אוניברסיטת
אריאל
בשומרון

דוח מסכם

מבוא למידת חיזוקים - תשפ"ה סמסטר ב'

המחלקה להנדסה תעשייה וניהול

מספר קבוצה:	**מרצה:**	**נושא:**
1	ד"ר לזבניק טדי	סודוקו

מגישים ותעודות זהות		
אדר סבן - 313174120	אליה זגורי - 207313131	מתן וידל - 206508426

אוניברסיטת
אריאל
בשומרון

אוניברסיטת
אריאל
בשומרון

1. סקירת הפרויקט

1.1 רקע כללי

סודוקו הוא פאזל לוגי הדורש מהשחקן למלא לוח בגודל 9×9 במספרים מ-1 עד 9, כך שבכל שורה, עמודה ותת-ריבוע 3×3 יכילו כל ספרה בדיוק פעם אחת. מדובר בבעיה מסוג Constraint Satisfaction Problem , בעלת מרחב מצבים עצום, בה מרבית הרצפים האפשריים מובילים לפתרון לא תקין או חסום. לפיכך, פתרון יעיל של סודוקו דורש הבנה של אילוצים לוגיים, תכנון מהלכים קדימה ותגובה לתוצאות הביניים.

1.2 מטרת הפרויקט

מטרת הפרויקט היא לפתח סוכן למידת חיזוקים (Reinforcement Learning Agent) המבוסס על אלגוריתם DQN, שיוכל לפתור לוחות סודוקו באמצעות אינטראקציה עם הסביבה בלבד – ללא הנחיה מפורשת על אילו צעדים נכונים או שגויים. הסוכן לומד באמצעות חיזוקים בלבד, על בסיס הפעולות שהוא מבצע, ונדרש לגבש מדיניות המובילה לפתרון תקין של הלוח.

1.3 מוטיבציה

בחירת סודוקו כבעיה ללמידת חיזוקים נובעת מהשילוב בין פשטות חיצונית למורכבות פנימית: החוקים ברורים, אך מרחב הפתרונות עצום והמשוב המיידי מוגבל. זהו אתגר קלאסי ללמידת חיזוקים – במיוחד כאשר אין תגמול מיידי מוחלט לפעולה, אלא רק שילוב מורכב של צעדים רבים שתורמים (או פוגעים) בדרך לפתרון.

בנוסף, פתרון סודוקו מחייב את הסוכן להתמודד עם אילוצים, עדכון מדיניות תוך כדי תנועה, ויכולת לתכנן צעדים עתידיים – אתגרים שמתאימים בדיוק לאופי הקורס ולמטרותיו.

1.4 היקף העבודה

בתחילת הדרך ניסינו ליישם גישת Q-Table קלאסית, אך נתקלנו בקושי מהותי: מאגר המצבים בסודוקו הוא עצום, שכן יש מיליוני מצבים אפשריים בלוח 9×9, מה שהופך את שמירת ערכי Q עבור כל מצב ופעולה לבלתי ישימה מבחינה חישובית וזיכרונית. לכן, עברנו לשיטת DQN (Deep Q-Network), שבה הרשת העצבית לומדת להעריך את ערכי Q באמצעות הכללה על פני מצבים דומים. כך ניתן להתמודד עם המורכבות של סביבת סודוקו מבלי לשמור טבלה מלאה של כל הצירופים האפשריים.

במהלך הפרויקט נבנתה סביבה מותאמת אישית המדמה לוחות סודוקו ברמות קושי שונות. הסוכן לומד לפעול בה על ידי הכנסת מספרים או תנועה בין תאים, תוך קבלת חיזוקים בהתאם לתקינות הפעולות. האלגוריתם שנבחר הוא DQN בשילוב Replay Buffer, Target Network ודעיכת ε. המודל נבנה באמצעות ספריית PyTorch, תוך שימוש ברשת Fully Connected פשוטה לפלט של ערכי Q.

הביצועים הוערכו באמצעות גרפים המציגים ניקוד מצטבר, מגמות רגרסיה ודעיכת אפסילון. הסוכן הצליח להגיע לשיפור מדורג לאורך זמן, בעיקר בלוחות ברמה קלה ובינונית.

אוניברסיטת
אריאל
בשומרון

2.ניסוח הבעיה

2.1 ניסוח פורמלי של הבעיה

פתרון סודוקו באמצעות למידת חיזוקים מהווה אתגר מורכב בשל מרחב המצבים העצום, המשוב הלא מיידי והאילוצים הלוגיים הקשיחים של הפאזל. מרבית הרצפים האפשריים מובילים לטעויות או מבוי סתום, ולעיתים השפעת פעולה ניכרת רק לאחר מספר צעדים. הסוכן פועל בסביבה חלקית ודינמית, שבה עליו לגבש מדיניות פעולה ללא הדרכה חיצונית, אלא מתוך חיזוקים בלבד.
כאמור סודוקו מציב אתגר ייחודי בלמידת חיזוקים מכמה סיבות מרכזיות:

מרחב מצבים עצום ודל:
מספר האפשרויות למילוי לוח הוא אסטרונומי, אך רק קומץ קטן מהן מוביל לפתרון חוקי. מרבית הרצפים מובילים למבוי סתום.
משוב לא מיידי:
לא תמיד ברור לסוכן מיד אם פעולה מסוימת היא חיובית או שלילית. למשל, הכנסת ספרה עשויה להיות תקינה בזמן הצעד, אך להתברר כשגויה רק לאחר צעדים נוספים.
אילוצים לוגיים מורכבים:
כל פעולה מושפעת מהקשרים מרחביים — לא רק התא עצמו, אלא גם השורה, העמודה ותת־הריבוע. זה יוצר תלות הדדית בין פעולות, אשר מקשה על הסוכן להבין את השפעת פעולתו.
אילוצים לוגיים מורכבים:
כל פעולה מושפעת מהקשרים מרחביים — לא רק התא עצמו, אלא גם השורה, העמודה ותת־הריבוע. זה יוצר תלות הדדית בין פעולות, אשר מקשה על הסוכן להבין את השפעת פעולתו.

2.2 מרחב המצבים

בכל צעד, מצב הסביבה מוגדר על ידי:

- לוח סודוקו בגודל 9×9, עם ערכים שלמים בין 0 (ריק) ל־9.
- רשימת תאים נעולים (שהוזנו מראש בפאזל) ורשימת תאים שבהם ביקר הסוכן במהלך הפרק.
- אין ייצוג מפורש של מיקום — כל פעולה כוללת בתוכה את כתובת התא והערך.

המצב מקודד כוקטור חד־ממדי באורך 81 (כל הלוח שטוח), ומועבר כקלט לרשת העצבית של הסוכן. קידוד זה מאפשר שמירה על פשטות חישובית אך אינו כולל הקשרים מרחביים מפורשים - על הרשת ללמוד אותם בעצמה.

2.3 מרחב הפעולות

מרחב הפעולות בפרויקט זה הוא דיסקרטי, בגודל 9×9×9=729 פעולות אפשריות. כל פעולה מקודדת שלישיית ערכים:
i - אינדקס שורה (0-8)
j - אינדקס עמודה (0-8)
v - ערך מספרי שאותו הסוכן מכניס לתא/
הסוכן בוחר פעולה אחת בכל צעד, והסביבה מפענחת אותה לשלישיית פעולה.
לא מתבצעת תנועה בין תאים מרחב פעולה זה יוצר אתגר חקירה לא טריוויאלי, במיוחד לאור כך שלסוכן אין מנגנון שגרתי לסנן פעולות חוקיות מראש, אלא עליו ללמוד אותן מתוך התנהגות ותגמולים בלבד.הסוכן פשוט "מצביע" על תא וערך בכל פעם, ללא תלות במיקום קודם.

2.4 פונקציית התגמול

פונקציית התגמול נועדה לכוון את הסוכן לעבר פתרון חוקי של לוח הסודוקו. היא מבוססת על כללים מוגדרים מראש אשר משתנים בהתאם לרמת הקושי (easy / medium / hard), ומופעלים בכל צעד על פי תוצאת הפעולה שנבחרה.
במהלך כל צעד, הסביבה בודקת:

1.האם הפעולה חוקית לפי חוקי הסודוקו (ללא כפילויות בשורה/עמודה/ריבוע).
2.האם התא היה ריק וטרם בוצעה בו פעולה (visited positions).
3.האם הפעולה תרמה לפתרון הלוח.

<u>דוגמא לערכי תגמול ברמת קושי "medium":</u>

פעולה	תגמול
הכנסת ערך חוקי	+20
הכנסת ערך לא חוקי	-10
ניסיון לשנות תא נעול	-15
חזרה לתא שכבר בוצעה בו פעולה	-5
פתרון מלא של הלוח	+500

העיצוב של פונקציית התגמול הוא קריטי, שכן הוא מכתיב את אופי המדיניות שהסוכן מפתח. בפועל, גילינו כי איזון עדין בין תגמולים חיוביים לשליליים משפיע רבות על קצב ההתקדמות ואיכות הלמידה.

2.5 דינמיקת הסביבה

הסביבה פועלת כמערכת סגורה אשר מקבלת פעולה מהסוכן, מפעילה אותה על הלוח הנוכחי, ומחזירה שלושה ערכים: `state`, `reward`, `done`.

בכל צעד של אפיזודה:

1. כאמור, הסוכן בוחר פעולה בדידה מתוך 729 הפעולות האפשריות, המקודדת כשלישייה: שורה, עמודה, וערך.
2. הסביבה מפענחת את הפעולה, מנסה לעדכן את הלוח בהתאם, ומבצעת בדיקה:

- האם הפעולה חוקית?
- האם התא נעול?
- האם הסוכן כבר ניסה לפעול באותו תא?

3. מחושב תגמול בהתאם, תוך שקלול של מספר פרמטרים (תקינות, חזרתיות, הצלחה).
4. מתבצעת עדכון של מצב הלוח, רשימת התאים שבהם הסוכן כבר פעל (`visited_positions`), וסטטוס הסביבה (`done`).
5. הלוח החדש מועבר לרשת כוקטור קלט חדש.

<u>תנאי סיום יוגדרו בצורה הבאה:</u>

- **הצלחה:** הלוח נפתר בצורה תקינה לחלוטין — כל שורה, עמודה ותת-ריבוע מכילים את כל הספרות מ־1 עד 9.

- **כישלון:** הסוכן לא מצליח להשלים את הלוח לאחר מספר צעדים מוגדר מראש (למשל, 200 צעדים).

המעבר בין המצבים תלוי לחלוטין בפעולות שנבחרו, כאשר כל פעולה משנה באופן פוטנציאלי את התנהגות הסוכן בפרקים הבאים. מבנה זה של סביבה מאפשר התנהגות לא דטרמיניסטית במובן הרחב, אך כן נשלטת על ידי כללי הסודוקו.

הפרויקט מגדיר את פתרון הסודוקו כבעיה של למידת חיזוקים בדידה, שבה הסוכן נדרש ללמוד מדיניות פעולה בסביבה מורכבת, חלקית ורועשת. מרחב המצבים כולל את לוח הסודוקו, התאים הנעולים והיסטוריית הפעולות, ואילו מרחב הפעולות כולל את כל האפשרויות החוקיות (והלא חוקיות) להזנת ערכים בלוח.

פונקציית התגמול עוצבה כך שתאזן בין עידוד צעדים נכונים לבין ענישה מדורגת על שגיאות — ובכך תכוון את הסוכן להתנהגות יעילה ופתרון הלוח. דינמיקת הסביבה מגיבה לכל פעולה בהתאם לחוקיותה, אך אינה מונעת מהסוכן לטעות – כך שנדרש תהליך חקירה ולמידה הדרגתי.

מורכבות הבעיה אינה טמונה רק בגודל מרחב האפשרויות, אלא גם בתלות ההדדית בין תאים, חוסר המשוב הישיר על טעויות, והצורך לבנות הבנה מצטברת של מבנה הפאזל. כל אלה הופכים את פתרון הסודוקו לאתגר משמעותי, המשלב למידה אסטרטגית עם תגמול מבוסס תוצאה.

3.שיטת הלמידה ואלגוריתם

בפרויקט זה יישמנו את אלגוריתם DQN (Deep Q-Learning) לפתרון פאזלי סודוקו באמצעות למידת חיזוקים. מטרת האלגוריתם היא ללמד את הסוכן לבחור את הפעולה האופטימלית (בחירת ערך לתא מסוים) בכל שלב, מתוך מרחב פעולה של 729 אפשרויות, באופן שמביא לפתרון חוקי של הלוח.
הבעיה מנוסחת כך שהסוכן מקבל תגמול על פעולות שמקדמות אותו לעבר פתרון חוקי, ועניש על שגיאות כמו כפילויות או ניסיון לשנות תא נעול.

3.1 מבנה הרשת העצבית

הסוכן משתמש ברשת נוירונים מסוג Fully Connected אשר מקבלת כקלט את הלוח הנוכחי כוקטור באורך 81 (לוח סודוקו שטוח), ומחזירה וקטור באורך 729 – כל ערך מייצג את ה־Q-value של פעולה אחת (שיבוץ ערך מסוים בתא מסוים).

מבנה הרשת:

- שכבת קלט: 81 נוירונים
- שכבות חבויות: 256 → 128 → 64 עם הפעלות ReLU
- שכבת פלט: 729 נוירונים (פעולות מקודדות מסוג $i\times81 + j\times9 + v$)

הרשת לומדת את הערך המשוער של כל פעולה בלוח מסוים, בהתאם להצלחת הסוכן בעבר, וכך מכוונת אותו לבחירות טובות יותר.

3.2 מדיניות הפעולה - epsilon greedy

כדי לאזן בין חקירה של פעולות חדשות לניצול הידע הקיים, הסוכן משתמש במדיניות ε-greedy. בתחילת האימון ערך ε הוא 1, כך שהפעולות נבחרות באקראי. אנו בחרנו שהערך ירד בקצב של 0.9999 בכל אפיזודה, עד לערך מינימלי של 0.1. בכל צעד, בהסתברות ε הסוכן בוחר פעולה אקראית חוקית (exploration), ובהסתברות 1−ε בוחר את הפעולה עם ערך Q הגבוה ביותר (exploitation). שיטה זו מאפשרת לסוכן ללמוד אילו פעולות משתלמות לאורך זמן, במיוחד בסביבת סודוקו שבה מרחב הפעולות עצום ורק חלק קטן מהן חוקי בכל מצב.
מניתוח גרפי של שיטה זו נעזרנו ע"מ להבין מתי הסוכן לומד והאם משתפר בפתרון.

3.3 Replay Buffer ולמידה מהעבר

כדי לשפר את היציבות והכלליות של הלמידה, הסוכן משתמש בזיכרון חוויות (Replay Buffer) המאחסן טרנזיציות מהצורה: (state, action, reward , next_state, done)

בכל שלב למידה, נדגמת קבוצה אקראית של 64 טרנזיציות מתוך זיכרון של עד 50,000.
שיטה זו מאפשרת למידה ממוזגת (batch) ומפחיתה הטיות שמקורן ברצף זמני של פעולות.
על ידי שימוש ב־Replay Buffer, הסוכן יכול ללמוד מדגימות שהתרחשו בשלבים שונים של האפיזודה (ולעיתים בלוחות שונים), מה שמאפשר לו לזהות דפוסים כלליים יותר — לדוגמה: איפה כדאי למלא קודם, באילו תאים קל לטעות, או אילו מבנים מניבים תגמולים גבוהים.

שיטה זו תורמת לכך שהסוכן לא "יתחפר" בלוח מסוים או ברצף טעויות חוזר, אלא ילמד נ
שמתאימה למגוון לוחות ורמות קושי.

3.4 עדכון רשתות

עדכון הרשת מתבצע לפי עקרונות Q-Learning תוך שימוש בשתי רשתות נפרדות:

- **רשת פעולה (q_eval)**: משמשת להערכת הפעולה במצב הנוכחי.
- **רשת מטרה (q_target)**: משמשת להפקת ערכי ה־Q של המצב הבא, ומתעדכנת כל 100 צעדים כדי לייצב את הלמידה.

הערך המטרה (target) מחושב לפי הנוסחה:

(Q_target) = reward + γ * max(Q_target(next_state)

הפסד מחושב לפי MSE בין הערך החזוי לערך המטרה, והאופטימיזציה נעשית בעזרת אלגוריתם Adam כאשר learning rate = 0.001.

בפתרון סודוקו, כל פעולה בודדת עשויה להשפיע על מספר מבנים בלוח (שורה, עמודה, תת־ריבוע), ולעיתים השפעתה ניכרת רק לאחר מספר צעדים. לכן, השימוש ברשת מטרה נפרדת תורם ליציבות ומונע תנודות חדות מדי בלמידה.

במיוחד במצבים שבהם הפעולה הנוכחית חוקית אך מובילה בהמשך למבוי סתום, רשת המטרה מאפשרת לסוכן ללמוד לאורך זמן איזה רצפים באמת משתלמים — לא רק בטווח הקצר, אלא בראייה כוללת של פתרון הלוח.

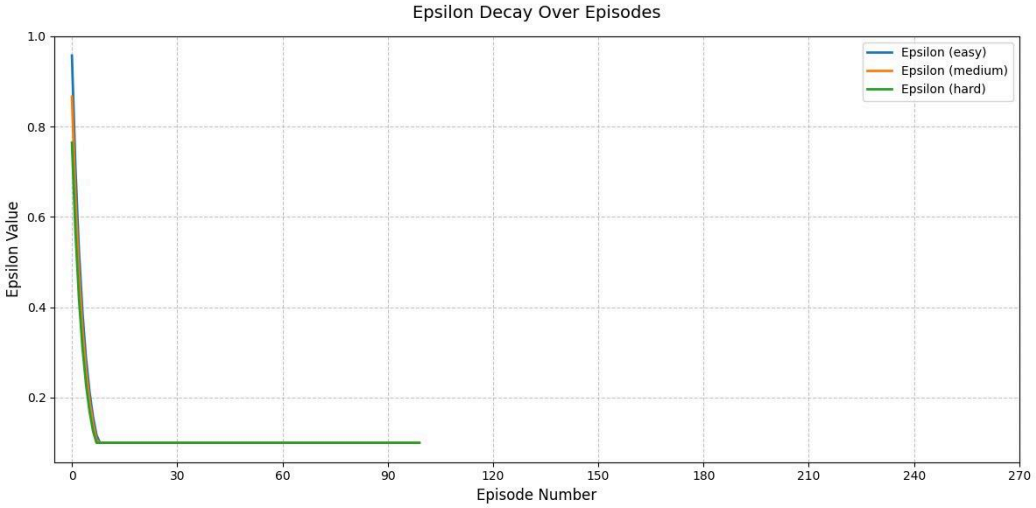
4. תוצאות

4.1 תוצאות כמותיות

בשלב זה נציג מדדים מספריים מרכזיים שנמדדו במהלך הניסוי, המטרה היא להציג בצורה ברורה את ביצועי הסוכן בהיבטים שונים של הלמידה.

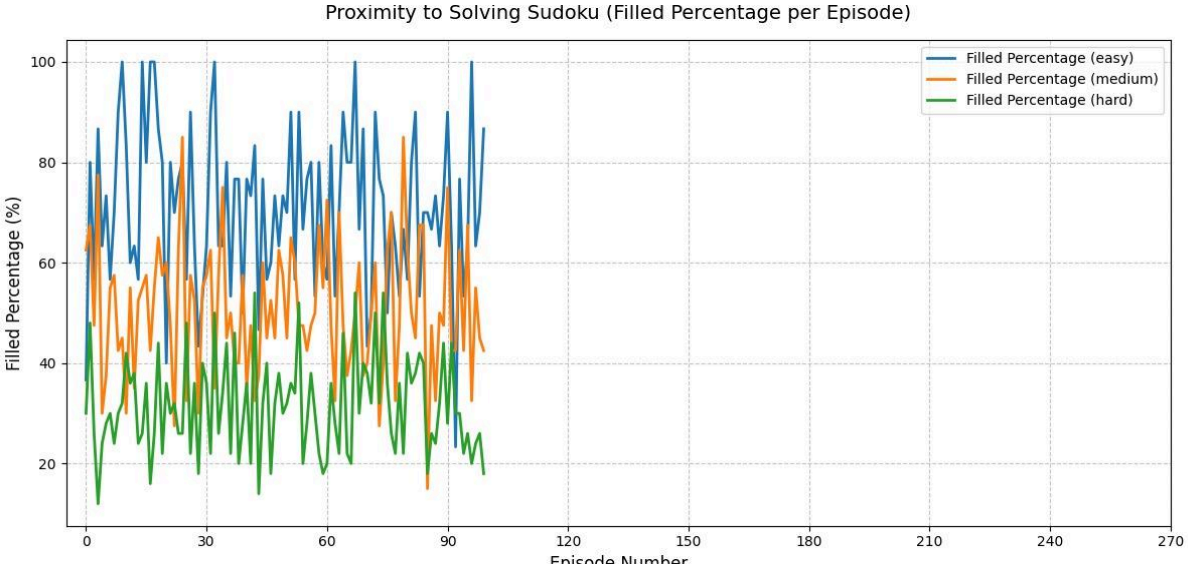
4.1.1 ירידת אפסילון לאורך זמן

מדד: ערך ε לאורך אפיזודות.
פירוט: אפסילון התחיל מ־1.0 וירד בהדרגה עד 0.1 לפי קצב של 0.9995. ערכי ε נאספו לאורך 150 אפיזודות לכל רמה.
מה נמדד בפועל: ערכי ε נשמרו ברשימת `eps_history` ונצפו בגרף הבא:



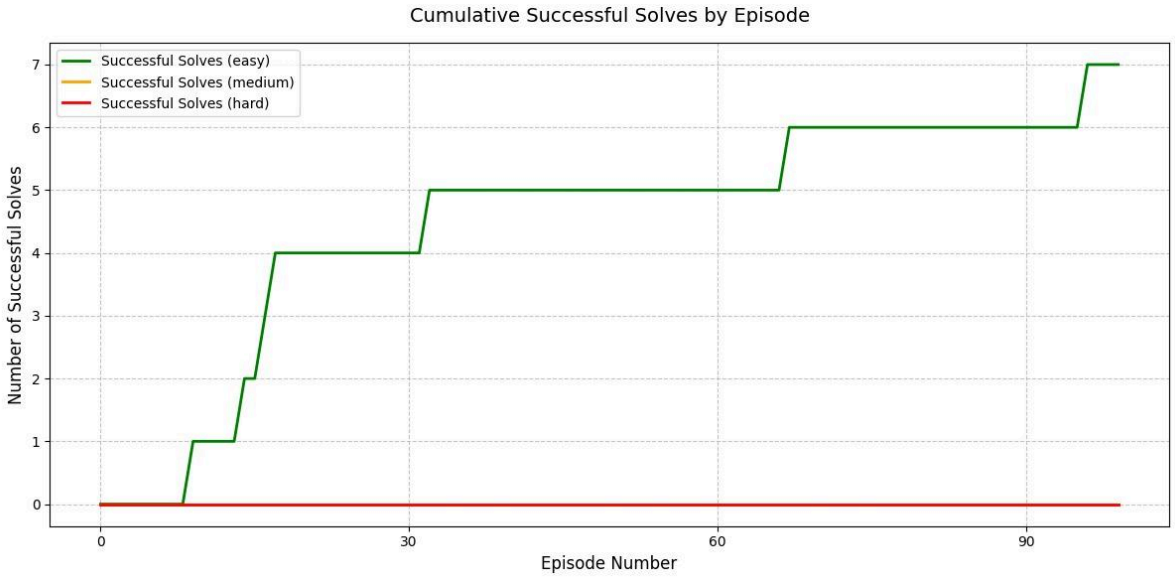
4.1.2 אחוז תאים שמולאו נכון

מדד: אחוז המילוי התקין מתוך הלוח המלא.
פירוט: מדד זה משקף את רמת הקרבה של הסוכן לפתרון מלא. הוא נמדד עבור כל אפיזודה, ונע בין 0% ל־100%.
מה נמדד בפועל: השוואה בין לוח הסוכן לבין הפתרון האמיתי בכל צעד, אחוז תאים נכונים.



4.1.3 מספר פתרונות מוצלחים (מצטברים)

מדד: פתרונות מלאים וחוקיים שזוהו על ידי הסוכן.
פירוט: מספר האפיזודות שבהן הסוכן הצליח להשלים את הלוח באופן מלא, ללא שגיאות. נמדד מצטבר לאורך האפיזודות.
מה נמדד בפועל: בדיקה אם הלוח פתור לחלוטין לפי חוקי סודוקו (unique valid solution), וספירת הצלחות.



4.2 ניתוח איכותי של התוצאות

הניתוח האיכותי עוסק בפרשנות של התוצאות הכמותיות, תוך בחינת ההתנהגות הלמידתית של הסוכן, ההבדלים בין רמות הקושי, והגורמים שהשפיעו על הביצועים.

4.2.1 אפסילון – איזון בין חקירה לניצול

גרף האפסילון מדגים כיצד הסוכן עבר מחקירה אקראית להסתמכות על מדיניות למידה.

- כבר באפיזודה 30 ירד ערך אפסילון כמעט למינימום (0.1), מה שמעיד על הפסקת החקירה בשלב מוקדם.
- ברמת "קל", הירידה הזו הייתה מספקת כי מרחב הפעולות קטן יותר.
- ברמות "בינוני" ו"קשה", שלב החקירה לא הספיק לחשוף את כל האפשרויות – מה שפגע ביכולת הלמידה.
- האלגוריתם לא התאים את קצב הירידה למורכבות הסביבה, ולכן הסוכן "ננעל" מוקדם מדי על מדיניות חלקית.

4.2.2 אחוזי מילוי הלוח – איכות הפתרון החלקי.

הגרף מראה עד כמה הסוכן הצליח למלא את הלוח בצורה חוקית, גם אם לא פתר אותו במלואו.

- ברמת "קל" נצפה שיפור יציב עם אחוזי מילוי גבוהים לאורך זמן.
- ב"בינוני" ו"קשה", נרשמה תנודתיות חדה – לעיתים צניחות לאזור 20%–40%.
- הסיבה: התגמול ניתן ברמת פעולה בודדת, ואינו מכוון ליצירת רצף תקף של פעולות.
- כתוצאה מכך, הסוכן לא הצליח לזהות דפוסים גלובליים או להימנע משגיאות חוזרות.

4.2.3 פתרונות מלאים – מבחן ההצלחה הסופי

הגרף האחרון מודד הצלחות בפועל – פתרון מלא של לוח הסודוקו.

- ברמת "קל" הסוכן פתר 7 לוחות באופן מלא, עם שיפור מדורג לאורך הזמן.
- ב"בינוני" ו"קשה" לא נרשם אפילו פתרון תקין אחד.
- הסיבה: האלגוריתם מסתמך על תגמול סופי בלבד – ולכן לא מצליח ללמוד מהצלחות חלקיות.
- בלי תכנון קדימה או חיזוק ביניים, הסוכן מתקשה לבנות רצף שלם של פעולות חוקיות.

4.3 סיכום התובנות מהתוצאות

שלושת הגרפים שבחנו לאורך הפרויקט מציירים תמונה ברורה של יכולות הלמידה של הסוכן, תוך הבחנה חדה בין רמות הקושי. גרף ירידת האפסילון מראה שהסוכן הפסיק לחקור בשלב מוקדם יחסית, מה שעבד היטב ברמת "קל" – שם מרחב האפשרויות מצומצם יותר – אך לא הספיק ברמות "בינוני" ו"קשה", שם היה צורך בזמן חקירה ארוך יותר כדי להבין את מבנה הבעיה. גם כשבחנו את אחוזי המילוי של הלוח, ראינו שהסוכן מצליח לייצר פתרונות חלקיים יחסית יציבים רק ברמה הקלה, בעוד שבשאר הרמות הביצועים היו תנודתיים ולא עקביים. הגרף השלישי, שהציג את מספר הפתרונות המלאים, חיזק את התמונה: רק ברמת "קל" הסוכן הצליח להגיע לפתרונות שלמים, ואילו בשתי הרמות האחרות – לא הצליח אפילו פעם אחת. מכל אלה עולה שהסוכן הצליח ללמוד התנהגות חיובית כשמרחב האפשרויות היה פשוט, אך התקשה מאוד כשנדרש ממנו לתכנן מהלכים מורכבים לאורך זמן.

5. סיכום

5.1 מה עבד בפרויקט

הסוכן הצליח ללמוד מדיניות אפקטיבית לפתרון פאזלים ברמת קושי קלה. נצפתה למידה הדרגתית, שיפור עקבי באחוזי המילוי, ואף פתרונות מלאים בפועל. רכיבי האלגוריתם המרכזיים – כולל Replay Buffer, רשת נוירונים, מנגנון ε-greedy – פעלו כנדרש והשתלבו בהצלחה במבנה סודוקו מותאם. תהליך האימון סיפק תוצאות מדידות שהעידו על הבנה מצטברת של כללי הפאזל.

5.2 מה היה מאתגר

הסוכן התקשה להתמודד עם פאזלים ברמות בינונית וקשה. ברמות אלה, מרחב המצבים גדול משמעותית, מספר התאים הריקים רב, ותלות האילוצים גבוהה. הירידה המוקדמת באפסילון הגבילה את שלב החקירה, והסוכן לא הצליח לגבש רצף פעולות שמוביל לפתרון מלא. בנוסף, העובדה שהתגמול ניתן רק על פעולות מקומיות או על פתרון שלם הגבילה את היכולת ללמוד מדיניות טובה לאורך זמן.

5.3 מה למדנו מהתהליך

הפרויקט המחיש את מגבלות אלגוריתם DQN במצבים שבהם אין משוב מיידי חזק, או כאשר נדרש תכנון לטווח ארוך. למדנו את החשיבות של תכנון מבנה תגמול שמתמרץ גם פתרונות חלקיים, ושל התאמה בין פרמטרי הלמידה לבין רמת המורכבות של הבעיה. יחד עם זאת, למדנו גם כיצד מודל פשוט יחסית מצליח להגיע להצלחות של ממש בתנאים נכונים.

5.4 תיקון לאחר הפרזנטציה

בהתאם למשוב שקיבלנו לאחר שלב הפרזנטציה, עדכנו את האלגוריתם ששימש אותנו במהלך הפרויקט ל־**DQN מלא**. נוספו רכיבים חסרים והקוד תוקן כך שיכלול Replay Buffer בגודל 50,000, מדגם למידה (batch) בגודל 64, וירידת אפסילון מדורגת. כל הרכיבים יושמו מחדש ותוקנו בהתאם להערות שקיבלנו — כפי שנדרש.

5.5 סיכום כללי

הפרויקט הדגים את הפוטנציאל של למידת חיזוקים גם במשימות עם אילוצים מורכבים, לצד מגבלות שדורשות פתרונות מתקדמים יותר. ברמות פשוטות, השיטה הצליחה היטב, אך ככל שרמת הקושי עלתה – נדרשה התאמה עמוקה יותר של מנגנון הלמידה והתגמול. הפרויקט סיפק הבנה עמוקה של למידה התנהגותית בסביבות מורכבות כמו סודוקו, והיווה בסיס להמשך שיפור עתידי.