

תרגול מס' 2 – שכבת קישור הנותנים



רואי' מת'ריה

□ טיפול שגיאות

- בדיקת זוגיות (Parity Checking).
- סכום הביקורת (Checksum).
- בדיקת יתרות מחזורית (CRC).

□ ALOHA

- שיטת ALOHA .Pure ALOHA
- שיטת ALOHA .*slotted*

□ תרגילים לדוגמא.

טיפול שגיאות



קוויים סטטיקליים / סינקלרים

רעיון כללי:

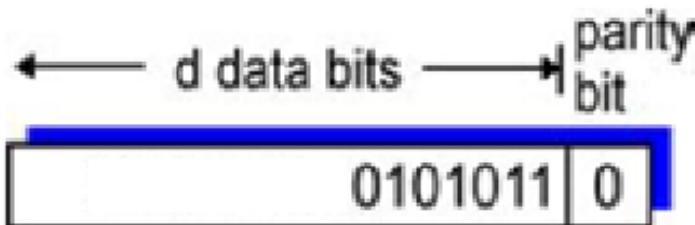
- ▣ השיטה העיקרית לזיהוי שגיאות היא לשלוח עם המידע המקורי איזשהו מידע עודף המכיל נתונים כלשהם על המידע הנשלח (redundant information).
- ▣ מידע עודף זה יעזר לנו לזהות טעויות בקלט שהגיע.
- ▣ במקרה הקיצוני נשלח כל בית פעמיים:
 - עבור הקלט 1011001010 נשלח את הודעה 11001111000011. אם מקבל יזהה איזשהו בית שלא הגיע ב"זוג" הוא ידע שיש שגיאה.
- ▣ שאלה: אופציה שנייה עבור הקלט היא להכפיל את הקלט ולשלוח את הודעה 10110011011001.
- ▣ תשובה: עבור מקרה שבו נאבד 5 ביטים רצופים (10110), באופציה הראשונה נאבד את המידע ואופציה השנייה לא!
 - חסרונות אופציה השנייה הבדיקה היא יותר איטית !
 - אבל גם עבור האופציה הראשונה זהו כמובן פיתרון מסורבל וקר בזמן שליחה וזכרון ולא יזהה שגיאה שתתרחש בשני ביטים רצופים! כמו כן הוא לא יכול לתקן שגיאה שהוא כי הוא לא ידע איזה בית נכנן ואיזה שגוי.
 - ניתן לכתוב כל בית 3 פעמים ואז בזיהוי שלשה לא זהה נדע נדע שיש טעות ואף נוכל לתקן אותה בסבירות גבוהה. גם שיטה זו יקרה מאד.

אֲקִיקָת זָמַנְיָה - One/Two Dimensional Parity Checking

□ **במימד אחד:** הרעיון הוא להוסיף בית אחד לכל בית (7 ביטים) באופן שיישלים למספר זוגי של אחדות. שיטה זו מאפשרת לנו לזהות כל מספר אי זוגי של שגיאות בכל בית. (אם מקבל רואה שיש לו מס אי זוגי של אחדות אז הוא יודע שיש 1/3/5... שגיאות במידע).

Single Bit Parity:

Detect single bit errors



□ מס זוגי של שגיאות לא יזהה
מכיוון שמס' האחדות יישאר זוגי

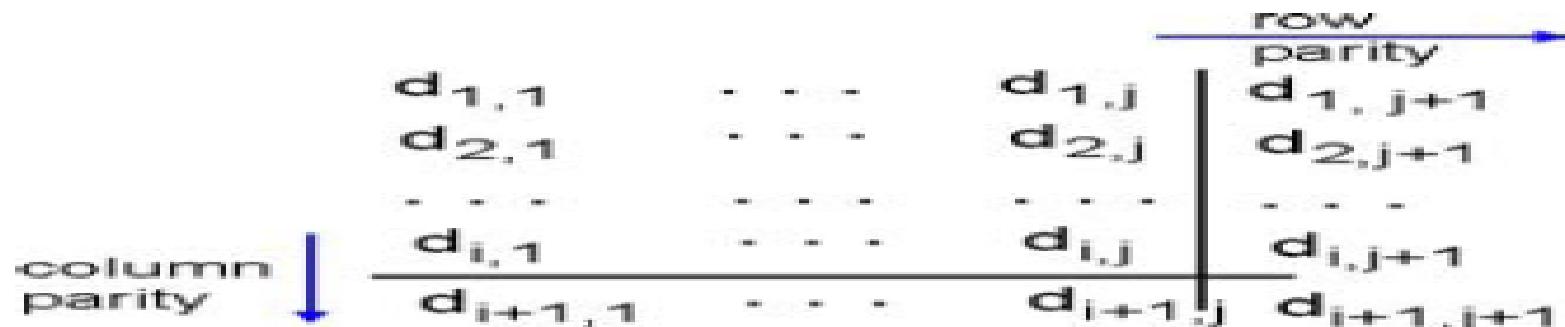
אֲקִיקָת זְוַיֵּת

בשני מידים אנחנו מוסיף בית נוסף על כל X בתים. בבית זה אנחנו רושמים ביטים המשלימים למספר זוגי של אחדות בסכום העמודות בתים שלפנייהם.

DATA	0 1 0 1 0 0 1	1
	1 1 0 1 0 0 1	0
	1 0 1 1 1 1 0	1
	0 0 0 1 1 1 0	1
	0 1 1 0 1 0 0	1
	1 0 1 1 1 1 1	0
Parity byte	1 1 1 1 0 1 1	0

Two Dimensional Bit Parity:

Detect and correct single bit errors



1	0	1	0	1	1
1	1	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0
<i>no errors</i>					

1	0	1	0	1	1
1	0	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0
<i>correctable single bit error</i>					

תיקוּן

שגיאה אחת

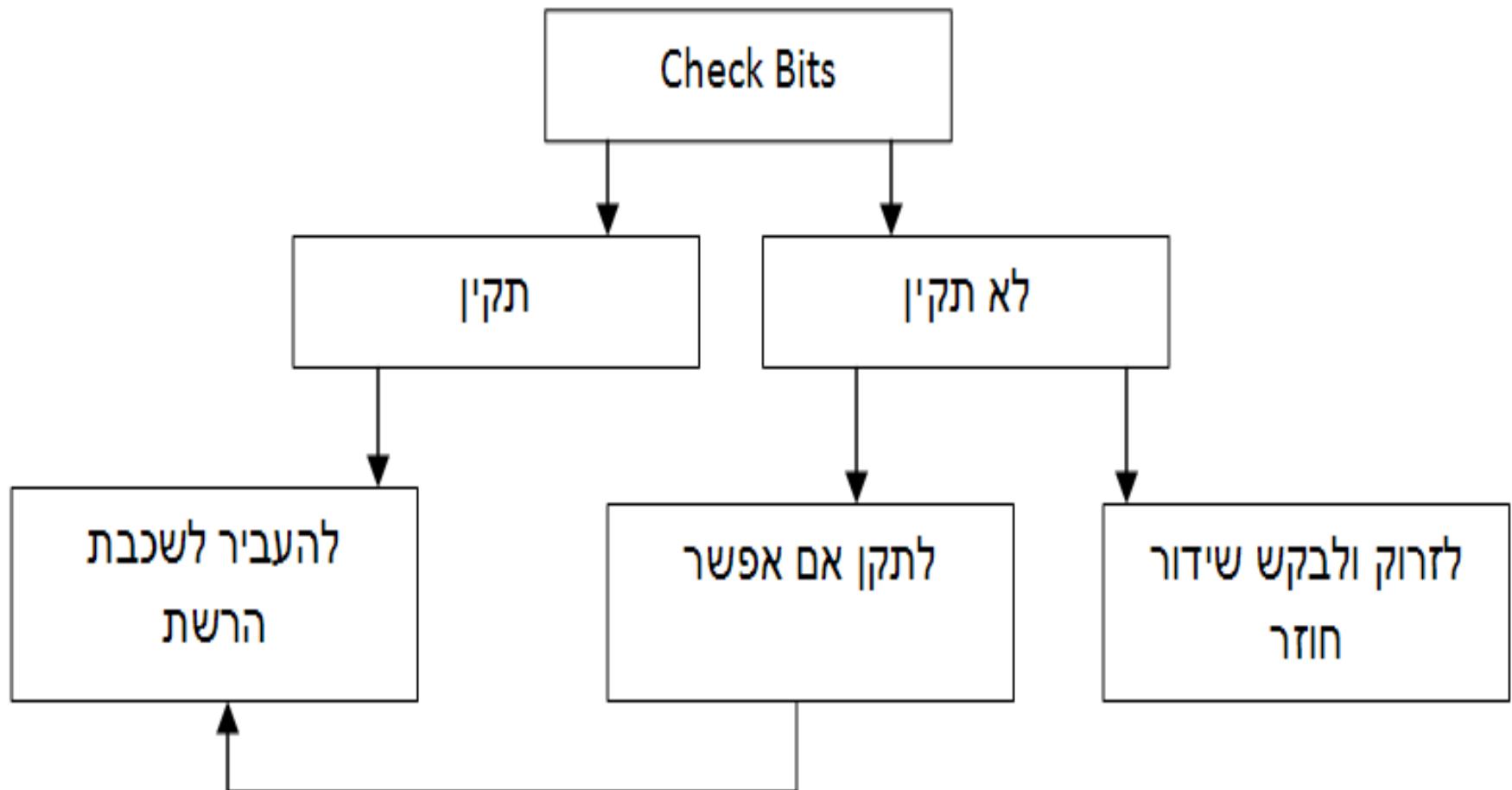
גלוּי

שגיאה אחת

שתי שגיאות (באותה שורה או עמודה)

שלוש שגיאות (תמיד נגלה אבל יתכן וייחשב לשגיאה אחת)

סיכום אפקט לוגית



סיכון איקאום – Checksum

- הרעיון: סוכרים את כל המילים שרצים להעביר ומעבירים גם את התוצאה (התוצאה של חיבור כל המילים נקראת checksum). מקבל מבצע את אותו חישוב על המידע ומשווה את התוצאה שקיבל ל-checksum. אם קיימת שגיאה במידע או אפילו ב-checksum שנשלח, שגיאה זו תגלה כי התוצאות יהיו שונות.
- קיימות המונן דרכיים למימוש:
 - המילים באורך 2^{16} .
 - מחברים אותם יחד בשיטת המשלים לאחד.
- תזכורת: בשיטת המשלים לאחד X – נחשב המשלים של X כЛОMER כל בית של X מתחOPER
- אם בחיבור היה carry אז הוא מתווסף לתוצאה.

קואקסיאליים ביצועים

וניח שהבלוק הבא מכיל 16 סיביות (ביטים) הנשלחים ומשתמשים בסיכום ביקורת (checksum) של 8 סיביות.

10101001 00111001

נחבר את המספרים תוך שימוש בשיטת המשלים ל-1 (one's complement).

10101001

00111001

Sum **סיכום** **11100010**

Checksum **סיכום ביקורת** **00011101**

התבנית אשר תשלח: **10101001 00111001 00011101**

זיהוי סיכון ויקאות (האפק)

עתה שהקולט מקבל את התבנית שנשלחה ואין שגיאות שנוצרו בשידור נניח

10101001 00111001 00011101

כאשר הקולט מבצע סיכום של שלושת הקטעים שנשלחו, הוא אמור לקבל רצף של 1-ים, ואחריו חישוב המשלים ל-1 נקלט רצף של 0-ים אשר יראה כי לא נוצרו שגיאות בשידור.

10101001

00111001

00011101

11111111

Sum

Complement

00000000 means that the pattern is OK.

אפקט מילוי אמצעים – CRC

- קוד נפוץ לגילוי שגיאות, המתווכם בהרבה מסיבית הציגות הוא **Cyclic Redundancy Code (CRC)**.
- לדוגמה **CRC** של 16 סיביות מאפשר לגלוות:
 - כל השגיאות של 2 ביטים.
 - כל מספר אי-זוגי של שגיאות.
 - כל השגיאות ברצף של 16 ביטים או פחות.
- 99.99% של רצפים של 17 ביטים או יותר.
- פרוטוקול קצת מרכיב אבל מאד טכני בין השולח למקבל.

אלגוריתם:

- **(X) D** – פולינום מדרגה χ המיצג $1 + \chi$ ביטים מהקלט. את הפולינום בונים ע"י הכפלת של כל ביט בערך 1 בחזקת מקדם הביט בקלט. דוגמה:

$$10011010 \rightarrow D(x) = X^7 + X^4 + X^3 + X^1$$

- **(X) G** – מחלק פולינומיyalי מדרגה 2.
 - ישן תבניות קבועות של מחלקים כאלה וכל אחת מהן בעלת תכונות מסוימות שטובות לזיהוי טוויות מסוימים.
 - דוגמה – ה Ethernet משתמש ב $G(X)$ מדרגה 32.

CRC-8:	$x^8 + x^2 + x^1 + 1$	דוגמאות נוספות:
CRC-10:	$x^{10} + x^9 + x^5 + x^4 + x^1 + 1$	
CRC-12:	$x^{12} + x^{11} + x^3 + x^2 + 1$	
CRC-16:	$x^{16} + x^{15} + x^2 + 1$	
CRC-CCITT:	$x^{16} + x^{12} + x^5 + 1$	
CRC-32:	$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$	

אלג'יט:

- **(X)T** – ההודעה אותה שולח השולח. הودעה זו היא בגודל $r + (1+h)$ ביט.
- **(X)T** מחלק בדיק בפולינום $(X)G$, ומכיל בתוכו למעשה את $(X)D$, ממנו יפענח המקבל את ההודעה המקורית.
- תזכורת – $(X)T$ מחלק ב $(X)G$ אם המעלת של $(X)T \leq$ מהעלת של $(X)G$.
- **(X)R** – השארית המתקבלת מחלוקת $(X)T$ ב $(X)G$. (אם יש צזו).

CRC-בְּאֵינִיֶּסֶת קָנָכִז

1. יצירת ההודעה ע"י השולח:

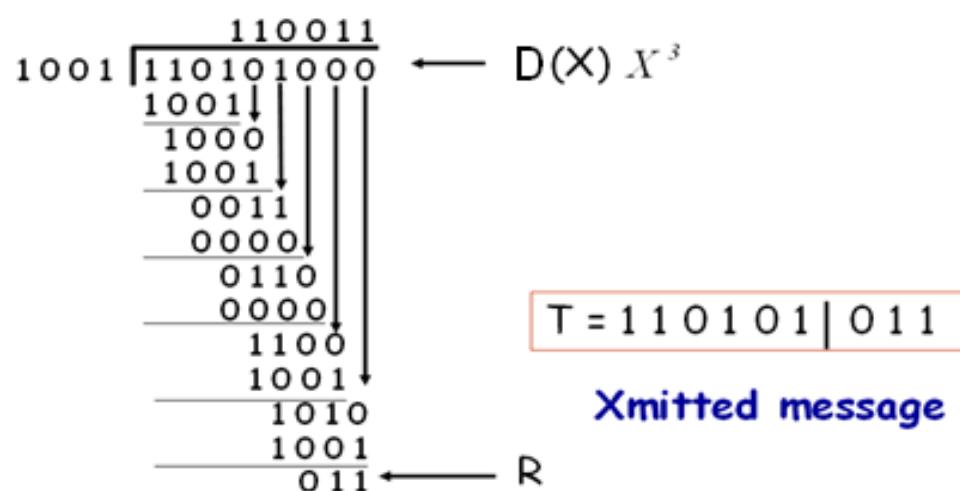
$T(X) = D(X)$ (adds r zeros at the message end) ■

r הוא המעלה של $G(X)$. ■

ב- :Divide $T(X)$ by $G(X)$ and find the remainder $R(X)$ ■

המעלה של $G(X)$ היא 3 ולכן מוסיפים 3 אפסים מימין ל $D(X)$ ויצרים את הרצף המוחולק: ■

Example: $D = 110101 \quad D(x) = X^5 + X^4 + X^2 + I$ ■
 $G = 1001 \quad G(x) = X^3 + I$ ■
 בחלוקת עושים XOR בין הביטים. ■
 טבלת XOR: ■



p	q	$p+q$
0	0	0
0	1	1
1	0	1
1	1	0

CRC – אינינס קאנס

ג

$$T(X) = R(X) - T(X)$$

מחסירים את השארית ומקבלים את $(X)T$ מוקן לשילחה.

$(X)T$ זהה מתחלק בדיק ב $(X)G$ ומכיל את $(X)D$ שלו יש מושרש מימין $(X)R$.

2. קבלת ההודעה:

המקבל מכיר את $(X)G$ כי הוא קבוע ומוסכם מראש.

הוא מחלק את $(X)T$ ב- $(X)G$ ומתוך $(X)R$ – השארית.

אם השארית היא 0 – משמע שאין שגיאות בהערכה.

אם השארית היא לא 0 – משמע שיש שגיאות בהערכה.

נחזיר דוגמא הבא:

$$\begin{array}{r} 110011 \\ 1001 \overline{)110101011} \\ \underline{1001} \\ 01000 \\ \underline{1001} \\ 00011 \\ \underline{0000} \\ 0110 \\ \underline{0000} \\ 1101 \\ \underline{1001} \\ 01001 \\ \underline{1001} \\ R = 0000 \end{array}$$

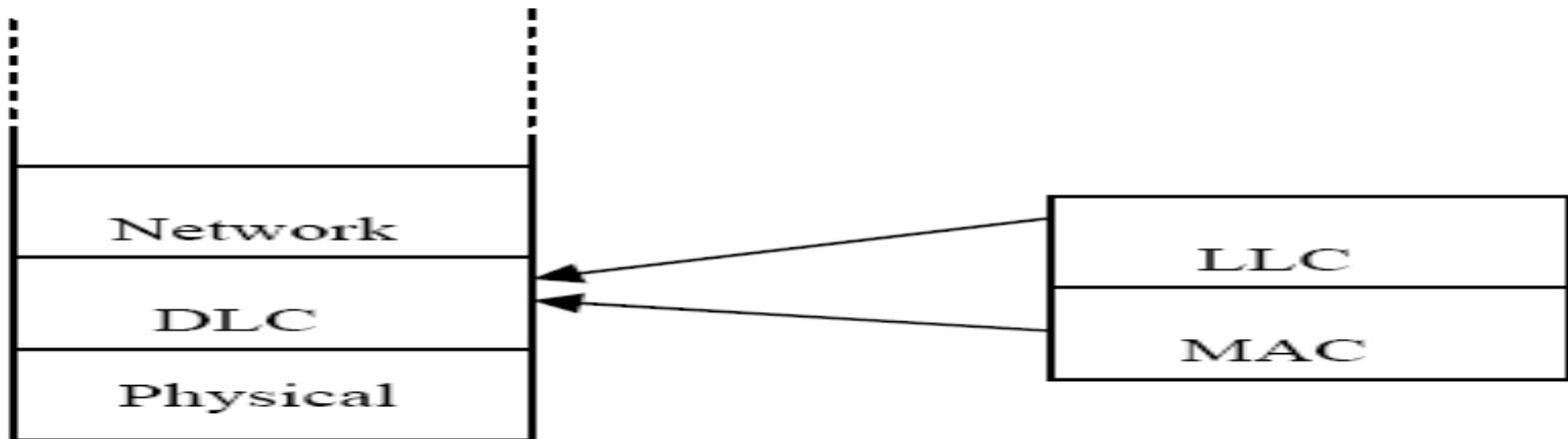
□ ע"מ להוציא מההודעה שהתקבלת את ההודעה המקורי על השולח להסיר את z הביטים הימניים (מה שהוא $(X)R$), והוא מותר עם ה – $(X)D$ המקורי – ההודעה!

ALOHA



אנו

- תחילת נמקם את הנושא במודל השכבות:



- שכבה ה-*DLC* (Data Link Control Layer) - מורכבת משתי רמות:
 - שכבה ה-*LLC* (Logical Link Control) - האחראית למעבר תקין של חבילות מהמקור ליעד. מספור החבילות, הגעתם לפי הסדר וכו'.
 - שכבה ה-*MAC* (Medium Access Control) - בקורת הגישה לערוץ משותף. כאשר ערוץ התקשרות משותף יש צורך בקורת גישה לערוץ, וכך נדון בשיטת *ALOHA*.

אפק כ传达

- מספר צרכנים אינסופי $\infty \rightarrow N$
- שתי הודעות המשודרות בפרק זמן חופפים מתנגשות (collision) והתקלה מתגלת ע"י התחנות בסיום שידור הודעה.
- פרט להתנגשיות אין שגיאות נוספות בערוצ (אילו היי, הדבר היה נפתר ברמת ה-LLC).
- תהליך הופעת החבילות לשידור (הודעות חדשות + שידורים חוזרים) הוא פואסוני עם ממוצע G חבילות ליחידת זמן.
- כל החבילות באורך קבוע ונקבע ייח' זמן צמן שידור חבילה אחת.
- **תזכורת :** נוסחת ההתפלגות הפואסונית:

$$\Pr(k, \lambda, t) = \frac{(\lambda t)^k}{k!} e^{-\lambda t}$$

- מתארת הסתברות k מופעים של תהליכי הפואסוני אם פרמטר λ בפרק זמן t .

Pure ALOHA סינט

- כל תחנה משדרת הודעה מתי שהיא רצאה, ככלומר צומת משדר מיד כשייש לו חבילת לשידור.
- במקרה של התנגשות יש לשדר את ההודעה מחדש.
- בחירת מועד השידור הבאה מתבצעת בצורה אקראית.
- **נחשב תüberה (נצחיות) עבור המודל הבא:**
 - זמן מסגרת – frame time – כמות הזמן הנדרשת להעברת מסגרת באורך קבוע.

$$זמן מסגרת = \frac{\text{אורך מסגרת (בטים)}}{\text{קצב שידור}}$$

- מס' הודעות המידע הנשלחות בפרק זמן T או קצב ניסיונות השידור: G
- קצב הגעת הודעות חדשות וישנות הוא פואסוני:

$$P_k(t) = \frac{(G)^k}{k!} e^{-G}$$

כאשר k הוא מספר המסגרות.

Pure ALOHA סיבת

- נחשב את ההסתברות להצלחת שידור בפרק זמן:

$$P_{\text{success}} = P\left(\frac{\text{לא תשלוח אף הודעה לרויין}}{2T \text{ בפרק זמן}}\right) = P\left(\sum_{k=0}^{\infty} \text{אף הודעה}\right) = e^{-2G}$$

- כל שידור במהלך תקופת הפגיעה יגרום ל - *collision*
- קצב הייצור מסגרות במוחץ, השווה למספר לנצלות של התעבורה:

$$\eta = S = Ge^{-2G}$$

- המקיים מתקובל עבור $G = 0.5$ ושוה $S = 0.18$

סלאט אולואה-slotted ALOHA

הפרוטוקול:

- כמו ALOHA רק שבנוסף קיימן מגנון לחירוץ הזמן, כלומר הזמן מחולק למרוחכי זמן שווים של Zeit' זמן (*Slots*).
- צומת שיש לו חבילה (חדרה או שידור חוזר) לשידור משדר אותה בתחילת חריצ הזמן הקרחב ביותר, כלומר השידור מתבצע רק מהתחלת *slot*.

чисוב תüberה עבור מקרה זה

- תקופת הפגיעה מצטמצמת ל- T (התחנה המשדרת אינה רשאית לשלוח מידע כל הזמן. ונאלצת לחכות לתחילת מקטע הזמן הבא)

$$P_{\text{success}} = P_{N=0}(t = T) = e^{-G}$$

$$\eta = S = G \cdot e^{-G}$$

$$S = e^{-1} \approx 0.36 \quad G = 1$$

תורת התקשורת

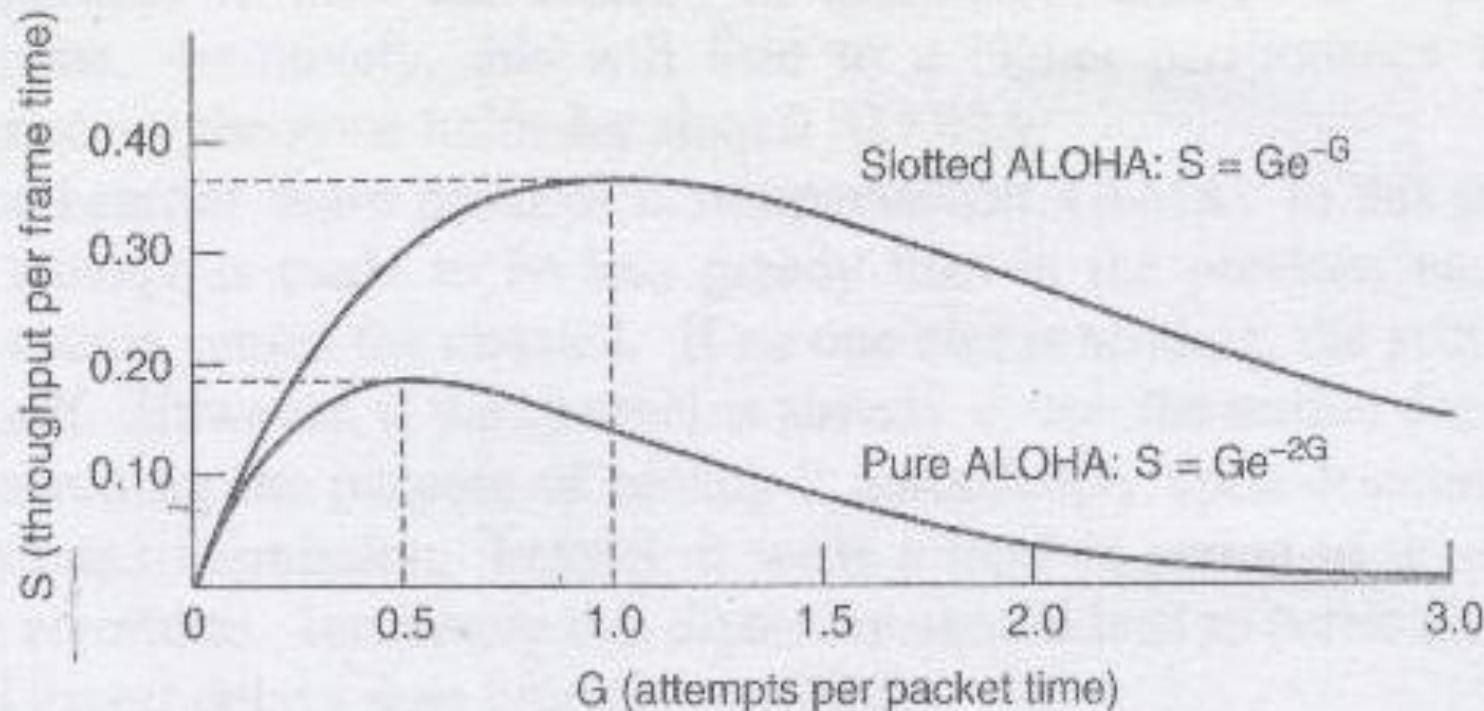


Fig. 4-3. Throughput versus offered traffic for ALOHA systems.

סלאט אולואה Slotted ALOHA

- N - תחנות.
- מס' חבילות (חדש ו חוזרת) ב ממוצע המופיעות בתחנה i במשך זמן חריצ (T יחס' זמן).
- S_i - קצב היוצרות מסגרות ב ממוצע או מס' חבילות ב ממוצע שתחנה i משלילה לשדר במשך זמן חריצ (T יחס' זמן).
- הנחת המודל : במצב יציב , תהילר הופעת חבילות (חדש + חוזרת) בחריצ הוא ברמוני עם פרמטר G_i

$$S_i = \left(\frac{\text{הסתברויות כי}}{\text{תחנה } i \text{ לא ישדרו}} \right) \cdot \left(\frac{\text{הסתברויות כי}}{\text{פרט } i \text{ לא ישדר}} \right) = G_i \prod_{j=1, j \neq i}^N (1 - G_j)$$

$$\left[\frac{\text{חbillות}}{\text{slot}} \right] \text{ קצב שידור בראשת } G = \sum_{i=1}^N G_i$$

$$\left[\frac{\text{חbillות}}{\text{slot}} \right] \text{ התובלה בראשת } S = \sum_{i=1}^N S_i$$

כיתת slotted ALOHA סוף תחרות

$$G_i = \frac{G}{N}$$

□ נבחן מקרה פרטי (מעניין)

$$S_i = \frac{S}{N}$$

בו כל התחנות זהות:

$$S_i = G_i \prod_{\substack{j=1 \\ j \neq i}}^N \left(1 - G_j\right) = \frac{G}{N} \left(1 - \frac{G}{N}\right)^{N-1}$$

$$S = N \cdot S_i = G \left(1 - \frac{G}{N}\right)^{N-1}$$

□ עבור $\infty < N$ נקבע זהה בדיקת במקרה של **slotted ALOHA** עבור ∞ תחנות, והמקרים מתקובל עבור $G = 1$.

$$\left(1 + \frac{1}{n}\right)^{\frac{n}{1}} \rightarrow e$$

$$\left(1 - \frac{1}{n}\right)^{\frac{n}{1}} \rightarrow e^{-1}$$

$$\Rightarrow G \left(1 - \frac{G}{N}\right)^{(N-1)} = G \left[\left(1 - \frac{G}{N}\right)^{\frac{N}{G}} \right]^{\frac{G}{N}(N-1)} \rightarrow G e^{-G}$$

$$\frac{G}{N}(N-1) \rightarrow G$$

$$\left[\left(1 - \frac{G}{N}\right)^{\frac{N}{G}} \right] \rightarrow e^{-1}$$

איך החלטה

את מכיר ות שולח Slotted Aloha

- נתונה רשת Slotted Aloha עם מספר אינסופי של תחנות. אורך כל חרץ הוא שנייה אחת. אורך כל הודעה היא 0.9 שנייה בהסתברות $P=1.1$ שנייה בהסתברות ($P=1$).
- תוך שימוש בסימן $\Pr(k, \lambda, t)$ כדי לציין את ההסתברות ל- k מופעים, ב- t יחידות זמן כאשר התפלגות המופעים היא פואסונית עם פרמטר λ , כפונקציה של מספר מופעים מסווץ בחרץ (G) בלבד, נוענה על השאלה: מה הניצולות של הערז עבור $P=1$ (כל ההודעות הן באורך 0.9)?

$$\Pr(k, \lambda, t) = \frac{e^{-\lambda t} (\lambda t)^k}{k!}$$

מצולות: מספר חנויות המועברות לשכנת הרשות בפרק
זמן השווה לו מני שידור מסגרת אחת

- תשובה:
- כל ההודעות הן באורך 0.9 ולכן ניתן להתגש רק עם הודעות מסוימו חרץ. בחרץ הנוכחי ישדרו תחנות שקיבלו החלטה לצאת לשידור בחרץ הקודם. נסמן ב- k את מס' המסדרות המשודרות בחרץ בהצלחה.

$$k = 1 \cdot \Pr(\text{successful slot}) = \Pr(1, G, 1) = Ge^{-G}$$

הסביר:

$$\Pr(k = 1, \lambda = G, t = 1) = \frac{e^{-G \cdot 1} (G \cdot 1)^1}{1!} = Ge^{-G}$$

אלאק המכוי שוטט Slotted Aloha סילוט

□ נחשב את הניצולת ע"פ ההגדרה:

$$\begin{array}{c} \boxed{1 \text{ slot} - Ge^{-G}[\text{frames}]} \\ \rightarrow \\ \boxed{T_i - S[\text{frames}]} \end{array} \quad \begin{array}{c} \rightarrow \\ \boxed{1[\text{sec}] - Ge^{-G}[\text{frames}]} \\ \boxed{0.9[\text{sec}] - S[\text{frames}]} \end{array} \quad \rightarrow S = 0.9 \cdot Ge^{-G}$$

תרגולים לדוגמה



מכניזם 1

- נתון דגל 01111110, משתמשים בשיטת ה- bit stuffing (אחרי 5 אחדות רצופים מכניםם 0) עבור זרם הביטים הבא:
 - ממש את השיטה על זרם הביטים הבא:
 - 011011110011111101011111111101111010
- הנח שזרם הביטים הבא מגיע:
01111101111101100111110011111011100011111010111110
 - ▣ הוציא את ה-bit stuffing והראה היכן הדגלים.
- נתון דגל 01111110, משתמשים בשיטת ה- bit stuffing (הנח שבמקרים שאחרי 5 אחדות רצופים מכניםם 0, משתמשים בשיטה הבא: אחרי 111110 מכניםם 0).
הנח שזרם הביטים הבא מגיע:
.011011110111111011111010111110
 - ▣ הוציא את ה-bit stuffing והראה היכן הדגלים.

כטלוון מכם 1

פתרונות סעיף א':

dagel: 01111110 (אחרי 5 אחדות רצופים מכנים 0)

מקור: 011011111001111110101111111101111010

תוצאה:

01111110

0110111110001111101010 111110111111001111010

דגלים מסומנים באדום

תוספת אפסים מסומנת בצהוב

פתרונות סעיף ב':

מקור: 011111101111101100111110011111011111011000111111010111110

נסמן את הדגלים:

011111101111101100111110011111011111011000111111010111110

דגלים מסומנים באדום

11111110011111011111111111100

תוצאה:

פתרונות סעיף ג':

dagel: 01111110 (אחרי 5 אחדות רצופים מכנים 0)

מקור: 0110111110111111101111110101111110

נסמן את הדגלים:

01101111101111110111110101111110

דגלים מסומנים באדום

1 11111

תוצאה:

טכני 2

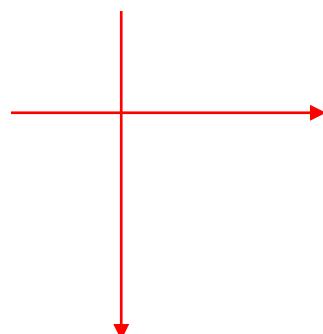
- נתון מידע באורך של D סיביות, כאשר מחלקים את המידע ל- τ שורות בעלות אורך של τ סיביות (הנחו כי D ניתן לחלוקת ב- τ וב- τ ללא שארית). מוצע לקודד את המידע ע"י סיבית זוגיות שמחושבת עבור כל שורה וסיבית זוגיות שמחושבת עבור כל עמודה.
- א- הוכחו שכל שגיאה אחת יכולה להתגלות ואף להיות מתוקנת ע"י הקוד המוצע.
 - ב- מצאו דוגמא של שתי שגיאות שניתנות לגילוי. האם כל קומבינציה של שתי שגיאות ניתנות לתקן ?
 - ג- מהי התקורתה של הקוד המוצע, וכי怎 ניתן להקטין את התקורתה הזו למינימום. הצעו ביטוי מתמטי המבטיח תקורתה מינימלית.

כטלוון מצלים 2

□ פתרון סעיף א':

כל שגיאה בודדת שמרתחשת במטריצת הנתונים גורמת לשגיאה בביט הזוגיות של שורה אחת ועמודה אחת. החיתוך בין השורה לעמודה מצין את הביט השגוי וכך ניתן לתקןו.

לדוגמה :


$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & X & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

כטלוון מרכז'יף 2

□ פתרון סעיף ב':

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ X & 0 & X & 0 \\ 0 & 0 & 0 & 0 \\ \downarrow & 0 & 0 & 0 \end{pmatrix}$$

דוגמא להצלחת האלגוריתם רק בגילוי שגיאה:

לא כל קומבינציה ניתנת לתקן , למשל :

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ X & 0 & X & 0 \\ 0 & 0 & 0 & 0 \\ \downarrow & 0 & 0 & 0 \end{pmatrix}$$

במקרה שמתואר באIOR ניתן רק לגלות את המיקום של העמודות בהם קיימת השגיאה, אך לא ניתן לקבל את המיקום המדויק של השגיאה.

אתגר מס' 2

פתרון סעיף ג':

התקורה של הקוד המוצע הינה: $i+j+i$, כאשר i סיביות מגיניות על העמודות ו- j סיביות מגיניות על השורות, סיבית נוספת משותפת להגנה של העמודה והשורה האחרונות במטריצה. על מנת לקבל תקורה מינימלית צריך לבצע: $\min \{i + j + 1\}$ תחת ההנחה ש- $D = j \cdot i$.

מבצע זאת ע"י הצבה גזירה והשוואה ל- 0 ונקבל:

$$\begin{aligned} \min_{i,j=D} \{i + j + 1\} &\Rightarrow \min_i \left\{ i + \frac{D}{i} + 1 \right\} = \left\{ 1 - \frac{D}{i^2} \right\} = 0 \Rightarrow i^2 = D \Rightarrow i = \sqrt{D} \\ &\Rightarrow \min \{i + j + 1\} = \min \left\{ \sqrt{D} + \sqrt{D} + 1 \right\} = 2\sqrt{D} + 1 \end{aligned}$$

נוסחה זו נcona בהנחה שניית להוציא שורש שלם ל- D . במקרה ולא ניתן, כלומר מתקבל מספר שאינו שלם, נחפש את המספר השלם הראשון שהci קרוב ל- \sqrt{D} שగודל או קטן מ- \sqrt{D} שמתחלק ב- D ללא שארית. המטרה לקבל מטריצה שתשאף להיות ריבועית.

תרגיל 3

□ אנו רוצים לשדר את הודעה 1100100100101011 באמצעות CRC8 ($G=100000111$).

- א- מהי מילת הקוד שתשדר ?
- ב- הניחו כי הביטים השלישי, הרביעי והחמישי בספירה משמאל התחפכו. מה תהיה תוצאה בדיקת CRC מצד מקבל ?
- ג- איך יודע המქבל כי הייתה שגיאה ?

3 מטלות

$$\begin{array}{r}
 1100101101011011 \\
 100000111 \overline{)110010010010101100000000} \\
 \underline{100000111} \\
 0100101010 \\
 \underline{100000111} \\
 000101101101 \\
 \underline{100000111} \\
 00110101001 \\
 \underline{100000111} \\
 0101011101 \\
 \underline{100000111} \\
 00101101000 \\
 \underline{100000111} \\
 00110111100 \\
 \underline{100000111} \\
 0101110110 \\
 \underline{100000111} \\
 00111000100 \\
 \underline{100000111} \\
 0110000110 \\
 \underline{100000111} \\
 R = 010000001
 \end{array}$$

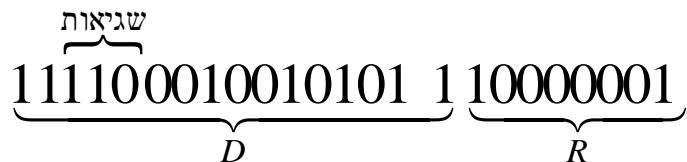
פתרונות סעיף א': □
 נתון: CRC8
 הודעה 1100100100101011
 $G = 100000111$
 נחשב את R כאשר $r=8$

$$R = \frac{D \cdot 2^r}{G}$$

• מילת הקוד לכן תהיה:
 $\underbrace{1100100100101011}_{D} \underbrace{10000001}_{R}$

כטלוון מרכזים 3

פתרון סעיף ב':



□ המקבל קיבל את המידע הבא:

□ המקבל יודע את האורך של *R* ומכאן כמובן גם את האורך של *D*,

ויחשב את אותו חישוב

$$R = \frac{D \cdot 2^r}{G}$$

□ אם תהיה תוצאה שונה עבור *R* יידע שעשה טעות, *G* כאמור ידוע ל-2 הצדדים.

111100111110000

100000111)11110010010101100000000

100000111

0111001010

100000111

0110011011

100000111

0100111000

100000111

000111111101

100000111

0111110101

100000111

0111100100

100000111

0111000110

100000111

0110000010

100000111

0100001010

100000111

0000011010 000

R = 11010000

כטלוון מרכזים 3

פתרון סעיף ג':

- כאמור בಗלל השגיאה קיבלנו תוצאה שונה שונה עבור R , המקיים משווה את ה- R שקיבל מהשולח לעומת ה- R שהчисב בעצמו ובגלל שהם לא שוויים ידע שהיה שגיאה בהודעה.

תיכונית 4

- א- איזה בעיה בא לפטור אלגוריתם ALOHA ?
- ב- מה ההבדל בין אלגוריתם מסוג Partitioning ? Random Access
- ג- הסבוו כיצד פועל אלגוריתם Pure ALOHA . מה החסרון והיתרון העיקריים ?

כטלוון מרכזים 4

פתרונות סעיף א':

- בא לפטור את בעיית ה- MULTI ACCESS CHANNEL בו יש ערוץ שידור יחיד אבל משדרים רבים.
- אם יותר מתחנה אחת משדרות בו זמנית יש היתנגשות והרס של השדר.
- הבעיה היא לתת אלגוריתם מבוזר אשר יתאים את תזמון השדרים של התchnות ככה שהיה fairness ניצולת (יעילות או efficiency) מקסימלית של הערז.

כטלוון מרכזים 4

פתרונות סעיף ב':

- ב- **CP** (Channel Partitioning) מחלקים את הערזץ ל- N תת-ערוצים אשר מאפשר שידור בו זמני בלי היתגשויות. דוגמאות הם TDMA ו-FDMA.
- ב- (Random Access) **RA** כל תחנה משדרת בערך מתי שבא לה ואם מזיהה היתגשות משדרת שוב בהסתברות מסוימת. דוגמאות הם slotted\pure CSMA\CD ו-ALOHA.

כטלוון מרכזים 4

פתרון סעיף ג':

- כישיש לתחנה מה לשדר היא משדרת מיידית. שייהו התנגשויות, כאשר המוגשות המתנגשות ירטסו. קיים משוב מהערוץ, ולכן התחנה המשדרת יודעת לזהות התנגשות על ידי האזנה לערוץ. במידה והמוגשת נהרסה, המשדר מכחכה פרק זמן אקראי, ושולח את המוגשתשוב.
- היתרון של ALOHA Pure שהוא מבוזר לחלווטין ולא דורש שום סינכרון מראש.
- החסרון שהיעילות מאד נמוכה בשל הרסתברות הגדולה להtanגשות.