A white paper with black writing

Description automatically generated with low confidence***שאלה 1***

1. *מכונת המצבים הינה:*
2. *הגרף מחלקת קשירות אחת, לכן הגרף Irreducible, וקיימת מחלקת קשירות אחת*
3. *נתבונן בגרף שקיבלנו, נחשב את הזמן מחזור של כל מצב:*

*לפיכך ניתן לקבוע כי השרשרת היא aperiodic.*

1. *נחשב את ההתפלגות הסטוציונרית:*
2. *נחשב את ה expected return time לכל מצב:*

*הראנו כי השרשרת אי פריקה ומחזורית, לכן המצבים הינם Positive recurrent , לפי הנלמד בשיעור, לכל מצב מתקיים:*

*לכן:*

1. *נגדיר מטריצה P’ באופן הבא:*

*נקבל:*

*A picture containing text, sky, whiteboard, flying

Description automatically generated*

*נחשב את ההתפלגות הסטוציונרית:*

*נקבל:*

*וכמובן, מתקיים:*

***שאלה* 3**

1. *נגדיר MDP:*

*כאשר:*

* *x הינו קבוצת הslots הפנויים, שלא שובצו מספרים בהם (יכול לקבל כל תת קבוצה של*
* *הינו המספר שהוגרל, ונדרש לשיבוץ באחד ה slots*

*בנוסף:*

*ז"א, בהינתן מצב , קבוצת הפעולות האפשריות (ז"א הצבת מספר שהוגרל באחד מהתאים הריקים מתוך s), הינו האוסף:*

*ונגדיר את פונקציית הרווח עבור ו*

*ז"א, עבור הצבת המספר בתא j, נקבל "תוספת" רווח למספר הסופי מהצורה הנ"ל.*

*נגדיר את מטריצת המעברים-   
עבור פעולה המציבה את המספר בתא j, יש לנו 10 מצבים שונים שניתן לעבור אליהם (בהתאם להגרלת המספר הבא) כל אחד בסיכוי :*

*אז, עבור מעבר ממצב , ביצענו את הפעולה , נעבור למצב באופן הבא:*

*לדוגמה, עבור:*

*ומצבי ההתחלה יהיו:*

*בנוסף הבעיה היא finite horizon, לכן*

1. *למדנו בשיעור כי:*

*עבור MDP עם finite horizon, לפיכך המדיניות האופטימלית עבור מצב s בזמן t איננו מושפע מהגרלת המספרים קודמת, ואיפה הם שובצו אלא רק מהמצב הנתון – קבוצת ה slots הריקים והמספר הרנדומי שדורש שיבוץ.*

*נתבונן על החלטה של מדיניות האופטימלית עבור מצב , אז לפי הטענה שציינו לעיל, המדיניות האופטימלית עבור מצב תהיה זהה למדיניות כאילו היה המצב ההתחלתי, עבור בעיה קטנה יותר.*

*נראה כי קיימת מדיניות אופטימלית יחידה-*

*עבור 2 מצבים שונים (כאשר הוצבו מספרים שהוגרלו בעבר, בתאים שונים, אך בשני המצבים נשארו m תאים ריקים), אנו נדרשים להציב מספר שהוגרל באחד מהתאים הריקים מתוך m תאים.*

*הרי כי מצבים אלו שקולים, עד כדי הפעלת פונקציה מונוטונית על הרווח שהצטבר עד כה-   
נניח כי הרווח המקסימלי עבור יתבצע ע"י הצבת המספר שהוגרל בתא j, אז המדיניות עבור תניב רווח מקסימלי ע"י השמת המספר בתא j’, כמובן ברווח יהיה כפולה של המספר 10.*

*נראה את המדיניות בעזרת אינדוקציה - על כמות המקומות הריקים – slots שנמצאים ב"משחק":*

***מקרה בסיס-*** *נותר slot אחד פנוי, אז ברור כי המדיניות האופטימלית תבצע השמה של המספר i שהוגרל בתא הריק.*

***הנחה-***

*עבור m תאים פנויים, נניח כי בוצע מדיניות אופטימלית של השמת מספרים בתאים , נסמן אותה כ , והרווח שלה כ (כאשר s’ מתאר את המצב בו אנו נמצאים, ואיזה מספר הוגרל).*

***צעד-***

*עבור m+1 תאים פנויים, נראה כי המדיניות האופטימלית איננה תלויה בעבר, אלא רק בתאים הריקים והמספר שהוגרל:*

*לפיכך ניתן לראות כי עבור פונקציית הרווח, היא איננה תלויה בשיבוצים הקודמים, כל עוד בוצע השמה אופטימלית, אלא רק בכמה תאים פנויים יש, ומה המספר שהוגרל.*

1. *נשתמש ב backwards recursion –* 
   1. *עבור הספרה האחרונה , יש לנו בדיוק מקום אחד לשים אותה, לפיכך:*
   2. *עבור הספרה לפני אחרונה, ישנם 3 מצבים אפשריים*

*\_ \_ x, \_ x \_ , x \_ \_*

*נחשב את הפעולה האופטימלית עבור כל מצב:*

*\_ \_ x :*

*ז"א, אם המספר שהוגרל קטן מ5, נשים אותו בתא האמצעי (אינדקס 1), אחרת נשים אותו בתא השמאלי (אינדקס 0).*

*\_ x \_ :*

*\_ \_ x:*

* 1. *עבור הספרה הראשונה, נדרש להחליט היכן למקם אותה:*

A picture containing text, whiteboard

Description automatically generated