

עץ משחק

עץ משחק הוא ייצוג מפורש של כל המהלכים האפשריים במשחק.

שורש העץ הוא המצב ההתחלתי של המשחק.

הבנים של השורש (successors) הם המצבים אליהם יכול להגיע השחקן הראשון ע"י ביצוע מהלך אחד. הבנים של בנים אלה הם המצבים היכולים להתקבל כתוצאה מתגובת השחקן השני וכך הלאה..

עלים – הם מצבים המציינים ניצחון, הפסד או תיקו.

כל מסלול משורש העץ לאחד העלים מייצג משחק מלא שונה.

משחק שני שחקנים

- נגדיר משחק כבעיית חיפוש בעלת המרכיבים הבאים:
- מצב התחלתי – מכיל את מצב (הלוח) ההתחלתי ואינדיקציה לגבי השחקן שיבצע את המהלך הבא.
- קבוצת אופרטורים – מגדירה מהם המהלכים החוקיים עבור שחקן.
- מבחן סיום – קריטריון לסיום המשחק.
- פונקצית תשלום – מתאימה ערך מספרי לתוצאת המשחק.

משחק שני שחקנים - המשך

השחקנים יקראו Min ו-Max. משחק ראשון. 

Max יחפש את האסטרטגיה שתוביל אל מצב סיום בו הוא מנצח ללא תלות במהלכי השחקן Min. 

היריב (Min) יבחר תמיד את הצעד הגרוע ביותר עבור השחקן הראשון (Max). 

אסטרטגית Minimax

V – פונקצית התשלום.



האלגוריתם:



אם הצומת הנוכחי הוא עלה

החזר את ערכו $V(s)$.

פתח את צומת s (צור את בניו s_1, s_2, \dots, s_b).

חשב את ערכי הבנים $V(s_1), V(s_2), \dots, V(s_b)$.

אם s הוא צומת MAX

החזר את האופרטור קס עבורו:

$$V(s) = \text{Max}\{V(s_1), V(s_2), \dots, V(s_b)\}$$

אם s הוא צומת MIN

החזר את האופרטור קס עבורו:

$$V(s) = \text{Min}\{V(s_1), V(s_2), \dots, V(s_b)\}$$

אסטרטגית Minimax - שיפור

נחליף את פונקצית התשלום V בפונקצית הערכה יוריסטית E .
פונקציה זו מחזירה הערכה לגבי התשלום שיניב משחק מהמצב הנתון. 

פונקצית ההערכה תחזיר ערכים גבוהים עבור מצבים המוערכים כטובים לשחקן Max ונמוכים עבור מצבים שמוערכים כרעים לשחקן (טובים עבור Min). 

פונקצית ההערכה חייבת להסכים עם פונקצית התשלום לגבי מצבי הסיום. 

פונקצית ההערכה אמורה לשקף בצורה מדויקת את הסיכוי לניצחון. 

אסטרטגית Minimax - המשך

בעיה: בדרך כלל עץ המשחק מאוד גדול! ↕

פתרון: נפתח את עץ המשחק עד לעומק מסוים (לא בהכרח נגיע למצב הסיום) ↕

העומק נקבע לפי הזמן שנקבע לביצוע מערך.

חסרון של הפתרון המוצע: יתכן כי מסלול משחק בעומק מסוים יראה כניצחון למרות שמעשית כאשר מעמיקים יותר הוא מוביל להפסד! ↕

מסקנה: נרצה העמקה הדרגתית מתוחכמת יותר: גיזום. ↕

גיזום - α - β pruning

שיפור: נותר על פיתוח ענפים שאינם יכולים לשנות את ערך המינימקס של השורש. 


עץ רגיל ועץ גזום יחזירו את אותו ערך. 


העיקרון המנחה: 

בהינתן צומת n כלשהו בעץ, כך שהשחקן יכול לעבור אליו במהלך הבא. אם לשחקן קיימת אלטרנטיבה טובה יותר, m בצומת האב של n או בכל נקודת החלטה גבוהה יותר, אזי לעולם המשחק לא יגיע לצומת n .

מסקנה: ברגע שגילינו כי לא נגיע לצומת n (באמצעות פיתוח בניו) – ניתן לגזום אותו. 

גִּיזוּם - α - β pruning - המשך

גִּיזוּם רדוד (shallow cutoff) – האינפורמציה לחיתוך ענף באה מהאחים או דודים שלו (ולא מרמות גבוהות יותר). 

גִּיזוּם עמוק (deep cutoff) - האינפורמציה לחיתוך ענף באה מהרמות הגבוהות (אבות, סבים..). 

גיזום - α - β pruning - המשך

חסמי החיתוך



חסם α - חסם החיתוך לצומת j מסוג Min. זהו הערך הגבוה ביותר שקיים עד עתה לכל אבות ה- Max של j . ניתן להפסיק את פיתוח j ברגע שהערך שלו שווה או קטן מ- α .

חסם β - חסם החיתוך לצומת i מסוג Max . הערך הנמוך ביותר שקיים עד עתה לכל אבות ה- Min של i . ניתן להפסיק את פיתוח i ברגע שהערך שלו שווה או גדול מ- β .

גיזום - α - β pruning - המשך

האלגוריתם:

```
Minimax-alpha-beta(board, depth, type, alpha, beta)
if depth = 0 then return (evaluate(board))
else
  if type = Max then
    cur-max  $\leftarrow$  -infinity
    loop for b in succ(board)
      b-val  $\leftarrow$  Minimax-alpha-beta(b,depth-1,min, alpha,beta)
      cur-max  $\leftarrow$  max(b-val, cur-max)
      alpha  $\leftarrow$  max(cur-max,alpha)
      if cur-max  $\geq$  beta then finish loop
    end loop
    return cur-max
  else ;; type=Min
    cur-min  $\leftarrow$  infinity
    loop for b in succ(board)
      b-val  $\leftarrow$  Minimax-alpha-beta(b,depth-1,max, alpha,beta)
      cur-min  $\leftarrow$  min(b-val, cur-min)
      beta  $\leftarrow$  min(cur-min,beta)
      if cur-min  $\leq$  alpha then finish loop
    end loop
    return cur-min
```

גיזום - α - β pruning - המשך

שיפור נוסף: סידור הילדים ע"י שימוש בפונקצית הערכה. 

חסרונות: 

אפקט האופק (the horizon effect) – פתרון שנראה טוב
לטווח קרוב אך בעצם דוחה את המכשול לשלב מאוחר יותר.