podobnosti, kjer se model obnaša manj patološko. Model minimaksa pri manjši stopnji vejitve in veliki stopnji podobnosti ni patološki ($g = 0$), zato je model minimin, po tej meri, neskončno slabši. Zato na sliki 4.11 ni ustrezne primerjave za to območje.



Slika 4.11: Kvocient točk grafov modela minimin in minimaksa.

Definirajmo novo mero za primerjavo obeh grafov, ki pokaže povprečno stopnjo skaliranja enega grafa, da je odstopanje od drugega grafa čim manjše. Definirajmo matriko G_{ij} , ki vsebuje mejne vrednosti g pri vejitvi i in podobnosti j minimin modela (slika 4.10b). Podobno definiramo matriko H_{ij} , le da so v njej mejne vrednosti g iz minimaksa modela (slika 4.10a). Iščemo realno vrednost S , ki predstavlja faktor skaliranja minimin grafa po g osi, pri kateri je skaliran graf minimina po nekem kriteriju najbolj podoben grafu minimaksa. Vrednost S si lahko razlagamo tudi kot mero, za koliko je en graf bolj patološki od drugega. Pri izračunu bomo uporabili $S' = \frac{1}{S}$, ki je le inverz vrednosti S . Definirajmo kriterijsko funkcijo, ki jo bomo minimizirali, z izrazom 4.3

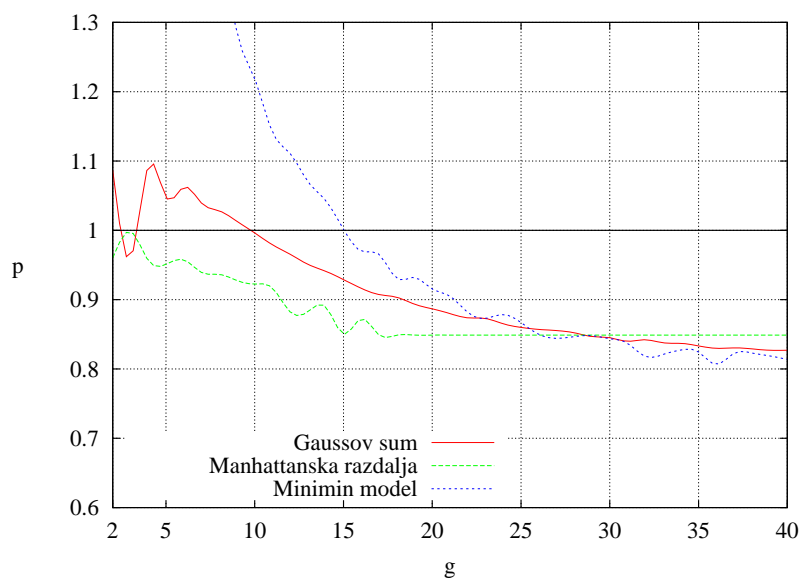
$$f = \frac{\sum_{i=1}^N \left(\frac{H_{ij} - S' * G_{ij}}{\frac{H_{ij} + S' * G_{ij}}{2}} \right)^2}{N}, \quad (4.3)$$

kjer je N število točk na grafu. Kriterijska funkcija je definirana podobno kot varianca [31], le da ne opazujemo odstopanja od aritmetične sredine krivulje, ampak odstopanja ene ploskve od druge. Ulomek $\frac{H_{ij} + S' * G_{ij}}{2}$ smo uvedli z namenom, da normaliziramo razlike med dvema vrednostima. Brez te uteži bi večji poudarek namenili večjim vrednostim, saj te povzročijo večjo napako, manjše pa bi imele premajhen vpliv.

Naša naloga je najti tak S , kjer vrednost izraza 4.3 doseže minimalno vrednost. Optimalno vrednost smo določili experimentalno in znaša $S = 2.43$. Za primerjavo smo preverili, kakšen je faktor skaliranja S pri primerjavi minimaks modela z urejanjem naključnih vrednosti (slika 4.10a) in ostalih dveh minimaks modelov v [17, 20]. Primerjava z modelom pridobljenim s porazdeljevanjem vrednosti vrne $S = 1.2$, kar je pričakovano, saj sta si modela zelo podobna. Pri modelu z normalizirano vsoto pa je $S = 1,9$, saj sta si grafa meje zrnatosti manj podobna. Minimim model je torej res bolj patološki kot model minimaksa, po zgornjih izračunih je v povprečju kar 2.4 krat bolj.

4.5 Primerjava z igro osmih ploščic

Poglejmo še primerjavo poteka patološkosti pri igri osmih ploščic in našega modela. Za igro osmih ploščic so na voljo [24] podatki o povprečni vejitvi, stopnji podobnosti merjeni s faktorjem grozdenja in izračunano stopnjo patološkosti za večje število zrnatosti. Nad našim modelom smo pognali simulacijo z enakimi naborom parametrov. Preslikavo iz podane vrednosti faktorja grozdenja v stopnjo podobnosti modela smo opravili s tabelo A.1, ki jo najdemo v prilogah. Na sliki 4.12 lahko vidimo potek patološkosti pri igri osmih ploščic in minimim modelu. Igra uporablja dve hevristični funkciji, za oceno položajev. Patološkost igre pri uporabi obeh funkcij je prikazana na grafu. Iz slike je razvidno, da model relativno dobro modelira potek stopnje patološkosti v igri, saj je potek vseh krivulij kvalitativno podoben. Le pri manjši zrnatosti je napaka modela večja.



Slika 4.12: Primerjava minimin modela z igro osmih ploščic.

Poglavje 5

Sklepne ugotovitve

V zadnjem poglavju ponovimo namen diplomskega dela in preverimo skladnost naših izsledkov z zastavljenimi cilji. Patologija enoagentnega preiskovanja je raziskana veliko slabše od patologije minimaksa. Vzrok temu morda leži v velikem zanimanju za patologijo minimaksa, saj je bila odkrita veliko prej. Veliko truda je bilo vložene v razlage tega pojava in mogoče je zmanjkalo zagona za raziskave patologije enoagentnega preiskovanja.

V tej nalogi smo ugotavljali, ali relacije v minimaks modelu veljajo tudi v modelu minimin. **Ugotovimo, da za oba modela kvalitativno veljajo enake relacije.**

Pri minimaksu igra vejitev pomembno vlogo, saj se s povečevanjem vejitve patološkost okrepi. Podobno obnašanje smo zaznali tudi pri analiziranju minimin modela.

Naslednji pomemben dejavnik je število pravih vrednosti v vozliščih drevesa. Povečevanje zrnatosti stopnjo patološkosti oslabi. Pri neodvisnem modelu opazimo, da se število vrednosti, ki so potrebne za odpravo patologije, z naraščanjem vejitve eksponentno povečuje. Tu je mogoče zaznati pomembno razliko med minimaks in minimin modeloma. Pri slednjem je namreč zadostno število vrednosti za odpravo patologije bistveno večje, kot pri minimaks modelu. Ta razlika je najbolj opazna pri večjih vejitvah.

Zadnji in eden najpomembnejših dejavnikov pa je odvisnost vozlišč v drevesu. Povečevanje odvisnosti pozitivno vpliva na patološkost v modelu, saj jo ublaži. S simulacijami smo pregledali prostor vseh treh parametrov in ugotovili mejne vrednosti zrnatosti, pri katerih patologija izgine. Po pričakovanjih, je za odpravo patologije pri povečevanju vejitve potrebna večja odvisnost dreves.

Za končni model smo s faktorjem grozdenja in korelacijo izračunali stopnjo odvisnosti. Opravili smo tudi primerjavo minimin modela z igro osmih ploščic

in ugotovili, da model precej dobro modelira stopnjo patologije v igri, le pri manjši zrnatosti je napaka modela večja.

Iz rezultatov naloge lahko sklepamo, da patologije v realni domeni ni za pričakovati, saj je navadno odvisnost bratskih vozlišč dovolj velika in tudi hevristična funkcija vrne dovolj veliko število različnih vrednosti. Pri odvisnosti, ki jo najdemo v igri osmih ploščic je v modelu za odpravo patologije dovolj že zrnatost $g = 15$. Ob nepazljivosti lahko na patologijo vseeno naletimo, a če se držimo smernic, ki so podane v tej nalogi, z njo ne bi smeli imeti večjih problemov.

V tem delu smo poskušali pojasniti nekaj dejavnikov, ki prispevajo k pojavu patologije in s tem dodatno razjasniti patološkost enoagentnega preiskovanja. Ostaja pa še nekaj odprtih vprašanj za nadaljne delo. Minimin model je v primerjavi z minimaksom precej bolj patološki, saj je potrebno število pravih vrednosti za odpravo patologije precej večje. Zanimivo bi bilo preučiti vzroke za ta pojav. Poleg tega pa bi lahko model nadgradili in vanj vpeljali uporabo pravih hevrističnih funkcij, kar bi omogočalo dodatno primerjavo z realnimi problemi.

Dodatek A

Tabele

s / b	2	3	4	5	6	7	8	9	10
0	0,680	0,768	0,800	0,816	0,826	0,832	0,837	0,840	0,843
0,1	0,660	0,742	0,773	0,788	0,797	0,802	0,806	0,811	0,814
0,2	0,635	0,712	0,741	0,755	0,764	0,771	0,778	0,784	0,788
0,3	0,604	0,675	0,704	0,718	0,728	0,736	0,743	0,750	0,757
0,4	0,570	0,634	0,660	0,674	0,685	0,693	0,701	0,707	0,714
0,5	0,532	0,582	0,609	0,623	0,634	0,640	0,649	0,654	0,663
0,6	0,492	0,519	0,548	0,559	0,571	0,579	0,585	0,590	0,595
0,7	0,450	0,452	0,468	0,481	0,495	0,499	0,509	0,511	0,516
0,8	0,404	0,376	0,383	0,394	0,406	0,407	0,418	0,417	0,423
0,9	0,368	0,279	0,270	0,272	0,273	0,285	0,281	0,285	0,299
1	0,319	0,160	0,110	0,063	0,052	0,034	0,028	0,024	0,018

Tabela A.1: Izmerjene vrednosti faktorja grozdenja cf pri modelu z urejanjem naključnih vrednosti (obravnavan je v razdelku 4.3).

s / b	2	3	4	5	6	7	8	9	10
0	0,695	0,568	0,476	0,408	0,354	0,309	0,274	0,244	0,219
0,1	0,709	0,587	0,499	0,434	0,385	0,347	0,317	0,294	0,275
0,2	0,725	0,608	0,527	0,469	0,427	0,396	0,371	0,353	0,338
0,3	0,743	0,633	0,558	0,507	0,471	0,444	0,425	0,408	0,376
0,4	0,763	0,660	0,592	0,546	0,515	0,492	0,475	0,462	0,451
0,5	0,784	0,689	0,628	0,588	0,560	0,541	0,526	0,515	0,506
0,6	0,807	0,723	0,668	0,633	0,608	0,591	0,579	0,570	0,564
0,7	0,834	0,761	0,714	0,683	0,663	0,648	0,638	0,631	0,626
0,8	0,863	0,807	0,769	0,744	0,728	0,717	0,708	0,702	0,698
0,9	0,897	0,867	0,842	0,826	0,815	0,807	0,802	0,798	0,795
1	0,937	0,962	0,970	0,973	0,973	0,972	0,972	0,972	0,972

Tabela A.2: Izmerjene vrednosti korelacije r pri modelu z urejanjem naključnih vrednosti (obravnavan je v razdelku 4.3).

Slike

1.1	Drevo igre pri preiskovanju z miniminom.	6
2.1	Drevo, na katerem je Bulitko pokazal patologijo enoagentnega preiskovanja.	13
3.1	Levo preiskovalno drevo je vedno nepatološko. Desno pa je v določenih primerih lahko patološko.	18
3.2	Poravnava mej podintervalov.	19
3.3	Prikaz stopnje patološkosti p v primeru, ko se hevrstične ocene lahko porazdelijo v dodatne podintervale in takrat, ko jih umestimo v zadnji podinterval.	21
3.4	Prikaz stopnje patološkosti neodvisnih dreves. Prikazana je v odvisnosti od zrnatosti g in vejitve b	22
4.1	Stopnja patologije pri vejitvi $b = 6$ in naključnem vstavljanju listov odvisnega drevesa.	25
4.2	Stopnja patologije pri vejitvi $b = 6$ in dodajanju odvisnih listov na robove neodvisnega drevesa.	26
4.3	Stopnja patologije pri vejitvi $b=6$ in dodajanju odvisnih listov z uporabo trikotniške porazdelitve $H_s(i)$	27
4.4	Stopnja patologije pri vejitvi $b = 6$ in dodajanju odvisnih listov z uporabo gladke porazdelitvene funkcije.	28
4.5	Povprečne vrednosti stopnje patološkosti za model z naključnim vstavljanjem in gladko porazdelitvijo, ter razmerje med obema krivuljama. Krivulje so prikazane v odvisnosti od stopnje s . . .	29
4.6	Vrednost zrnatosti g , ki je potrebna, da meja patologije pade pod $p = 1$ v modelu z urejanjem naključnih vrednosti.	30
4.7	Stopnja patologije v minimin modelu preiskovanja.	31
4.8	Prikaz stopnje odvisnosti v modelu z urejanjem naključnih vrednosti, izmerjene s faktorjem grozdenja.	33

4.9	Prikaz stopnje odvisnosti v modelu z urejanjem naključnih vrednosti, izmerjene s korelacijo.	34
4.10	Mejne vrednosti g , pri katerih je $p = 1$	35
4.11	Kvocienat točk grafov modela minimin in minimaks.	36
4.12	Primerjava minimin modela z igro osmih ploščic.	38

Tabele

3.1	Število pravih vrednosti g , pri katerih patološkost doseže vrednosti $p = 0,98$, $p = 1$, $p = 1,02$	21
A.1	Izmerjene vrednosti faktorja grozdenja cf pri modelu z urejanjem naključnih vrednosti (obravnava je v razdelku 4.3).	41
A.2	Izmerjene vrednosti korelacije r pri modelu z urejanjem naključnih vrednosti (obravnava je v razdelku 4.3).	42

Algoritmi

1	<i>minimin(node, depth)</i>	6
2	<i>error_level(V_r, V_n)</i>	17

Literatura

- [1] Ivan Bratko. Prolog Programming for Artificial Intelligence, 3rd edition. Addison-Wesley, Boston, ZDA.
- [2] R.E.Korf. Real-time heuristic search. *Artificial Intelligence*, 42(2,3):100–107, 1990.
- [3] Donald F. Beal. An analysis of minimax. V zborniku *Advances in Computer Chess 2*, str. 103–109. Edinburgh University Press, 1980.
- [4] V. Bulitko, Lihong Li, R. Greiner, I. Levner. Lookahead pathologies for single agent search. V zborniku *18th International Joint Conference on Artificial Intelligence*, str. 1531–1533, 2003.
- [5] V. Bulitko. Lookahead Pathologies and Meta-level Control in Real-time Heuristic Search.. V zborniku *15th Euromicro Conference on Real-Time Systems*, str. 113-16, 2003.
- [6] Mitja Lustrek, Vladim Bulitko. Lookahead Pathology in Real-Time Path-Finding.. V zborniku *Proceedings of AAI, Learning for Search Workshop*, str. 108-114, 2006
- [7] M. Lustrek. Pathology in single agent search. V zborniku *8. mednarodne multikonference Informacijska družba*, str. 345–348, 2005.
- [8] Mitja Lustrek, Vadim Bulitko. Thinking Too Much: Pathology in Pathfinding. V zborniku *ECAI 2008: 18th European Conference on Artificial Intelligence*, str. 899–900, 2008.
- [9] A. Sadikov, I. Bratko. Pessimistic heuristics beat optimistic ones in real-time search. V zborniku *European Conference on Artificial Intelligence (ECAI)*, str. 148–152, 2006.

- [10] Donald F. Beal. Benefits of minimax search. V zborniku *Advances in Computer Chess 3*, str. 17–24. Pergamon Press, 1982.
- [11] Ivan Bratko in Matjaž Gams. Error analysis of the minimax principle. V zborniku *Advances in Computer Chess 3*, str. 1–15. Pergamon Press, 1982.
- [12] Dana S. Nau. *Quality of decision versus depth of search on game trees*. Doktorska disertacija, Duke University, 1979.
- [13] Dana S. Nau. An investigation of the causes of pathology in games. *Artificial Intelligence*, 19(3):257–278, 1982.
- [14] Dana S. Nau. Pathology on Game Trees Revisited, and an Alternative to Minimizing. *Artificial Intelligence*, 21(1,2):257–278, 1983.
- [15] John von Neumann. Zur theorie der gesellschaftsspiele. *Mathematische Annalen*, (100):295–320, 1928.
- [16] Boštjan Kaluža, Mitja Luštrek in Matjaž Gams. Patologija minimaksa v sintetičnih drevesih in Pearlovi igri. V zborniku *Proceedings of the 11th International Multiconference Information Society (IS 2007)*, zv. A, str. 84–87, 2007.
- [17] Boštjan Kaluža, Mitja Luštrek, Matjaž Gams in Aleš Tavčar. Pathology in minimax searching. V zborniku *Proceedings of the Sixteenth International Electrotechnical and Computer Science Conference (ERK 2007)*, zv. B, str. 107–110, 2007.
- [18] Boštjan Kaluža in Aleš Tavčar. *Seminarska naloga pri predmetu UISP: Patologija pri preiskovanju z minimaksom*. Seminarska naloga, Fakulteta za računalništvo in informatiko Univerze v Ljubljani, Institut “Jožef Stefan”, 2007.
- [19] Mitja Luštrek. *Patologija v hevrističnih preiskovalnih algoritmih*. Doktorska disertacija, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, 2007.
- [20] Boštjan Kaluža. *Analiza patoloških modelov minimaksa in Pearllove igre*. Diplomsko delo, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, 2008.

- [21] Mitja Luštrek, Matjaž Gams in Ivan Bratko. Is real-valued minimax pathological? *Artificial Intelligence*, 170(6):620–642, 2006.
- [22] Matt Donadio. *How to Generate White Gaussian Noise*. <http://www.dspguru.com/howto/tech/wgn.htm>, 15. 8. 2008.
- [23] Nicholas Metropolis in Stanislaw M. Ulam. The Monte Carlo Method. *Journal of the American Statistical Association*, 44(247):335–341, 1949.
- [24] Rok Piltaver. *Patologija preiskovanja v igri osmih ploščic*. Diplomaska naloga, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, 2008.
- [25] Rok Piltaver, Mitja Luštrek, Matjaž Gams. Search pathology of 8-puzzle. V zborniku *Proceedings of the 10th International Multiconference Information Society (IS 2007)*, zv. A, str. 8–12, 2007.
- [26] Aleksander Sadikov. *Širjenje napak hevrističnih ocen v preiskovanju grafov iger*. Doktorska disertacija, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, 2005.
- [27] Aleksander Sadikov, Ivan Bratko in Igor Kononenko. Bias and pathology in minimax search. *Theoretical Computer Science*, 349(2):268–281, 2005.
- [28] Wikipedia. *Alpha-beta pruning*. http://en.wikipedia.org/wiki/Alpha_beta_pruning, 15. 8. 2008.
- [29] Wikipedia. *Correlation*. <http://en.wikipedia.org/wiki/Correlation>, 15. 8. 2008.
- [30] Wikipedia. *N-puzzle*. <http://en.wikipedia.org/wiki/N-puzzle>, 20. 2. 2009.
- [31] Wikipedia. *Varianca*. <http://sl.wikipedia.org/wiki/Varianca>, 19. 8. 2009.
- [32] Wikipedia. *Automated theorem proving*. http://en.wikipedia.org/wiki/Automated_theorem_proving, 20. 2. 2009.
- [33] Wikipedia. *Travelling salesman problem*. http://en.wikipedia.org/wiki/Travelling_salesman_problem, 20. 2. 2009.
- [34] Wikipedia. *Shannon number*. http://en.wikipedia.org/wiki/Shannon_number, 20. 2. 2009.

IZAJAVA O AVTORSTVU

diplomskega dela

Spodaj podpisani Aleš Tavčar,
z vpisno številko 63030068,
sem avtor diplomskega dela z naslovom:

PATOLOGIJA MINIMIN PREISKOVANJA

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom akad. prof. dr. Ivana Bratka
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki "Dela FRI".

Ljubljana, 24. 8. 2008

Aleš Tavčar