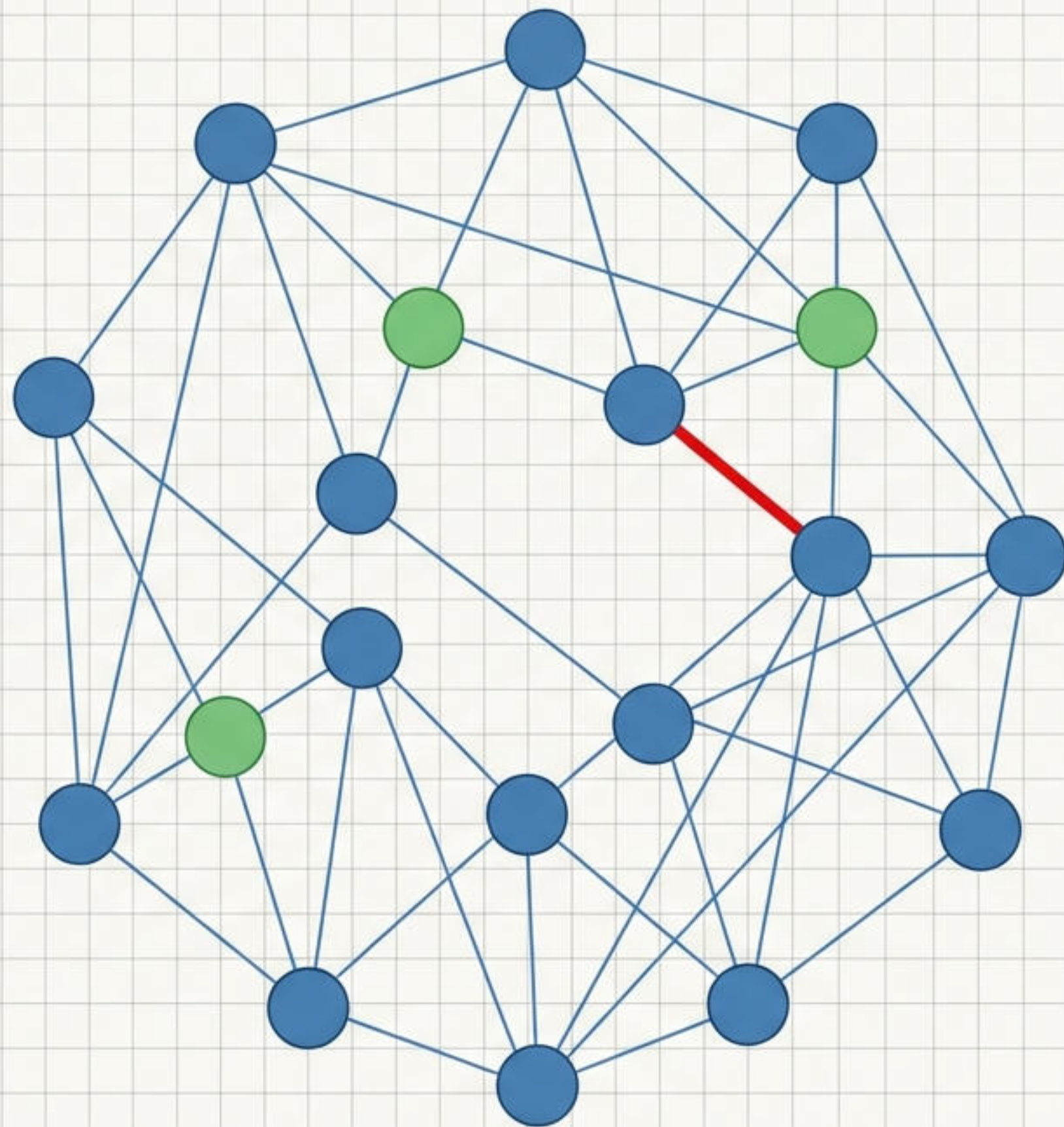


מבוא לבינה מלאכותית: שליטה בבעיות סיפוק אילוצים (CSPs)

מתיאוריה לפרקטיקה: המדריך
החזותי לפתרון בעיות, כולל ניתוח
מקרי בוחן.



האנטומיה של בעיית CSP: שינוי פרדיגמה בחיפוש



אילוצים (Constraints - C)

החוקים והתנאים שחייבים
שחייבים להתקיים בין
המשתנים.



תחומים (Domains - D)

קבוצת הערכים האפשריים
שניתן להקצות לכל
משתנה.



משתנים (Variables - X)

סט קבוע של משתנים
המרכיבים את מצב
הבעיה (State).

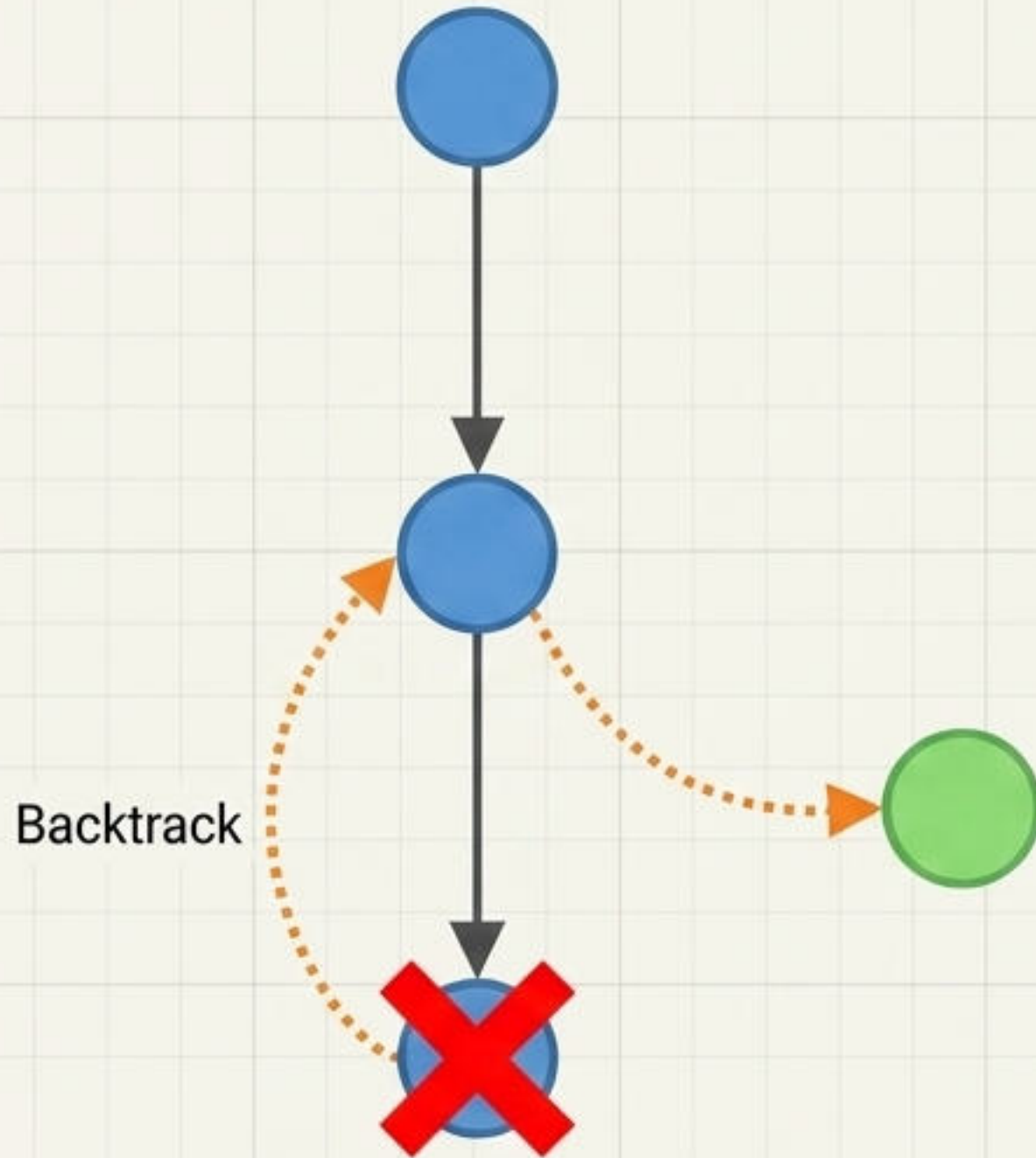
בניגוד לאלגוריתמי חיפוש קלאסיים (כמו DFS/BFS), ב-CSPs המדינה (State) מוגדרת
במדויק על ידי ערכי המשתנים, ומבחן המטרה הוא סיפוק כל האילוצים במלואם.

מנוע החיפוש הבסיסי: Backtracking

הגדרה: Backtracking הוא אלגוריתם חיפוש לעומק (Depth-First Search) משודרג.

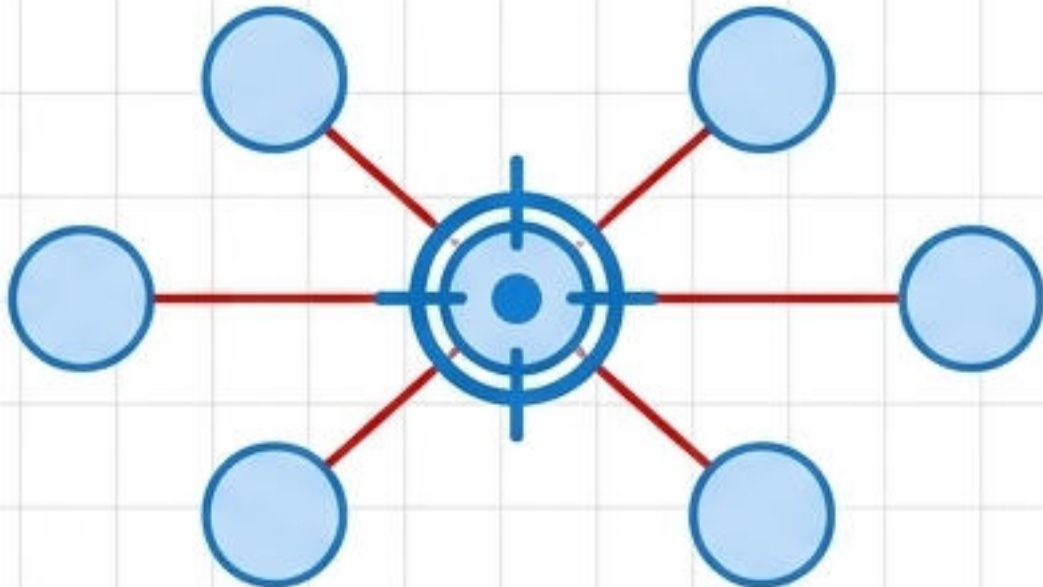
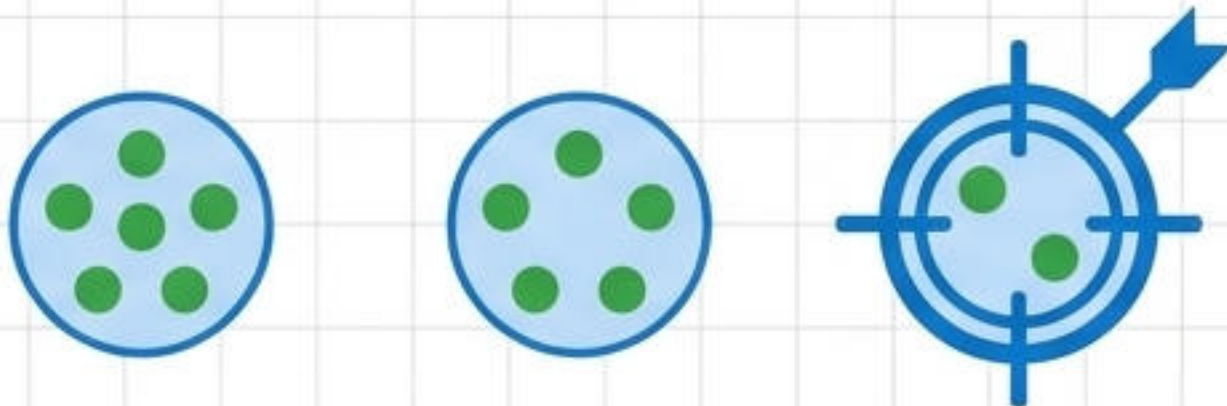
פעולה: בכל צומת (Node) בעץ החיפוש, אנו מקצים ערך למשתנה אחד בלבד.

מנגנון הנסיגה: ברגע שזיהינו הפרת אילוץ - או שמשתנה נותר ללא ערכים חוקיים בתחום שלו - האלגוריתם "חוזר אחורה" צעד אחד (Backtracks) ומנסה הקצאה חלופית למשתנה הקודם.



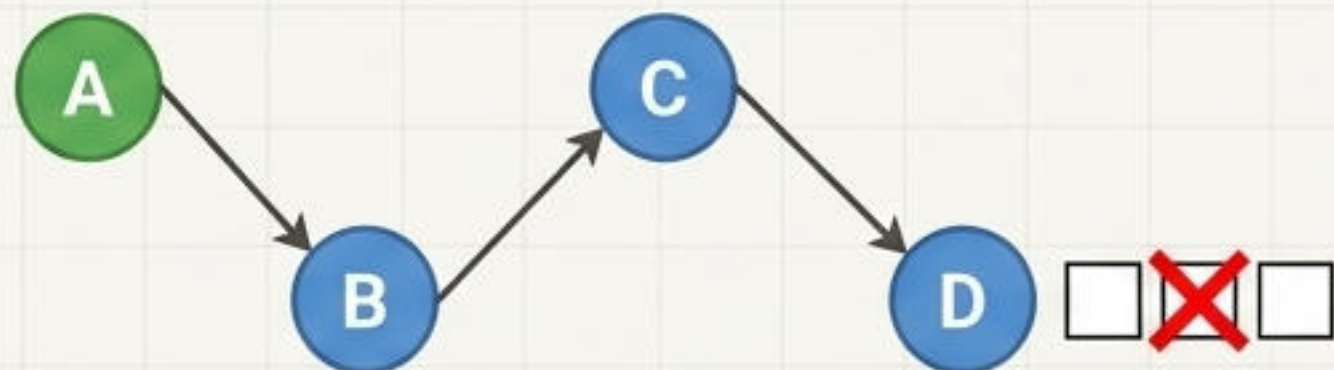
יוריסטיקות לבחירת משתנים: מי הבא בתור?

במקום לבחור משתנים באקראי, יוריסטיקות מייעלות דרמטית את זמן הריצה.

Degree Heuristic (יוריסטיקת הדרגה)	Minimum Remaining Values (MRV)
	
הכלל: בוחרים את המשתנה שמעורב בהכי הרבה אילוצים מול משתנים שטרם הוקצו להם ערכים. ההיגיון: "לפתור את הפקק המרכזי". משמש לרוב כשובר שוויון (Tie-breaker) כאשר ל-MRV יש מספר משתנים עם אותו גודל תחום.	הכלל: בוחרים את המשתנה בעל התחום המצומצם ביותר (הכי מעט ערכים חוקיים שנותרו). ההיגיון: "הכי כדאי להיכשל מהר". טיפול מוקדם בצווארי בקבוק מונע בניית עצי חיפוש שנדונים לכישלון.

סינון מראש (Filtering): צמצום אפשרויות כדי למנוע טעויות

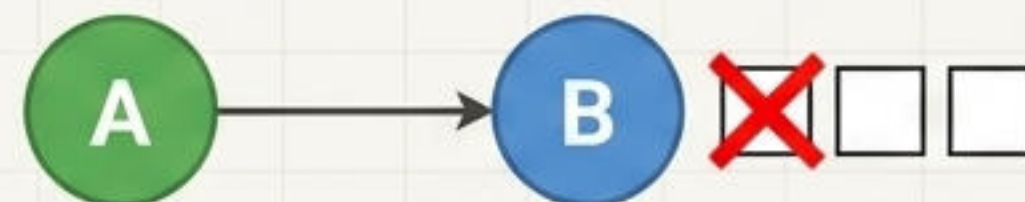
עקביות קשתות (Arc Consistency)



פעולה: מבצע עבודה עמוקה יותר. בודק שלא רק לשכנים ישירים יש ערכים חוקיים, אלא שכל זוג משתנים קשורים בגרף מקיימים תאימות הדדית (יכולים לספק את האילוצים יחד).

יתרון: מזהה אי-התאמות מורכבות יותר מוקדם יותר, אך דורש יותר משאבי חישוב.

בדיקה קדימה (Forward Checking)

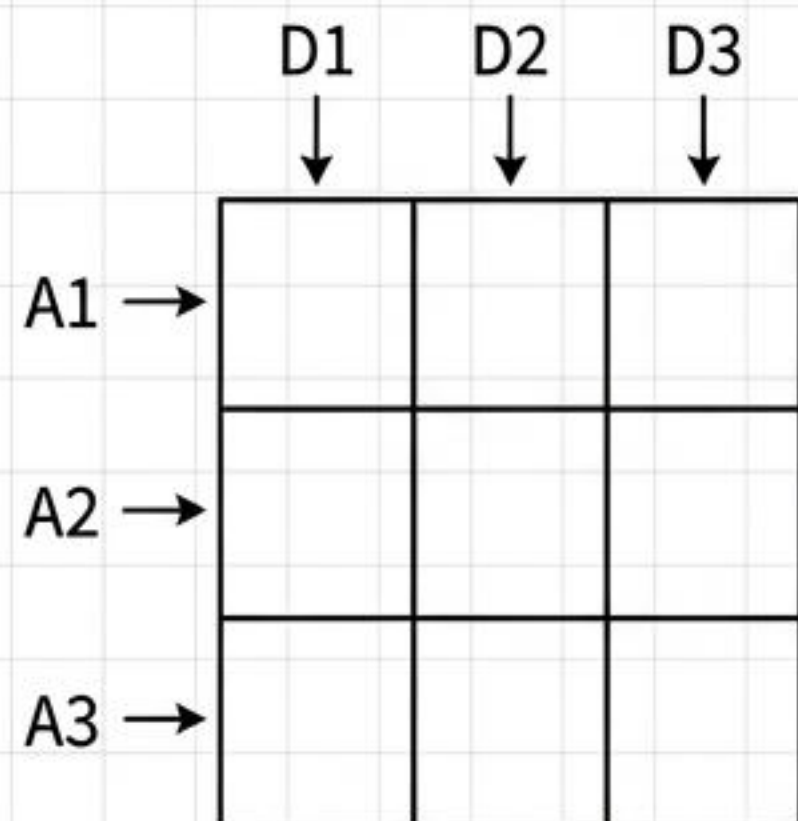


פעולה: מיד לאחר הקצאת ערך למשתנה, מסירים מהתחומים של כל שכניו הישירים (שטרם הוקצו) כל ערך שמפר את האילוץ האילוץ.

יתרון: מונע באופן אקטיבי ענפי חיפוש שיובילו לכישלון ודאי.

מקרה בוחן 1: ייצוג תשבץ כבעיית CSP

המטרה: שיבוץ 6 מילים באורך 3 אותיות מתוך מאגר נתון של 40 מילים.



מאגר מילים (Word Bank)

add, ado, age, ago, aid, ail,
aim, air, and, any, ape,
apt, arc, are, ark, arm,
art, ash, ask, auk, awe,
awl, aye, bad, bag, ban,
bat, bee, boa, ear, eel,
eft, far, fat, fit, lee, oaf,
rat, tar, tie

אפשרויות ייצוג (Formulation Options)

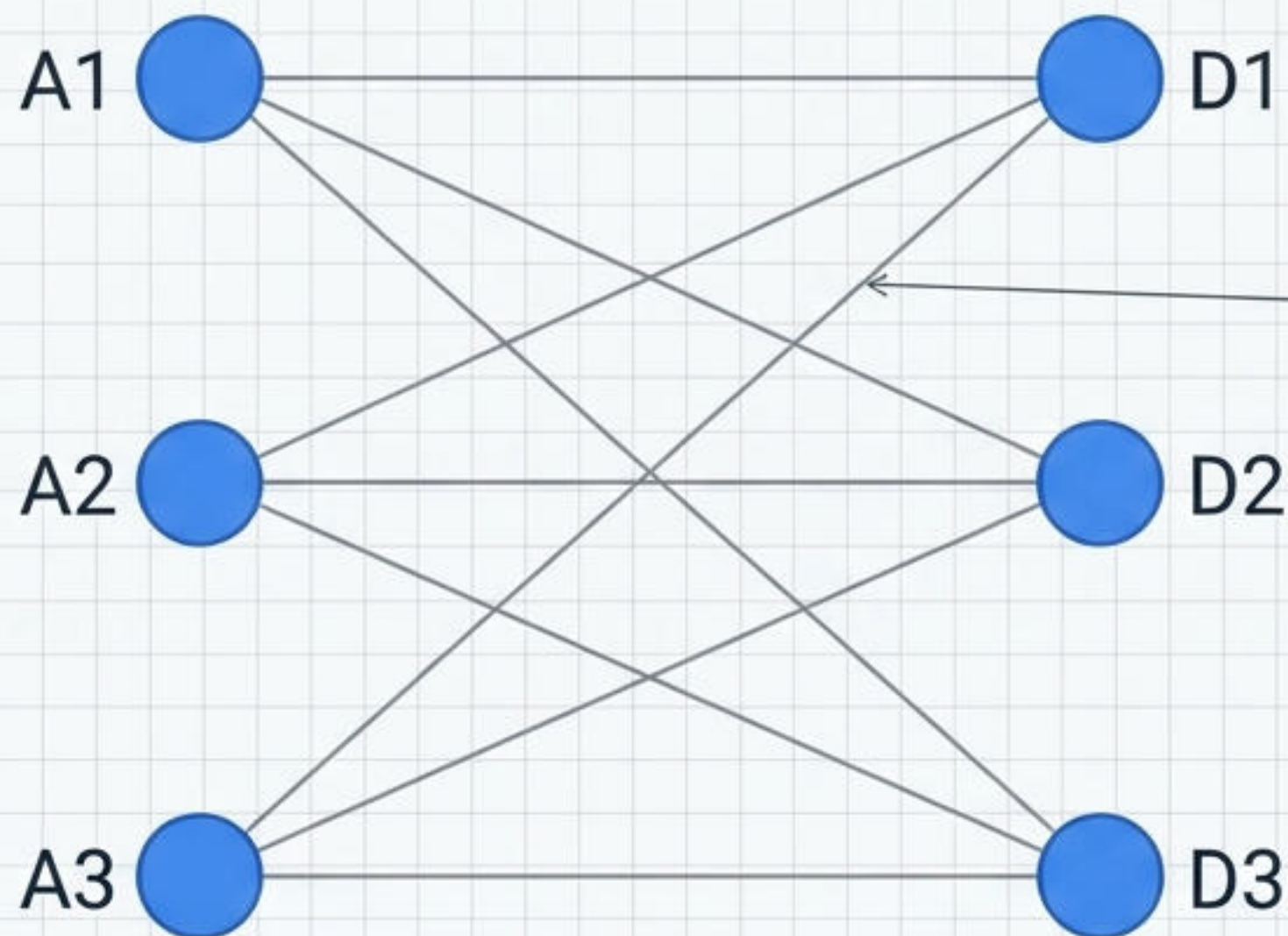
1 
אופציה א' (המועדפת):
משתנים = המילים במלואן (A1, A1, D1, A2, A1).
(A3, A2, A3, D1, D2, D3).
תחום (Domain) = 40 המילים שבמאגר.

2 
אופציה ב' (נפסלה):
משתנים = תאים בודדים בתשבץ (9 משתנים).
תחום = 22 אותיות הא"ב האנגלי.

ייצוג מילים כמשתנים מצמצם דרמטית את מרחב החיפוש.
האילוצים במקרה זה יהיו רק **נקודות החיתוך בין מילה אופקית לאנכית** (חייבות להכיל את אותה אות), ואילוץ AllDiff המבטיח שלא נשתמש במילה פעמיים.

מבנה הגרף: איך המחשב 'רואה' את התשבץ

AllDiff



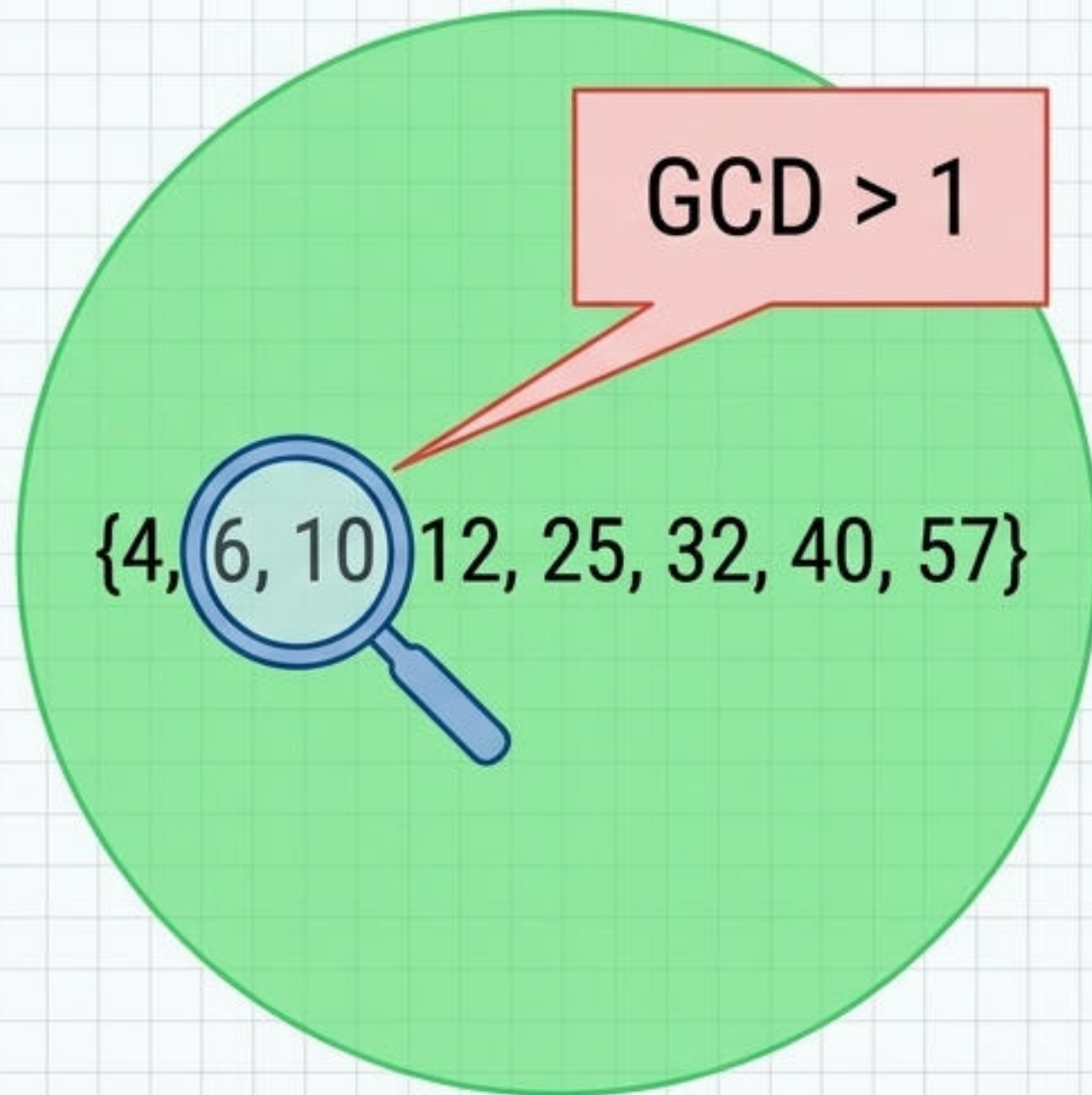
הצמתים (Nodes): 6 המשתנים שלנו - 3 מילים אופקיות (A) ו-3 אנכיות (D).

הקשתות (Edges): כל קשת מייצגת אילוץ חיתוך. בגלל שכל מילה אופקית חותכת כל מילה אנכית, מתקבל "גרף דו-צדדי מלא" (Complete Bipartite Graph).

אילוץ גלובלי (Global Constraint): תיבת ה-AllDiff העוטפת מציינת אילוץ-על החל על כל 6 המשתנים יחד - אסור להקצות את אותו ערך מהתחום לשני משתנים שונים.

מקרה בוחן 2: בעיית המספרים הזרים וטריק ה-Boolean

בהינתן קבוצת מספרים, עלינו למצוא תת-קבוצה בגודל k שבה אף זוג מספרים לא חולק מחלק מעותק משותף גדול מ-1 (מספרים זרים).



ניסוח כבעיית סיפוק אילוצים (CSP)

המשתנים (X) : כל מספר בקבוצת הקלט מוגדר כמשתנה נפרד $(4, 10, \dots)$.

התחום (D) : בוליאני $\{True, False\}$. כלומר, ההחלטה היא רק "האם המספר הזה נכנס לתת-קבוצה או לא?".

האילוצים (C):

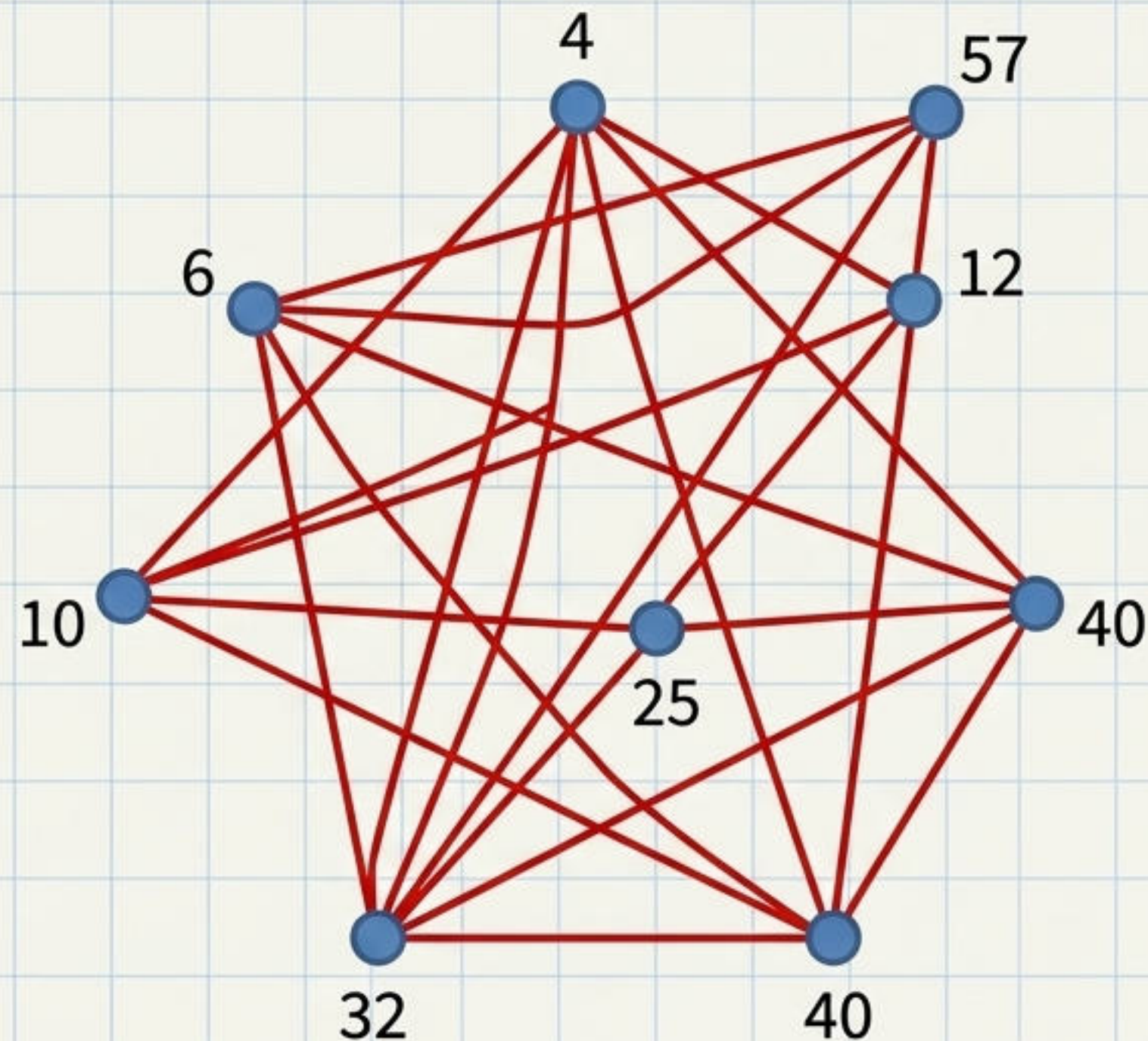
1. בדיוק k משתנים חייבים לקבל את הערך $True$.
2. אילוץ מניעה (NAND): אם לשני מספרים יש מחלק משותף, הם לא יכולים לקבל שניהם את הערך $True$ במקביל.

מיפוי הקונפליקטים: בניית גרף האילוצים

משמעות הקשתות: כל קשת בגרף משקפת קונפליקט פוטנציאלי. מתחנו קו בין מתננו קו בין שני מספרים רק אם הם חולקים מחלק משותף.

תרגום לאילוץ: קשת מתוחה משמעותה ששני הצמתים בקצותיה אינם יכולים להיות מוקצים ל-True בו זמנית.

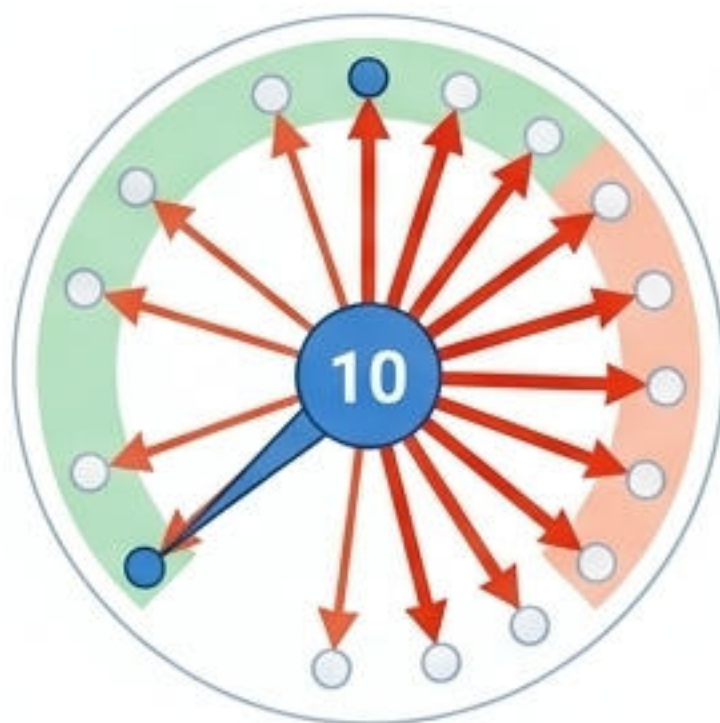
מבט חזותי על הגרף חושף מיד מי המשתנים "הבעייתיים" ביותר בעלי הפוטנציאל הגבוה ביותר ליצור קונפליקט (אלו עם הכי הרבה קשתות).



The Problem-Solvers Canvas

ניתוח יוריסטיקות במבחן המציאות

Degree



הצלחת ה-Degree: יוריסטיקת הדרגה מחפשת את המשתנה בעל המספר הרב ביותר של קשתות (אילוצים) למשתנים שטרם הוקצו.

מסקנה פרקטית: בגרף שלנו, נבחר את המספר שחולק מחלקים משותפים עם הכי הרבה מספרים אחרים, כדי לפתור את מוקד הקונפליקט המרכזי כבר בתחילת החיפוש.

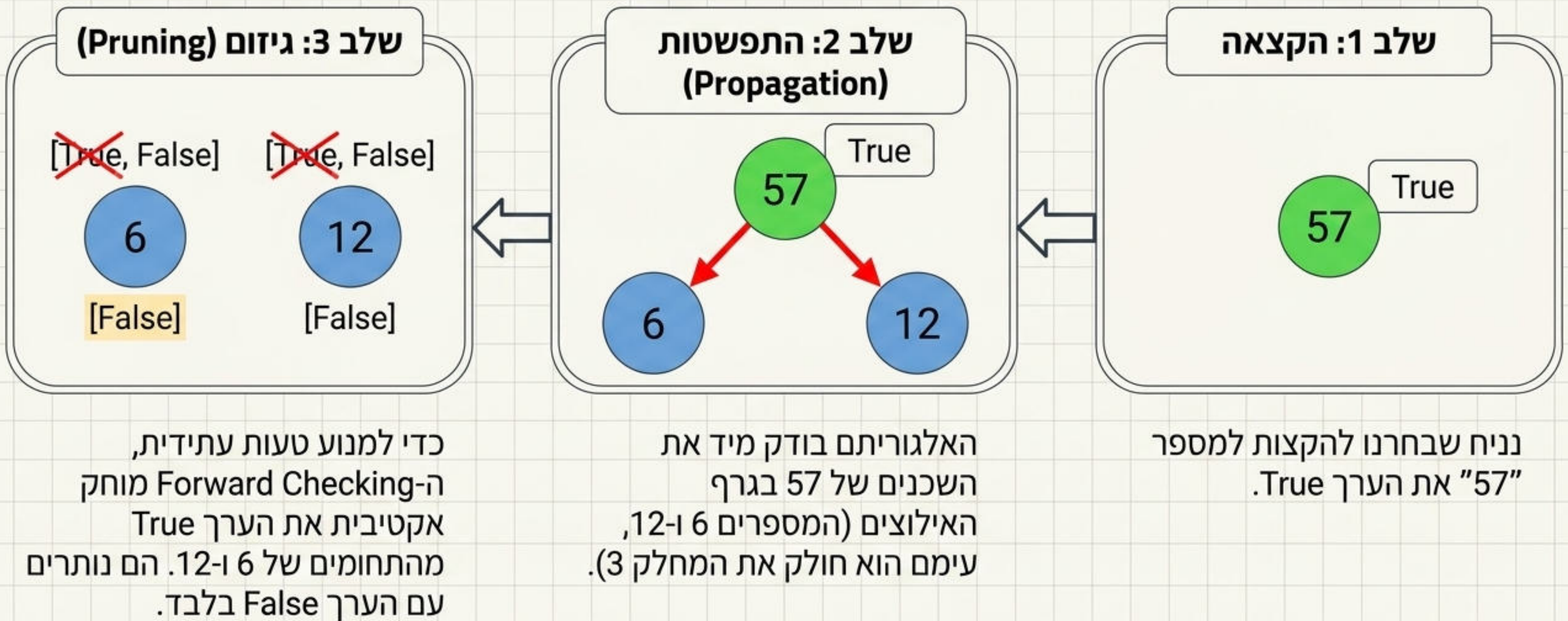
MRV



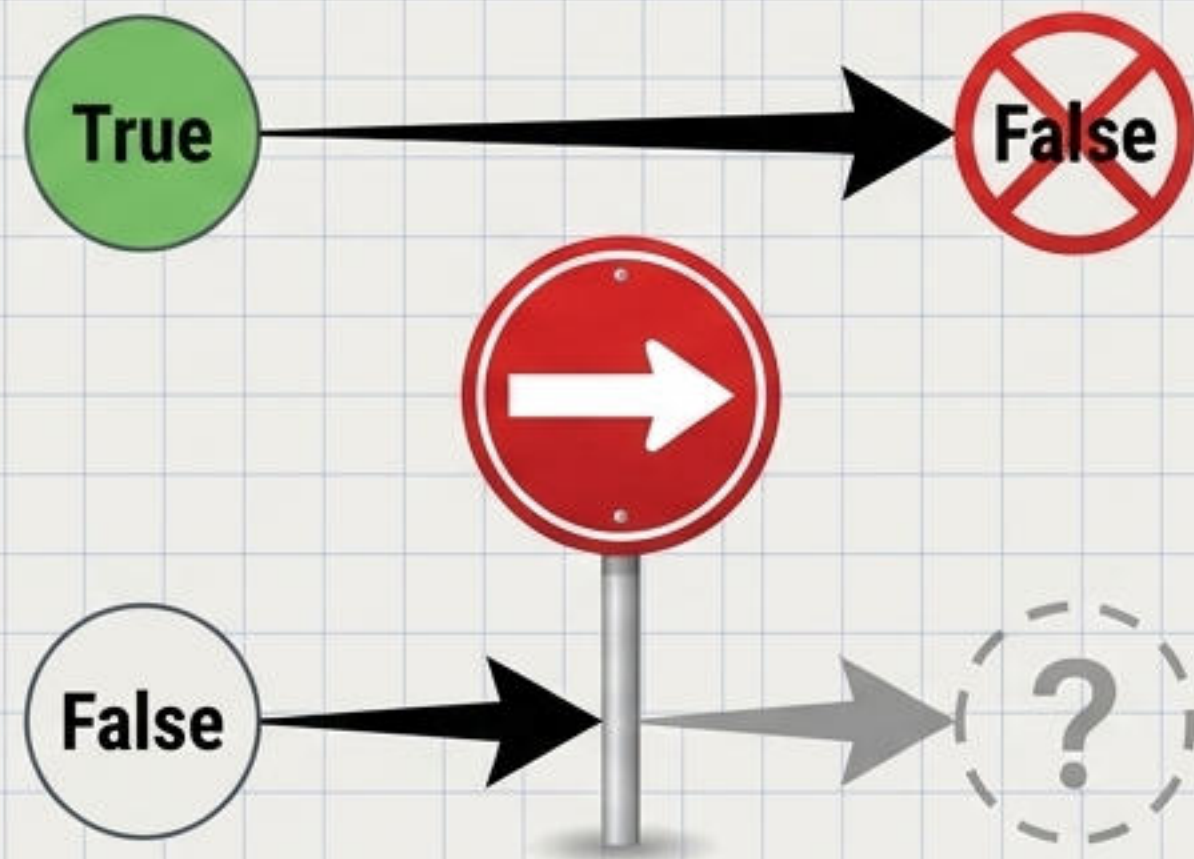
כישלון ה-MRV: בתחילת החיפוש, התחום של כל המשתנים זהה בגודלו (גודל 2: {True, False}).

מסקנה: יוריסטיקת MRV (המחפשת את התחום המצומצם ביותר) לא מספקת שום מידע הכרעתי ויש צורך בשובר שוויון.

מעקב דינמי: Forward Checking בפעולה



הגבולות של עקביות קשתות (Arc Consistency)



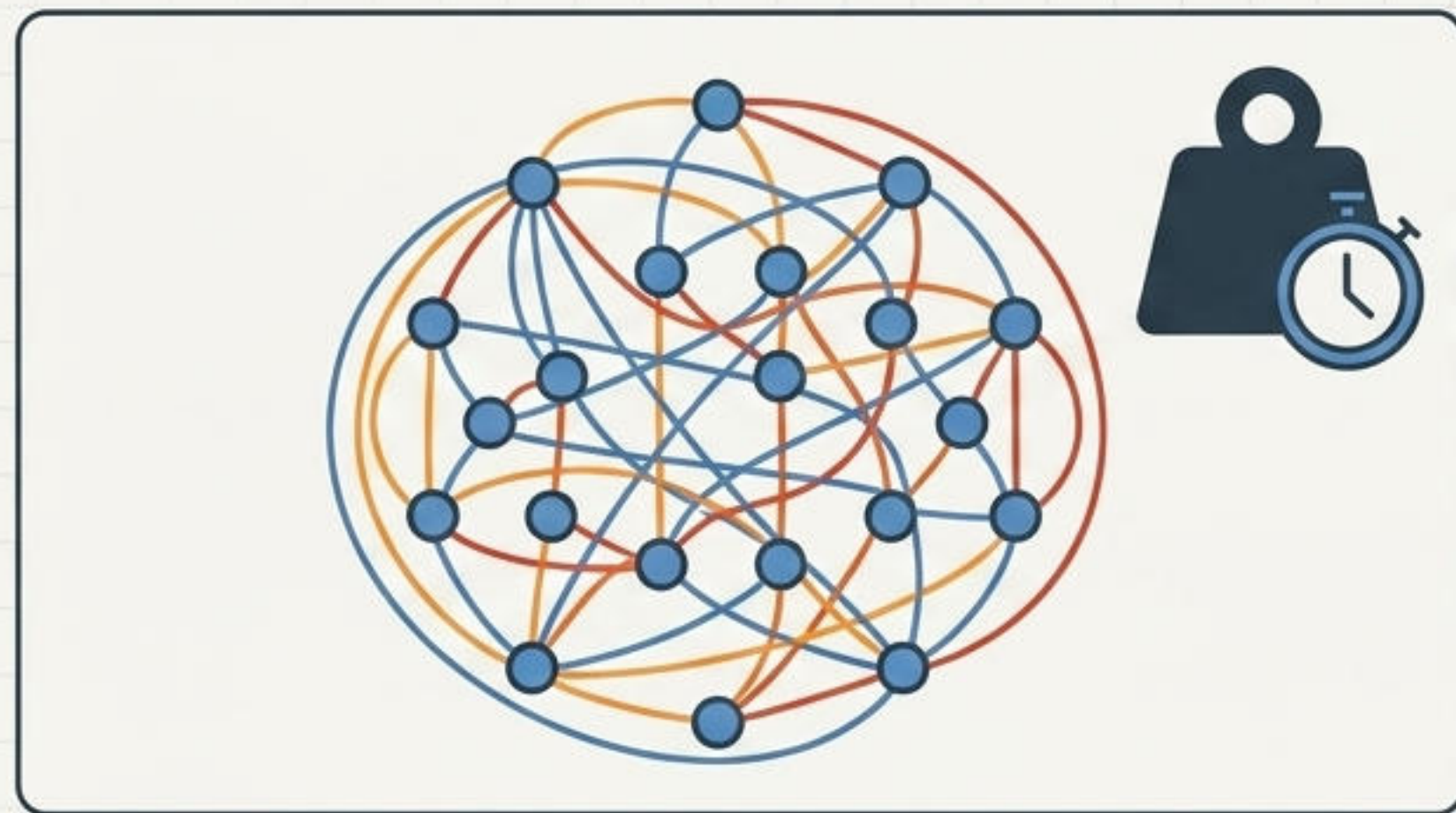
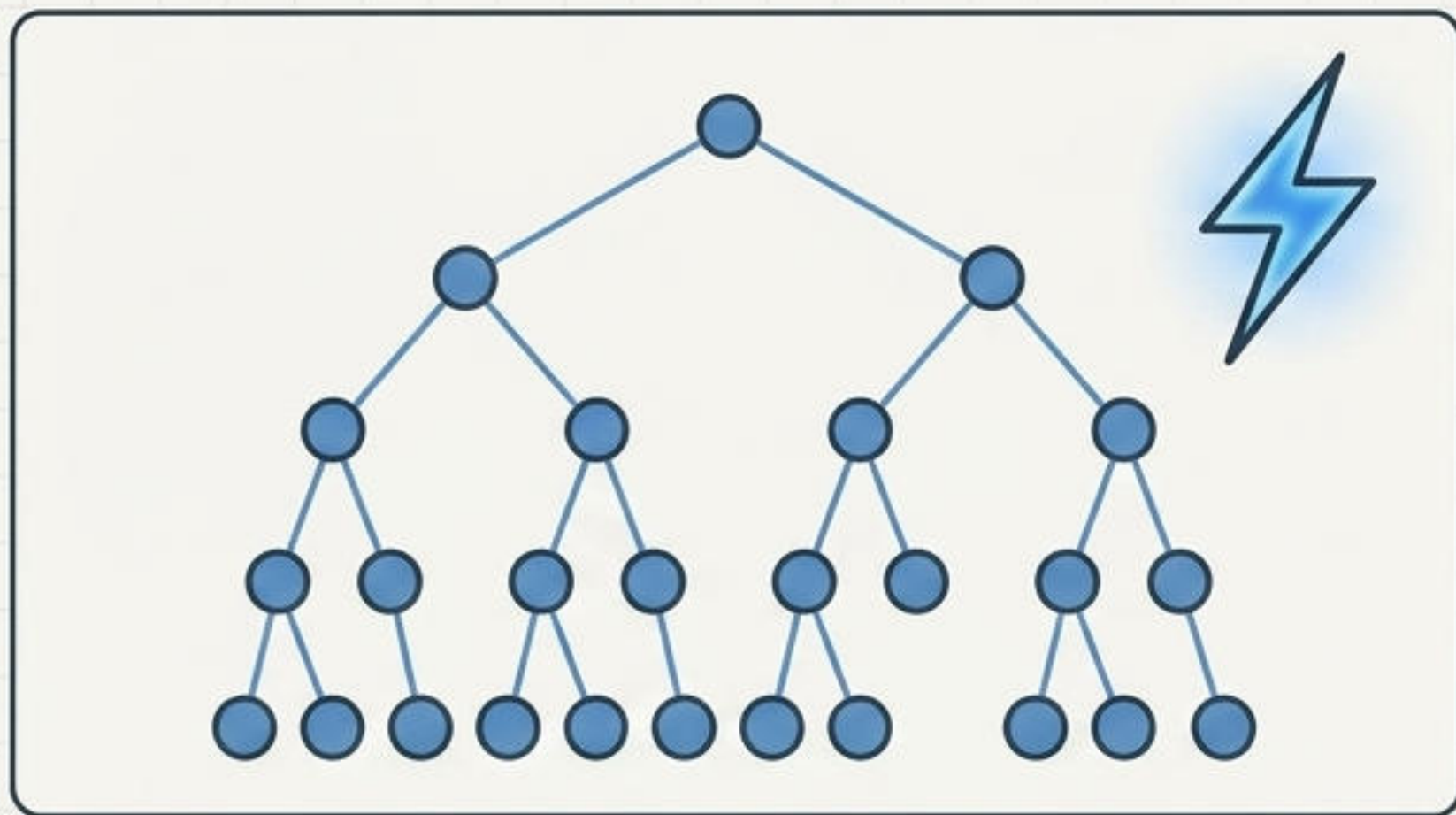
האם Arc Consistency יוסיף כוח ניבוי מעבר ל-Forward Checking בבעיה זו?

לא. במקרה זה, AC מיותר ולא שווה את זמן החישוב.

- התחומים מכילים רק שני ערכים: {True, False}.
- האילוץ פועל רק בכיוון אחד: אם X הוא True, שאר שכניו חייבים להיות False.
- אולם, אם פסלנו Trie ממשתנה, נותר לו רק False. מכיוון שהאילוץ לא אוסר על שכנים לקבל False יחד, אין להשמה זו כל השפעה על התחומים של שכניו. השרשרת נעצרת מיד.

מבנה הגרף קובע הכל: הכוח של עצים

מה ניתן ללמוד מייצוג חזותי של CSP?



אך, כפי שנלמד בשיעור - אם הגרף **מעוצב כעץ** (Tree-structured) **ללא מעגלים**, ניתן לפתור את הבעיה **בזמן ליניארי** ביחס למספר המשתנים.
זיהוי מבנה כזה מאפשר שימוש **באלגוריתמים מהירים** **בהרבה מ-Backtracking** סטנדרטי.

אם גרף האילוצים שלנו מכיל **מעגלים וקשרים** סבוכים (כמו בבעיית המספרים הזרים), הבעיה לרוב **קשה חישובית**.

סיכום: ארגז הכלים המלא לפתרון בעיות CSP

