

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

**ANALIZA PATOLOŠKIH MODELOV
MINIMAKSA IN PEARLOVE IGRE**

Boštjan Kaluža

Ljubljana, 2008

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

ANALIZA PATOLOŠKIH MODELOV MINIMAKSA IN PEARLOVE IGRE

Boštjan Kaluža

Delo je pripravljeno v skladu s Pravilnikom o podeljevanju Prešernovih
nagrad študentom, pod mentorstvom akad. prof. dr. Ivana Bratka in
somentorstvom dr. Mitje Luštreka.

Ljubljana, 2008

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .

Povzetek

Programi za igranje iger, ki temeljijo na načelu minimaks, običajno z globljim preiskovanjem dobijo boljše rezultate. Ko so raziskovalci poskušali z matematičnimi modeli to tudi formalno pojasniti, so naleteli na nenavaden pojav: pri določenih pogojih globlje iskanje povzroči slabše rezultate. Pojav so poimenovali *patologija minimaksa*.

V nalogi se ukvarjamo s patološkimi modeli preiskovanja z minimaksom in dejavniki, ki vplivajo na pojav patologije. Najpomembnejši dejavniki so vejitev drevesa igre, število možnih pravih vrednosti položaja v vozliščih drevesa, verjetnost poraza za igralca na potezi in medsebojna odvisnost vrednosti vozlišč. Namen naloge je osvetliti njihove vplive na patologijo. V nalogi zgradimo neodvisni model preiskovanja in ga kombiniramo z različnimi odvisnimi modeli. Pri tem uporabimo že predstavljene odvisne modele in pa nov odvisni model, ki ga konstruiramo v nalogi. Pojav patologije analiziramo tudi pri delno odvisnih drevesih. Zaključimo, da se analizirani odvisni modeli obnašajo kvalitativno podobno. Patološkost upada s povečevanjem odvisnosti in povečevanjem števila pravih vrednosti v vozliščih. Naraščajoča vejitev drevesa patološkost okrepi v patoloških območjih in oslabi v nepatoloških območjih.

Izsledke analize praktično preverimo na Pearlovi igri, ki so jo tudi v preteklosti že uporabili za preučevanje patologije. Izvorno igro nadgradimo z dejavniki, ki vplivajo na patologijo, in na njej preizkusimo delovanje različnih hevrističnih funkcij. Pri izbrani hevristični funkciji *povprečje* analiziramo pojav patologije in potrdimo vpliv odvisnosti pravih vrednosti in vejitve drevesa na pojav patologije. Ugotovimo, da pri izbrani hevristični funkciji število vrednosti ne vpliva na patološkost, kot napoveduje model, temveč je zaznati celo nasproten trend.

Ključne besede:

minimaks, patologija minimaksa, drevo igre, odvisna drevesa, Pearlova igra

Abstract

Computer games based on the minimax principle usually produce better results when searching deeper. When researchers attempted to explain this formally using mathematical models, they found that under certain conditions the minimax behaves unexpectedly: deeper search produced worse results. This phenomenon was termed *minimax pathology*.

We examine pathological models of minimax search and factors influencing on the pathology. The most important factors are the branching of the game tree, the number of position values, the probability of loss for player on the move and the dependence between nearby positions. Our goal is to elucidate their impact on the pathology. We design an independent minimax model and combine it with several dependent models. One of them is originally constructed in this work, others were already presented by other authors. The pathology is also analyzed on partly dependent trees. We conclude that the analyzed models behave qualitatively similarly. The pathology is decreasing with increasing dependence and increasing number of position values. Increasing branching strengthens pathology in pathological regions and weakens it in non-pathological regions.

Models are experimentally verified on Pearl game, which was already used in the pathology research in the past. The original game is expanded with several factors and tested with several heuristic functions. The pathology is analyzed with the selected heuristic function *average* confirming influence of the branching factor and dependence between nearby positions. The number of position values using the selected heuristic function does not impact the pathology as predicted by the model.

Key words:

minimax, minimax pathology, game tree, dependence game tree, Pearl game

Kazalo

1	Uvod	1
1.1	Patologija minimaksa	2
1.2	Namen naloge in rezultati	3
1.3	Organizacija naloge	4
2	Pregled sorodnega dela	5
2.1	Odkritje patologije	5
2.1.1	Bealov model patologije	5
2.1.2	Nauov model patologije	6
2.2	Vpliv dejavnikov na uspešnost preiskovanja z minimaksom	7
2.2.1	Povzetek	9
3	Neodvisni model	10
3.1	Zgradba modela	10
3.2	Patologija v modelu	13
4	Odvisni modeli	15
4.1	Mere odvisnosti	15
4.1.1	Faktor grozdenja	15
4.1.2	Korelacija	16
4.2	Verjetnost poraza	17
4.3	Postopni modeli odvisnosti	19
4.3.1	Model s porazdeljevanjem pomožnih vrednosti	20
4.3.2	Model z normalizirano vsoto	22
4.4	Model z urejanjem naključnih vrednosti	24
4.4.1	Delno urejanje	25
4.4.2	Trikotniška porazdelitev	26
4.4.3	Gladka porazdelitev	28
4.5	Primerjava modelov	31

5	Pearlova igra	34
5.1	Opis in razširitev igre	34
5.2	Hevristične funkcije	36
5.2.1	Sintetična hevristična funkcija	37
5.2.2	Povprečje	37
5.2.3	Mediana	38
5.2.4	Vsota absolutnih razlik	39
5.2.5	Maksmin	39
5.2.6	Povzetek	40
5.3	Meritve in rezultati	41
6	Sklep	43
6.1	Patološki modeli preiskovanja z minimaksom	43
6.2	Pearlova igra	44
A	Tabele	45
	Seznam slik	50
	Seznam tabel	52
	Seznam algoritmov	53
	Literatura	54

Seznam uporabljenih kratic in simbolov

Oznake

b	vejitev drevesa
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 (nivo)
d_{max}	višina drevesa (največja globina)
f	faktor grozdenja
h	poljubna hevristična funkcija
h_a	hevristična funkcija <i>povprečje</i> pri Pearlovi igri
h_d	hevristična funkcija <i>vsota absolutnih razlik</i> pri Pearlovi igri
h_m	hevristična funkcija <i>mediana</i> pri Pearlovi igri
h_{mm}	hevristična funkcija <i>maksmin</i> pri Pearlovi igri
h_s	sintetična hevristična funkcija
k	verjetnost poraza za igralca na potezi
k_i	verjetnost poraza na nivoju i
p	stopnja patološkosti
q_i	razmerje med verjetnostjo poraza in zmage na nivoju i
r_{xy}	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
\mathcal{V}	množica polj iz preostalega dela plošče po delitvi
$E(d, h)$	verjetnost napačne izbire poteze pri izbrani globini d in hevristični funkciji h

$H_s(i)$	trikotna porazdelitev, ki določa verjetnost zamenjave lista z indeksom i
$H'_s(i)$	gladka porazdelitev, ki določa verjetnost zamenjave lista z indeksom i
$N(v)$	množica polj, ki so sosednja polju v
$Q(c_{b,s})$	standardni odklon vseh q_i pri izbranem $c_{b,s}$
σ	standardni odklon pri normalni porazdelitvi
ζ	začasna vrednost vozlišča

Slovarček izrazov

backed-up value	vzratna vrednost, povratna vrednost
branching factor	faktor vejitev, vejitev, vejanje, faktor vejanja
clustering factor	faktor grozdenja
degree of pathology	patološkost, stopnja patologije
granularity	zrnatost
heuristic function	hevristična funkcija
incremental game	postopna igra
move error	napaka poteze
pathology	patologija
position error	napaka položaja, napaka pozicije
similarity	podobnost

Poglavje 1

Uvod

Pri reševanju mnogih problemov si lahko pomagamo s preiskovanjem prostora stanj, katerega predstavimo z usmerjenim grafom. Vozlišča predstavljajo možna stanja podanega problema, veje pa ponazarjajo dovoljene prehode med stanji. Posebna stanja grafa so začetno stanje in eno ali več končnih stanj z rešitvijo. Reševanje problema prevedemo na potovanje po grafu in iskanje najboljše rešitve. Velikost grafa eksponentno narašča s številom stanj, zato je preiskovanje celotnega grafa pogosto prezamudno, hkrati pa presega zmogljivosti računalnika. Z različnimi pristopi lahko omejimo preiskovanje in izgradnjo grafa.

Poseben primer usmerjenih grafov so drevesa. Računalnikarji imamo navado obrniti drevesa na glavo in vrh drevesa poimenujemo koren, ki obenem predstavlja začetno stanje. Končna stanja se nahajajo v listih, torej končnih elementih drevesa.

V nalogi se bomo osredotočili na preiskovanje dreves iger, na katerih temelji večina programov za igranje iger dveh nasprotnikov s popolno informacijo, kot je npr. šah. Zgradbo drevesa si oglejmo na primeru igre dveh igralcev. V korenu drevesa je na potezi prvi igralec, poimenujmo ga Maks, ki želi izbrati najboljšo potezo, torej takšno, ki bi ga vodila k zmagi. Korenu priredimo maksimum vrednosti sinov ali z drugimi besedami, vrednosti položajev, do katerih lahko pride igralec Maks. Za ovrednotenje sinov pa je potrebno ovrednotiti vozlišča na nižjem nivoju. Tu je na potezi nasprotni igralec, poimenujmo ga Min, ki izbira za igralca Maks slabe poteze, zato vozliščem priredimo minimum sinov. Nivoji min in maks se izmenjujejo do listov, kjer so prave vrednosti položajev. Vrednosti v listih sta lahko le dve, npr. 1 – zmaga in 0 – poraz, lahko pa jih je tudi več, kot je prikazano na sliki 1.1. Vrednosti se nato nato preko min in maks nivojev prenesejo v koren drevesa. Igralec Maks izbere potezo, ki vodi k

sinu z največjo pravo vrednostjo.

Neformalno predstavljeni algoritem za določanje prave vrednosti položaja se imenuje *minimaks* in je formalno zapisan v algoritmu 1.

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

Require: *node*, *depth*

Ensure: real value of *node*

```

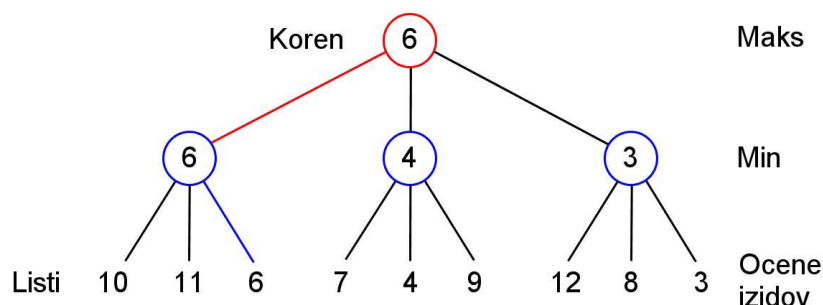
if node is a leaf node or depth = 0 then
  return value of node
else
  if depth is even then
     $\alpha \leftarrow -\infty$ 
    for all child of node do
       $\alpha \leftarrow \max(\alpha, \text{minimax}(\text{child}, \text{depth} - 1))$ 
    end for
  else
     $\alpha \leftarrow \infty$ 
   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

```

Na sliki 1.1 je prikazano drevo igre. Vozlišča maks so obarvana rdeče, vozlišča min so modra in izbrana poteza v korenu drevesa je obarvana rdeče. Igralec Maks bo v korenu izbral potezo, ki ga vodi k vozlišču z največjo vrednostjo 6.

1.1 Patologija minimaksa

Algoritem minimaks je leta 1928 objavil John Von Neumann [7] za izbiro optimalne poteze pri igranju šaha. V primeru, ko lahko preiščemo celotno drevo igre, kakršno dobimo pri npr. igranju šaha, z minimaksom vedno izberemo najboljšo potezo. A drevesa so običajno prevelika, zato drevo igre preiščemo do neke globine in s hevristično funkcijo ocenimo položaje. Ocenjene vrednosti imenujemo statične hevristične vrednosti, njihovo napako pa statična napaka. Statične vrednosti nato z minimaksom prenesemo pod koren drevesa, s čimer



Slika 1.1: Drevo igre pri preiskovanju z minimaksom.

dobimo povratne hevristične vrednosti. Globlje preiskovanje bi za igralca pomenilo *premisлити* več potez naprej in programi v praksi igrajo bolje, ko preiščejo večji del drevesa in hevristično ocenjevanje uporabijo globlje. Iz tega je mogoče sklepati, da so povratne hevristične vrednosti zanesljivejše od statičnih.

Ko so raziskovalci poskušali pojasniti, zakaj globlje preiskovanje z minimaksom v praksi deluje, so naleteli na nenavaden pojav. Matematični model je pokazal, da naj bi bile ocene izidov igre, ki so z minimaksom določene z globljim preiskovanjem, manj zanesljive od tistih, ki so rezultat plitvejšega preiskovanja. Z drugimi besedami to pomeni, da s *premislekom* več potez vnaprej igramo slabše oz. manj zanesljivo. To razhajanje med praktičnimi izkušnjami in teoretičnimi analizami so poimenovali *patologija minimaksa*.

Napake, ki povzročajo nezanesljive ocene izidov igre, ločimo na *napako položaja* in *napako poteze*. Pri prvi napaki gre za odstopanje hevristične vrednosti položaja od prave. Pri drugi vrsti napake pa v nekem vozlišču izberemo potezo, ki ne vodi v sina z najboljšo pravo vrednostjo. Do slednjega pride zaradi napake ocene položaja v sinovih. Patološko obnašanje je lahko prisotno pri obeh vrstah napak, ni pa nujno, da se pri obeh pojavi hkrati.

1.2 Namen naloge in rezultati

Namen naloge je s pomočjo modela sintetičnih dreves preučiti dejavnike, ki vplivajo na patološkost. Osredotočimo se na gradnjo različnih modelov, ki upoštevajo odvisnost položajev med igro. Naloga preučuje delno odvisna drevesa v odvisnosti od stopnje odvisnosti, vejitve drevesa in števila vrednosti položajev.

V nalogi pokažemo, da so različni modeli odvisnih dreves kvalitativno po-

dobni. S povečevanjem odvisnosti dreves in števila vrednosti v položajih patološkost upada. Povečana vejitev ojača pojav patologije v patoloških in zmanjša v nepatoloških območjih.

Drugi cilj naloge je praktičen preizkus izsledkov analize na pravi igri. Pearlovo igro [5] razširimo z dejavniki, ki vplivajo na patološkost, ter preučimo patološkost pri uporabi različnih hevrističnih funkcij. Ugotovimo, da vejitev drevesa in odvisnost vrednosti vplivata na patološkost skladno z napovedmi modela, medtem ko število vrednosti na patološkost nima pričakovanega vpliva.

Pričujoče delo je nastalo kot plod dvoletne raziskave na Odseku za inteligentne sisteme na Institutu "Jožef Stefan" in povzema ključne rezultate raziskave. Večino izsledkov v tem delu smo objavili na konferencah [8, 9], nekatere modele odvisnih dreves pa smo obravnavali v seminarski nalogi [10] na diplomskem študiju. Pričujoče delo predstavlja enoten pregled celotne raziskave in vključuje neobjavljene izsledke.

1.3 Organizacija naloge

V diplomski nalogi v drugem poglavju pregledamo sorodna dela, ki so se ukvarjala s preučevanjem patologije. Predstavimo ključne avtorje in njihova odkritja, ki so botrovala razvoju pojasnjevanja patologije.

V tretjem poglavju predstavimo neodvisni večvrednostni model preiskovanja z minimaxsom. Podamo ključne predpostavke modela, opišemo njegovo zgradbo in analiziramo pojav patologije v modelu.

Z odvisnimi modeli se ukvarjamo v četrtem poglavju. V začetku predstavimo mere za določanje odvisnosti v drevesih in postopek za določanje verjetnosti poraza, ki je potreben za zagotavljanje zanimivih dreves iger. V nadaljevanju opišemo zgradbo postopnih modelov odvisnih dreves in analiziramo rezultate meritev patološkosti. Predstavimo odvisni model z urejanjem naključnih vrednosti in težave, na katere smo naleteli pri njegovem načrtovanju. Poglavje zaključimo s primerjavo in povzetkom analize odvisnih modelov.

Peto poglavje je posvečeno analizi Pearllove igre. Izvirno Pearlovo igro razširimo s predpostavkami modela in preizkusimo različne hevristične funkcije. Na spremenjeni igri analiziramo pojav patologije in komentiramo rezultate.

Zaključimo s sklepom v šestem poglavju, kjer povzamemo opravljeno delo in izsledke celotnega dela, ter podamo smernice za nadaljnje preučevanje pojava patologije.

Poglavje 2

Pregled sorodnega dela

V tem poglavju je povzet kratek pregled sorodnega dela na področju pojasnjevanje patologije. Opisani so ključni modeli in ideje, ki so botrovali razvoju področja. Poglavje podaja izhodiščne točke za nadaljnje delo.

2.1 Odkritje patologije

V sedemdesetih letih prejšnjega stoletja je raziskovalce mučilo vprašanje, zakaj programi za igranje iger igrajo bolje z globljim preiskovanjem. Pri preučevanju modelov iskanja z minimaksom sta patologijo minimaksa neodvisno odkrila Dana S. Nau leta 1979 [4] in Donald F. Beal leta 1980 [1]. V nadaljevanju bomo predstavili njune ugotovitve.

2.1.1 Bealov model patologije

Beala je zanimalo, zakaj so z minimaksom pridobljene hevristične vrednosti zanesljivejše od statičnih hevrističnih vrednosti.

Da bi dobil odgovor, je sestavil enostaven matematični model iskanja z minimaksom [1]. Temelji na naslednjih predpostavkah:

1. Drevo igre ima konstanto vejitev b .
2. Vrednosti vozlišč so bodisi *zmaga* bodisi *poraz*.
3. Vrednosti vozlišč so na posameznem nivoju razporejene tako, da je verjetnost poraza za igralca na potezi enaka na vseh nivojih in je enaka c_b (c_b razložimo v razdelku 2.2).

4. Vrednosti vozlišč znotraj nivoja so neodvisne.
5. Statična napaka hevristične funkcije predstavlja verjetnost, da se poraz zamenja za zmago ali obratno, ter je neodvisna od prave vrednosti vozlišča in globine preiskovanja.

Beal je ugotovil, da napaka v korenu drevesa igre z globino d narašča. Pri vsakem dodatnem nivoju preiskovanja napaka naraste za faktor $\frac{c_b b}{1-c_b}$, kar je pri velikih b mogoče aproksimirati z $\log b$.

2.1.2 Nauov model patologije

Nau se je podobno kot Beal spraševal, zakaj programi igrajo bolje z globljim preiskovanjem. Sestavil je svoj matematični model preiskovanja z minimaksom [4] z naslednjimi predpostavkami:

1. Drevo ima konstantno vejitev $b = m + n$ (m in n razložimo v nadaljevanju).
2. Vrednosti vozlišč sta prav tako lahko *zmaga* ali *poraz*.
3. Vozlišča so izgubljena, če so na nivoju maks vsi sinovi izgubljeni¹, dobljena, če so vsi sinovi na min nivoju dobljeni, ali kritična, če imajo sinovi mešane vrednosti. Položaji so zanimivi v slednjem primeru, saj so sicer vse poteze enakovredne. Razmerje med dobljenimi in izgubljenimi vrednostmi sinov določata števili m in n , kjer je m število sinov vozlišča, ki imajo pravo vrednost enako pravi vrednosti starša, n pa število sinov s pravo vrednostjo, različno od prave vrednosti starša.
4. Vrednosti hevristične funkcije je končno mnogo in ležijo med 0 in r .
5. Napaka je predstavljena kot verjetnost za izbiro neoptimalne poteze.

Če poteze izbiramo naključno, je verjetnost izbire optimalne poteze v kritičnem vozlišču enaka $\frac{m}{m+n}$. Nau je pokazal, da pri uporabi hevristične funkcije h pri preiskovanju do globine d velja izrek o patologiji (2.1), ki govori o verjetnosti za izbiro optimalne poteze $D_{m,n}(h, d)$.

$$\lim_{d \rightarrow \infty} D_{m,n}(h, d) = \frac{m}{m+n} \quad (2.1)$$

¹Izgubljena vrednost v vozlišču pomeni poraz, dobljena pa zmago.

Z naraščajočo globino preiskovanja verjetnost optimalne izbire optimalne poteze konvergira k verjetnosti, ki jo dobimo pri naključnem izbiranju poteze. Pri velikem d so namreč hevristične vrednosti vseh sinov korena enake in ni možno smiselno izbirati.

Nau je ugotovitve praktično preizkusil na Pearlovi igri (igro podrobneje razložimo v razdelku 5.1). Pri izbrani hevristični funkciji je pokazal, da je igra pri dovolj velikem d in b patološka.

2.2 Vpliv dejavnikov na uspešnost preiskovanja z minimaksom

Preiskovanje z minimaksom je v praksi uspešno, kar je v nasprotju s teoretičnimi spoznanji izpred dveh desetletij. Nadaljnje raziskave so bile posvečene temu razhajanju. Raziskovalce je predvsem zanimalo, katere lastnosti pravih iger botrujejo uspešnosti minimaksa oz. katere lastnosti, ki preprečujejo patologijo, manjkajo matematičnim modelom minimaksa, ki so zaradi lažje analize poenostavljeni.

Beal [2] je v nadaljnjih analizah dopolnil svoj model s faktorjem grozdenja f (angl. *clustering factor*), ki predstavlja delež vozlišč, katerih sinovi imajo enako vrednost. Poimenoval jih je *grozdi* (angl. *cluster*). Pokazal je, da se pri majhnem f napaka z vsakim dodatnim nivojem preiskovanja povečuje in prispeva k pojavu patologije, medtem ko pri velikem f napaka z dodatnimi nivoji preiskovanja pada, zato je v tem primeru preiskovanje z minimaksom koristno.

Bratko in Gams [3] sta sprostila nekatere predpostavke v Bealovem modelu [1]. Najprej sta preverila tretjo predpostavko o konstantni verjetnosti poraza k . Poraz v nekem vozlišču je vedno posledica samih zmag v njegovih sinovih. Izračun deleža porazov na sosednjih nivojih sledi iz enačbe 2.2.

$$k_i = (1 - k_{i+1})^b, k_i \in [0, 1], 1 \leq i < d_{max} \quad (2.2)$$

Pri konstantni verjetnosti poraza na vseh nivojih drevesa igre mora za vsak i veljati $k_i = c_b$. Takrat je napaka v korenu drevesa enaka c_b in patologija nastopi. Ko je rešitev enačbe blizu 0 ali 1, napaka v korenu drevesa konvergira k 0 in patologija ni prisotna. Avtorja sta na podlagi izračunov pokazala, da patologija v Bealovem modelu nastopi tudi pri različnih deležih k .

V nadaljevanju sta sprostila drugo predpostavko o dveh vrednostih v vozliščih. Dve vrednosti sta nadomestila s poljubnim številom enakomerno porazdeljenih vrednosti. Spremenjeni model se je prav tako obnašal patološko.

Bratko in Gams sta neodvisno² prišla do podobnih zaključkov kot Beal [2]. V zadnjem koraku sta sprostila predpostavko o neodvisnih pravih vrednostih v vozliščih z argumentom, da se vrednosti bratskih vozlišč ne spremenijo popolnoma z le eno potezo. Igro sta razdelila na dele, kjer so vsi položaji dobljeni, dele, kjer so vsi položaji izgubljeni, in dele z mešanimi pravimi vrednostmi položajev. Prva dva dela sta poimenovala t.i. *stabilizacijska jedra*, ki stabilizirajo preiskovanje z minimaksom, saj je hevristična ocena v takih vozliščih zanesljivo ocenjena. Pokazala sta, da je globlje preiskovanje z minimaksom v tako spremenjenem modelu koristno.

Nau [6] je v nadaljnjih raziskavah v Pearlovo igro vpeljal odvisnost položajev in sicer tako, da jo je spremenil v postopno.³ Povezavam v drevesu igre je nključno pripisal vrednosti -1 ali 1 , nato pa je v vsakem listu seštel vrednosti povezav od korena do izbranega lista. Če je bila dobljena vsota pozitivna, je listu pripisal 1 , sicer pa 0 . S Takšnim načinom vpeljave odvisnosti položajev je odpravil patologijo. Nauov pristop bomo analizirali v razdelku 4.3.2.

Pearl [16] je sestavil svoj teoretični model preiskovanja z minimaksom in ga dobro podprl z matematično analizo. Menil je, da ključ za uspeh minimaksa tiči v zelo slabih potezah, ki predčasno zaključijo igro. Poimenoval jih je *pasti* (angl. *traps*). Takšne položaje je mogoče oceniti brez napake in minimaks jih z globljim preiskovanjem najde vse več. S tem je prišel do enakih zaključkov kot Bratko in Gams [3].

Scheucher in Kaindl [21] sta pri analizi modelov definirala *mirnost* (angl. *quiescence*). Vozlišče je mirno, če je njegova povratna hevristična vrednost enaka (ali dovolj blizu) statični hevristični vrednosti. V drevesa igre sta uvedla delež vozlišč, ki so mirna do določenega nivoja. Napaka v mirnih vozliščih je bila manjša kot v nemirnih in je v večini poizkusov odpravila patologijo. V kasnejših raziskavah [20] sta se osredotočila na število hevrističnih vrednosti. Predpostavila sta, da so hevristične vrednosti na nižjih nivojih bolj razpršene, kar je v kombinaciji z odvisnostjo pripomoglo k odpravi patologije.

Luštrek idr. [13, 12, 11] so z matematičnim dokazom pokazali, da je predpostavka Scheucherja in Kaindla [21] o razpršenih vrednostih posledica lastnosti minimaksa. Pokazali so tudi, da se z uporabo realnih vrednosti v položajih in uporabo hevristične funkcije, ki doda enak statičen šum pri vseh globinah

²Beal [1] ter Bratko in Gams [3] so prišli do podobnih ugotovitev o vplivu podobnih vrednosti na uspešnost preiskovanja z minimaksom neodvisno. Prispevke so predstavili celo na isti konferenci z naslovom The 2nd International Conference on Advances in Computer Chess.

³Vrednosti vozlišč se postopno spreminjajo z globino drevesa, s čimer dosežemo, da imajo bratska vozlišča vrednosti odvisne od starša. S postopnimi modeli odvisnosti se podrobneje ukvarjamo v razdelku 4.3.

preiskovanja, izognemo patologiji, četudi vrednosti pretvorimo v dvojiške. Na nižjih nivojih so skrajne vrednosti (zelo velike in zelo majhne) pogostejše kot na višjih in posledično je na nižjih nivojih dvovrednostna napaka manjša, saj se pri dveh vrednosti, ki sta bolj narazen, manj pogosto zmotimo. Iz tega sledi manjša statična napaka, kar odpravi patologijo. Luštrek [11] je vpeljal odvisnost v drevo igre s postopnim spreminjanjem vozlišč na podoben način kot Nau [6] in se prav tako izognil patologiji.

Sadikov idr. [18, 19] so analizirali šahovske končnice in pokazali, da se patologija pojavi, kadar je vejitev med igralcema neenaka. To se zgodi v primeru, ko ima igralec na potezi bistveno večje število možnih potez, kot njegov nasprotnik. Sadikov je opazoval napako položaja, ki je sicer patološka, a na napako poteze ni imela bistvenega vpliva.

2.2.1 Povzetek

Vpliv vejitve na patološkost so opazili vsi raziskovalci. Povečevanje vejitve pripomore k patološkosti, a raziskave kažejo, da vejitev ni edini dejavnik. V modelih bomo vejitev vsekakor upoštevali.

Vpeljava različnih odvisnosti položajev v drevo igre je v večini raziskav pripomogla k odpravi patologije. Nau [5] ter Luštrek idr. [11, 13] so uporabili postopno spreminjanje vrednosti vozlišč, Bratko in Gams [3] ter Pearl so v drevo igre uvedli delež zanesljivo ocenjenih vozlišč, Beal [2] pa je v drevo vpeljal omejitve med brati.

Raziskovalci si glede števila vrednosti v vozliščih drevesa igre niso bili enotni. Prvi modeli Beala [1] in Naua [4] so uporabljali le dve pravi vrednosti in dve (Beal) oz. končno mnogo (Nau) hevrističnih vrednosti. Patologija je bila prisotna pri obeh modelih. Bratko in Gams [3] sta jo opazila tudi pri drugačnem številu pravih in hevrističnih vrednosti. Scheucher in Kaindl [21] ter Luštrek idr. [13, 11] so različno število vrednosti šteli za prepreko patologiji. V nalogi bomo pri gradnji modelov upoštevali ta dejavnik in preverili njegov vpliv na patologijo.

Vsi raziskovalci pa so bili enotni glede uravnavanja deleža porazov za igralca na potezi. S tem so se omejili na drevesa igre, ki niso preveč v prid enemu od igralcev. V modelih bomo uporabili vrednost c_b oz. podobno vrednost, s katero bomo dosegli enak učinek.

Poglavje 3

Neodvisni model

V poglavju je predstavljena zgradba neodvisnega modela preiskovanja z minimaksom. Razložene so osnovne predpostavke modela in podan je postopek za gradnjo dreves. Predstavljeni so rezultati meritev in pojav patologije v modelu.

3.1 Zgradba modela

Neodvisni model je zgrajen po zgledu večvrednostnega modela za preiskovanje z minimaksom Luštreka idr. [13, 12]. Model upošteva različno število pravih in hevrističnih vrednosti v vozliščih in predstavlja nadgradnjo Bealovega modela (opisan je v razdelku 2.1.1). Osnovne predpostavke modela lahko strnemo v pet točk:

1. Drevo igre ima konstanto vejitev b in globino d_{max} .
2. Prave in hevristične¹ vrednosti vozlišč lahko zavzamejo $g \in [2, \infty]$ pravih realnih vrednosti z intervala $[0, 1]$. Parameter g poimenujemo *zrnatost* (angl. *granularity*).
3. Vrednosti vozlišč so na posameznem nivoju razporejene tako, da je pri pretvorbi realnih vrednosti v poraze in zmage verjetnost poraza za igralca na potezi enaka na vseh nivojih.
4. Vrednosti vozlišč znotraj nivoja so neodvisne.

¹Število vrednosti hevristične funkcije ne vpliva na patološkost.

5. Statična napaka hevristične funkcije je modelirana kot normalno porazdeljen šum ter je neodvisna od prave vrednosti vozlišča in globine preiskovanja.

V razdelku 1.2 smo omenili, da obstajata dve vrsti napake, napaka položaja in napaka poteze. V nadaljevanju se bomo ukvarjali z napako poteze in sicer bomo opazovali delež napačno izbranih potez v korenu drevesa igre pri izbrani globini preiskovanja d in hevristični funkciji h . Izbira poteze je napačna, če ni optimalna. Napako označimo z $E(d, h)$. Do patologije pride, ko pri globljem preiskovanju naredimo več napak kot pri plitvejšem. Patološkost p definiramo z izrazom 3.1:

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

Patologija je odsotna, ko je $p < 1$, saj je v tem primeru pri globljem preiskovanju delež napačno izbranih potez manjši, in obratno, patologija je prisotna, ko je $p > 1$, saj nam globlje preiskovanje ne zmanjša napake.

Analize modelov smo se lotili s simulacijo. Tvorili smo veliko število sintetičnih dreves, na njih izračunali količine, ki nas zanimajo, in rezultate povprečili. Patološkost smo opazovali pri izbranih globinah preiskovanja $d_1 = 1$ in $d_2 = 5$.

Gradnja drevesa poteka po naslednjem postopku: Najprej zgradimo drevo igre z ustrezno vejitvijo b in globino d_{max} . Listom dodelimo eno izmed g ključnih vrednosti, pri čemer upoštevamo verjetnost poraza c_b (upoštevanje c_b za večvrednostni model razložimo v nadaljevanju). Nato z minimaksom izračunamo prave vrednosti vozlišč. Statično napako hevristične funkcije simuliramo tako, da pri preiskovanju do globine $d < d_{max}$ pravim vrednostim vozlišč na nivoju d dodamo šum in s tem dobimo hevristične vrednosti. Za dodajanje šuma uporabimo Gaussov šum [14] s standardnim odklonom $\sigma = 0,1$. Hevristične vrednosti z algoritmom alfabet² prenesemo pod koren drevesa in izračunamo $E(d_1, h)$, $E(d_2, h)$ in p . Celoten postopek je formalno zapisan v algoritmu 2 – *build_tree(b, d)*.

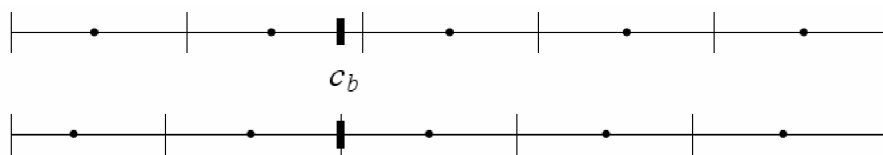
Število pravih vrednosti v vozliščih uravnavamo s parametrom g . Interval $[0, 1]$ razdelimo na g podintervalov enake dolžine. Pri vejitvi $b = 2$ tako vrednosti na podintervalu $[0, 0,5)$ predstavljajo poraz, vrednosti na podintervalu $[0,5, 1]$ pa zmago. Težava pri tej razdelitvi je neupoštevanje vrednosti c_b ,

²Alfabet [22] je izboljšava minimaksa, ki da enake rezultate, a preišče manj vozlišč. Običajno se uporabi pri implementaciji tovrstnega iskanja, saj namesto $O(b^{\frac{d_{max}}{2}})$ obišče le $O(\sqrt{b^{d_{max}}})$ vozlišč.

Algoritem 2 $build_tree(b, d)$

Require: branching factor b , max. tree depth d **Ensure:** pathology degree p $root \leftarrow init\ tree\ with\ b$ **for all** $child$ **of** $root$ **do** $real\ value\ of\ child \leftarrow minimax(child, d - 1)$ $backed-up\ heuristic\ value\ from\ depth\ 5\ of\ child \leftarrow alphabeta(child, 5)$ $backed-up\ heuristic\ value\ from\ depth\ 1\ of\ child \leftarrow alphabeta(child, 1)$ **end for** $error_5 \leftarrow compute\ move\ error\ E(5, noise)$ $error_1 \leftarrow compute\ move\ error\ E(1, noise)$ $p \leftarrow error_5 / error_1$

saj je v tem primeru verjetnost poraza vedno $k = 0,5$. Da bi podintervali upoštevali c_b , je potrebno premakniti notranje meje podintervalov na mejo c_b . Na c_b premaknemo mejo, ki mu je najbliže. V primeru $b = 2$ je parameter $c_b = 0,382$ in podinterval $[0, 0,382)$ predstavlja poraz, podinterval $[0,382, 1]$ pa zmago. Na sliki 3.1 je zgoraj prikazana enakomerna razdelitev intervala $[0, 1]$ na pet podintervalov, spodaj pa poravnava mej na c_b .

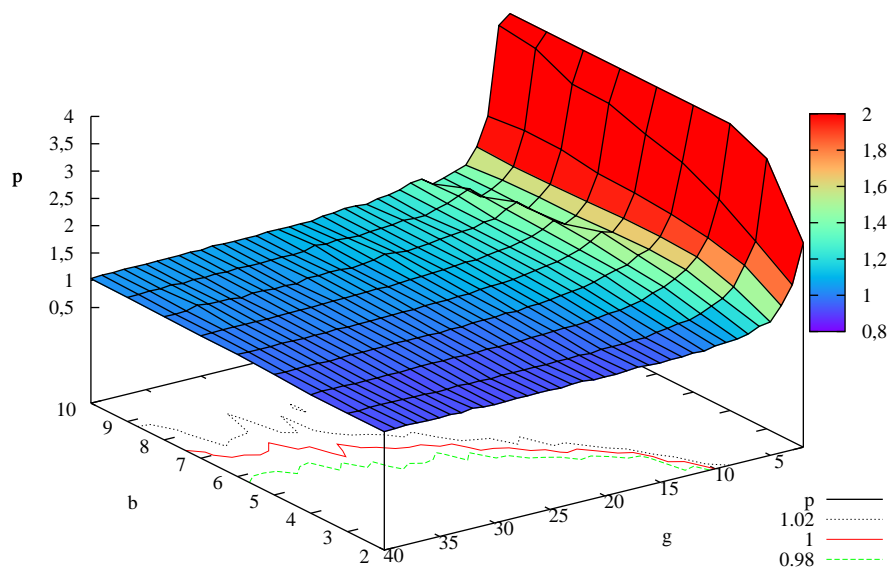
Slika 3.1: Poravnava mej podintervalov na vrednost c_b .

Število pravih vrednosti mora biti natanko g , zato posamezno vrednost s podintervala nadomestimo s srednjo vrednostjo podintervala. Le-ta je na sliki 3.1 v vsakem podintervalu označena s piko.

3.2 Patologija v modelu

Patološkost v modelu smo ugotavljali s simulacijo Monte Carlo³. Za vsak nabor parametrov b iz množice $\{2, 3, \dots, 10\}$ in g iz množice $\{2, 3, \dots, 40\}$ smo tvorili 10.000 dreves z globino $d_{max} = 5$. Pri povečevanju vejitve čas računanja eksponentno narašča, zato smo vse izračune omejili na izbrani interval vejitve. Poskuse pri večjih vejitvah smo opravili v posameznih primerih in z manjšim naborom dreves.

Na sliki 3.2 je s ploskvijo prikazana stopnja patološkosti p v odvisnosti od vejitve drevesa b in števila pravih vrednosti g . Intenzivnost p je ponazorjena z barvo.



Slika 3.2: Stopnja patološkosti p v odvisnosti od vejitve drevesa b in števila pravih vrednosti g . Pri mejnih vrednostih g , pri katerih je patološkost $p = 1$, se patološkost poslabša za 2 % ($p = 1,02$) in izboljša za 2 % ($p = 0,98$), so projicirane na spodnjo ploskev z rdečo, modro in zeleno krivuljo.

Število pravih vrednosti g , potrebnih za odpravo patološkosti, pri večjem

³Metodo Monte Carlo sta leta 1949 predstavila N. Metropolis and S. Ulam [15]. Uporablja se za razumevanje problemov s pomočjo simulacije. Zaradi zaupanja v ponovljivost rezultatov in naključnih števil se metoda široko uporablja za simulacijo matematičnih in fizikalnih problemov s pomočjo računalnika.

b eksponentno narašča. Posamezne mejne vrednosti smo pridobili z ločeno simulacijo. Z ekstrapolacijo znanih mejnih vrednosti smo določili okvirno mejno območje in ga nato podrobneje preiskali. Velikost dreves namreč z vejitvijo strmo narašča in preiskovanje celotnega prostora postane časovno neučinkovito. Mejne vrednosti so povzete v tabeli 3.1.

p/b	2	3	4	5	6	7	8	9	10
0,98	10	12	16	20	26	31	40	46	53
1,00	11	13	18	23	30	39	57	85	110
1,02	13	14	20	26	41	61	90	152	-

Tabela 3.1: Mejne vrednosti g , pri katerih je dosežena patološkost $p = 1$.

Iz slike 3.2 je razvidno, da povečanje števila pravih vrednosti prepreči patologijo. Povečevanje vejitve okrepi patološkost in za odpravo patologije je potrebnih vse več pravih vrednosti v vozliščih.

Rezultati meritev patološkosti so skladni z obstoječimi rezultati Luštreka idr. [11, 13, 12], saj povečevanje števila pravih vrednost odpravi patologijo. Skladni so tudi z ugotovitvami Beala [2] in Naua [4], da večja vejitev povečuje patološkost.

Poglavje 4

Odvisni modeli

Položaji se pri igrah spreminjajo postopno, saj običajno z eno potezo ne pridemo do zmage ali poraza. V poglavju bomo analizirali dva postopna modela odvisnih dreves in predstavili svojega, ki uporablja povsem drug pristop. Preučili bomo delno odvisna drevesa, njihovo zgradbo in stopnjo odvisnosti. Poglavje zaključimo s primerjavo in ovrednotenjem modelov.

4.1 Mere odvisnosti

Stopnjo odvisnosti drevesa so raziskovalci merili z različnimi merami. Zavoljo primerljivosti rezultatov in umestitve raziskave na realnih igrah (Sadikov [18]) v modele smo uporabili dve meri. Podrobneje sta opisani v nadaljevanju.

4.1.1 Faktor grozdenja

Beal [1] je uporabil faktor grozdenja f (omenjen v 2.2), definiran kot delež vozlišč, katerih sinovi imajo enako vrednost. Ker pa je uporabljal model z le dvema pravima vrednostima v vozliščih, definicije ne moremo neposredno uporabiti pri večvrednostnem modelu.

Sadikov [18] je faktor grozdenja razširil in prilagodil za večvrednostne modele. Definiran je kot količnik med povprečnim standardnim odklonom vrednosti bratov¹ in standardnim odklonom celotnega drevesa. Z izračunom je mogoče zaznati, kdaj so si bližnja vozlišča nadpovprečno podobna. Pri dovolj velikem drevesu je standardni odklon celotnega drevesa neodvisen od stopnje

¹Števec v izrazu 4.1 pravzaprav ni čisto pravi povprečni standardni odklon med vrednostmi bratskih vozlišč, saj je vsota kvadratov razlik med posamezno vrednostjo bratskega vozlišča in povprečno vrednostjo bratskih vozlišč povprečena pred operacijo korenjenja.

odvisnosti vozlišč, medtem ko je povprečen standardni odklon sinov pri večji odvisnosti manjši. Izračuna se z izrazom 4.1.

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

Naj še razložimo oznake, ki nastopajo v izrazu (4.1): N je število vseh notranjih vozlišč, M je število vseh vozlišč, v_{ij} je prava vrednost j -tega sina i -tega vozlišča, v_i je prava vrednost i -tega vozlišča, \bar{v}_i je povprečje vrednosti sinov vozlišča i in \bar{v} je povprečna vrednost vseh vozlišč.

Faktor grozdenja je med 0 in 1. Manjši f pomeni močnejšo odvisnost med vozlišči in posledično večji odstotek odvisnih vozlišč.

Takšna definicija (4.1) faktorja grozdenja ni neposredno primerljiva z Bealovo [1]. V nadaljevanju bomo pod izrazom faktor grozdenja imeli v mislih Sadikovo definicijo.

4.1.2 Korelacija

Drugi pristop za ugotavljanje stopnje odvisnosti v drevesu je z izračunom korelacije [23] med očetom in sinovi. Vrednost očeta je enaka eni izmed vrednosti sinov. V primeru, ko so vrednosti odvisne in posledično blizu skupaj, je njihova koreliranost večja. Nasprotno pa je pri neodvisnih, naključnih vrednostih sinov korelacija manjša.

Korelacijo računamo z izrazom 4.2. Za pare (x_i, y_i) vzamemo vozlišče v_i in njegove sinove v_{ij} , npr. pri vejitvi $b = 2$ obravnavamo para $(v_i, v_{i1}), (v_i, v_{i2})$. V izraz za korelacijo (prvi del izraza 4.2) vstavimo pare (x_i, y_i) in izpeljemo izračun korelacije na vseh notranjih vozliščih drevesa med vrednostmi očetov in sinov.

$$\begin{aligned}
r_{xy} &= \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{\sqrt{n \sum_{i=1}^n x_i^2 - \left(\sum_{i=1}^n x_i\right)^2} \sqrt{n \sum_{i=1}^n y_i^2 - \left(\sum_{i=1}^n y_i\right)^2}} = \\
&= \frac{Nb \sum_{i=1}^N \sum_{j=1}^b v_i v_{ij} - b \sum_{i=1}^N v_i \left(\sum_{i=1}^N \sum_{j=1}^b v_{ij}\right)}{\sqrt{Nb^2 \sum_{i=1}^N v_i^2 - \left(b \sum_{i=1}^N v_i\right)^2} \sqrt{Nb \sum_{i=1}^N \sum_{j=1}^b v_{ij}^2 - \left(\sum_{i=1}^N \sum_{j=1}^b v_{ij}\right)^2}} \quad (4.2)
\end{aligned}$$

Vrednost korelacija r_{xy} je na intervalu $[-1, 1]$. Večji r_{xy} predstavlja močnejšo odvisnost vozlišč, medtem ko je faktor grozdenja definiran ravno nasprotno. Ko je r_{xy} blizu 0, so vozlišča neodvisna.

V nadaljevanju se izkaže, da je razpon vrednosti med odvisnimi in neodvisnimi drevesi pri merjenju s korelacijo večji kot pri merjenju s faktorjem grozdenja. Razlike so razvidne v razdelku 4.3.1 na slikah 4.4 in 4.5.

4.2 Verjetnost poraza

Verjetnost poraza, ki se za igralca na potezi ohranja na vseh nivojih in je enaka c_b , se pri delno odvisnih drevesih razlikuje od verjetnosti poraza pri neodvisnih drevesih. Enačba 2.2 se ne pokorava zakonitostim delno odvisnih dreves.

Luštrek idr. [13, 12, 11] so pri preučevanju delno odvisnih dreves definirali c'_b kot število, od katerega je prava vrednost v korenu drevesa z verjetnostjo c_b manjša in z verjetnostjo $1 - c_b$ večja. Tvorili so x dreves², jih uredili naraščajoče glede na velikost prave vrednosti v korenu in izbrali $c_b x$ -to po vrsti. S tem so zagotovili, da položaj v korenu ni preveč v prid enemu izmed igralcev.

Ta pristop upošteva verjetnost poraza c_b v korenu drevesa, a ne zagotavlja ohranjanja verjetnosti poraza po nivojih. Namenili smo se uporabiti izračun, ki bi upošteval tudi slednje.

Najprej definirajmo mero odvisnosti³ $s \in [0, 1]$, $s \in \mathcal{R}$ (angl. *similarity*), ki določa delež odvisnih listov v drevesu. Za izračun novih vrednosti c_b , ki so

²Število x je odvisno od vejitve in nikoli ni manjše od 1000. Pri večji vejitvi so namreč drevesa večja in preiskovanje počasnejše.

³V tuji literaturi se uporablja izraz *similarity*, kar pomeni podobnost. Glede na način gradnje naših odvisnih dreves so vrednosti v vozliščih med seboj odvisne in posledično tudi

poleg b odvisne tudi od s , smo se ponovno zatekli k simulaciji. Definirajmo verjetnost poraza $c_{b,s} \in \mathcal{R}$ pri vejitvi b in stopnji odvisnosti s . Za drevo z izbranimi parametroma b in s zgradimo veliko število dreves z različnimi parametri $c_{b,s} \in [0, 1]$. Pri tem iščemo tako vrednost $c_{b,s}$, pri kateri se verjetnost poraza za igralca na potezi ohranja na vseh nivojih. Z drugimi besedami, verjetnost poraza za igralca na potezi na lihih nivojih mora biti enaka verjetnosti zmage na sodih nivojih.

Nalogo zapišemo kot optimizacijski problem, kjer želimo minimizirati standardni odklon količnika med verjetnostjo poraza in zmage na lihih ter obratne vrednosti na sodih nivojih. Razlog, da smo namesto standardnega odklona k_i in vrednosti $1 - k_i$ na sodih nivojih uporabili kvocient, je v dopustnosti rešitve $c_{b,s} = 0$ in $c_{b,s} = 1$. V teh primerih je igra že na začetku dobljena oz. zgubljena in za preiskovanje nezanimiva. Z uporabo kvocienta so vrednosti v okolici 0 in 1 nestabilne (blizu nič ali zelo velike) in se jim lažje izognemo. Optimizacijski problem formalno zapišemo v (4.3).

$$\begin{aligned} & \operatorname{argmin}_{c_{b,s} \in [0,1]} Q(c_{b,s}), \\ Q(c_{b,s}) &= \sqrt{\frac{1}{N} \sum_{i=1}^{d_{max}} (q_i - \bar{q})^2}, \\ q_i &= \begin{cases} \frac{k_i}{1-k_i}, & \text{če je } i \text{ lih} \\ \frac{1-k_i}{k_i}, & \text{če je } i \text{ sod} \end{cases} \end{aligned} \quad (4.3)$$

Parameter k_i predstavlja verjetnost poraza na nivoju i . Izkaže se, da je prostor preiskovanja konveksne oblike in ga je mogoče učinkovito preiskati. Zaradi nezveznosti prostora in manjšega nihanja vrednosti, k iskanju nismo pristopili z običajnimi metodami za optimizacijo konveksnih funkcij, temveč smo razvili novo.

Vrednosti parametra $c_{b,s}$ lahko zavzamejo vrednosti z intervala $[0, 1]$. Z začetnega intervala izberemo t ekvidistančnih točk $c_{b,s}^i$ in izračunamo vrednosti $Q(c_{b,s}^i)$. V naslednjem koraku interval iskanja omejimo, pri čemer za središče vzamemo točko $c_{b,s}^{i_{min}}$, pri kateri smo dosegli minimalno vrednost. Iskanje na zmanjšanem intervalu ponovimo. Postopek $find.c(b, s)$ je formalno zapisan v algoritmu 3.

Verjetnost poraza $c_{b,s}$ izračunamo na tri decimalna mesta natančno v sedmih korakih, pri čemer naredimo 30 izračunov $Q(c_{b,s})$. Vsak izračun $Q(c_{b,s})$ je

podobne. Z izrazom *odvisnost* bolje opišemo podobnost listov in glede na njegovo uporabo v slovenski literaturi [11], ga bomo uporabljali tudi v tem delu.

Algoritem 3 *find_c(b, s)*

Require: branching factor b , similarity degree s **Ensure:** best c using criteria $\operatorname{argmin}_{c_{b,s} \in [0,1]} Q(c_{b,s})$ $precision \leftarrow \{0,1, 0,05, 0,02, 0,01, 0,005, 0,002, 0,001\}$ $from \leftarrow 0$ $to \leftarrow 1$ $min\ ratio \leftarrow \infty$ **for all** $step$ **in** $precision$ **do** $interval \leftarrow \{from, from + step, from + 2step, \dots, to\}$ **for all** c **in** $interval$ **do** $build\ tree\ with\ b, s, c$ $ratio \leftarrow compute\ Q(c_{bs})$ **if** $min\ ratio < ratio$ **then** $min\ ratio \leftarrow ratio$ $best\ c \leftarrow c$ **end if****end for** $from \leftarrow best\ c - step$ $to \leftarrow best\ c + step$ **end for**

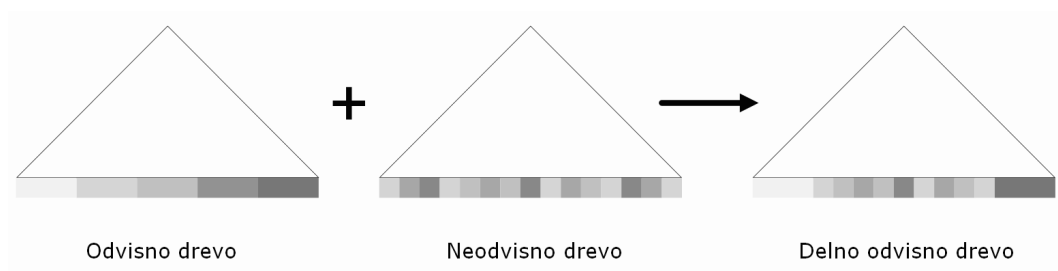
sestavljeno iz nabora 10.000 dreves. Vrednosti $c_{b,s}$ izračunamo za določen model odvisnosti samo enkrat. Numerične vrednosti za posamezne modele so zbrane v tabelah v prilogi A.

4.3 Postopni modeli odvisnosti

Skupna lastnost postopnih modelov odvisnosti je način gradnje odvisnih dreves. Korenu se pripiše neka začetna začasna vrednost, ki se nato postopno spreminja in prenaša na sinove po drevesu navzdol, vse do listov. V listih začasne vrednosti postanejo prave. Tako prenesene vrednosti se v bratskih listih razlikujejo le za en nivo spreminjanja. Modeli se razlikujejo v načinu spreminjanja in prenašanja pomožnih vrednosti.

Pristop gradnje delno odvisnih dreves je povsem enak. Drevo, zgrajeno s postopnim modelom odvisnosti, vzamemo kot popolnoma odvisno drevo z odvisnostjo $s = 1$. Nasprotno vzamemo drevo, zgrajeno z neodvisnim modelom, kot drevo z odvisnostjo $s = 0$. Da bi dobili delno odvisno drevo z odvisnostjo $s = s_0$, vozliščem na predzadnjem nivoju z verjetnostjo s_0 dodelimo sinove

z odvisnimi pravimi vrednostmi in z verjetnostjo $1 - s_0$ neodvisne sinove z naključnimi pravimi vrednostmi. Na sliki 4.1 je ponazorjeno kombiniranje odvisnih in neodvisnih sinov.



Slika 4.1: Delno odvisna drevesa zgradimo s kombinacijo popolnoma odvisnega in neodvisnega drevesa.

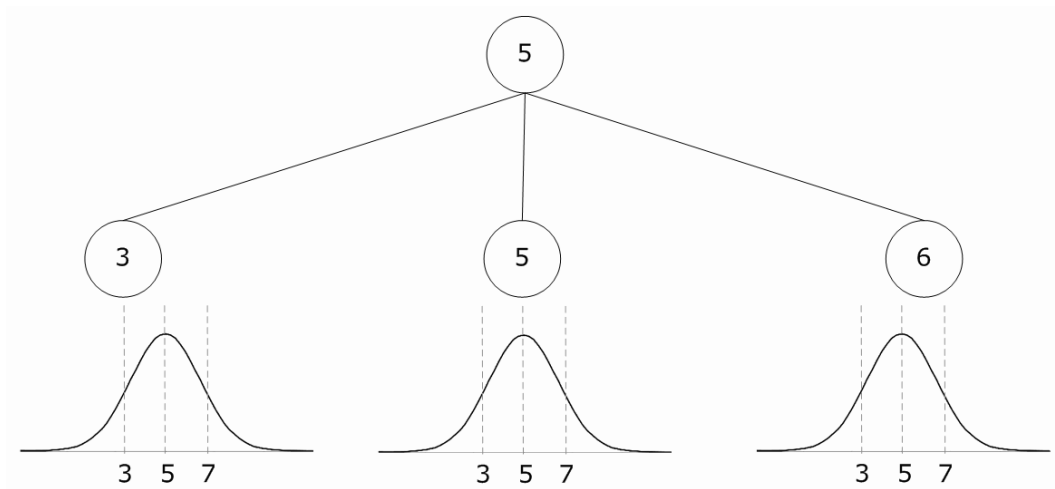
4.3.1 Model s porazdeljevanjem pomožnih vrednosti

Gradnja popolnoma odvisnega modela je povzeta po Luštreku [11]. Popolnoma odvisno drevo zgradi tako, da korenu pripiše pomožno vrednost ζ_i , ki jo nato uporabi pri določanju prave vrednosti sinov. Pri vsakem prehodu na nižji nivo se vrednosti sinov ζ_{ij} porazdelijo okrog starša z normalno porazdelitvijo [14]. Bratski listi se tako med seboj razlikujejo le za en nivo porazdeljevanja.

Začetna vrednost v korenu je 0,5. Pri prenosu na nižji nivo se vrednosti normalno porazdelijo s $\sigma = 1,413$. Vrednosti, ki po porazdelitvi segajo izven intervala $[0, 1]$, ponovno porazdelimo, dokler ne pristanejo v intervalu. Parameter σ je izbran tako, da je takšnih vrednosti pri globini drevesa $d_{max} = 5$ natanko 5 %. Na sliki 4.2 je prikazano porazdeljevanje vrednosti sinov okrog starša.

Gradnja delno odvisnih dreves poteka na način, opisan v razdelku 4.3. Na predzadnjem nivoju z verjetnostjo s dodelimo sinovom odvisne vrednosti oz. z verjetnostjo $1 - s$ naključne vrednosti. Dobljenemu drevesu nato preslikamo realne vrednosti listov v prave vrednosti, kar storimo s poravnavo mej g podintervalov na $c_{b,s}$ in preslikavo realnih vrednosti v srednje vrednosti podintervala. Postopek poravnave mej in preslikave smo opisali v razdelku 3.1.

Izračun stopnje patološkosti smo pridobili s simulacijo. V razdelku 3.2 opisanim poizkusom smo dodali še dimenzijo odvisnosti. Drevesa smo tvorili za nabor parametrov $b \in \{2, 3, \dots, 10\}$, $g \in \{2, 3, \dots, 40\}$, $s \in [0, 1]$, korak pri s je 0,1.



Slika 4.2: Korenu dodelimo začetno pomožno vrednost. Pomožne vrednosti sinov se porazdelijo okrog starša in v listih postanejo prave.

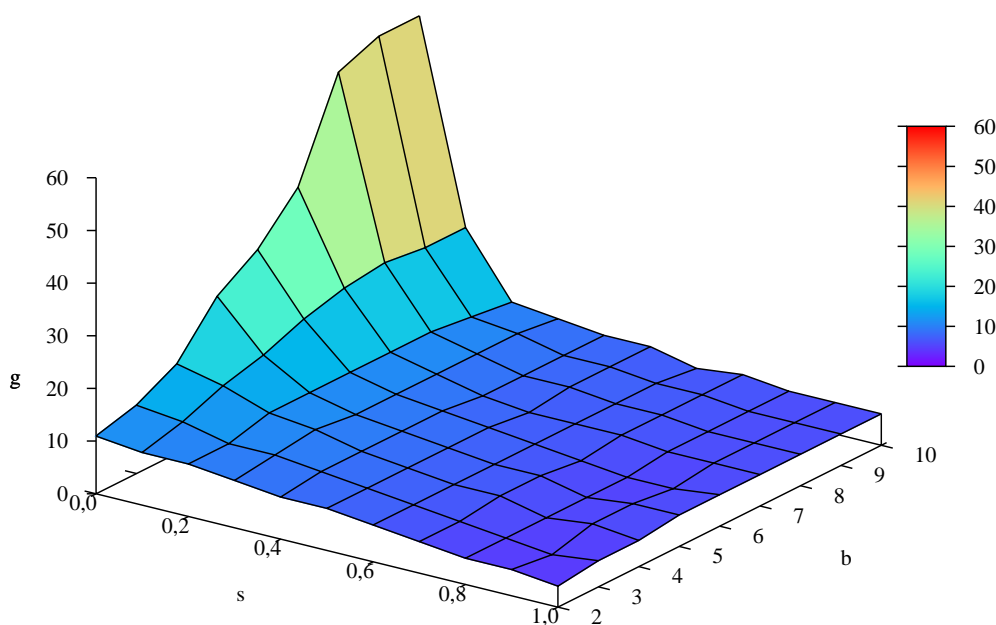
Na sliki 4.3 je prikazana meja zrnatosti g , pri kateri je patološkost $p = 1$. Območje pod ploskvijo je patološko, nad ploskvijo pa nepatološko.

Rezultati so kljub nekoliko drugačnem izračunu⁴ $c_{b,s}$ kvalitativno primerljivi z rezultati, ki so jih dobili Luštrek idr. [13, 11]. Število vrednosti g , potrebnih za odpravo patologije (meja $p = 1$), pri majhni odvisnosti eksponentno narašča z vejitvijo, medtem ko je pri odvisnosti $s \geq 0,2$ povečini enako, neodvisno od vejitve. Pri povečanju odvisnosti stopnja patologije upada, a kljub majhni zrnatosti patologija ne izgine povsem. Presenetilo nas je dejstvo, da se zrnatost in patološkost v modelu z odvisnostjo $s = 0,2$ obnašata povsem enako kot v popolnoma odvisnem modelu z odvisnostjo $s = 1$. Pa dec patološkosti že pri majhni stopnji odvisnosti nakazuje razlog za odsotnost patologije v igrah.

Stopnjo odvisnosti modela smo izmerili z obema merama, faktorjem grozdenja f in korelacijo r_{xy} . Grafični prikaz stopnje odvisnosti je za faktor grozdenja prikazan na sliki 4.4 in za korelacijo na sliki 4.5.

Sliki 4.4 in 4.5 sta precej podobni in poleg nasprotne definicije za močno odvisnost ni opaziti bistvene razlike v merah. Koordinatni osi b in s sta zaradi boljšega prikaza podobnosti ravno obrnjeni. Pri posamezni vejitvi b je prehod od neodvisnega k odvisnemu drevesu linearen, medtem ko se s povečevanjem vejitve f večja oz. r_{xy} manjša. Numerične vrednosti so zbrane v prilogi A in sicer vrednosti f v tabeli A.8, vrednosti r_{xy} pa v tabeli A.5.

⁴Luštrek idr. so uporabili c'_b , opisan je v 4.2.



Slika 4.3: Meja zrnatosti g , pri kateri je patološkost $p = 1$ v odvisnem modelu s porazdeljevanjem pomožnih vrednosti.

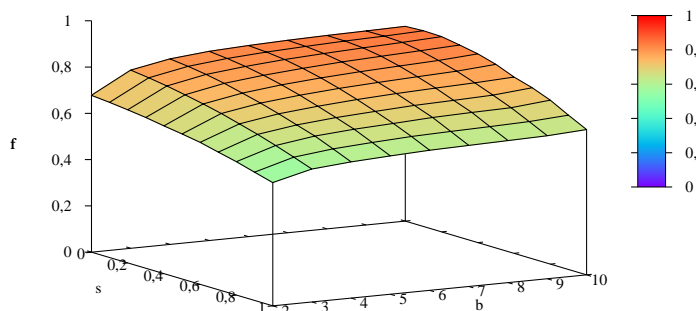
4.3.2 Model z normalizirano vsoto

Ideja za popolnoma odvisno drevo z normalizirano vsoto je povzeta po Nauovi vpeljavi odvisnosti v Pearlovo igro [6]. Spomnimo se, da povezavam v drevesu igre naključno dodeli vrednosti -1 ali 1 in v listih sešteje pot od lista do korena. Prava vrednost v listu postane 1 ob pozitivni vsoti, oz. 0 ob negativni vsoti.

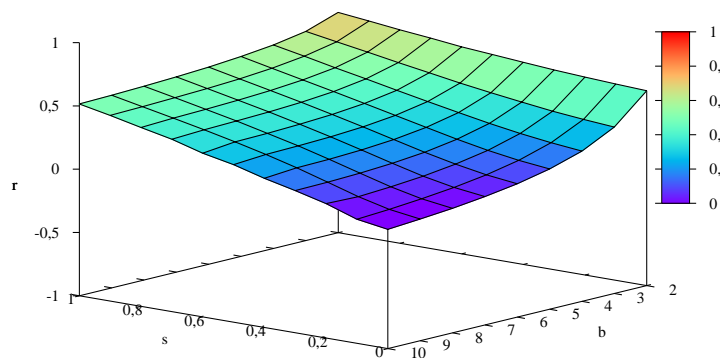
Nauov pristop smo za večvrednostni model prilagodili. V korenu drevesa definiramo pomožno vrednost $\zeta_0 = 0$. Sinovom v_{ij} vozlišča v_i dodelimo pomožno vrednost $\zeta_{ij} = \zeta_i + \zeta'_0$, kjer je ζ_i pomožna vrednost starša, ζ'_0 pa naključna porazdeljena spremenljivka s srednjo vrednostjo 0 in odklonom σ . Vrednosti, ki po porazdeljevanju segajo izven intervala 3σ , ponovno porazdelimo.

V naslednjem koraku poiščemo maksimalno in minimalno vrednost v listih ter normaliziramo vse liste na interval $[0, 1]$. Prave vrednosti v listih določimo z opisanim postopkom premikanja mej podintervalov (3.1).

Rezultati simulacije so kvalitativno podobni modelu s porazdelitvijo po-



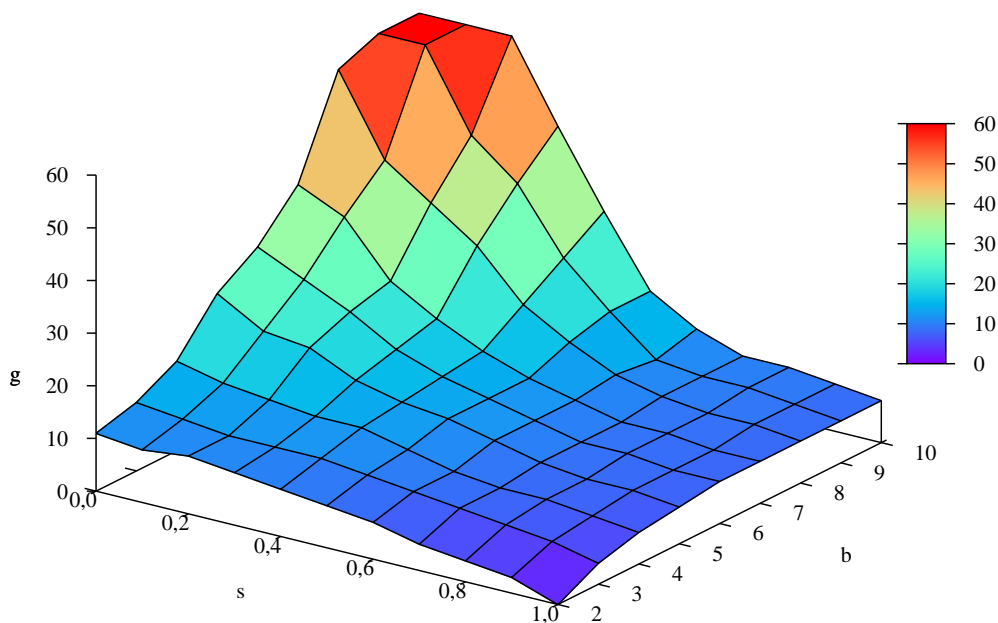
Slika 4.4: Stopnja odvisnosti v modelu s porazdeljevanjem pomožnih vrednosti, izmerjena s faktorjem grozdenja f . Prikazana je v odvisnosti od stopnje s in vejitve b .



Slika 4.5: S korelacijo r_{xy} izmerjena stopnja odvisnosti v modelu s porazdeljevanjem pomožnih vrednosti. Prikazana je v odvisnosti od stopnje s in vejitve b .

možnih vrednosti. Prikazani so na sliki 4.6. Meja zrnatosti g , pri kateri patološkost doseže $p = 1$ z naraščajočo odvisnostjo upada in se povečuje z

naraščajočo vejitevijo. Meja patologije se vzdolž s spreminja bolj postopno kot pri modelu s porazdeljevanjem pomožnih vrednosti.



Slika 4.6: Meja zrnatosti g , pri kateri je patološkost $p = 1$ v odvisnem modelu s porazdeljevanjem pomožnih vrednosti.

Patološkost s povečano vejitevijo upada in pri majhni zrnatosti in vejitvi tudi povsem izgine. Popolnoma odvisna drevesa dosega jo večjo odvisnost glede na mere f in r_{xy} . Vrednosti f in r_{xy} so zbrane v prilogi v tabelah A.5 in A.7.

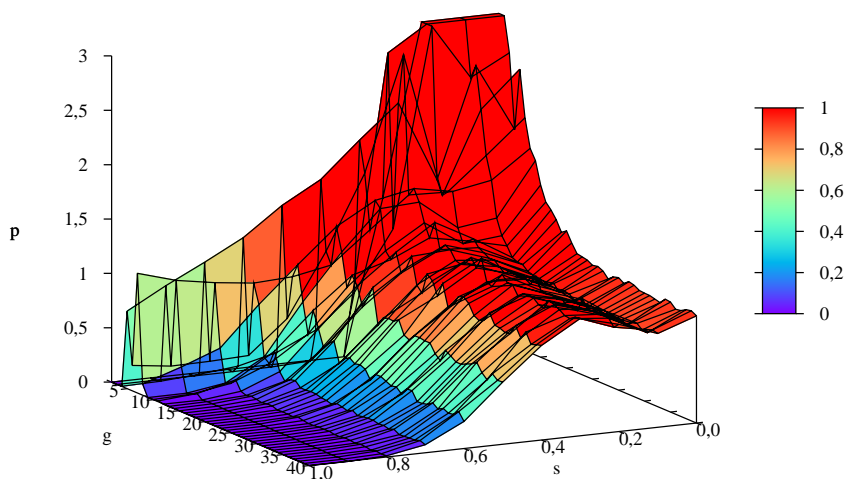
4.4 Model z urejanjem naključnih vrednosti

Postopna modela odvisnosti sta si po zgradbi precej podobna. Namenili smo se zgraditi nov model odvisnosti, ki bo uporabljal povsem drugačen pristop gradnje odvisnih dreves.

Gradnja popolnoma odvisnega drevesa poteka na naslednji način: Listom dodelimo naključne pomožne vrednosti, nato pa jih sortiramo po velikosti. Listi imajo po naraščajočih indeksih naraščajoče pomožne vrednosti. Dobljeno drevo označimo kot drevo z odvisnostjo $s = 1$.

Delno odvisno drevo z odvisnostjo $s = s_0$ zgradimo z vstavljanjem listov odvisnega drevesa v neodvisno na naključna mesta, dokler ne zamenjamo delež s_0 listov. To storimo na enak način, kot pri postopnih modelih v razdelku 4.3.

Pristop je sprva kazal vzpodbudne rezultate, vendar so se pojavile težave. Npr. pri $b = 6$ in $s = 0,3$ je bila patologija prisotna pri vseh zrnatostih in se vzdolž s ni spreminjala monotono. Na sliki 4.7 je prikazana stopnja patologije v odvisnosti od zrnatosti in odvisnosti pri konstantni vejitvi $b = 6$. Iz slike je razvidna omenjena težava.



Slika 4.7: Patologija je pri vejitvi $b = 6$ in stopnji odvisnosti $s = 0,3$ venomer prisotna, poleg tega pa se patološkost vzdolž s ne spreminja monotono.

Domnevali smo, da razlog za takšno obnašanje tiči v ponesrečeni gradnji delno odvisnih dreves. Vrednosti, ki jih dodamo na sredino, imajo zaradi narave delovanja minimaksa manjši vpliv. Posledično patologija začne upadati šele pri odvisnosti $s > 0,3$.

4.4.1 Delno urejanje

Da bi se izognili težavi, ki jo povzroča premalo dodanih vrednosti na robove neodvisnega drevesa, smo pred vstavljanjem odvisnega drevesa v neodvisno naraščajoče uredili veje neodvisnega drevesa na prvem nivoju. Naključna po-

razdelitev pravih vrednosti v listih drevesa ostane nespremenjena, saj smo veje le obravnavali v drugačnem vrstnem redu.

Iz odvisnega drevesa vzamemo delež $\frac{s_0}{2}$ prvih ter delež $\frac{s_0}{2}$ zadnjih listov⁵ in jih vstavimo na ista mesta v neodvisno drevo. Drevo je tako bolj odvisno na skrajnih robovih, pri povečevanju s pa postaja vse bolj odvisno tudi v srednjem delu.

Na sliki 4.8 je zgoraj prikazano neodvisno drevo, ki mu zamenjamo veji na prvem nivoju. Tako urejenim vejam z leve in desne zamenjamo delež $\frac{s}{2}$ vrednosti listov. Spodaj je prikazano neodvisno drevo z zamenjanimi vejami, pod njim pa so listi odvisnega drevesa z naraščajoče urejenimi vrednostmi, ki jih z leve in desne vstavimo na pripadajoča mesta v neodvisno drevo. Stopnja odvisnosti bratskih listov se poveča, hkrati pa dodajanje na ta način zagotavlja, da je več vrednosti dodanih na robove.

Težavo s stalno prisotnostjo patološkosti (npr. pri $b = 6$ in $s = 0,3$) smo odpravili, vendar pa je odvisnost delno odvisnih dreves premočna. Patologija namreč izgine že pri $s = 0,3$. Prehod med neodvisnimi in odvisnimi drevesi je nekoliko prehiter, da bi lahko opazovali, kaj se dogaja z delno odvisnimi drevesi. Na sliki 4.9 je prikazana stopnja patološkosti pri $b = 6$. Iz slike je razviden oster padec stopnje patološkosti in njena odsotnost pri večjem s in g .

4.4.2 Trikotniška porazdelitev

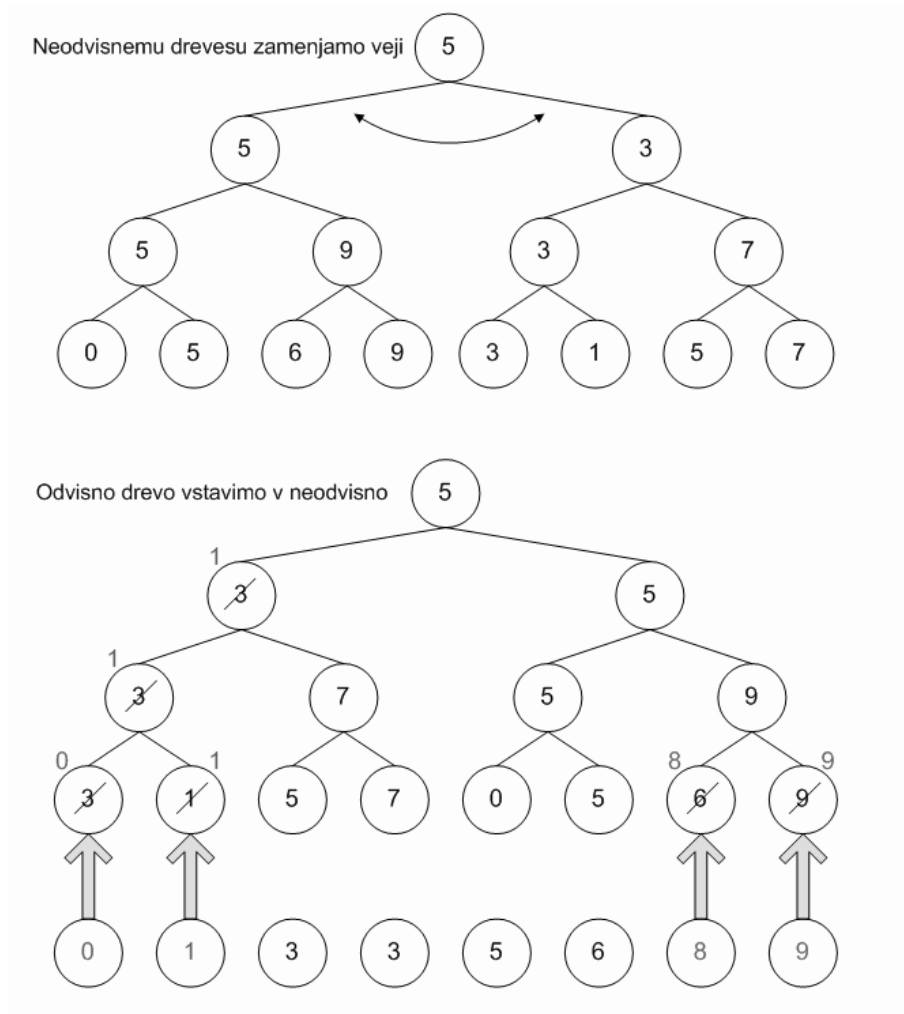
Za ublažitev prehoda med neodvisnimi in odvisnimi drevesi bi potrebovali postopno dodajanje listov odvisnega drevesa k neodvisnemu. Namesto, da zamenjamo delež skrajnih $\frac{s}{2}$ listov na obeh straneh, bi zamenjali delež s vseh listov, pri čemer bi bila verjetnost zamenjave lista na robu večja kot za list v sredini drevesa.

Definirajmo funkcijo $H_s(i)$ kot verjetnost zamenjave i -tega lista v neodvisnem drevesu z i -tim listom odvisnega drevesa. Funkcijo poimenujemo *trikotniška porazdelitev* in je odvisna od stopnje odvisnosti s . Sestavljena je iz unije dveh polovic trikotnika, povprečna višina unije pa je natanko s .

Na sliki 4.10 je prikazana funkcija $H_s(i)$ pri različnih s . Na vodoravni osi so indeksi listov i , normalizirani na interval $[0, 1]$, na navpični osi pa verjetnost zamenjave $H_s(i)$. Verjetnost zamenjave skrajnih indeksov na robu je večja kot v sredini. Pri $s = 1$ zamenjamo vse liste, medtem ko pri $s = 0$ ostane drevo nespremenjeno.

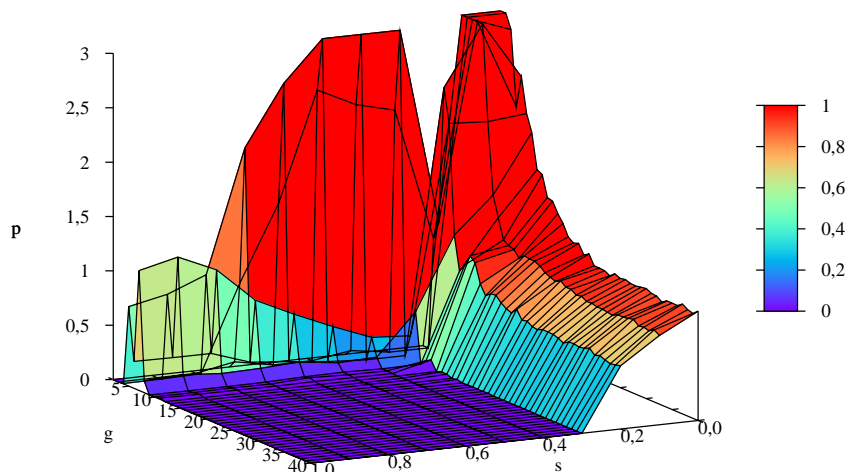
Meritve so pokazale zmernejši prehod med odvisnimi in neodvisnimi drevesi. Pri stopnji odvisnosti $s = 0,5$ je zaznati grobo spremembo v naklonu

⁵Listi odvisnega drevesa so urejeni v naraščajočem vrstnem redu glede na pravo vrednost.

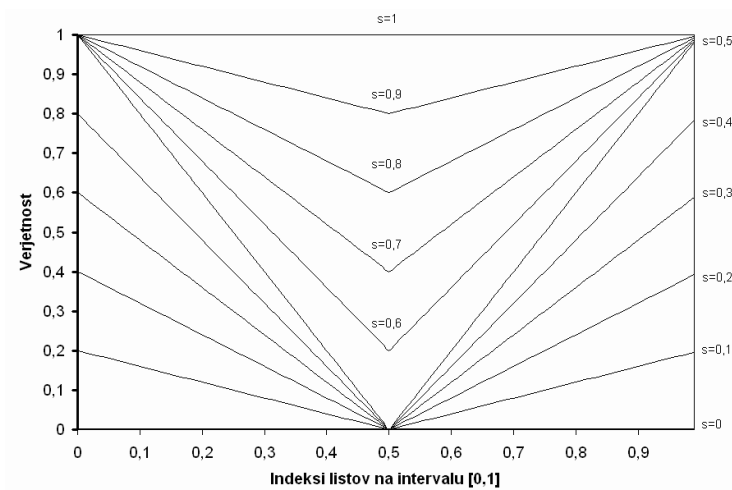


Slika 4.8: Delno odvisno drevo zgradimo z naraščajočo ureditvijo vej neodvisnega drevesa na prvem nivoju in zamenjavo dela naključnih pravih vrednosti v listih z odvisnimi pravih vrednostmi.

stopnje patološkosti. Nezveznost same trikotniške porazdelitve pri $s = 0,5$, kjer se prične dvigovati srednja točka (slika 4.10), se namreč odraža tudi na patološkosti. Na sliki 4.11 je prikazana patologija pri vejitvi $b = 6$. Iz slike je razviden prelom p pri $s = 0,5$.



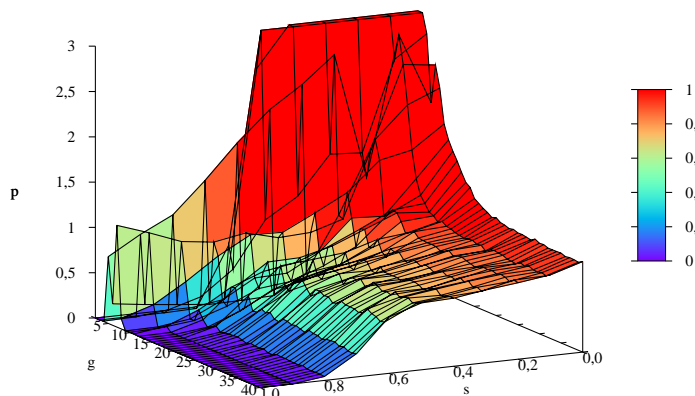
Slika 4.9: Grob prehod med neodvisnimi in odvisnim drevesi pri vejitvi $b = 6$ v modelu z urejanjem naključnih vrednosti pri dodajanju odvisnih listov z leve in desne.



Slika 4.10: Trikotniška porazdelitev $H_s(i)$.

4.4.3 Gladka porazdelitev

V izogib dvigovanju srednje točke porazdelitve $H_s(i)$ smo definirali novo porazdelitev $H'_s(i)$. Definirali smo jo na naslednji način: Liste odvisnega drevesa



Slika 4.11: Prehod med neodvisnimi in odvisnim drevesi se pri modelu z urejanjem naključnih vrednosti in uporabi trikotne porazdelitve $H(s)$ pri vejitvi $b = 6$ in odvisnosti $s = 0,5$ prelomi.

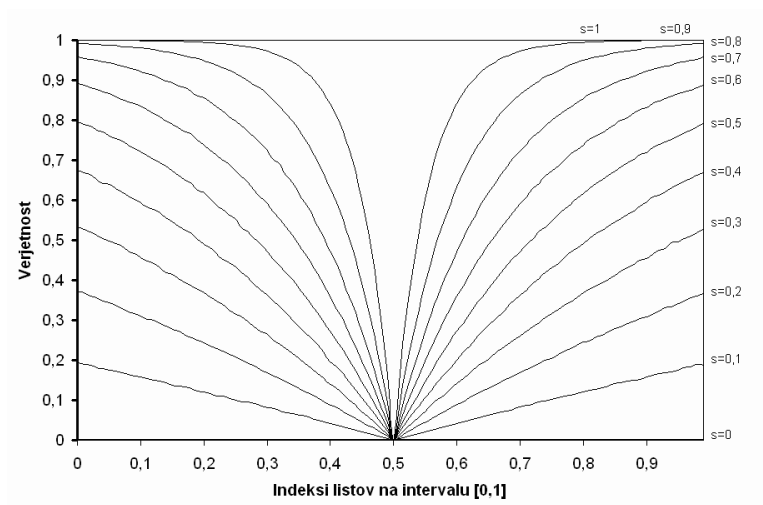
dodamo v neodvisno na mesto i z verjetnostjo $H_{0,5}(i)$. To ponavljamo, dokler drevo ne vsebuje delež s odvisnih vozlišč. Tako dosežemo, da se več vrednosti zamenja na robovih. Verjetnost zamenjave vozlišča i pri stopnji odvisnosti s (oz. deležu s zamenjanih listov) določa gladka porazdelitvena funkcija $H'_s(i)$.

Zaradi hitrejših izračunov patološkosti smo novo porazdelitev $H'_s(i)$ pri posameznem s izračunali v ločeni simulaciji in med gradnjo delno odvisnih dreves uporabili izračunane vrednosti. Gladka porazdelitev H'_s je za različne stopnje s prikazana na sliki 4.12.

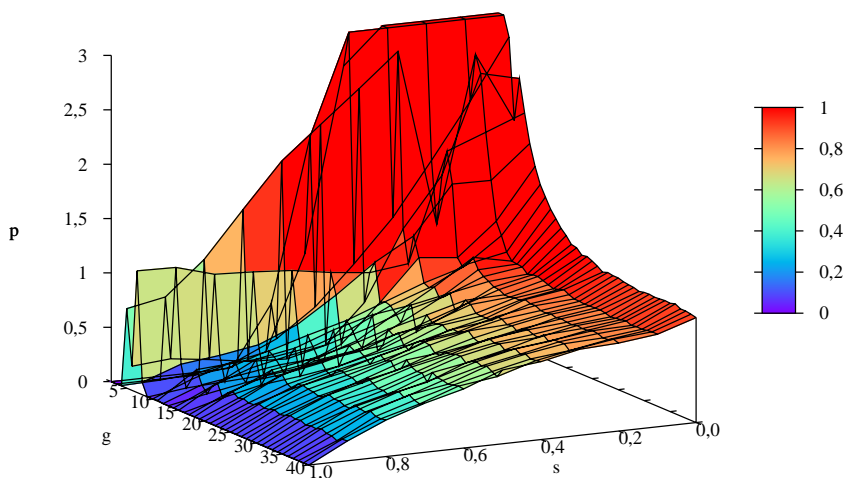
Zvezen prehod v načinu tvorjenja delno odvisnih dreves se pozitivno odraža na rezultatih. Stopnja patološkosti enakomerno pada s povečevanjem odvisnosti. Za primerjavo s slikami 4.7, 4.9 in 4.11 je na sliki 4.13 prikazana stopnja patološkosti pri vejitvi $b = 6$.

Na sliki 4.14 so prikazani končni rezultati meritev patološkosti. Prikazana je meja zrnatosti, pri kateri doseže $p = 1$. Model je kvalitativno primerljiv s postopnimi modeli odvisnosti. Patologija pri popolni odvisnosti in pri manjših vejitvah popolnoma izgine, medtem ko je pri večjih vejitvah kljub odvisnosti venomer prisotna. S povečevanjem odvisnosti je zaznati rahel padec meje patološkosti. Meja patologije $p = 1$ pa, tako kot pri modelu s porazdeljevanjem pomožnih vrednost, pade že pri majhni stopnji odvisnosti.

V primerjavi s postopnimi modeli (sliki 4.4 in 4.5) faktor grozdenja in

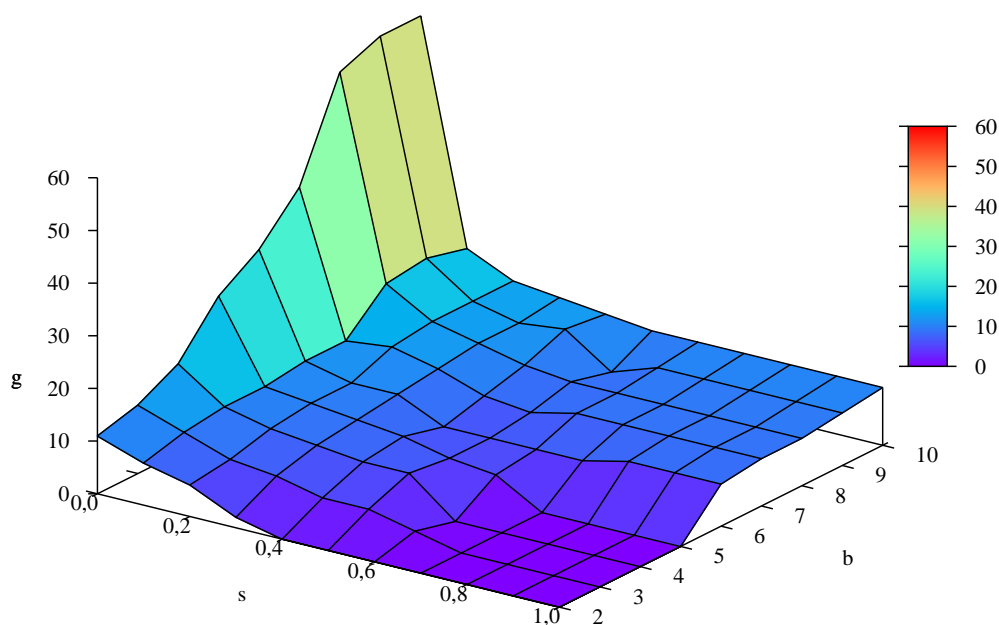


Slika 4.12: Gladka porazdelitev $H'_s(i)$.



Slika 4.13: Enakomeren prehod med neodvisnimi in odvisnimi drevesi v modelu z urejanjem naključnih vrednosti pri vejitvi $b = 6$ in uporabi gladke porazdelitvene funkcije $H'_s(i)$.

korelacija dosežata večjo odvisnost. Numerične vrednosti se nahajajo v prilogi



Slika 4.14: Mejne vrednosti g , pri katerih je $p = 1$ v modelu z urejanjem naključnih vrednosti.

A v tabelah A.6 in A.9.

4.5 Primerjava modelov

V četrtem poglavju smo analizirali tri odvisne modele preiskovanja. Kljub korenito drugačnemu pristopu h gradnji odvisnih in delno odvisnih dreves so rezultati kvalitativno podobni, kot je razvidno na slikah 4.3, 4.6 in 4.14. Meja zrnatosti, pri kateri je patološkost $p = 1$, pri modelu s porazdeljevanjem pomožnih vrednosti in modelu z urejanjem naključnih vrednosti bliskovito pade že pri majhni stopnji odvisnosti. Pri odvisnosti $s > 0,2$ je povečini enaka in rahlo raste s povečevanjem vejitve. Pri modelu z normalizirano vsoto je padec meje patološkosti vzdolž s počasnejši in enakomeren.

Vrednosti v listih modela z urejanjem naključnih vrednosti dosegajo večjo odvisnost glede na meri f in r_{xy} kot v postopnih modelih odvisnosti. Velik razpon odvisnosti glede na meri je zelo zaželen za primerjavo in umestitev

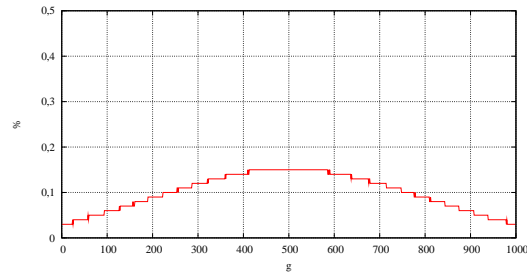
pravih iger v modele. Sadikov [18] je pokazal, da šahovske končnice dosegajo veliko stopnjo odvisnosti glede na mero f . Postopni modeli bolje posnemajo ustvarjanje odvisnih položajev v drevesu prave igre, saj se tudi situacija med igro spreminja postopoma. Nasprotno pa model z naraščajoče urejenimi praviimi vrednostmi nima neposredne primerjave s spreminjanjem položajev med igro.

Na sliki 4.15 je prikazana primerjava porazdelitve pravih vrednosti v listih. Prave vrednosti v listih modela z urejanjem naključnih vrednosti so razporejene enakomerno preko vseh g (slika 4.15c). Razporeditev v postopnih modelih odvisnosti pa je bližje potlačeni normalni porazdelitvi (slika 4.15a in 4.15b).

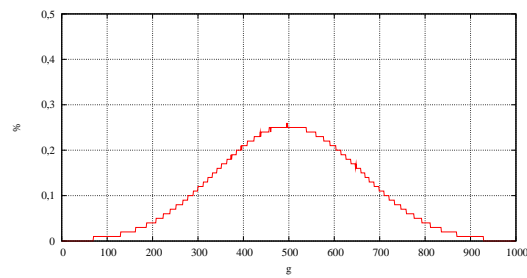
Vsi modeli imajo določene težave pri povečani vejitvi. V območju zrnatosti $[2, b]$ in povečani odvisnosti patološkost zelo niha, je nestabilna glede na b in s . Pojav je prisoten pri vseh modelih.

V okolici zrnatosti $g = b$ in odvisnostni $s > 0,6$ se bolj ali manj izrazito pojavi nerazjasnjena grbina (slika 4.14). Prisotna je pri vseh modelih in patološkosti v tem območju s povečevanjem odvisnosti narašča. Razlogom za pojav grbine nismo uspeli priti do dna in glede na njeno pojavitev v vseh treh modelih lahko le izključimo napako v programski kodi simulatorja.

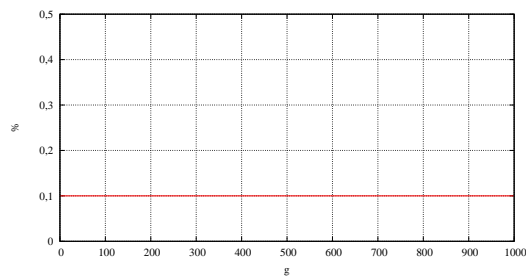
Rezultati analize modelov so skladni z obstoječimi raziskavami iz drugega poglavja. Skladni so z Bealom [1, 2] in Nauom [5], da povečanje vejitve povzroča močnejšo patološkost, a zmanjšanje vejitve ni dovolj, da patologija povsem izgine. K odpravi patologije pripomore večje število pravih vrednosti v listih drevesa igre, kot so že ugotovili Scheucherj in Kaindl [20] ter Luštrek idr. [13, 12, 11]. Patološkost uspešno zmanjšuje tudi vpeljava odvisnosti v drevesa igre, kar se sklada z ugotovitvami večine raziskovalcev [2, 3, 5, 11, 21].



(a) Model s porazdeljevanjem pomožnih vrednosti.



(b) Model z normalizirano vsoto.



(c) Model z urejanjem naključnih vrednosti.

Slika 4.15: Primerjava porazdelitve $g = 1.000$ pravih vrednosti v listih popolnoma odvisnih modelov.

Poglavje 5

Pearlova igra

V tem poglavju preverimo izsledke analize modelov na realni igri. Predstavljena je Pearlova igra in njena razširitev na vse parametre modela. Preizkusimo različne hevristične funkcije in opazujemo napako ter pojav patološkosti.

5.1 Opis in razširitev igre

J. Pearl [16] je za potrebe analize učinkovitosti različnih preiskovalnih algoritmov drevesa igre razvil razred iger za dva igralca s popolno informacijo. Pearlovo igro, znano tudi kot *P-igra*, so raziskovalci že v preteklosti uporabili za analizo iskanja z minimaksom. Drugo različico igre je razvil Dana S. Nau [5] in jo poimenoval *N-igra* (omenili smo jo v razdelku 2.2). V njej je vpeljal pojem odvisnosti, a več o tem v nadaljevanju.

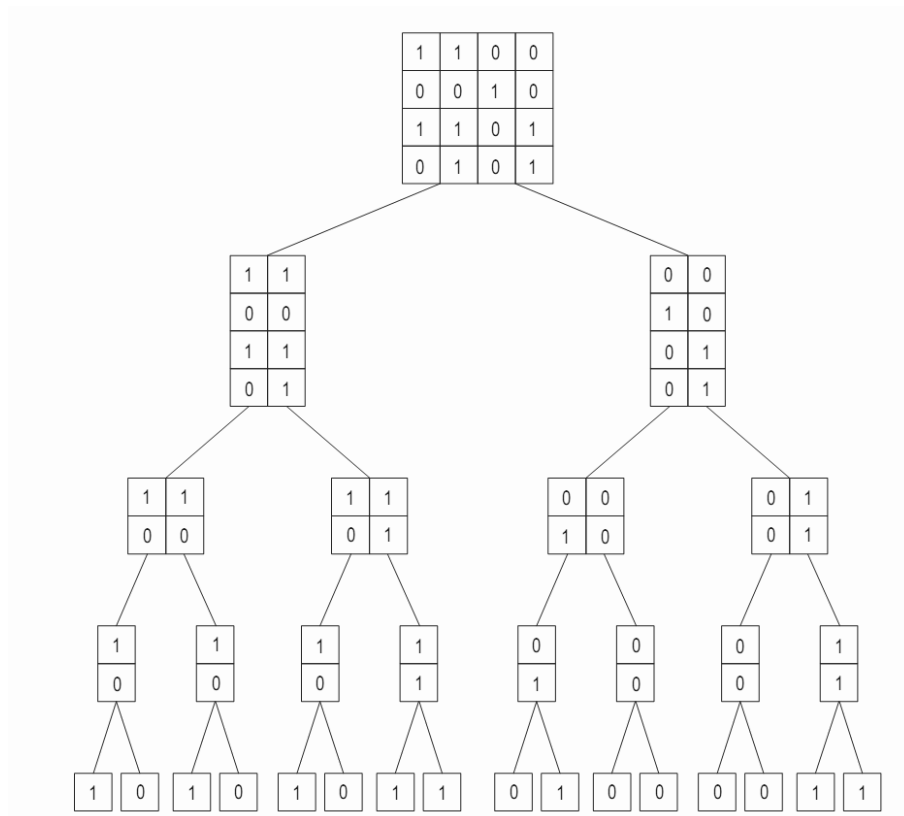
Pearlova igra se igra na plošči velikosti $2^{\lceil \frac{d}{2} \rceil} \times 2^{\lfloor \frac{d}{2} \rfloor}$ polj, $d \in \mathcal{Z}^+$, kjer d pomeni število potez do konca igre. Naloga prvega igralca je, da ploščo navpično razdeli na polovico in izbere bodisi levo bodisi desno polovico. Polja iz neizbrane polovice plošče so izločena iz nadaljnje igre. Drugi igralec razdeli ploščo vodoravno in izbere bodisi zgornjo bodisi spodnjo polovico. Igralca izmenično vlečeta poteze, dokler na plošči ne ostane eno samo polje. Če igralcu, ki je naredil zadnjo potezo, ostane polje z vrednostjo 1, je zmagovalec, sicer zmaga njegov nasprotnik.

V posamezni potezi igralec razdeli ploščo na dva dela, torej je vejitev drevesa igre $b = 2$. Poljem na plošči se pred začetkom igre naključno pripišejo vrednosti 0 ali 1 tako, da je število enic in ničel v razmerju¹ $(1 - c_2) : c_2$.

Na sliki 5.1 je prikazana igralna plošča s štirimi potezami do konca igre

¹Verjetnost poraza nastavimo na vrednost c_b , kar je v našem primeru c_2 .

($d = 4$) in delitvijo plošče na dva dela ($b = 2$). Vse možne delitve plošče in nadaljnje poteze so predstavljene v drevesu igre.



Slika 5.1: Drevo igre pri igralni plošči z $b = 2$ in $d = 4$.

Igra v osnovi predstavlja dvovrednostni² neodvisni model. Razširitev na večvrednostni model je dokaj enostavna, saj lahko poljem pripišemo več vrednosti. V nasprotju z dvovrednostnim modelom sedaj ne moremo več govoriti o zmagi in porazu, lahko pa vrednosti interpretiramo kot točke, kjer si prvi igralec prizadeva za čim višje in drugi igralec za čim nižje število točk. V splošnem poljem pripišemo g vrednosti, vendar pa to ne zadostuje nujno za doseg zrnatosti g . Hevristična funkcija lahko v splošnem vrne več kot g vrednosti, zato je potrebno te vrednosti preslikati v predstavnike intervalov zrnatosti, kot smo opisali v razdelku 3.1.

²Ime *dvovrednostni model* se nanaša na število pravih vrednosti v listih drevesa igre. V primeru, ko sta vrednosti dve, govorimo o dvovrednostnem modelu, ko imamo več vrednosti, pa ga poimenujemo večvrednostni model.

Pri izvorni igri igralca v vsaki potezi delita ploščo na polovico, kar predstavlja vejitev $b = 2$. Če želimo doseči drugačno vejitev b , morata igralca v vsaki potezi razdeliti ploščo na b delov in izbrati enega. Začetna velikost plošče je v tem primeru $b^{\lceil \frac{d}{2} \rceil} \times b^{\lfloor \frac{d}{2} \rfloor}$.

Odvisnost med posameznimi polji je vpeljal že Nau in igro poimenoval *N-igra*. Nauova odvisnost se nekoliko razlikuje od obravnavanih modelov, zato odvisnost v igro vpeljemo s postopnim modelom s porazdelitvijo pomožnih vrednosti (opisan je v razdelku 4.3.1). V drevesu igre korenu dodelimo začasno vrednost in jo pri prenosu na sinove porazdeljujemo z normalno porazdelitvijo. V listih začasne vrednosti postanejo prave vrednosti v poljih.

Glede na izvorno *P-igro* smo igri dodali večvrednostni izid in vpeljali postopno odvisnost vrednosti v poljih. Tako spremenjeno igro poimenujemo *K-igra* in je bolj splošna od *P-igre* in *N-igre*. Slednji igri sta le posebna primera *K-igre*, ko je $b = 2$, $g = 2$ in $s = 0$ (*P-igra*) oz. $b = 2$, $g = 2$ in $s = 1$ (*N-igra*). Slednji primer pravzaprav ni povsem enak *N-igri*, saj je odvisnost definirana drugače.

Simulacija *K-igre* je potekala na igralni plošči $b^4 \times b^3$, kar pomeni 7 potez in drevo igre globine $d_{max} = 7$. Globina drevesa je v igri večja kot v modelu, ker ne uporabljamo sintetične hevristične funkcije, ki s šumom simulira napako pravih vrednosti, temveč prave hevristične funkcije, kakršne bi lahko uporabljali programi za igranje iger. Te funkcije pa na spodnjih dveh nivojih ne delajo napak.

Za izračun pravih vrednosti vozlišč smo igro odigrali do konca in vrednosti prenesli pod koren. Hevristične vrednosti smo pridobili tako, da smo igro odigrali do izbrane globine ($d = 1$ ali $d = 5$) in uporabili hevristično funkcijo. Napako pri izbrani globini preiskovanja in patologijo smo izračunali na enak način kot v neodvisnem modelu (izraz 3.1).

5.2 Hevristične funkcije

Napaka poteze in posledično patologija sta v veliki meri odvisni od hevristične funkcije. Ker za *P-igro* ni splošno sprejete hevristične funkcije, smo preizkusili nekaj hevrističnih funkcij, preverili njihove napake in ocenili patologijo.

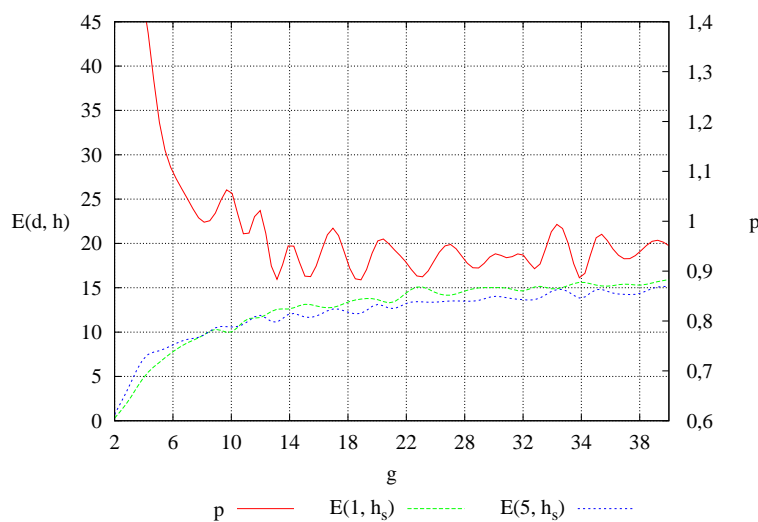
Poizkuse s hevrističnimi funkcijami smo poganjali za nabor parametrov $b = 2$, $g \in \{2, 3, \dots, 40\}$ in $s = 0$. Za vsako kombinacijo parametrov smo tvorili 3000 dreves³ in rezultate povprečili.

³Pri modelih smo za vsak nabor parametrov tvorili 10000 dreves. Večje število dreves smo si lahko privoščili zaradi nižje višine dreves in posledično krajšega časa preiskovanja.

5.2.1 Sintetična hevristična funkcija

Najprej smo za hevristično funkcijo preizkusili sintetično hevristično funkcijo, ki smo jo uporabili že v modelu. Sintetična hevristična funkcija h_s simulira napako ocene tako, da vrednostim na izbranem nivoju doda šum z Gaussovo porazdelitvijo.

Za primerjavo z ostalimi, v nadaljevanju obravnavanimi hevrističnimi funkcijami je na sliki 5.2 z rdečo krivuljo prikazana stopnja patologije, ki se nanaša na desno navpično os. Zelena in modra krivulja predstavljata napako $E(d, h_s)$ pri preiskovanju do nivoja $d = 5$ oz. $d = 1$ in se nanašata na levo navpično os. Na vodoravni osi je število različnih vrednosti g v poljih tabele.



Slika 5.2: Sintetična hevristična funkcija h_s .

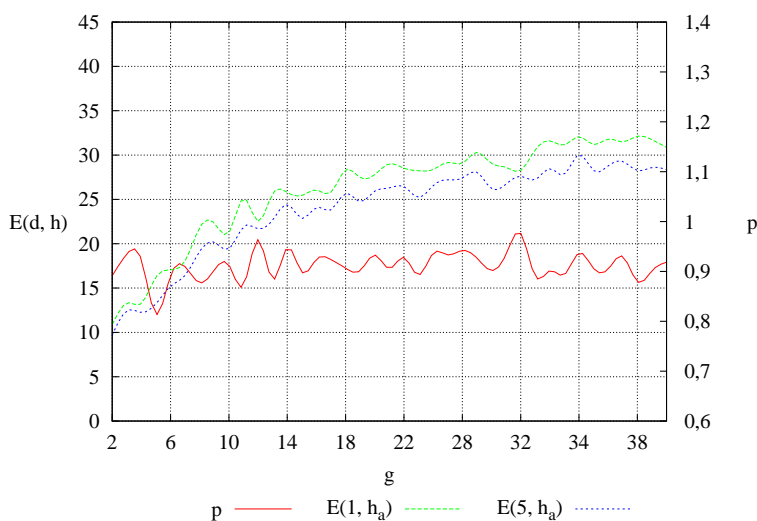
Rezultati so povsem skladni z modelom, kar smo pri uporabi h_s tudi pričakovali. Patologija je prisotna pri majhnih g , b in s . S povečanjem zrnatosti g patologija izgine.

5.2.2 Povprečje

Nau je za hevristično ocenjevanje položajev uporabljal povprečje [5], ki ga je izračunal iz vrednosti na preostalem⁴ delu plošče. Igralec, ki želi večjo vre-

⁴Preostali deli plošče so tisti deli, ki jih igralec na potezi ustvari z razdelitvijo plošče. V primeru, ko igralec ploščo navpično razdeli na dva dela, sta preostala dela leva in desna polovica plošče. Igralec nato oceni, kateri del je zanj ugodnejši, in izbere enega.

dnost, bo izbral del plošče z večjim povprečjem in obratno, njegov nasprotnik, ki želi manjšo vrednost, bo izbral del plošče z manjšim povprečjem. Hevristično funkcijo z izračunom povprečja smo označili s h_a .



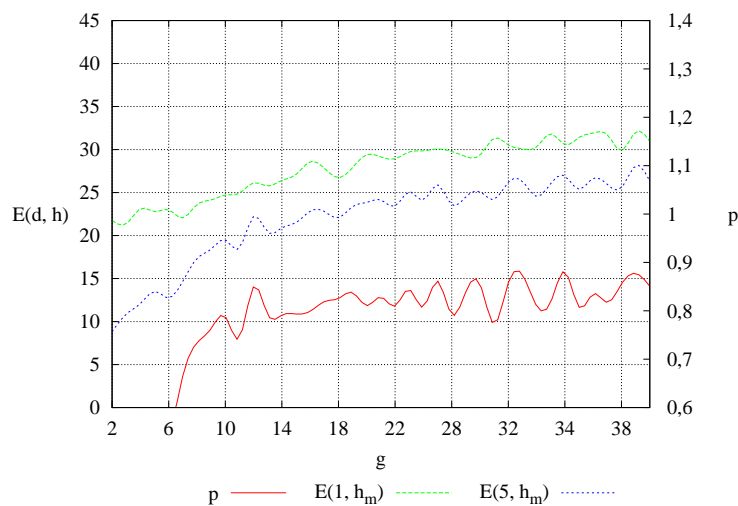
Slika 5.3: Hevristična funkcija povprečje h_a .

Na sliki so prikazane napaka na prvem in petem nivoju, ki se nanašata na levo navpično os, in stopnja patologije, ki se nanaša na desno navpično os. Patološkost je pri vejitvi $b = 2$ in odvisnosti $s = 0$ odsotna, kar se sklada z rezultati, ki jih je dobil Nau [5]. Nau je v svojem delu pokazal obstoj patologije pri globljem preiskovanju oz. pri večji vejitvi b , medtem ko je pri teh vrednostih parametrov d , b in s ni našel.

5.2.3 Mediana

Naslednja hevristična funkcija, ki smo jo preizkusili, je mediana. Izhajali smo iz podobnih predpostavk kot pri povprečju, torej igralec, ki želi večjo vrednost, teži k preostalem delu plošče z večjo mediano (in obratno velja za njegovega nasprotnika).

Izkaže se, da je hevristična funkcija z mediano h_m pri ocenjevanju na globini 1 enako dobra kot ocenjevanje s povprečjem, pri ocenjevanju na globini 5 pa je pri manjši zrnatosti bistveno slabša, saj je napaka skoraj dvakrat večja. Stopnja patologije je v tem delu bistveno nižja. Na sliki 5.4 so prikazane stopnja patološkosti p in napaki poteze $E(5, h_m)$ ter $E(1, h_m)$.

Slika 5.4: Hevristična funkcija *mediana* h_m .

5.2.4 Vsota absolutnih razlik

Pri naslednji hevristični funkciji smo izhajali iz dejstva, da je za igralca ugodno, če so sosednje vrednosti polj po delitvi čim bolj enake in čim večje za prvega igralca ter čim manjše za drugega. S tem postane vseeno, kakšno potezo bo izbral nasprotnik.

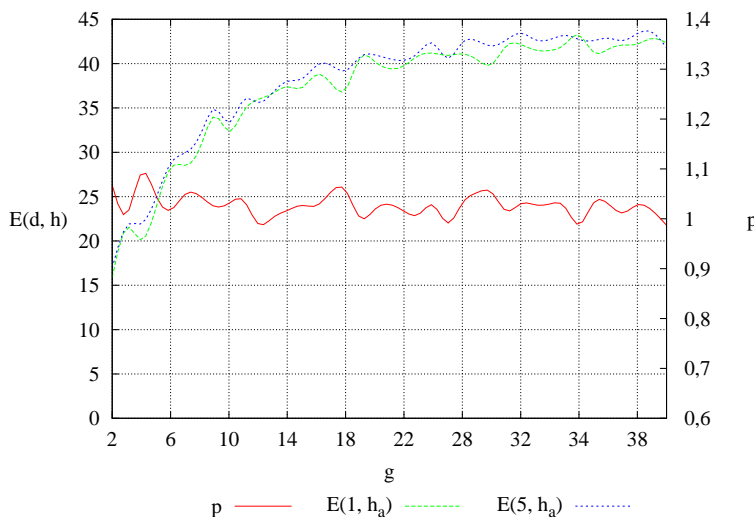
Hevristična funkcija h_d na preostalih delih plošče izračuna vsoto absolutnih razlik vseh polj do svojih sosedov. Izračun hevristične ocene h_d je formalno zapisan v izrazu 5.1, kjer so \mathcal{V} polja iz preostalega dela plošče, $N(v)$ pa množica polj, ki so sosednja polju v .

$$h_d = \sum_{v \in \mathcal{V}} \sum_{v_i \in N(v)} (|v_i - v|) \quad (5.1)$$

Graf napak $E(5, h_m)$ ter $E(1, h_m)$ in stopnje patologije p pri uporabi hevristične funkcije h_d je prikazan na sliki 5.5. Zanimivo je, da je napaka poteze na nivoju 1 vedno nekoliko manjša od napake poteze na nivoju 5. Posledično je patologija vedno prisotna.

5.2.5 Maksmin

Hevristična funkcija maksmin h_{mm} skuša dobiti vpogled v nadaljnji potek igre. Na preostalih b delih plošče izračuna minimalne vrednosti stolpcev in vrne



Slika 5.5: Hevristična funkcija vsota absolutnih razlik h_d .

maksimalno vrednost. Vrednosti h_{mm} izračunamo z izrazom 5.2, kjer so R_i polja v i -tem stolpcu na preostalem delu plošče.

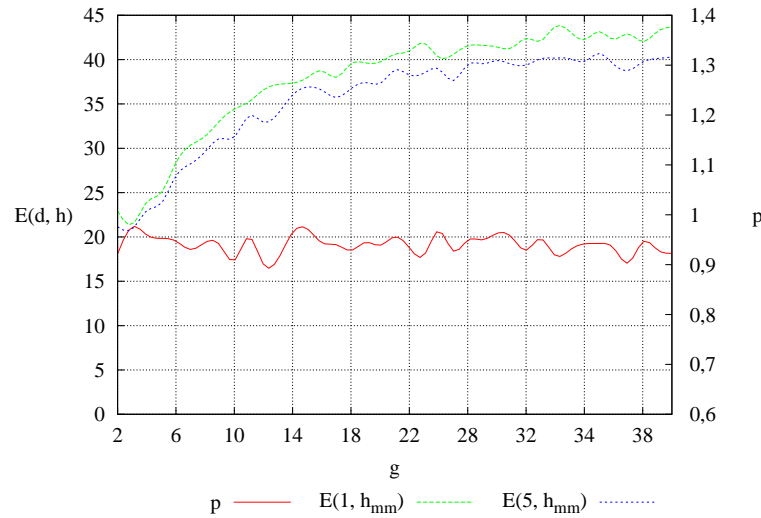
$$h_{mm} = \max_i \min_{v \in R_i}(v) \quad (5.2)$$

Grafični prikaz napak in stopnje patologije je prikazan na sliki 5.6. Funkcija h_{mm} se obnaša precej podobno kot hevristična funkcija *povprečje* (slika 5.3), le da sta obe napaki v povprečju višji za 10 odstotnih točk.

5.2.6 Povzetek

Izbira hevristične funkcije močno vpliva na velikost napake poteze $E(d, h)$ in posledično na stopnjo patologije. Odločitev, katero hevristično funkcijo bomo uporabili za nadaljnje meritve, smo sprejeli na osnovi velikosti napake poteze pri izbrani hevristični funkciji.

V praksi si namreč želimo, da bi imela hevristična funkcija čim manjšo napako poteze. Kot najprimernejša se izkaže hevristična funkcija *povprečje*, saj ima v povprečju najmanjši $E(d, h)$. Ostale preizkušene hevristične funkcije imajo bodisi v celoti bodisi deloma večjo napako, nekatere so celo vedno patološke.

Slika 5.6: Hevristična funkcija *maksmin* h_{mm} .

5.3 Meritve in rezultati

Pri nadaljnjih meritvah patološkosti v K -igri smo uporabljali hevristično funkcijo *povprečje*. Za vsak nabor parametrov b , s in g smo odigrali 5000 iger. Opazovano napako $E(d, h)$ in patološkost smo povprečili po igrah.

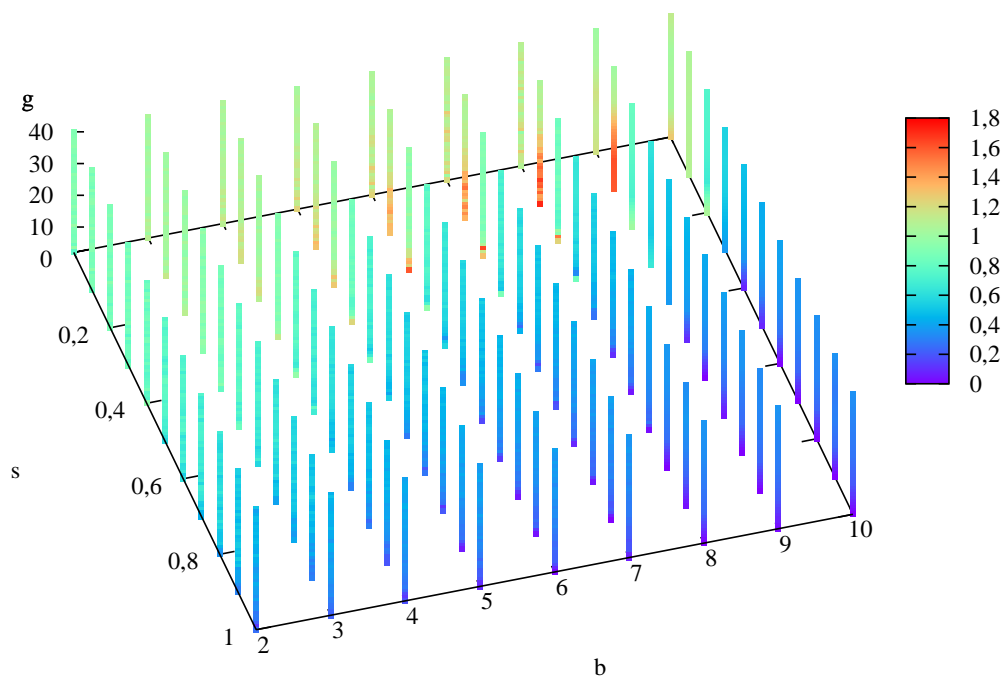
Na sliki 5.7 je z barvno intenzivnostjo prikazana stopnja patološkosti v odvisnosti od vejitve, zrnatosti in odvisnosti. Legenda intenzivnosti p se nahaja na desni strani. Numeričnih vrednosti zaradi obsega nismo vključili v priložo⁵.

Patologija se pojavi šele pri vejitvi $b > 2$, kar se sklada z rezultati Naua [5]. Stopnja patološkosti p po napovedih modela z odvisnostjo pada. Vpliv vejitve ustreza napovedim modela, saj patološkost pri povečevanju vejitve upada v nepatoloških in narašča v patoloških področjih. Vejitev tako ojača pojav patološkosti oz. nepatološkosti. Skladno z ugotovitvami Naua [5] je patologija pri vejitvi $b \geq 3$ in odvisnosti $s = 0$ vedno prisotna.

Šele s povečanjem odvisnosti se povsem izognemo patologiji. Na sliki 5.7 je s prehodom barve viden padec patološkosti pri povečevanju odvisnosti. Pri tem lahko potrdimo ugotovitve Naua [5], da popolnoma odvisna drevesa odpravijo patologijo v Pearlovi igri.

Nekoliko presenetljivo pa zrnatost na stopnjo patologije nima vpliva. Pri večji stopnji odvisnosti je mogoče z naraščajočo zrnatostjo zaznati celo manjši

⁵Graf je sestavljen iz $9 * 11 * 39 = 3762$ numeričnih vrednosti. Vsako vrednost smo izračunali iz povprečja 5000 odigranih iger.



Slika 5.7: Stopnja patološkosti p v K -igri v odvisnosti od vejitve b , zrnatosti g in odvisnosti s .

trend povečevanja stopnje patologije, vendar je stopnja patologije še vedno v nepatološkem območju. Pojav je najbolj izrazit pri večji stopnji odvisnosti $s > 0,5$ in zrnatosti $2 \leq g \leq 10$. Na sliki 5.7 je pojav razviden kot prehod iz temno modrega v svetlo modro območje. Razlog za odsotnost vpliva zrnatosti na patološkost verjetno tiči v izbiri hevrstične funkcije.

Iz povedanega lahko povzamemo, da se patološkem obnašanju K -igre pri uporabi hevrstične funkcije *povprečje* lahko izognemo le z odvisnostjo polj na plošči. Povečana vejitev ima negativen vpliv le pri majhni odvisnosti, medtem ko pri večji odvisnosti pozitivno prispeva k učinkovitosti preiskovanja z minimaksom.

Poglavje 6

Sklep

V zadnjem poglavju povzamemo opravljeno delo in komentiramo ključne izsledke naloge. Ti so vpliv vejitve na patološka in nepatološka območja, temeljita preiskava celotnega prostora vejitve, zrnatosti in odvisnosti z različnimi modeli, ter vpeljava zrnatosti v Pearlovo igro. Podamo sklepne ugotovitve in načrtamo smernice za nadaljnje delo na področju pojasnjevanja pojava patologije.

6.1 Patološki modeli preiskovanja z minimaksom

V nalogi smo preučili pomembne dejavnike, za katere so raziskovalci domnevali, da vplivajo na patologijo. Vsi so se strinjali, da vejitev vpliva na patološkost. Podobne rezultate je pokazalo tudi naše modeliranje, da naraščajoča vejitev okrepi patološkost v patoloških območjih in oslabi v nepatoloških, česar v preteklih raziskavah niso zaznali. Z naraščajočo odvisnostjo vpliv vejitve pada.

Modeli so upoštevali različno število pravih in hevrističnih vrednosti v položajih drevesa. V skladu s Scheucherjem in Kaindlom [20] ter Luštrekom idr. [11, 13] povečanje števila vrednosti zadostuje za odpravo patologije. Število vrednosti, ki so potrebne za izogib patologiji, z naraščajočo vejitvijo eksponentno narašča, po drugi strani pa povečanje odvisnosti uspešno prispeva k zmanjšanju te meje.

Največ pozornosti smo posvetili vplivu odvisnosti položajev na patološkost. Zgradili smo tri odvisne modele preiskovanja z minimaksom, analizirali delno odvisna drevesa in izmerili stopnjo odvisnosti z različnimi merami. Naloga za različne modele odvisnosti temeljito raziše vpliv patologije v celotnem pro-

storu vejitve, zrnatosti in odvisnosti. Zaključili smo, da se vsi modeli obnašajo kvalitativno podobno; patološkost s povečevanjem odvisnosti v položajih upada. Pričakovali smo postopen padec meje zrnatosti, pri kateri patologija izgine vzdolž odvisnosti, kar se tudi pokaže v modelu z normalizirano vsoto. V preostalih dve modelih pa je za padec meje patologije dovolj že majhna stopnja odvisnosti, kar je nekoliko presenetljivo.

Pri vseh modelih smo opazili težave z nestabilnostjo patološkosti v območju visoke odvisnosti $s \geq 0,6$ in zrnatosti $2 \leq g \leq b$. Pojava ne znamo razložiti.

6.2 Pearlova igra

Rezultate analize modelov smo praktično preverili na Pearlovi igri. Definirali smo *K-igro*, ki je razširitev izvirne Pearlove igre na večvrednostni odvisni model s porazdeljevanjem pomožnih vrednosti. Preizkusili smo različne hevristične funkcije in izbrali povprečje, saj ima najmanjšo povratno hevristično napako. Z izbrano hevristično funkcijo smo simulirali igre pri različnih parametrih vejitve, zrnatosti in odvisnosti.

Potrdili smo rezultate, ki jih je dobil Nau [5]. Igra je namreč patološka pri vejitvi $b \geq 3$ in majhni stopnji odvisnosti. Vpliv vejitve se sklada z rezultati v modelih, saj ima na patološkost predviden vpliv. Odvisnost pripomore k zmanjšanju patološkosti in v igri tudi odpravi patologijo.

Število vrednosti v položajih na patološkost ni imelo posebnega vpliva. Pričakovali smo, da bo tako kot v modelih bistveno prispevalo k zmanjšanju patološkosti in odpravi patologije. Patološkost je povečini nespremenjena z naraščajočim številom vrednosti, v nekaterih delih pa je zaznati celo rahlo naraščanje patološkosti, kar je v nasprotju z rezultati v modelih. Domnevamo, da je razlog za takšno obnašanje izbira hevristične funkcije *povprečje*. Ni povsem jasno ali je vpliv zrnatosti odsoten le pri uporabi izbrane hevristične funkcije ali pa se pojavi samo v modelih. Raziskave na igri osmih ploščic [17], ki se sicer ukvarjajo s patologijo enoagentnega preiskovanja, so potrdile vpliv zrnatosti na patologijo. V nadaljnjem delu bi veljalo preizkusiti še druge hevristične funkcije in igre, s katerimi bi ugotovili vpliv števila vrednosti na patologijo.

Dodatek A

Tabele

s / b	2	3	4	5	6	7	8	9	10
0	0,382	0,319	0,276	0,245	0,222	0,203	0,188	0,176	0,165
0,1	0,396	0,338	0,301	0,273	0,252	0,235	0,221	0,210	0,212
0,2	0,410	0,359	0,326	0,304	0,285	0,271	0,261	0,253	0,241
0,3	0,423	0,381	0,354	0,332	0,318	0,308	0,300	0,299	0,281
0,4	0,432	0,397	0,373	0,359	0,345	0,339	0,336	0,334	0,324
0,5	0,441	0,407	0,393	0,381	0,370	0,366	0,362	0,356	0,344
0,6	0,448	0,425	0,406	0,398	0,390	0,380	0,375	0,373	0,370
0,7	0,457	0,437	0,417	0,410	0,404	0,395	0,395	0,390	0,388
0,8	0,457	0,437	0,417	0,410	0,404	0,395	0,395	0,390	0,390
0,9	0,470	0,452	0,441	0,433	0,420	0,417	0,415	0,410	0,408
1	0,470	0,452	0,441	0,433	0,420	0,417	0,415	0,410	0,409

Tabela A.1: Verjetnost poraza $c_{b,s}$ pri modelu s porazdelitvijo pomožnih vrednosti (obravnavan je v razdelku 4.3.1).

s / b	2	3	4	5	6	7	8	9	10
0	0,382	0,319	0,276	0,245	0,222	0,203	0,188	0,176	0,165
0,1	0,405	0,337	0,295	0,265	0,241	0,222	0,206	0,192	0,181
0,2	0,415	0,354	0,315	0,287	0,264	0,244	0,227	0,213	0,201
0,3	0,422	0,371	0,337	0,311	0,289	0,270	0,254	0,239	0,226
0,4	0,433	0,385	0,359	0,336	0,318	0,300	0,284	0,270	0,257
0,5	0,435	0,401	0,380	0,362	0,347	0,332	0,319	0,306	0,294
0,6	0,444	0,415	0,400	0,386	0,376	0,366	0,356	0,346	0,337
0,7	0,453	0,428	0,419	0,411	0,404	0,395	0,389	0,383	0,376
0,8	0,455	0,443	0,436	0,431	0,427	0,424	0,421	0,419	0,415
0,9	0,460	0,453	0,450	0,449	0,447	0,444	0,445	0,442	0,440
1	0,470	0,469	0,466	0,465	0,464	0,464	0,463	0,463	0,462

Tabela A.2: Verjetnost poraza $c_{b,s}$ pri modelu z normalizirano vsoto (obravnavan je v razdelku 4.3.2).

s / b	2	3	4	5	6	7	8	9	10
0	0,382	0,319	0,276	0,245	0,222	0,203	0,188	0,176	0,165
0,1	0,392	0,339	0,296	0,265	0,252	0,233	0,218	0,216	0,205
0,2	0,412	0,349	0,326	0,295	0,282	0,273	0,278	0,286	0,275
0,3	0,432	0,369	0,346	0,325	0,332	0,323	0,328	0,336	0,345
0,4	0,452	0,389	0,376	0,365	0,372	0,373	0,378	0,386	0,385
0,5	0,472	0,439	0,416	0,395	0,402	0,403	0,408	0,416	0,415
0,6	0,512	0,499	0,446	0,445	0,452	0,453	0,458	0,466	0,465
0,7	0,532	0,519	0,526	0,515	0,492	0,493	0,498	0,506	0,505
0,8	0,532	0,519	0,526	0,515	0,492	0,493	0,498	0,506	0,505
0,9	0,552	0,539	0,546	0,535	0,522	0,523	0,528	0,506	0,505
1	0,552	0,539	0,546	0,535	0,522	0,523	0,528	0,506	0,505

Tabela A.3: Verjetnost poraza $c_{b,s}$ pri modelu z urejanjem naključnih vrednosti (obravnavan je v razdelku 4.4)

s / b	2	3	4	5	6	7	8	9	10
0	0,678	0,770	0,802	0,818	0,826	0,832	0,836	0,839	0,842
0,1	0,671	0,760	0,793	0,810	0,821	0,829	0,835	0,840	0,846
0,2	0,662	0,748	0,781	0,799	0,812	0,821	0,829	0,836	0,844
0,3	0,652	0,733	0,766	0,784	0,797	0,808	0,816	0,823	0,831
0,4	0,639	0,717	0,747	0,766	0,779	0,789	0,797	0,804	0,816
0,5	0,627	0,699	0,727	0,744	0,757	0,766	0,774	0,781	0,794
0,6	0,614	0,677	0,704	0,720	0,732	0,741	0,748	0,754	0,762
0,7	0,595	0,655	0,679	0,694	0,704	0,713	0,718	0,724	0,739
0,8	0,577	0,630	0,654	0,667	0,675	0,684	0,689	0,693	0,709
0,9	0,555	0,603	0,622	0,633	0,642	0,647	0,651	0,656	0,667
1	0,533	0,575	0,589	0,598	0,606	0,610	0,614	0,618	0,628

Tabela A.4: Faktor grozdenja f pri modelu s porazdelitvijo pomožnih vrednosti (obravnavan je v razdelku 4.3.1).

s / b	2	3	4	5	6	7	8	9	10
0	0,681	0,766	0,801	0,817	0,823	0,831	0,837	0,842	0,844
0,1	0,667	0,756	0,792	0,809	0,819	0,826	0,833	0,836	0,838
0,2	0,656	0,745	0,783	0,802	0,813	0,822	0,828	0,831	0,833
0,3	0,644	0,734	0,776	0,796	0,808	0,817	0,823	0,827	0,828
0,4	0,632	0,723	0,768	0,791	0,803	0,812	0,819	0,823	0,825
0,5	0,617	0,709	0,759	0,784	0,798	0,808	0,815	0,819	0,821
0,6	0,602	0,694	0,746	0,774	0,790	0,800	0,807	0,813	0,814
0,7	0,584	0,671	0,727	0,758	0,776	0,788	0,796	0,801	0,803
0,8	0,556	0,634	0,695	0,728	0,749	0,763	0,772	0,778	0,780
0,9	0,518	0,574	0,628	0,662	0,683	0,698	0,709	0,717	0,720
1	0,464	0,452	0,450	0,444	0,437	0,431	0,426	0,421	0,414

Tabela A.5: Faktor grozdenja f pri modelu z normalizirano vsoto (obravnavan je v razdelku 4.3.2).

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.6: Faktor grozdenja f pri modelu z urejanjem naključnih vrednosti (obravnavan je v razdelku 4.4).

s / b	2	3	4	5	6	7	8	9	10
0	0,540	0,315	0,188	0,109	0,054	0,015	-0,015	-0,038	-0,057
0,1	0,551	0,334	0,212	0,135	0,083	0,046	0,018	-0,003	-0,019
0,2	0,564	0,357	0,243	0,172	0,126	0,093	0,071	0,055	0,063
0,3	0,579	0,387	0,281	0,218	0,178	0,153	0,137	0,128	0,164
0,4	0,598	0,419	0,325	0,270	0,236	0,216	0,205	0,200	0,179
0,5	0,617	0,454	0,370	0,323	0,295	0,278	0,269	0,263	0,245
0,6	0,636	0,493	0,417	0,376	0,351	0,336	0,326	0,321	0,294
0,7	0,663	0,532	0,466	0,428	0,405	0,391	0,382	0,375	0,362
0,8	0,687	0,573	0,512	0,478	0,456	0,442	0,432	0,425	0,413
0,9	0,715	0,613	0,561	0,528	0,507	0,493	0,483	0,474	0,467
1	0,741	0,653	0,608	0,579	0,558	0,543	0,531	0,522	0,516

Tabela A.7: Korelacija r_{xy} pri modelu s porazdelitvijo pomožnih vrednosti (obravnavan je v razdelku 4.3.1).

s / b	2	3	4	5	6	7	8	9	10
0	0,542	0,318	0,186	0,110	0,056	0,015	-0,013	-0,034	-0,054
0,1	0,567	0,350	0,218	0,134	0,079	0,037	0,000	-0,026	-0,041
0,2	0,587	0,377	0,245	0,158	0,101	0,053	0,019	-0,009	-0,025
0,3	0,607	0,403	0,269	0,183	0,121	0,074	0,038	0,009	-0,007
0,4	0,624	0,426	0,294	0,207	0,146	0,098	0,060	0,030	0,010
0,5	0,644	0,453	0,325	0,238	0,174	0,124	0,085	0,052	0,031
0,6	0,663	0,488	0,363	0,277	0,215	0,164	0,123	0,085	0,064
0,7	0,686	0,534	0,415	0,332	0,269	0,217	0,174	0,138	0,112
0,8	0,718	0,600	0,494	0,415	0,352	0,299	0,254	0,216	0,189
0,9	0,761	0,697	0,619	0,552	0,495	0,445	0,400	0,361	0,332
1	0,817	0,847	0,840	0,830	0,821	0,810	0,799	0,791	0,789

Tabela A.8: Korelacija r_{xy} pri modelu z normalizirano vsoto (obravnavan je v razdelku 4.3.2).

s / b	2	3	4	5	6	7	8	9	10
0	0,532	0,310	0,184	0,104	0,050	0,010	-0,019	-0,042	-0,061
0,1	0,560	0,353	0,236	0,161	0,111	0,075	0,048	0,028	0,013
0,2	0,589	0,407	0,302	0,235	0,192	0,165	0,150	0,142	0,133
0,3	0,627	0,470	0,378	0,323	0,295	0,276	0,268	0,266	0,272
0,4	0,667	0,538	0,465	0,421	0,405	0,391	0,393	0,392	0,399
0,5	0,709	0,618	0,560	0,518	0,519	0,496	0,512	0,505	0,563
0,6	0,749	0,705	0,655	0,653	0,632	0,611	0,628	0,609	0,627
0,7	0,789	0,778	0,769	0,757	0,739	0,753	0,717	0,747	0,736
0,8	0,830	0,844	0,845	0,837	0,824	0,835	0,806	0,830	0,821
0,9	0,858	0,913	0,924	0,923	0,928	0,918	0,927	0,926	0,909
1	0,894	0,969	0,987	0,995	0,997	0,998	0,999	0,999	1,000

Tabela A.9: Korelacija r_{xy} pri modelu z urejanjem naključnih vrednosti (obravnavan je v razdelku 4.4).

Slike

1.1	Drevo igre pri preiskovanju z minimaksom.	3
3.1	Poravnava mej podintervalov na vrednost c_b	12
3.2	Stopnja patološkosti p v odvisnosti od vejitve drevesa b in števila pravih vrednosti g . Pri mejnih vrednostih g , pri katerih je patološkost $p = 1$, se patološkost poslabša za 2 % ($p = 1,02$) in izboljša za 2 % ($p = 0,98$), so projicirane na spodnjo ploskev z rdečo, modro in zeleno krivuljo.	13
4.1	Delno odvisna drevesa zgradimo s kombinacijo popolnoma odvisnega in neodvisnega drevesa.	20
4.2	Korenu dodelimo začetno pomožno vrednost. Pomožne vrednosti sinov se porazdelijo okrog starša in v listih postanejo prave.	21
4.3	Meja zrnatosti g , pri kateri je patološkost $p = 1$ v odvisnem modelu s porazdeljevanjem pomožnih vrednosti.	22
4.4	Stopnja odvisnosti v modelu s porazdeljevanjem pomožnih vrednosti, izmerjena s faktorjem grozdenja f . Prikazana je v odvisnosti od stopnje s in vejitve b	23
4.5	S korelacijo r_{xy} izmerjena stopnja odvisnosti v modelu s porazdeljevanjem pomožnih vrednosti. Prikazana je v odvisnosti od stopnje s in vejitve b	23
4.6	Meja zrnatosti g , pri kateri je patološkost $p = 1$ v odvisnem modelu s porazdeljevanjem pomožnih vrednosti.	24
4.7	Patologija je pri vejitvi $b = 6$ in stopnji odvisnosti $s = 0,3$ venomer prisotna, poleg tega pa se patološkost vzdolž s ne spreminja monotono.	25
4.8	Delno odvisno drevo zgradimo z naraščajočo ureditvijo vej neodvisnega drevesa na prvem nivoju in zamenjavo dela naključnih pravih vrednosti v listih z odvisnimi pravimi vrednostmi.	27

4.9	Grob prehod med neodvisnimi in odvisnim drevesi pri vejitvi $b = 6$ v modelu z urejanjem naključnih vrednosti pri dodajanju odvisnih listov z leve in desne.	28
4.10	Trikotniška porazdelitev $H_s(i)$	28
4.11	Prehod med neodvisnimi in odvisnim drevesi se pri modelu z urejanjem naključnih vrednosti in uporabi trikotne porazdelitve $H(s)$ pri vejitvi $b = 6$ in odvisnosti $s = 0,5$ prelomi.	29
4.12	Gladka porazdelitev $H'_s(i)$	30
4.13	Enakomeren prehod med neodvisnimi in odvisnimi drevesi v modelu z urejanjem naključnih vrednosti pri vejitvi $b = 6$ in uporabi gladke porazdelitvene funkcije $H'_s(i)$	30
4.14	Mejne vrednosti g , pri katerih je $p = 1$ v modelu z urejanjem naključnih vrednosti.	31
4.15	Primerjava porazdelitve $g = 1.000$ pravih vrednosti v listih popolnoma odvisnih modelov.	33
5.1	Drevo igre pri igralni plošči z $b = 2$ in $d = 4$	35
5.2	Sintetična hevristična funkcija h_s	37
5.3	Hevristična funkcija <i>povprečje</i> h_a	38
5.4	Hevristična funkcija <i>mediana</i> h_m	39
5.5	Hevristična funkcija <i>vsota absolutnih razlik</i> h_d	40
5.6	Hevristična funkcija <i>maksmin</i> h_{mm}	41
5.7	Stopnja patološkosti p v K -igri v odvisnosti od vejitve b , zrna-tosti g in odvisnosti s	42

Tabele

3.1	Mejne vrednosti g , pri katerih je dosežena patološkost $p = 1$	14
A.1	Verjetnost poraza $c_{b,s}$ pri modelu s porazdelitvijo pomožnih vrednosti (obravnavan je v razdelku 4.3.1).	45
A.2	Verjetnost poraza $c_{b,s}$ pri modelu z normalizirano vsoto (obravnavan je v razdelku 4.3.2).	46
A.3	Verjetnost poraza $c_{b,s}$ pri modelu z urejanjem naključnih vrednosti (obravnavan je v razdelku 4.4)	46
A.4	Faktor grozdenja f pri modelu s porazdelitvijo pomožnih vrednosti (obravnavan je v razdelku 4.3.1).	47
A.5	Faktor grozdenja f pri modelu z normalizirano vsoto (obravnavan je v razdelku 4.3.2).	47
A.6	Faktor grozdenja f pri modelu z urejanjem naključnih vrednosti (obravnavan je v razdelku 4.4).	48
A.7	Korelacija r_{xy} pri modelu s porazdelitvijo pomožnih vrednosti (obravnavan je v razdelku 4.3.1).	48
A.8	Korelacija r_{xy} pri modelu z normalizirano vsoto (obravnavan je v razdelku 4.3.2).	49
A.9	Korelacija r_{xy} pri modelu z urejanjem naključnih vrednosti (obravnavan je v razdelku 4.4).	49

Algoritmi

1	<i>minimax</i> (<i>node</i> , <i>depth</i>)	2
2	<i>build_tree</i> (<i>b</i> , <i>d</i>)	12
3	<i>find_c</i> (<i>b</i> , <i>s</i>)	19

Literatura

- [1] Donald F. Beal. An analysis of minimax. V zborniku *Advances in Computer Chess 2*, str. 103–109. Edinburgh University Press, 1980.
- [2] Donald F. Beal. Benefits of minimax search. V zborniku *Advances in Computer Chess 3*, str. 1–15. Pergamon Press, 1982.
- [3] Ivan Bratko in Matjaž Gams. Error analysis of the minimax principle. V zborniku *Advances in Computer Chess 3*, str. 1–15. Pergamon Press, 1982.
- [4] Dana S. Nau. *Quality of decision versus depth of search on game trees*. Doktorska disertacija, Duke University, 1979.
- [5] Dana S. Nau. An investigation of the causes of pathology in games. *Artificial Intelligence*, 19(3):257–278, 1982.
- [6] Dana S. Nau. Pathology on Game Trees Revisited, and an Alternative to Minimaxing. *Artificial Intelligence*, 21(1,2):257–278, 1983.
- [7] John von Neumann. Zur theorie der gesellschaftsspiele. *Mathematische Annalen*, (100):295–320, 1928.
- [8] 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.
- [9] 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.

- [10] Boštjan Kaluža in Aleš Tavčar. *Seminarska naloga pri predmetu UISP: Patologija pri preiskovanju z minimaksom*. Seminarska naloga, Fakulteta za računalništvo in informatiko UL, Institut "Jožef Stefan", 2007.
- [11] Mitja Luštrek. *Patologija v hevrističnih preiskovalnih algoritmih*. Doktorska disertacija, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, 2007.
- [12] Mitja Luštrek, Matjaž Gams in Ivan Bratko. Is real-valued minimax pathological? *Artificial Intelligence*, 170(6):620–642, 2006.
- [13] Mitja Luštrek, Matjaž Gams in Ivan Bratko. Minimax Pathology: From Two to Real-Number Values. *Poslano v Journal of Artificial Intelligence Research (JAIR)*, 2008.
- [14] Matt Donadio. *How to Generate White Gaussian Noise*. <http://www.dspguru.com/howto/tech/wgn.htm>, 15. 8. 2008.
- [15] Nicholas Metropolis in Stanislaw M. Ulam. The Monte Carlo Method. *Journal of the American Statistical Association*, 44(247):335–341, 1949.
- [16] Judea Pearl. On the nature of pathology in game searching. *Artificial Intelligence*, 20(4):427–453, 1983.
- [17] Rok Piltaver. *Patologija preiskovanja v igri osmih ploščic*. Diplomaska naloga, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, 2008.
- [18] Aleksander Sadikov. *Širjenje napak hevrističnih ocen v preiskovanju grafov iger*. Doktorska disertacija, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, 2005.
- [19] Aleksander Sadikov, Ivan Bratko in Igor Kononenko. Bias and pathology in minimax search. *Theoretical Computer Science*, 349(2):268–281, 2005.
- [20] Anton Scheucher in Hermann Kaindl. Benefits of using multivalued functions for minimaxing. *Artificial Intelligence*, 99(2):187–208, 1998.
- [21] Anton Scheucher in Hermann Kaindl. The reason for the benefits of minimax search. V zborniku *International Joint Conference on Artificial Intelligence*, str. 322–327, 1989.

- [22] Wikipedia. *Alpha-beta pruning*. http://en.wikipedia.org/wiki/Alpha-beta_pruning, 15. 8. 2008.
- [23] Wikipedia. *Correlation*. <http://en.wikipedia.org/wiki/Correlation>, 15. 8. 2008.