

Примена Минимакс алгоритма на “Икс-окс”

Минимакс алгоритам служи за одлучивање о следећем потезу у играма нулте суме. Овај рад примењује тај алгоритам на игру “Икс-окс”. Пошто се типична имплементација минимакс алгоритма састоји од кодирања, генерисања, евалуације, пропагације, и селекције, осмотрићемо и нашу имплементацију по тим деловима.

3.1 Кодирање

Иако је за људе тривијално да виде таблу и друге елементе игре, манипулишу њима, што физички, што апстрактно, и да увиде када је игра завршена. Међутим, да би рачунари радили исто потребно је описати податке о игри тако да рачунар може њима да манипулише. Мада на први поглед то можда није тешко, у општем случају избор кодирања у великој мери утиче на перформансе због огромног броја позиција које рачунар мора да обради по јединици времена.

У специфичном случају игре “Икс-Окс” једноставним кодирањем табле као матрице димензија 3×3 чија поља могу садржати симболе “X”, “O”, или празан стринг постижу се задовољавајуће перформансе.

3.2 Генерисање

Рачунар, без обзира на изабрано кодирање, углавном једноставно генерише све легалне потезе играча који је тренутно на потезу. Затим генерише све противничке одговоре на сваки од тих потеза, и тако даље до одређене “дубине”. Број таквих потенцијалних потеза у општем случају експоненцијално расте са дубином предвиђања.

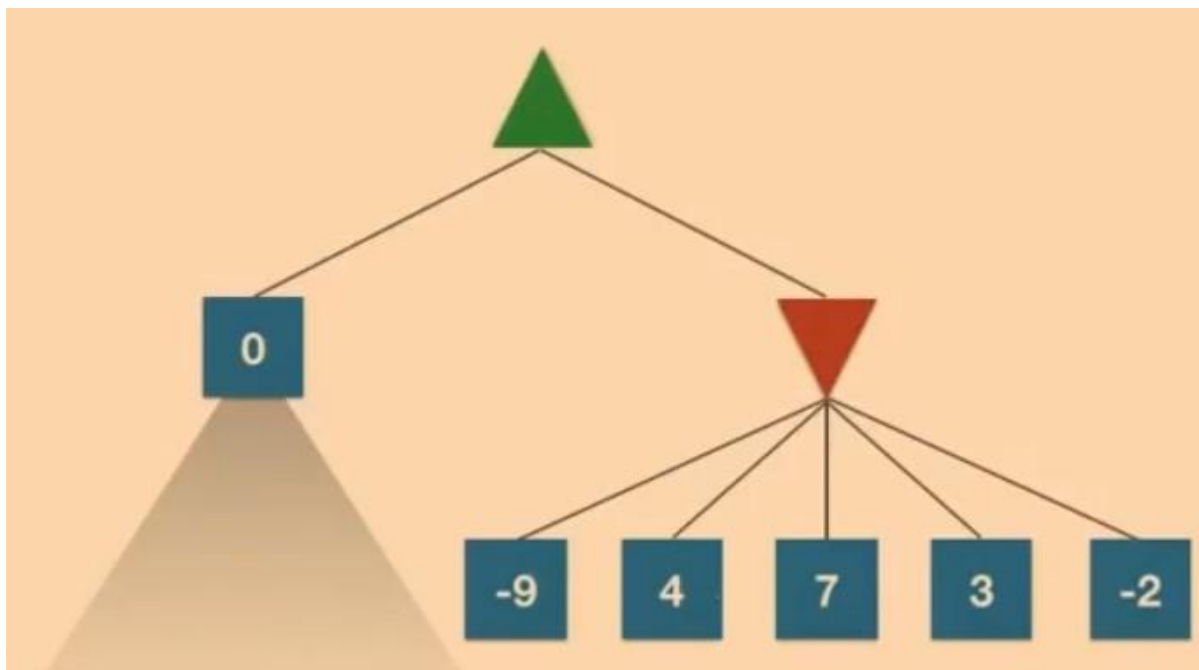
У случају “Икс-окс”-а генерисање легалних потеза је тривијално - свако поље са празним стрингом је легалан потез. Пошто је табла мала и игра кратка, укупан број могућих позиција је релативно мали (апсолутна горња граница, без узимања у обзир симетрије и правила игре је реда величине $9! = 362880$).

3.3 Евалуација и пропагација

У општем случају, Минимакс алгоритам ради са евалуацијама квалитета позиција, добијених неком хеуристиком. Евалуација одређене позиције је само апроксимација њеног квалитета - њен квалитет у ствари зависи од квалитета позиција које следе из ње. Зато, након што је изгенерисао и евалуирао одређени број позиција, програм мења евалуацију свих потеза осим најдубљих, на следећи начин.

Потенцијални потези формирају стабло. Почевши од “листова” стабла, најдубљих чворова графа, евалуације се пропагирају уназад. Пошто играчи наизменично играју, а функција евалуације је формирана тако да својим знаком показује који играч има већу вероватноћу да победи а својом вредношћу колика је та вероватноћа, евалуације се пропагирају тако што чворови “родитељи” на непарним дубинама узимају минималну вредност међу евалуацијама своје “деце”, а чворови на парним дубинама максималну (или обрнуто, у зависности од функције евалуације. Конвенционално, играч који први игра максимизује). То је наизменична минимизација и максимизација која даје алгоритму име. На Слици 1 је приказано поједностављено стабло потеза. Играч који

минимизује, који би био на потезу у позицији означеној црвеним троуглом, има пет потенцијалних потеза. Минимизујући играч изабрао би потез који ће га довести у позицију процењену најмањом вредношћу, -9. Зато би алгоритам позицији означеној црвеним троуглом доделио ту вредност. Међутим, пошто максимизујући играч има избор између те позиције и друге која је процењена нулом, он би изабрао већу вредност, и вредност зеленог троугла била би нула, па до позиције означене црвеним троуглом не би ни дошло.



Слика 1, Пример стабла потеза

Није нам позната ниједна хеуристика за оцењивање квалитета позиције у игри “Икс-Окс”, а због једноставности игре не постоји ни потреба за њом. Стабло могућих потеза је довољно мало да се дође до листова, тј. завршних стања. Ако се тим стањима доделе вредности, нпр. $-\infty$ за стање у ком побеђује “О” и $+\infty$ за победу “Х”-а, а за нерешено вредност 0, онда ће се једино те три вредности пропагирати уназад кроз стабло.

3.4 Селекција

У општем случају, због величине стабла потеза јављају се проблеми везани за избор грана које треба истраживати дубље него остале, и “проблем хоризонта”, који је последица ограничене дубине претраге. Међутим, као што је претходно споменуто, ова игра је довољно једноставна да ови проблеми не постоје, већ је комплетно стабло доступно у сваком тренутку.

Надаље, алгоритам какав је овде описан има две потенцијалне слабости у игри против људи. Као прво, потпуно је детерминистички - у истој позицији увек ће одиграти исти потез, чак и ако постоји још потеза са истом вредношћу. Алгоритам је дакле предвидљивији него што је неопходно.

Као друго, алгоритам претпоставља оптималну игру противника. Ако после потенцијалног потеза “А” сви одговори противника доводе до нерешене партије, а после потенцијалног потеза “Б” противник има само један одговор којим ремизира, а осталима губи партију, алгоритам ће изабрати потез “А” и тиме одузети себи шансу да победи, под претпоставком да противник игра оптимално, што углавном није случај.