

נשנה את האלגוריתם כך:

- נתחיל מחלוקה ריקה המאותחלת $(1,1:0)$
- ניצור את כל המצבים הנובעים ממצב קיים + חלוקת חפץ אחד
כאן יש שינוי: במקום להוסיף את הערך של החפץ שמקבלים, נכפיל אותו.
- נמחק מצבים מיותרים (pruning)
- מתוך כל המצבים הסופיים, נבחר מצב עם הערך המינימלי הגדול ביותר.

`Mul_egalitarian_allocation([[4, 5], [8, 7]])`

מצב התחלתי: $([1,1]:0)$

נתינת חפץ א': $([1,8]:1)$, $([4,1]:1)$

חסם פסימי: 4 בחלוקה אקראית שחקן 1 יקבל את חפץ א ושחקן 2 את חפץ ב

חסם אופטימי: 20 המינימום כאשר כולם מקבלים את כל החפצים שנשארו

חסם אופטימי < חסם פסימי

נתינת חפץ ב': $([1,56]:2)$, $([5,8]:2)$, $([4,7]:2)$, $([20,1]:2)$

$([4,1]:1)$

חסם פסימי: 0, בצורה אקראית שחקן א קיבל גם את החפץ השני והשחקן השני לא קיבל שום חפץ.

חסם אופטימי: 7, שחקן א מקבל את החפץ השני כפול הערך שיש לו= 20

שחקן ב מקבל את החפץ השני כפול הערך שיש לו= 7

חסם אופטימי < חסם פסימי

$([1,8]:1)$

חסם פסימי: 0, בצורה אקראית שחקן ב קיבל גם את החפץ השני ושחקן א לא קיבל שום חפץ.

חסם אופטימי: 5, שחקן א מקבל את החפץ השני כפול הערך שיש לו= 5

שחקן ב מקבל את החפץ השני כפול הערך שיש לו= 56

חסם אופטימי < חסם פסימי

- מתוך כל המצבים הסופיים נבחר את המצב עם הערך המינימלי הגדול ביותר: $([5,8]:2)$