



Универзитет „Св. Кирил и Методиј“ во Скопје  
**ФАКУЛТЕТ ЗА ИНФОРМАТИЧКИ НАУКИ И  
КОМПЈУТЕРСКО ИНЖЕНЕРСТВО**

# **Проектна задача по предметот „Вештачка Интелигенција“**

на тема

**„Имплементација на повеќе-агентна средина и  
алгоритми за спротиставено пребарување (MiniMax,  
Alpha-Beta поткастрување & ExpectiMax) во Game of  
the Amazons“**

Изработиле:  
Ангела Ангелеска, 221083  
Ева Смилеска, 221053

Ментор:  
Проф. д-р Соња Гиевска

ФИНКИ, февруари 2025

# Содржина

Содржина .....	1
Апстракт .....	2
Вовед .....	3
MiniMax алгоритам.....	5
Alpha-Beta поткастрување .....	6
ExpectiMax алгоритам .....	7
Споредба на алгоритмите.....	8
Опис на проблем .....	9
Резултати.....	11
Споредба на алгоритми .....	11
Заклучок .....	17
Референци .....	18

## Апстракт

Овој проект претставува имплементација на алгоритми за спротивставено пребарување како што се MiniMax, Alpha-Beta поткастрување и ExpectiMax, во контекст на стратешката игра Game of the Amazons. Играта претставува сложен домен за вештачка интелигенција, каде што агентите мораат да донесуваат оптимални одлуки во динамично опкружување.

Во рамките на проектот развиена е интерактивна средина за играње, која овозможува натпревар помеѓу различни интелигентни агенти. За донесување на најдобар можен потег се користат MiniMax и Alpha-Beta поткастрување како традиционални алгоритми за спротивставено пребарување, како и ExpectiMax кој воведува елемент на неизвесност преку пресметка на веројатносни исходи.

Истражувањето ги анализира перформансите на овие алгоритми преку споредба на ефикасноста, брзината на одлучување и квалитетот на избраните потези. Според добиените резултати, секој алгоритам има свои предности во различни сценарија. MiniMax генерира стабилни стратегии, но може да биде прескап во однос на пресметките, додека Alpha-Beta поткаструвањето значително го намалува бројот на разгледани состојби без да влијае на точноста на решението. ExpectiMax, од друга страна, овозможува модел на игра кој ги зема во предвид случајните фактори, што го прави пофлексибилен во сценарија со неизвесност. Спроведената анализа вклучува графички прикажани податоци, кои визуелно ја илустрираат разликата помеѓу алгоритмите во однос на бројот на победи, порази и просечно време на одлучување.

## Вовед

Спротиставеното пребарување и повеќе-агентните средини претставуваат клучни концепти во областа на вештачката интелигенција, особено значајни за моделирање и решавање на комплексни проблеми каде повеќе ентитети интерактивно донесуваат одлуки во конкурентни сценарија. Овие пристапи наоѓаат широка примена во различни домени, од стратешки игри до реални системи за донесување одлуки во динамични околности.

Во срцето на спротиставеното пребарување лежи идејата за оптимизација на стратегиите на еден агент во присуство на противници кои исто така се стремат кон оптимални резултати. Ова создава комплексна динамика каде секој чекор мора да се евалуира не само според неговата непосредна корист, туку и според потенцијалните контрапотези на противникот. Алгоритми како MiniMax, Alpha-Beta поткастрување и ExrectiMax се развиени за ефикасна навигација низ овој сложен простор на одлуки, секој со свои предности во одредени сценарија.

Повеќе-агентните средини, од друга страна, го прошируваат овој концепт на поширок спектар на интеракции, вклучувајќи не само антагонистички, туку и кооперативни и неутрални односи меѓу агентите. Овие системи се карактеризираат со висока комплексност поради многубројните интеракции и потенцијално различните цели на секој агент. Предизвиците во овие средини вклучуваат координација на активностите, балансирање на индивидуалните и колективните цели, како и адаптација на стратегиите во реално време како одговор на акциите на другите агенти.

Примената на овие концепти се протега и надвор од доменот на игрите. Во роботиката, на пример, спротиставеното пребарување се користи за планирање на движењата на работи во динамични средини каде има други подвижни објекти или противници. Во економијата, овие техники се применуваат за моделирање на пазарни стратегии и предвидување на економски трендови во присуство на конкурентни актери. Областа на сајбер безбедноста исто така значително профитира од овие концепти, користејќи ги за симулација на потенцијални напади и развој на робусни одбранбени стратегии.

Современите истражувања во оваа област се фокусираат на надминување на ограничувањата на традиционалните алгоритми, особено во поглед на скалабилноста и ефикасноста при работа со големи простори на состојби. Интеграцијата на техники од машинското учење, особено длабокото учење, со класичните алгоритми за спротиставено пребарување отвора нови можности за справување со уште покомплексни проблеми. Дополнително, развојот на методи за справување со

несигурност и нецелосни информации станува сè поважен како што овие системи се применуваат во реални сценарија каде перфектни информации ретко се достапни.

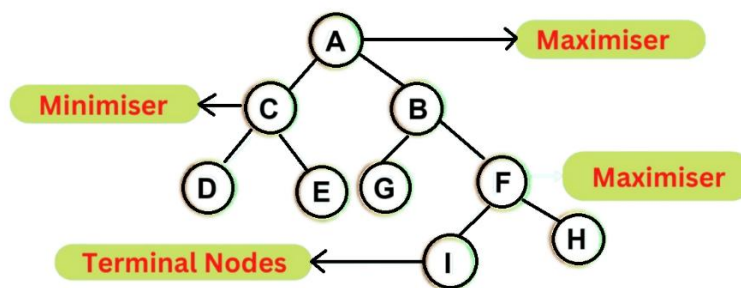
Етичките импликации на овие технологии, особено кога се применуваат во критични системи за донесување одлуки, исто така стануваат предмет на зголемено внимание. Истражувачите се соочуваат со предизвикот да развијат системи кои не само што се ефикасни, туку и транспарентни, праведни и усогласени со човековите вредности.

Како што технологијата продолжува да напредува, спротиставеното пребарување и повеќе-агентните системи ќе играат сè поважна улога во обликувањето на идните интелигентни системи. Нивната способност да моделираат комплексни интеракции и да оптимизираат стратегии во динамични средини ги прави незаменливи алатки во арсеналот на вештачката интелигенција, со потенцијал да револуционизираат широк спектар на индустрии и да придонесат за решавање на некои од најпредизвикувачките проблеми на нашето време.

## MiniMax алгоритам

MiniMax алгоритмот претставува математички пристап за оптимизација на одлуките во игри со двајца играчи каде што добивките на еден се изедначуваат со загубите на другиот (zero-sum игри). Овој алгоритам се базира на теоријата на игри и минимизирање на максималниот ризик, што е формално дефинирано со MiniMax теоремата – максималната минимална добивка на еден играч е еднаква на минималната максимална загуба на противникот. Всушност во рамките на дрвото, едниот играч е „max“ кој има за цел да го максимизира својот резултат, додека пак другиот е „min“ кој се стреми да го минимизира резултатот на противникот.

Механизмот на работа на MiniMax алгоритмот се состои од три главни чекори. Прво, се врши конструкција на дрво на игри каде се генерираат сите можни состојби на играта до одредена длабочина. Во ова дрво, секој јазол претставува одредена состојба на играта, додека гранките ги претставуваат можните потези. Второ, се применува евалуациска функција која на секој терминален јазол (лист) му доделува нумеричка вредност. Оваа вредност ја одразува предноста за максимизирачкиот играч. На пример, во игра како шах, оваа евалуација може да вклучува оцена на материјална предност или позициона контрола на таблата. Третиот чекор вклучува рекурзивна оцена на јазлите. Кај максимизирачките јазли, алгоритмот ја избира највисоката вредност од нивните деца, додека кај минимизирачките јазли се избира најниската вредност од нивните деца. Овој процес на рекурзивна оцена продолжува нагоре низ дрвото, овозможувајќи на алгоритмот да донесе оптимална одлука за следниот потег.



Слика 1. MiniMax алгоритам

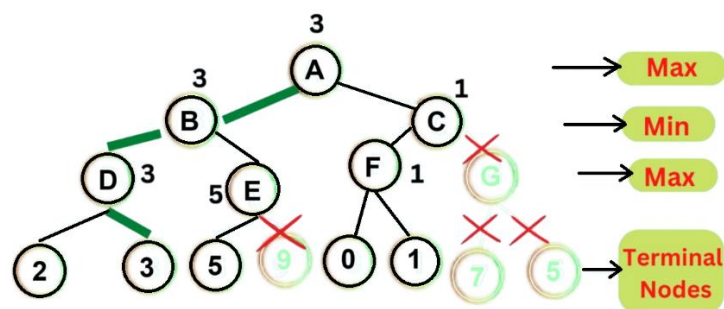
## Alpha-Beta поткастрување

Alpha-Beta поткаструвањето е софистицирана техника за оптимизација на MiniMax алгоритмот, која значително го намалува бројот на јазли што треба да се евалуираат во дрвото на игра. Оваа техника е особено корисна за игри со голем фактор на разгранување, каде што стандардниот MiniMax би бил непрактичен.

Принципот на работа на Alpha-Beta кастрењето се заснова на елиминирање на гранки од дрвото на игра кои не можат да влијаат на конечната одлука. Ова се постигнува преку одржување на два параметри: алфа ( $\alpha$ ) и бета ( $\beta$ ). Алфа ја претставува најдобрата (највисоката) вредност што максимизирачкиот играч може да ја гарантира на тековното ниво или погоре, додека бета ја претставува најдобрата (најниската) вредност што минимизирачкиот играч може да ја гарантира на тековното ниво или погоре.

Процесот на кастрење се одвива на следниов начин: за максимизирачки јазол, ако тековната вредност е поголема или еднаква на бета, пребарувањето се прекинува ( $\beta$ -отсекување). Од друга страна, за минимизирачки јазол, ако тековната вредност е помала или еднаква на алфа, пребарувањето се прекинува ( $\alpha$ -отсекување).

Важно е да се напомене дека во некои случаи, алфа-бета кастрењето може да не отстрани ниту една гранка. Ова се случува кога резултатите на дното на дрвото се подредени од најлошиот до најдобриот резултат за тековниот играч. Во таков случај, алгоритмот е принуден да ги истражи сите гранки. Спротивно на тоа, алгоритмот ќе отстрани најмногу гранки кога дното на дрвото е подредено од најдобриот до најлошиот резултат за тековниот играч. Во најдобар случај, Alpha-Beta кастрењето може да отстрани една половина од гранките, што го прави два пати побрз од стандардниот MiniMax алгоритам.



Слика 2. MiniMax алгоритам со алфа-бета кастрење

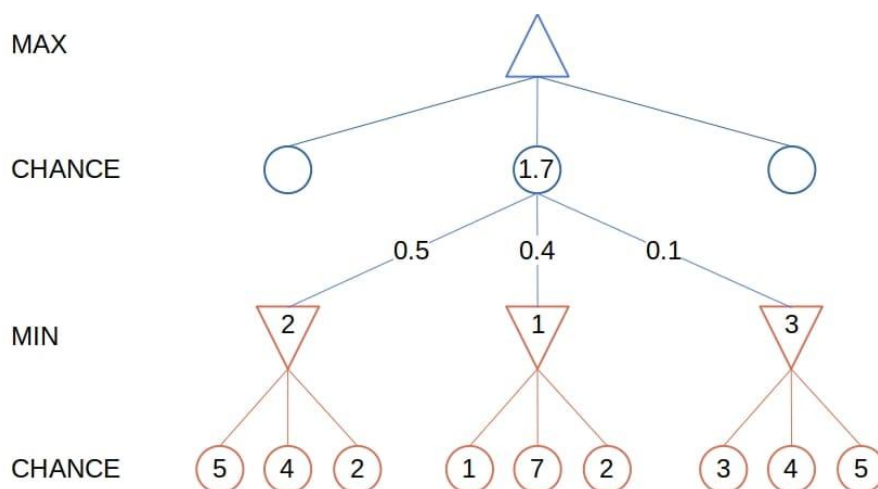
## ExpectiMax алгоритам

ExpectiMax алгоритмот претставува софистицирана модификација на MiniMax алгоритмот, специјално дизајнирана за справување со игри и ситуации кои вклучуваат елементи на случајност или несигурност. Овој алгоритам е особено корисен за вештачка интелигенција која се соочува со сценарија каде што исходот зависи не само од вештината на играчите, туку и од непредвидливи фактори.

Механизмот на работа на ExpectiMax е сличен на MiniMax, но со клучна разлика во структурата на дрвото на игри. Покрај стандардните „Max“ и „Min“ јазли, ExpectiMax вклучува и трет тип на јазли наречени „Chance“ јазли. Овие „Chance“ јазли ги претставуваат елементите на случајност во играта, како што е фрлањето на коцки или мешањето на карти.

При конструкцијата на дрвото на игри, алгоритмот ги генерира сите можни состојби до одредена длабочина, вклучувајќи ги и состојбите кои произлегуваат од случајни настани. Евалуацијата на јазлите се врши рекурзивно од дното нагоре. „Max“ јазлите ја избираат највисоката вредност од своите деца, „Min“ јазлите ја избираат најниската вредност, додека „Chance“ јазлите пресметуваат пондериран просек на вредностите на своите деца, земајќи ги во предвид веројатностите за секој можен исход.

Во областа на игрите, овој алгоритам е особено ефикасен за игри како табла, покер и 2048, каде што постигнува импресивни резултати. Надвор од доменот на игрите, алгоритмот се користи во вештачката интелигенција за донесување одлуки во несигурни околности и во роботиката за планирање движења во динамични средини.



Слика 3. ExpectiMax алгоритам



## Споредба на алгоритмите

MiniMax без Alpha-Beta поткастрување е основен алгоритам за донесување одлуки во игри со zero sum. Тој систематски ги истражува сите можни потези до одредена длабочина, претпоставувајќи оптимална игра од двата играчи. Меѓутоа, неговата временска сложеност е експоненцијална во однос на длабочината на пребарување, што го прави непрактичен за игри со голем фактор на разгранување. Овој алгоритам е најсоодветен за едноставни игри како што е ОХО, каде што просторот на состојби е релативно мал.

MiniMax со Alpha-Beta поткастрување претставува значително подобрување на основниот MiniMax алгоритам. Тој користи две вредности, алфа и бета, за да елиминира гранки од дрвото на пребарување кои не можат да влијаат на конечната одлука. Иако во најлош случај сè уште има експоненцијална сложеност, во пракса овој алгоритам може да ја намали длабочината на пребарување за фактор 2, што го прави погоден за посложени игри како шах. Ефикасноста на Alpha-Beta поткаструвањето во голема мера зависи од редоследот на евалуација на потезите.

ExpectiMax е варијанта на MiniMax алгоритмот дизајнирана за игри со елементи на случајност. Наместо да претпоставува најдобар потег за противникот, ExpectiMax пресметува очекувана вредност базирана на веројатностите на различните исходи. Ова го прави особено корисен за игри како табла или покер, каде што случајноста игра значајна улога. Временската сложеност на ExpectiMax е поголема од онаа на стандардниот MiniMax, бидејќи мора да ги земе во предвид сите можни случајни исходи. Како резултат на тоа, овој алгоритам може да биде пресметковно интензивен за игри со голем број на случајни настани.

Критериум	MiniMax	MiniMax + Alpha-Beta поткастрување	ExpectiMax
Сложеност	$O(b^d)$ , каде $b$ е просечен број на гранки по јазол, а $d$ е длабочина на пребарување	$O(b^{d/2})$ во најдобар случај, $O(b^d)$ во најлош, каде $b$ е просечен број на гранки по јазол, а $d$ е длабочина на пребарување	$O(b^d \cdot n)$ , каде $b$ е просечен број на гранки по јазол, $d$ е длабочина на пребарување, а $n$ е број на случајни исходи

Оптималност	Да (за детерминистички игри)	Да (за детерминистички игри)	Да (за игри со елемент на случајност)
Употреба	Мали игри (пр.: ОХО)	Сложени игри (пр.: шах)	Игри со случајност (пр.: друштвени игри со коцки)

Табела 1. Споредба на алгоритмите MiniMax без Alpha-Beta кастрење, MiniMax со Alpha-Beta кастрење и ExpectiMax

## Опис на проблем

Game of the Amazons (Играта на Амазонките) е апстрактна стратешка игра за двајца играчи, создадена во 1988 година од аргентинскиот математичар Walter Zamkuskas . Играта е позната по својата едноставност во правилата, но истовремено нуди огромна длабочина на стратегија и тактичко размислување. Оваа игра е одличен пример за сложеност што произлегува од едноставни правила и затоа е популарна за проучување и имплементирање на алгоритми за вештачка интелигенција.

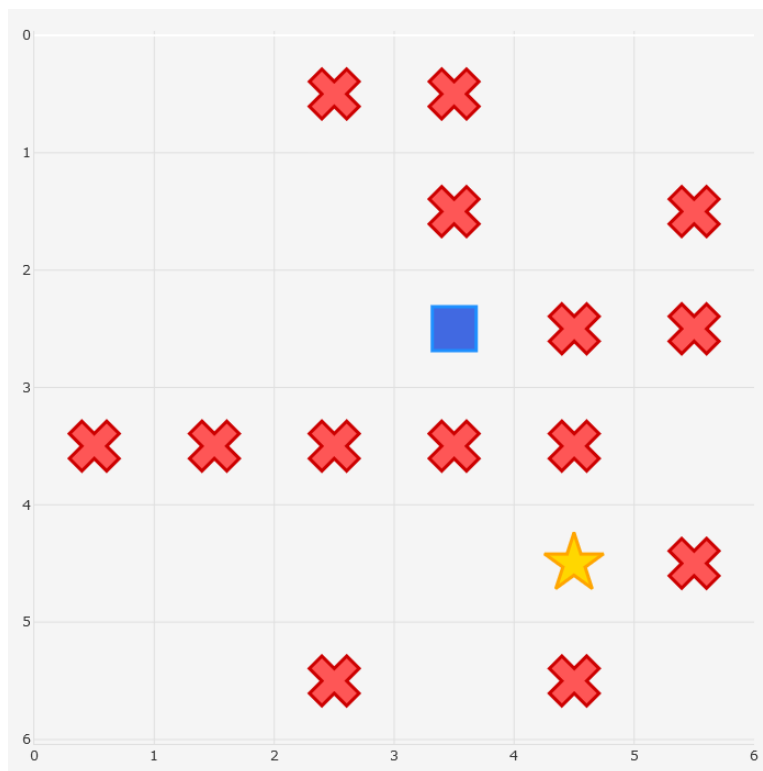
Во Game of the Amazons играчите се натпреваруваат на табла со големина 10x10, а секој од нив започнува со по четири фигури наречени „амазонки“. Целта на играта е да се контролира што поголем дел од таблата, така што противникот ќе остане без можност да направи легален потег. Играта започнува со празна табла (освен почетните позиции на амазонките). Секој играч наизменично изведува потег што се состои од два чекори:

1. Поместување на една од своите амазонки.
2. Поставување „стрела“ (што го блокира тој квадрат за понатамошна употреба).

Во оваа модифицирана верзија на Game of the Amazons, играчите ќе се натпреваруваат на табла со големина 6x6, наместо стандардната 10x10 табла. Секој играч ќе започнува со само една фигура наречена „амазонка“, наместо со четири како во оригиналната верзија.

Амазонките се движат како кралицата во шах – дијагонално, хоризонтално или вертикално, преку било кој број полиња, сè додека патот не е блокиран од друга амазонка или стрела. Откако амазонката ќе се помести, играчот мора да постави стрела на било кое поле до кое може да стигне таа фигура. Стрелата трајно го блокира тоа поле, што значи дека ниту една фигура не може да се движи или да стрела таму во текот на понатамошната игра. Играта завршува кога еден играч не може да направи легален потег.

Просторот на состојби во оваа игра ги опфаќа сите можни конфигурации на таблата, вклучително и позициите на амазонките, блокираните полиња (стрели) и слободните полиња. Оваа комплексност произлегува од фактот дека секој потег се состои од две фази – поместување на амазонката и поставување стрела – што експоненцијално го зголемува бројот на можни комбинации.



Слика 4. Пример за можна состојба во Играта на Амазонките

За секоја игра со дискретен простор на состојби, како што е Game of the Amazons, можеме да конструираме дрво на игра во кое секој јазол претставува можна состојба на таблата. Коренот го претставува почетокот на играта, додека внатрешните јазли ја прикажуваат состојбата што произлегува од потезите на играчите. Јазлите на парна длабочина ги претставуваат потезите на еден играч, додека оние на непарна длабочина ги претставуваат потезите на противникот. Терминалните јазли се состојби каде играта завршува – или еден играч останал без легални потези или таблата е целосно блокирана. Поради сложеноста и големиот број можни конфигурации, дрвото на игра за Game of the Amazons е огромно и бара ефикасни алгоритми за анализа и пребарување.

## Резултати

Во овој дел се прикажани резултатите добиени со симулации на играта Game of the Amazons со различните стратегии имплементирани во проектот.

## Споредба на алгоритми

Game of the Amazons е комплексна игра со голем фактор на гранење, што ја прави предизвикувачка за имплементација на AI алгоритми. Во оваа анализа, ги споредуваме перформансите на три клучни алгоритми: MiniMax, Alpha-Beta и ExpectiMax.

### *Опис на алгоритмите*

1. MiniMax: Покажува солидни резултати при основна имплементација, но има ограничувања поради големиот фактор на гранење во играта.
2. Alpha-Beta: Оваа оптимизација на претходиот алгоритам значително го подобрува перформансот преку ефикасно отстранување на непотребните гранки од дрвото на пребарување.
3. ExpectiMax: Внесува елемент на несигурност во одлучувањето, што е корисно за справување со неочекувани потези на противникот.

### *Методологија*

Со цел да се обезбеди разновидност во поединечните игри, во самата имплементација на играта воведен е случаен фактор во процесот на одлучување на алгоритмите. На овој начин ќе се избегне постојано повторување на истите сценарија при различните игри. Ова ја прави играта подинамична и пореалистична, бидејќи различни игри немаат идентичен тек на настани, дури и кога почетната состојба е иста.

Случајноста се применува на неколку клучни места во логиката на AI играчот:

1. Рандомизација при невалиден потег

Ако AI играчот предложи невалиден потег (на пример, обид за поместување на амазонката на недозволена позиција), алгоритмот автоматски избира случаен валиден потег наместо да заглави или да

направи грешка. Ова осигурува дека играта ќе продолжи без прекини, овозможувајќи му на AI играчот да реагира флексибилно на ограничувањата на таблата.

## 2. Случаен избор при еднакви проценки

Во ExpectiMax алгоритмот кога повеќе потези имаат идентична проценка, системот избира еден од нив на случаен начин. Ова спречува шаблонско однесување на AI играчот, намалувајќи ја предвидливоста и правејќи ја играта понеизвесна.

## 3. Мали случајни варијации во евалуацијата

ExpectiMax алгоритмот додава минимални случајни флуктуации во проценката на позициите. Ова ја прави стратегијата подинамична, бидејќи две идентични сценарија можат да резултираат со различни избори, што го спречува AI да игра строго според статичен шаблон. Овој пристап придонесува за пософистициран стил на игра, бидејќи се внесува природна непредвидливост во одлуките.

## 4. Случаен избор при еднакво добри стратегии

Кога Minimax или Alpha-Beta алгоритмот детектира дека два или повеќе потези имаат еднаква вредност според проценката, AI играчот избира еден од нив по случаен принцип. Ова дополнително го намалува предвидливото однесување на алгоритмот и го прави пофлексибилен во различни игровни ситуации.

Воведувањето на случајност во AI алгоритмите обезбедува поголема разновидност во играњето и спречува репетитивни партии со предвидливи исходи. Ова значително ја подобрува реалистичноста на симулираните натпревари и овозможува подинамични стратегии, правејќи ја играта попредиизвикувачка за играчите.

## *Влијание на редоследот на играње*

Во анализата на перформансите на различните алгоритми за играта Game of the Amazons, важно е да се разгледа влијанието на редоследот на играње врз резултатите. Овој аспект е особено значаен кога се споредуваат Alpha-Beta и MiniMax алгоритмите, а истиот има помало влијание кога овие алгоритми се натпреваруваат против ExpectiMax.

## 1. Alpha-Beta против MiniMax

Редоследот на играње има значително влијание врз перформансите на Alpha-Beta и MiniMax алгоритмите кога се натпреваруваат еден против друг. Оваа динамика произлегува од суштинските разлики меѓу двата алгоритма и природата на самата игра.

Alpha-Beta алгоритмот има предност во ефикасноста на пребарување благодарение на неговата способност за поткастрување. Ова му овозможува да истражи подлабоко во дрвото на игра за исто време во споредба со MiniMax. Кога Alpha-Beta игра прв, оваа предност се зголемува, бидејќи може подобро да ги подреди и евалуира потенцијалните потези.

Од друга страна, MiniMax, иако помалку ефикасен во пребарувањето, може да извлече корист од играњето прв. Ова му дава можност да воспостави рана стратегиска предност, што делумно ја компензира неговата помала ефикасност во споредба со Alpha-Beta.

Генерално, првиот играч во Game of the Amazons има мала вродена предност. Оваа предност се засилува кога се комбинира со ефикасноста на Alpha-Beta или со можноста на MiniMax да воспостави рана позициска предност.

## 2. Alpha-Beta/MiniMax против ExpectiMax

Кога Alpha-Beta или MiniMax се натпреваруваат против ExpectiMax во Game of the Amazons, влијанието на редоследот на играње е значително помало. Ова произлегува од фундаменталните разлики во дизајнот и функционирањето на овие алгоритми.

ExpectiMax е првенствено дизајниран за игри со стохастички елементи, што го прави помалку ефикасен во детерминистички игри како Game of the Amazons. За разлика од Alpha-Beta, ExpectiMax не користи техники на поткастрување, што резултира со помалку ефикасно истражување на дрвото на игра. Ова му дава значителна предност на Alpha-Beta, особено во подлабоко пребарување.

Експерименталните резултати потврдуваат дека ExpectiMax има многу мали шанси за победа против Alpha-Beta или MiniMax, без оглед на редоследот на играње. Ова укажува на тоа дека во детерминистички игри како Game of the Amazons, алгоритмите специјализирани за вакви сценарија (како Alpha-Beta и MiniMax) имаат значителна предност над пофлексибилните, но помалку специјализирани алгоритми како ExpectiMax.

Перформансите на алгоритмите значително се разликуваат во зависност од нивните карактеристики. Alpha-Beta најчесто доминира, благодарение на ефикасното поткастрување, но MiniMax може да биде конкурентен ако игра прв. ExpectiMax, пак, се покажува како најслаб алгоритам за оваа игра, бидејќи не е оптимизиран за детерминистички сценарија. Значајно е што редоследот на играње е критичен фактор за успехот на Alpha-Beta и MiniMax, но има многу малку влијание кога игра ExpectiMax.

### *Резултати од AI против AI натпревари*

Анализата се базира на 30 тест игри за секоја комбинација на алгоритми. Резултатите и анализата за секоја комбинација се следниве:

i. Alpha-Beta против MiniMax

	Alpha-Beta	MiniMax
Број на победи	26	4
Стапка на успех на Alpha-Beta: 86.67%		

Табела 2. Резултатот од натпреварите меѓу Alpha-Beta & MiniMax

Alpha-Beta значително ги надминува перформансите на MiniMax благодарение на Alpha-Beta поткаструвањето. Оваа техника му овозможува на алгоритмот да елиминира непотребни гранки во дрвото на пребарување, што резултира со подлабоко разгледување на потезите во ист временски период. Како резултат, Alpha-Beta носи пооптимизирани одлуки во споредба со MiniMax.

ii. MiniMax против Alpha-Beta (Разлика во редоследот на играчите)

	MiniMax	Alpha-Beta
Број на победи	26	4
Стапка на успех на MiniMax: 86.67%		

Табела 3. Резултатот од натпреварите меѓу MiniMax & Alpha-Beta

Во овој случај, MiniMax игра прв, што е клучен фактор за разликата во резултатите. Првиот играч има мала предност бидејќи може прв да постави тактичка основа за играта. Во вакви околности, MiniMax успева подобро да ги анализира можните потези и да ја искористи предноста на иницијативата.

iii. Alpha-Beta против ExpectiMax

	Alpha-Beta	ExpectiMax
Број на победи	27	3
Стапка на успех на Alpha-Beta: 90%		

Табела 4. Резултатот од натпреварите меѓу Alpha-Beta & ExpectiMax

Alpha-Beta остварува импресивна доминација над ExpectiMax. ExpectiMax не користи поткастрување и базира дел од своите одлуки врз стохастичка проценка, наместо на детерминистичка логика. Ова води кон послаби резултати во строго контролирани средини. Alpha-Beta е далеку поефикасен, носејќи пооптимални одлуки врз база на конкретни проценки наместо на веројатности.

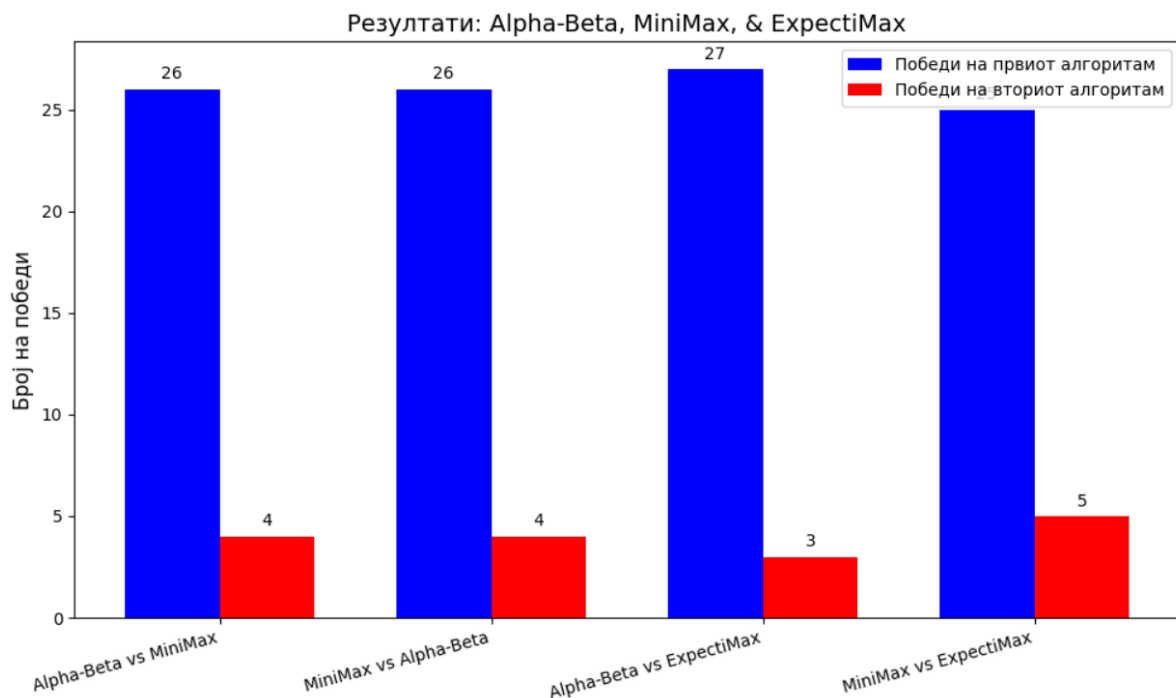
iv. MiniMax против ExpectiMax

	MiniMax	ExpectiMax
Број на победи	25	5
Стапка на успех на MiniMax: 83.33%		

Табела 5. Резултатот од натпреварите меѓу MiniMax & ExpectiMax

MiniMax покажува значително подобри перформанси во детерминистички услови. Неговата стратегија е насочена кон предвидливи сценарија, што му овозможува оптимално да ги анализира сите можни потези и да ја избере најдобрата опција. ExpectiMax е подобар за игри со случајни фактори, но во вакви контролирани услови, неговата ефикасност е значително помала.

Тестовите покажуваат дека Alpha-Beta е најефикасен алгоритам, особено кога игра прв. MiniMax е силен алгоритам во детерминистички средини, додека ExpectiMax е помалку ефикасен во строго дефинирани игри како Game of the Amazons. Редоследот на играње има значително влијание врз исходот, особено во натпреварите меѓу Alpha-Beta и MiniMax.

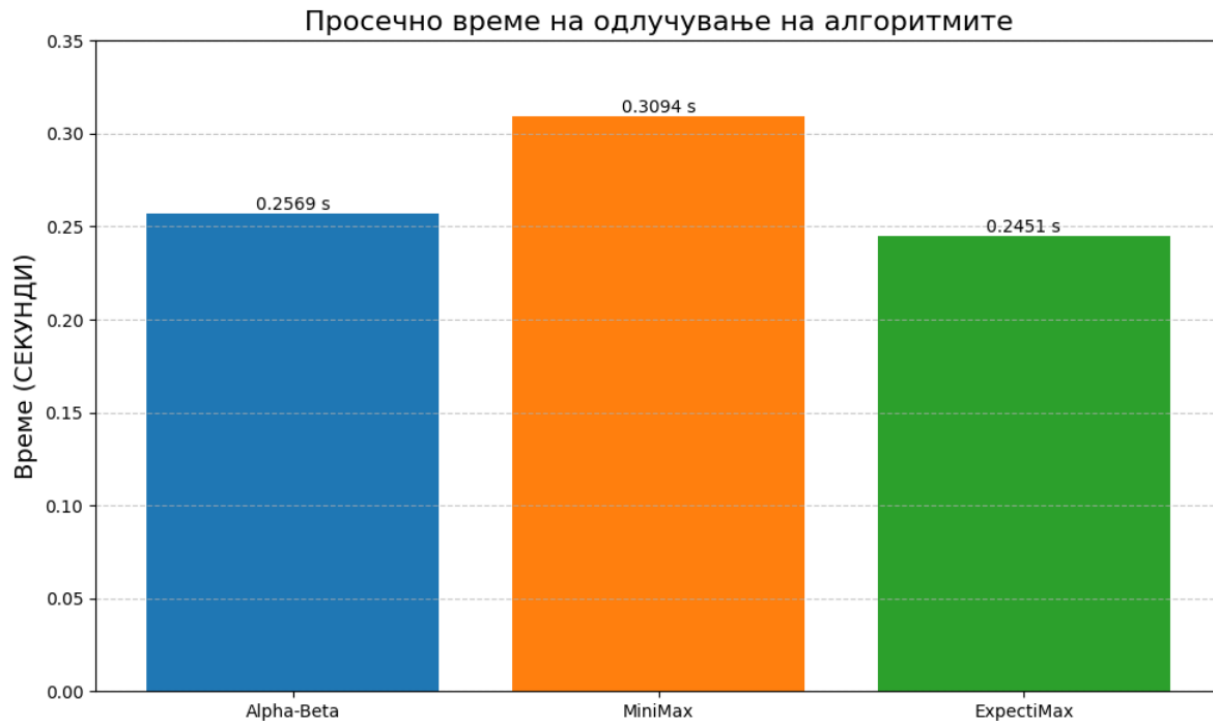


Слика 5. Резултати од сите натпревари помеѓу алгоритмите



### Просечно време на одлучување на секој од алгоритмите

- Alpha-Beta: 0.2569 секунди по потег
- MiniMax: 0.3094 секунди по потег
- ExpectiMax: 0.2451 секунди по потег



Слика 6. Просечно време на одлучување на секој од алгоритмите

### Човек против Вештачка Интелигенција (Human vs AI)

Покрај симулациите во кои се натпреваруваат два алгоритми за вештачка интелигенција, во рамките на проектот е имплементирана и можност за игра помеѓу човек и AI. Ова овозможува интерактивно тестирање на алгоритмите, каде што играчот може да ја испита стратегијата и тактичките одлуки на вештачката интелигенција во реално време.

Во овој мод, човекот може да игра прв, а потоа AI агентот пресметува оптимален потег користејќи го избраниот алгоритам за спротивставено пребарување. Играта може да се изведува со истите алгоритми кои беа наведени погоре (Alpha-Beta поткастрување, MiniMax, ExpectiMax).

За да се овозможи „Човек против AI“ модот, потребна е измена во иницијализацијата на играчите. Наместо да се постават два AI агенти, едниот играч е дефиниран како човек, а другиот како AI.

Оваа интерактивна функционалност овозможува детално тестирање и подобрување на алгоритмите преку директна човечка интеракција, што е корисно за анализа на стратегиите и подобрување на моделите за игра.

## Заклучок

Анализата на алгоритмите за Game of the Amazons открива комплексна динамика меѓу Alpha-Beta поткаструвањето и MiniMax алгоритмот. Иако Alpha-Beta генерално се истакнува со побрзо време на извршување и подлабоко пребарување, ефикасноста на двата алгоритми значително зависи од редоследот на играње.

Во директните натпревари меѓу Alpha-Beta и MiniMax, исходот често зависи од тоа кој алгоритам игра прв. Кога MiniMax има предност на првиот потег, тој може да постави стратегија која Alpha-Beta мора да ја следи, понекогаш водејќи до подеднакво ефикасни или дури и подобри резултати. Меѓутоа, кога Alpha-Beta игра прв, неговата способност за подлабока анализа му овозможува да ја искористи оваа предност и да ја контролира играта поефективно.

ExpectiMax се покажува како најмалку ефикасен за оваа игра, бидејќи неговиот пристап базиран на очекувани вредности не е соодветен за детерминистички услови.

Иако ефикасноста на Alpha-Beta и MiniMax варира во зависност од редоследот на играње, Alpha-Beta генерално демонстрира супериорност во брзината на извршување. Ова се должи на неговата способност селективно да ги истражува само релевантните гранки, ефикасно отстранувајќи ги непотребните делови од дрвото на пребарување. Оваа карактеристика му овозможува на Alpha-Beta да направи подлабоко пребарување во ист временски период, што често резултира со подобри и побрзи одлуки во комплексни игри како Game of the Amazons.

# Референци

[B. Sc. \(H\) Computer Science Semester VI](#)

[AI algorithm for game of the Amazons based on computer game playing](#)

[An Enhanced Solver for The Game of Amazons](#)

[MinMax Algorithm](#)

[Alpha Beta Pruning](#)

[Expectimax Algorithm in Game Theory](#)