**Kvantni križić-kružić i MiniMax bot s alfa-beta podrezivanjem**

Ena Škopelja, Mislav Žanić, Ivan Živković

# Pravila igre

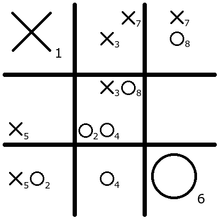
Kvantni križić-kružić je kvantna generalizacija običnog križić-kružića, koja uključuje kvantne fenomene:

1. superpoziciju (superposition)
2. zaplitanje (spregnutost, entanglement) i
3. raspad (collapse).

U svakom potezu, trenutni igrač označi dva polja sa svojim slovom (X ili O) indeksiranim s rednim brojem poteza (X1, O2, X3 itd.). Ta dva polja se tada nazivaju spregnutima (entangled). U slučaju da se na ploči zatvori ciklus spregnutih polja (na primjer, potez X1 povezuje polja 1 i 2, X5 polja 2 i 3 i O2 polja 3 i 1), igrač koji je sljedeći na potezu (u ovom primjeru nakon X5 to je igrač O) izvršava mjerenje ciklusa na način da odabere na koji način će se ciklus raspasti, za što uvijek postoje točno dva načina (u primjeru X5 na 2, X1 na 1 i O2 na 3 ili X5 na 3, X1 na 2 i O2 na 1). U slučaju da su neka polja iz ciklusa bila spregnuta s nekim poljima izvan ciklusa, i ta se polja raspadaju. U raspadnuta polja se više ne mogu igrati novi potezi. Prvi igrač koji postigne tri za redom u retku, stupcu ili na dijagonali (broje se samo raspadnuta polja) pobjeđuje, dok u slučaju da oba igrača dobiju tri za redom, pobjeđuje onaj koji u svojoj trojci ima najmanji maksimalni indeks (npr. ako je X dobio X1, X3 i X7 u istom retku, a O dobio O2, O4 i O6 u istom retku, pobjednik je O jer 6 < 7). Mogući rezultati u bodovima se označavaju sa 1-0 (ako X ima dva puta po tri za redom), 1-0 (ako X ima jedan put po tri za redom), 1-0.5 (ako i X i O imaju jedan put po tri za redom, ali X ima najmanji maksimalni indeks u svojoj trojci), 0-0 (ako niti jedan igrač nema tri za redom -> izjednačenje), 0.5-1, 0-1 i 0-1 (analogno za O umjesto X).

# Implementacija igre

Igru smo zbog jednostavnosti implementirali u Pythonu 3. Igra ima konzolno sučelje i podržava igru između dva ljudska igrača, između igrača i MiniMax bota i između dva MiniMax bota. Također je podržano pokretanje igre počevši od neke pozicije koja nije početna pozicija (prazna ploča).



# MiniMax bot s alfa-beta podrezivanjem

Minimax s alfa-beta podrezivanjem je implementiran koristeći dvije funkcije: minimax\_mark i minimax\_collapse. Ovakva implementacija je bila potrebna zbog mehanike raspadanja oznaka. Obje funkcije implementiraju klasični minimax uz dodatnu provjeru postoji li raspadno stanje ili ne. Također, moguće poteze (oblika (i, j), i, j = 1,…,9) ne pretražujemo abecedno već u redoslijedu {1, 3, 5, 7, 9, 2, 4, 6, 8} kako bi što prije došli do alfa i beta koji će podrezati što više grana. Za prva tri poteza bot koristi unaprijed izračunate optimalne poteze, jer inače ne-heuristički MiniMax s alfa-beta podrezivanjem traje predugo za interaktivnu igru protiv bota. Ostatak poteza se računa običnim MiniMax-om s alfa-beta podrezivanjem.

# Riješenost igre

Izvršavanje MiniMax algoritma na praznoj ploči uvijek daje pobjedu prvog igrača 1-0.5, što dokazuje jaku riješenost igre Kvantni križić-kružić kao pobjeda 1-0.5 za prvog igrača s optimalnim potezima s obje strane.