

מושגים:

- מידע **non-isochronous** - עם *packet switching*, מידע מסורתי, דף *e-mail, web*. מופיע בפרצים (זמן וגודל לא ידוע), שהייה מותרת, אין רגישות לאיבוד (ניתן לשדר מחדש) - רשתות מחשבים.
- מידע **isochronous** - עם *circuit switching* קול המועבר בשיחות טלפון, מופיע בקצב ידוע מראש, קיימת שהייה (קצרה) מקסימאלית, ולכן בדרי"כ לא מתאפשר שידור חוזר - רשתות טלפון.
- **Piggybacking** - במקום לשלוח מסגרת תגובה מיוחדת, מכניס המקבל את *RN* למסגרת המידע שהוא שולח (אם הוא שולח) אל הצד השני.

מודל השכבות:

- 5) **Application (FTP)** : end to end protocol
- 4) **Transport Layer (TCP,UDP)**
- תפקידה לוודא שהקובץ יגיע בצורה תקינה ליעד (לא כל החבילות עוברות באותו מסלול, ולא תמיד שכתב הקו מבטיחה אמינות - ולכן אי אפשר להסתפק באמינות שכתב הקו)
- מקבלת הודעה מהאפליקציה ומחלקת אותה למספר סגמנטים. **end to end protocol**

- 3) **Networks (IP)**
- תפקידה לנתב חבילה למחשב היעד, דרך מחשבי ביניים (routers).
- מקבלת סגמנט, ומוסיפה לו שדות שיוצרים חבילה. **Source Routing** - תחנת המקור מוסיפה ל-*header* את כל המסלול
- **Distributed Routing** - תחנת המקור מוסיפה ל-*header* את היעד.

- 2) **Data Link (DLC)** (*ARQ: stop & wait, GBN, SR*): מיומש בחומרה
- תפקידה לשדר הודעה ללא שגיאה לתחנת סמוכה.
- מקבלת חבילה ומוסיפה (*trailer & header*) שיוצרים מסגרת.
- בכל הערוצים - הוספת *delimiters* לזיהוי מסגרות.
- בערוצים רועשים - שימוש בקוד לגילוי ותיקון שגיאות. (טבעת, אסימון, *Ethernet (MAC sub-layer)*)
- דואגת שלא יהיו התנגשויות בערוץ פיזי משותף מרובה משתמשים.

- 1) **Physical** (רדיו, אופטי, רדיו)
- ייצוג ביטים, קביעת קצב שידור, זיהוי תחילת וסיום מסגרת, שידור באופן אמיין ביותר רצף של סיביות ממחשב אחד אל מחשב שני מחובר אליו באמצעות קו תקשורת (חוטני או אלחוטי).
- אמינות בשכבת התובלה היא חובה. אמינות בשכבת הקו היא אופציונלית - בקו עם סיכוי נמוך לשגיאה ניתן לוותר עליה.

גילוי ותיקון שגיאות בשכבת הקו

- הסתברות לשגיאה - הסתברות שיתחפף ביט
- כדי **לגלות** d שגיאות במילה, צריך שמרחק המינג בין כל שתי מילים יהיה לפחות $d+1$.
- כדי **לתקן** עד d שגיאות במילה, צריך שמרחק המינג בין כל שתי מילים יהיה גדול מ- $2d$.
- **קוד המינג לתיקון שגיאה אחת:** עבור m סיביות מידע r -1 סיביות הגנה $(m+r+1) \leq 2^m$ נובע מהדרישה $2^m \geq n = m + r + 1$
- **לוגיקה:** על כל אינדקס מגנות הספרות שהן פירומו הבינארי. סה"כ בפירוק ובספרה המניגה יהיה מס' זוגי של 1-ים.
- תיקון רצף של שגיאות ע"י שידור ההודעה טור אחר טור (פרץ שגיאות באורך טור)

Digit num.	1	3	5	7	9	11	13	15
1	1	3	5	7	9	11	13	15
2	2	3	6	7	10	11	14	15
4	4	5	6	7	12	13	14	15
8	8	9	10	11	12	13	14	15

* גילוי מול תיקון: תקורה של תיקון שגיאות גדולה מתקורה של גילוי שגיאות ושליחה חוזרת. לכן לרוב נעדיף גילוי בלבד.

- **Parity check** - גילוי מספר אי זוגי של שגיאות.
- גילוי מס' אי זוגי (לפעמים זוגי) ורצף של שגיאות שאורכן לא גדול מאורך שורה, **תיקון שגיאה בודדת**, ע"י הוספת סיבית זוגיות לכל שורה וטור.
- **קוד checksum:** סכום של כל הודעה בסופה.

קוד CRC:

- כללי: הודעות מתפרשות כפולינומים מעל בסיס 2: $1001 \equiv x^3 + 1$
- חיבור וחיסור מקבילים ל-*xor*: $(x^3 + 1) \pm (x^3 + x^2) = x^2 + 1$
- בוחרים פולינום יוצר $G(x)$ מדרגה g . רוצים לשדר את $M(x)$. המילה שתשודר: $T(x) = x^g M(x) - ((x^g M(x)) \% G(x))$
- שתתקבל היא: $T'(x) = T(x) + E(x)$ כאשר $E(x)$ שגיאה כלשהי
- נבדוק אם $T'(x) \% G(x) = 0$, אם לא יש טעות ונורוק את המסגרת, אחרת נקבל (לא לשכוח להוריד את g הביטים התחתונים).
- G עם שני איברים או יותר יגלה שגיאה בודדת.
- שתי שגיאות בביטים i, j יתגלו עבור $G(x) = (x^i - x^j) \% G(x)$.
- מס' אי זוגי של שגיאות יתגלה אם $G(x)$ מתחלק ללא שארית ב- $x+1$.
- פרץ שגיאות $x^r + \dots + x^a$ יתגלו עבור $a \geq r$
- הכפלת פולינומים נותנת פולינום עם היכולת של שני גורמי המכפלה.

זמן שידור - $T_i = \frac{\text{frame length (num of bits)}}{\text{transmit rate}}$

זמן התפשטות - $T_p = \frac{\text{channel length}}{\text{propagation rate}} = \left(\frac{T_p}{L}\right) \cdot L$

זמן הגעה - $T_{arrival} = T_{transmission} + T_{propagation}$

T_{out} - הזמן עד ההחלטה שחלה שגיאה (מקרה פרטי, משך שידור *ack* זניח ואז $T_{out} = 2T_p$)

RTT - הזמן הכולל להתפשטות מידע הלך וחזור $RTT = 2T_p + T_{ack}$ (optimal T_{out})

יעילות השידור - $\beta = RTT / T_i$

כמה פעמים נכנס זמן שידור בזמן ההמתנה. **תוחלת הזמן לשידור מוצלח** - $T_v = T_i \cdot E(k)$

$$= \sum_{k=0}^{\infty} P(k \text{ re-trans}) \cdot T_{k \text{ re-trans}} = T_{OK} + P_{fail} \cdot T_{penalty}$$

← הזמן הממוצע מתחילת שידור מסגרת עד לרגע בו מתחילים בשידור מסגרת הבאה.

$E(k)$ - תוחלת מס' השידורים עד להצלחה k

T_i - זמן מחזור שתלוי במדיניות השליחה.

חלון השליחה - החוצץ ששומר את המסגרות שנשלחו עד קבלת *ack*.

חלון המקבל - החוצץ ששומר את המסגרות המסובלות להפחתת שידורים חוזרים

P_{error} - ההסתברות שתהיה שגיאה לפחות בביט 1.

P_{loss} - ההסתברות לאיבוד מסגרת

ניצולת - זמן השידור יחסית לתוחלת זמן שידור מוצלח (שאינו מנוצל) - $S = T_i / T_v$

שכבת הקו

פירוטוקול ARQ - Automatic Repeat Request

השכבה המקבלת מזהה חבילות משובשות ומבקשת משכבת הקו בצד השני לשדר אותן שוב.

- הניצולת המוקסימלית האפשרית היא $1-P$
- P זו ההסתברות לכישלון המסגרת כולה.

Stop & Wait

השולח ממתיין ל-*ack*, אם לא הגיע תוך T_{out} ישלח שוב.

ack - יכול להשתבש / להאבד

בעיה: שכפול חבילות / פיתרון: מס' סידורי לחבילה. הסבר: אם שכבת הקו תזהה שזהו העתק תזרוק את המסגרת, אבל עדיין שולחת *ack*.

T_{out} יכול להיות קצר מידי

בעיה: קבלת "ack" על מסגרת שאבדה

פיתרון: מס' סידורי ל-*ack*.

• החבילה נשלחת עם SN ומחזרת עם $SN+1$

נכונות:

• כל השגיאות מתגלות, נשמר *FiFo*, בטיחות: לא נכנס למצב שגוי (חבילה לא נמסרת שלא לפי סדר, ולא נמסרת יותר מפעם אחת), חיות: חבילות הנמסרות מגיעות בסוף.

העבה: מספיק לשמור על $mod(N+1)$ מס' סידורים, כאשר N הוא גודל החלון המקבל.

ניצולת: $S = \frac{1-P}{1+\beta}$

GO-Back-N: GBN

השינוי בשולח-מותר לשלוח חלון של חבילות בגודל N מבלי לקבל *ack*. שידור חוזר:

א) כל החלון אם עבר T_{out} איבוד של *ack* יגרום לשליחת חלון שלם.

ב) משתמשים ב-*timer* עבור כל מסגרת, ובאיבוד מסגרת משדרים רק את המסגרת האובדת.

חלון אידאלי - בשולח הוא בגודל $\beta + 1$ (חלון גדול יותר יגרום לזריקת מסגרות תקינות). במקבל: 1

• $S \& W$ מקרה פרטי של *GBN* עם $N=1$.

ניצולת: א) $S = \frac{1-P}{1+\beta}$ ב) $S = \frac{1-P}{1+2\beta P(1-P)}$

Selective Repeat: SR

שינוי גם במקבל - שומר מסגרות תקינות שהגיעו לאחר מסגרת לא תקינה (קבלת מסגרות לא לפי סדר).

גודל החלון האופטימאלי של השולח והמקבל בהתאמה:

- כל החבילות מגיעות בצורה תקינה: $1 - \beta + 1$.
- שידור חוזר ראשון יגיע בצורה תקינה $2\beta + 1$ באופן כללי: $\beta + 1$ ו $m\beta + 1$

ניצולת אופטימאלית (אין שידור שואה) $1-P$ (עבור חלון אינסופי לשולח ולמקבל)

תורת התורים

התפלגות פואסון: $P_k(t) = \frac{(\lambda t)^k \cdot e^{-\lambda t}}{k!}$

הסיכוי ל- k מופעים ב- t יחידות זמן λ - קצב כניסה, μ - קצב עזיבה.

מספר הופעות ממוצע ב- t יחידות זמן הינו λt

קצב השירות הממוצע הוא μ לקוחות בשניה. זמן השירות הממוצע (אספוננציאל) הוא $1/\mu$ שניות. קצב השירות של m שרתים הוא $m \cdot \mu$ לקוחות בשניה.

חוק Little: $\bar{\lambda} \cdot \bar{T} = \bar{N}$

\bar{T} - זמן שהייה (המתנה) במערכת ממוצע. זמן המתנה בתור = זמן שהייה במערכת - זמן שירות

$\bar{N} = \sum i P_i$ - ממוצע מס' הצרכנים הממתנים כולל את זה שמקבל שירות.

$\bar{\lambda} = \sum \lambda_i P_i$ - קצב הגעה ממוצע.

משוואת מצב היציב: $\sum P_i = 1$ $\lambda P_0 = \mu P_1 \Leftrightarrow P_1 = \rho P_0$

מצב יציב: $P_n = \lim_{t \rightarrow \infty} P_n(t)$ קיים.

M/M/K/N: קצה הגעה ושירות פואסוני, K שרתים, N לקוחות במערכת לכל היותר. מערכת כזו תמיד יציבה (מספר צרכנים סופי)

M/M/1: חוצץ אינסופי. פתרון המשוואות: $P_0 = 1 - \rho$ $P_n = (1 - \rho) \rho^n$

$E(T) = 1/(\mu - \lambda)$, $E(N) = \rho/(1 - \rho)$

תנאי יציבות: $\lambda < \mu \Leftrightarrow \lambda/\mu = \rho < 1$

זמן שהייה במערכת: $\bar{T} = 1/(\mu - \lambda)$

זמן המתנה בתור: $T_Q = \bar{T} - 1/\mu$

מספר לקוחות ממוצע בתור: $N_Q = \lambda T_Q \neq \bar{N} - 1$

עבור שרת יחיד: $N_Q = \sum_{k=1}^{\infty} [(k-1) \cdot P_k]$

חלוקה של ערוץ - ברובח C ל- N תתי ערוצים עם רוחב של C/N לכל ערוץ

FDM - לכל תת ערוץ מוקצה תחום תדרים

TDM - מתן הערוץ כולו לכל תחנה למשך $1/N$

• קצב הכניסה כעת פואסוני עם ממוצע $\lambda \cdot N$

• קצב שידור $C \cdot N$

• זמן המתנה: $T_{FDM/TDM} = N/(\mu \cdot C - \lambda)$

• מוריד עומס פי N מגדיל זמן המתנה פי N

• טוב עבור שידור מסוג **isochronous**

• **מסבנות:** על מנת למוער שהייה במערכת כדאי לאחד משאבי חישוב, על מנת למוער את זמן התגובה במערכת כדאי לבזר משאבי חישוב.

• **מסבנות:** לרוב עדיף ערוץ אחד על פני N ערוצים שכל אחד מהם חלש פי N .

isochronous - Circuit Switching

הקצאה סטטית, תעבורה מסודרת. אין עיבוד מידע עקב עומס, ההשהיות קטנות מאד וקבועות.

non-isochronous - Packet switching

הקצאה דינאמית, לא מובטח לחבילה שירות תוד פרק זמן קצר.

נוסחאות עזר:

$$\sum_{i=0}^{\infty} x^i = \frac{1}{1-x} \quad \sum_{i=1}^{\infty} ix^{i-1} = \frac{1}{(1-x)^2}$$

$$\sum_{i=1}^{\infty} ix^i = \frac{x}{(1-x)^2} \quad \sum_{i=0}^k x^i = \frac{1-x^{k+1}}{1-x}$$

$$\sum_{i=m}^k x^i = \sum_{i=0}^{k-m} x^{i+m} = \frac{x^m - x^{k+1}}{1-x}$$

$$\sum_{i=1}^k ix^{i-1} = \frac{1+x^k(kx-k-1)}{(1-x)^2}$$

$$\lim_{x \rightarrow \infty} \left(1 - \frac{a}{x}\right)^x = e^{-a} \quad \sum_{k=0}^{\infty} \frac{x^k}{k!} = e^x$$

$10^{-6} \text{ KB} = 1024 \text{ byte}$

$10^{-3} \text{ MB} = 1024 \text{ KB}$

$8 \text{ GB} = 1024 \text{ MB}$

Ethernet - 802.3 - רשתות מקומיות גישה ל- shared bus באמצעות CSMA/CD 1-persistent

Peramble (8 bytes)	Dest Mac address (6 bytes)	Src Mac address (6 bytes)	Frame Type/Length (2 bytes)	Payload (0-1500 bytes)	PAD (0-46 bytes)	CRC (4 bytes)
סינכרו עם השעון 7 בייט: 10101010 1 בייט: 10101011	כתובת MAC יעד כתובת פיזית	כתובת MAC מקור כתובת פיזית	אורך מסגרת או סוגה	החבילה שמועדת לשידור	משה העיד יהיו לפחות 64 בתים לזיהוי התנגשות	קוד לניקוי שגיאות ללא תיקון

• אם משתמשים ב- switch לא משתמשים ב- CSMA/CD כי אין התנגשות הצורך ב- PAD: נגדיר τ זמן התפשטות מקסימאלי בערוץ - ולכן לזיהוי התנגשות דרוש שאורך המסגרת (זמן שידור) לפחות 2τ (ע"י חישוב פשוט, בקונפיגורציה ראשונה התקבל שנדרש 64 בתים לפחות). לכן, 2τ - זמן מקסימאלי שבו הערוץ יכול להיות רועש כתוצאה מהתנגשות אחת.

• כתובת יעד שמתחילה ב: 0 - ממענת תחנה בודדת, 1- קב' של תחנות, 111111 - broadcast

Repeater - רכיב פיזי המקשר בין שני שכבות פיזיות של רשתות שונות, ומבטיח כל סיגנל שמגיע אליו.

Hub - הוא repeater עם מספר ports, מהווה collision domain אחד ולכן אינו משפיע על ה- throughput.

Binary Exponential backoff (BEoff) - ניחוש ותיקון לקביעת זמן המתנה אקראי אחרי התנגשות אלגוריתם שמתבצע לאחר התנגשות ראשונה ע"י כל תחנה: לאחר ההתנגשות ה- i מגרילה כל תחנה מסי' שלם בין $[0, 2^i - 1]$ וממתינה בהתאם 2^i יח'. תחום הגרילה מקסימאל 1023, לאחר 16 התנגשות רצופות, זורקים את המסגרת (אם נדרשת אמירות ה- DLG תמסור ל- MAC שוב את ההודעה לשידור. לאחר שידור מוצלח - המסגרת הבאה מתחילה עם $i = 0$).

פרוטוקול ALOHA

- התנגשות מתגלה בסיוע שידור ההודעה, כאשר יש הפרש בין השידור לקליטה.
- מתאים לכל רשת שיכולה להאזין לערוץ בזמן שהיא משדרת, ולדעת אם באותו זמן יש תחנות נוספות שמשדרות.
- כל תחנה רשאית לשדר מתי שהיא רוצה. אם מתגלה התנגשות בין מסגרות, מנסים שוב לאחר המתנה אקראית.
- "מחלקים את ציר הזמן לחריצים ברוחב של זמן שידור מסגרת" - מותר להתחיל לשדר בכל עת.
- הנחות:
- קצב הכניסה מפולג פואסוני עם קצב S.
- מספר התחנות אינסופי.
- מספר המסגרות המשודרות לערוץ (מכל התחנות) מפולג פואסוני עם ממוצע G.
- אין שגיאות נוספות (נפתרות ברמת ARQ).
- כל החבילות באורך קבוע.
- ניצולת הקצב שבו מסגרות מגיעות ליעדן חלקי קצב השידור, או, תוחלת מספר המסגרות המשודרות בהצלחה בכל חריץ.

$$P_r(1, G, 1) \cdot P_r(0, G, 1) = G \cdot e^{-G} \cdot e^{-G} = G \cdot e^{-2G}$$

1 start in slot i
0 start before i

מקסימום מתקבל ב- $G=0.5 \quad S=1/(2e) \sim 18\%$

פרוטוקול Slotted Aloha

- בנוסף לתחנה מותר לשדר רק בתחילתו של חריץ זמן (הקרוב).
 - ניצולת תוחלת מסי' השידורים המוצלחים עם תחילת חריץ i. = תעבורה % * הזמן מתוך ה- slot שבאמת שולחים בו.
- $$P_r(1, G, 1) = G \cdot e^{-G}$$
- 1 want to transmit in slot i-1
- מקסימום מתקבל ב- $G=1 \quad S=e^{-1} = 0.36$

ניצולת עבור מסי' סופי של תחנות:

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

זיכרון לחישוב: $Pr_i(succ) = Pr_i(trans) \prod_{j \neq i} (1 - Pr_j(trans)) \rightarrow S_i = G_i \prod_{j \neq i} (1 - G_j)$

תנאי יציבות: מסי' חריצים גדול ממסי' הודעות משודרות במערכת

קול ב - GSM לפי תקן cellular switching

הרשת מחולקת לתאים, כאשר בכל תא יש תחנת בסיס, אליה משדרים הטלפונים וממנה הם קולטים. בכל תא משתמשים בתחום תדרים שונה משכניו, ויש צורך למנוע הפרעות רק בתוך התא.

frequency reuse - מסי' תתי התחומים האפשרי. תחום התדרים מחולק ל- 248 רצועות (FDMA), חצי לבסיס וחצי ממנו. כל רצועה מחולקת ל- 8 ערוצי קול (TDM) כל ערוץ מוקצה לשיחה. על כל רצועה 8 חריצי זמן שאורכו 148 סיביות.

GPMS - packet switching מעל virtual circuit הצורך לכך נובע עבור גלישה באינטרנט מטלפון סלולארי. הקצאת ערוץ לשליחת non-isochronous data נעשית ע"י הגדרת סוגים של logical channels מעל מסגרת GSM. בתחום תדרים מסתכלים על 52 מסגרות כאשר כל 4 שייכות ל- logical channel (חלק משמשים עבור reservation ALOHA - בקשות להקצאת רוחב פס לשיחה)

Reservation ALOHA

תחנה מתחרה על הערות ע"י פרוטוקול slotted ALOHA על מנת לשדר הודעת בקשה. הצליחה - X הסלטים (שדרשה) הבאים שלה, ואסור להפריע לה.

הפסידה - ממתנה זמן רנדומאלי של סלטים לא מוזמנים ומתחרה שוב.

העדיפות ל- slotted ALOHA על CSMA/CD כאשר הערוצים מנוהלים ע"י תחנה מרכזית.

הזמנת ערוץ המשך פרוטוקול מפת הביטים

- כל תחנה משדרת בתורה דגל (0) לא רוצה לשדר, (1- רוצה לשדר): זוהי מפת ביטים.
- כל התחנות שביקשו לשדר משדרות לפי הסדר.

$$S = \frac{k \cdot d}{N + k \cdot d}$$

גודל הודעה: d
מסי' תחנות: N
מסי' שידורים: k

CSMA (Carrier Sense Multiple Access)

- לפני שמשדרים, מאזינים אם שקט: משדרים.
- אם הערוץ לא שקט: אבחנה ביחס לשידור הראשון.
- 1-Persistent** - מנסה שוב לשדר כשהערוץ שקט.
- Non-persistent** - ממתנה פרק זמן אקראי ואז מאזינה.
- אם התנגשות: מחכה זמן אקראי (ע"י Binary Exponential backoff) ומקשיבה שוב אם שקט משדרת אם לא תחזור להמתין זמן אקראי.

CSMA/CD

יש מנגנון חומרה שמוזהה התנגשויות תוך כדי שידור ומפסיק שידור. ניתוח ביצועים:

k תחנות שרוצות לשדר, ההסתברות שתחנה תשדר בחריץ מסויים הוא P. ההסתברות להתחלה של שידור מוצלח בחריץ כלשהו הוא:

$$A = k \cdot P(1 - P)^{k-1} \quad (P = \lambda/k \quad A = \lambda e)$$

Contention Interval (נסמן CI) - פרק הזמן החל מסוף שידור מוצלח עד תחילת שידור מוצלח הבא.

ההסתברות לכך ש- Contention Interval יכיל j חריצים לא מנוצלים הוא:

$$E(CI) = 2\tau \sum_{j=1}^{\infty} j \cdot CI_j = 2\tau \cdot \frac{1-A}{A} \leftarrow CI_j = (1-A)^j A$$

$$S = throughput = \frac{T}{T + E(CI)} = \frac{T}{T + \frac{T}{A} \cdot \frac{1-A}{A}} = \frac{1}{1 + \frac{1-A}{A^2}}$$

- הניצולת קטנה ככל ש-T (זמן שידור) קטן או ש- τ גדל.
- T קטן כשמגדילים את הקצה השידור, מקטינים אורך מסגרת.
- τ גדל כשמגדילים את אורך הכבל.

CSMA/CA - לא ניתן להשתמש ב- CD כי:

- לא ניתן להקשיב ולשדר בו זמנית.
- ההסתברות לשגיאה בביט היא גבוהה מאד.

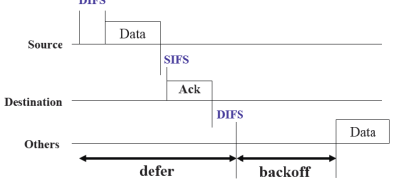
השינויים שבוצעו:

- תחנה מאזינה לערוץ, אם אין שקט, היא ממתנה זמן ראנדומאלי (BEoff) של סלטים שקטים ואז משדרת אם אין שידור אחר.
- מנגנון S&W - תחנה חייבת לשדר מייד ack
- תמיכה ב- fragmentation ברמת ה- MAC מעמים על המערכת - לכל שבור CRC ack משלו עקרונות ה- Binary Exponential Backoff: זמן המתנה - כפולה של slot time. מיני סלוט תלוי בקוטר התא - max round trip delay.
- Physical carrier sense** הפרוטוקול:
 - פעם I שתחנה רוצה לשדר היא ממתנה DIFS, אם יש שקט משדרת. אחרת, נכנסת למתנה.
 - תהליך המתנה** - מוחזק מונה W - מאותחל למסי' אקראי של מיני סלטים. בכל פעם שיש שקט קטן ב- 1, אין שקט - לא משתנה, לאחר תקופה כזו ממתין DIFS לפני שממשיך להקטין.
 - לאחר המתנה:**
 - אם שקט - משדרת. אם לא קיבלה ack תוך DIFS זמן, מכפילה את תחום החלון ומגרילה זמן המתנה.
 - אין שקט - מכפילה את תחום החלון ומגרילה זמן המתנה. SIFS < DIFS - מביטיח שידור ack ללא הפרעה במקרה של שידור מוצלח (ללא התנגשות).
 - מנגנון מבטיח שידור חוזר גם לאחר התנגשות וגם לאחר שגיאה בניגוד ל- CD (לאחר התנגשות) בעיות:
 - hidden terminal effect** - תחנה א- B אינם בטוח קליטה אחד של השנייה, אבל רוצות לשדר למשל ל- AP שבטוח קליטה של שניהם.
 - exposed terminal effect** - צמת נמנעת משידור בגלל שידור אחר למרות שהשידור אפשרי. המשך virtual carrier sense בעמ' 3.

WiFi/IEEE-802.11

- באלחוט אל הרשת, וליצור רשת מקומית אלחוטית.
- מגדיר שני סוגי התקנים:
 - תחנת קצה (לדוג' laptop).
 - Access Point (AP)** - תחנה המחברת את הרשת האלחוטית לרשת הייחית.
- כאשר שתי תחנות או יותר מזהות זו את זו הן מהוות **Basic Service Set (BSS)**.
- מגדיר שני מודים של פעולה:
 - Peer To Peer mode (P2P)** - התחנות בין הן IBSS "סוג של רשת מקומית" - הן בטוח האזנה זו מזו, וישובות על סגנון Ethernet משותף.
 - Infrastructure mode (Im)** - התקשרות בין התחנות היא תמיד דרך AP. ה- AP מהווה Ethernet switch לכל BSS תדר מילה. מוגדרים שני סוגי פרוטוקולים:
 - DCF** - ניתן להשתמש בו בשני סוגי המודים, נקרא גם CSMA/CA להלן ←
 - PCF** (עמ' 3) - ניתן להשתמש בו רק ב- Im.

Physical carrier sense



הערוץ העולה - (מתחנות הקצה), משתמש בפרוטוקול ה- MAC. מחולק לחריצי זמן, כאשר הסטטוס שלכל חריץ נקבע ע"י ה- CMTS והם: **החריץ פנוי** (contention slot) - **החריץ ממומן** לתחנה X.

הפרוטוקול: best effort

תחנה הרוצה לשדר מסגרת ממתנה לחריץ פנוי ומשדרת מסגרת הזמנה (דורשת מסי' חריצים), אם היא היחידה ששידרה בחריץ, אז ההזמנה מתקבל, אחרת "התנגשות" והחריץ מבוזבז. תחנה יודעת אם בקשתה התקבלה או אם הייתה התנגשות ע"י חיווי חוזר מה- CMTS (יכול לענות רק על חלק מהבקשה). אם ה- CMTS קיבל את הבקשה ולא החליט כמה להקצות יחזיר Data Grant pending. אם תחנה לא קיבלה אישור, היא תנסה שוב ע"י exponential back off אלגוריתם.

piggybacking - ניתן לצרף ל- data בקשות לשידור נוסף (היתרון פחות התנגשות, חיסרון הגינות).

Unsolicited Grant Service - הקצאה סדירה לתחנות Polling Service - מקצה חריץ בודד מדי פעם

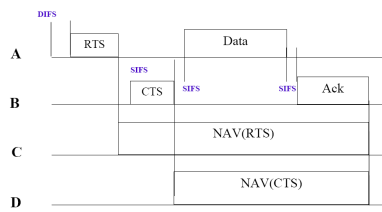
DOCSIS - פרוטוקול גישה ל- broadband access networks מעל תשתית הטלוויזיה והכבלים.

- Broadband access networks** - רשת טכנולוגית packet switching שמאפשרת למשתמשי קצה ביתיים גישה בכפוף תעבורה גבוהים ל- Internet.
- טופולוגיית עץ - טופולוגיה קשירה כשמי' הקווים קטן ב- 1 ממספר
- בראש העץ תחנה הנקראת head-end המשדרת את ערוצי הטלוויזיה לבתים - בה יושבת הלוגיקה CMTS (דומה GSM), האחראית בין השאר על הקצאת סלטים
- העץ - ערוץ משותף. אפשרות להתנגשויות. התחנות לא יכולות להאזין לשידורים של תחנות אחרות.

פרוטוקול ה- MAC

- שימוש ב- reservation ALOHA ע"י לוגיקה הנקראת CMTS (בדומה ל- GSM).
- מחולק לשני ערוצים: **ערוץ יורד** (מה- CMTS) אין צורך בפרוטוקול ה- MAC.

הפרוטוקול: virtual carrier sense



RTS - בקשה לשדר (ההודעה מכילה את אורך המסגרת).
CTS - אישור לשידור, מכיל פרמטרים מ-**CTS** כל תחנה ששומעת אחד מהנייל מסיקה את פרק הזמן עליה להימנע משידור.
NAV - פרק זמן שעל התחנות להמנע מלשדר.
 • אם **ack** לא מגיע הפרוטוקול מתחיל מחדש האם קיימת אפשרות להתנגשות?

	ack	data	CTS	RTS	
ack	לא	כן	כן	כן	RTS
data	לא	כן	כן	כן	CTS
CTS	לא	כן	כן	כן	data
RTS	לא	לא	לא	לא	ack

PCF + DCF

תחנות ניגשות לערוץ לפי **DCF**, אבל ל-**AP** יש עדיפות (פרקי זמן של המתנה לפני שידור).
SIFS - כמו קודם המתנה לאחר **ack** או **CTS**.
PIFS - אחריו מותר ל-**AP** לשדר הודעות בקרה של **PCF** (מצב ערוץ, או **Polling**).
DIFS - אחריו מותר לתחנות לשדר, לפי **DCF**.
SIFS < PIFS < DIFS

שאלה לעזר ב- DOCSIS:

החלק המוקצה להזמנת הנתון a , או $1-a$ מהחריצים בערוץ העולה משמים להעברת מידע. ההסתברות לשידור בודד בחריץ הזמנה מסוים היא:

$$\lim_{N \rightarrow \infty} N \cdot \frac{1}{N} \left(1 - \frac{1}{N}\right)^{N-1} = \frac{1}{e}$$

לכן, מתוך חריצי ההזמנה, רק a/e מכילים שידור בודד. ובגלל שההסתברות לשגיאה היא p , החלק של חריצי ההזמנה מוצלחת הינו $a \cdot (1-p)/e$

לכל שידור הזמנה מוצלחת, ה-**CMTS** מקצה T חריצי שידור מידע ולכן החלק של חריצי המידע הינו $\left[\frac{1}{e} \cdot a \cdot (1-p)\right] \cdot T$

לכן אם נשווה, נקבל

$$\left[\frac{1}{e} \cdot a \cdot (1-p)\right] \cdot T = 1 - a$$

$$a = \frac{e}{T(1-p) + e} \quad S = 1 - a = \frac{T(1-p)}{T(1-p) + e}$$

ניתוח ביצועים Go-Back-N יהיה x מספר הפעמים שצריך לשדר את המסגרת עם חבילה i באינטרוול $[t_i, t_{i+1}]$ - X הוא משתנה מקרי עם התפלגות גיאומטרית $\Pr(X = k) = p^{k-1} (1-p)$ - ב- $X-1$

מהפעמים האלו מכיל האינטרוול β מסגרות נוספות, המכילות את חבילות $(i+1, i+2, \dots, i+\beta)$ ואילו בפעם האחרונה הוא מכיל רק את i , ולכן: $\gamma = \sum_{x=1}^{\infty} p^{x-1} \cdot (1-p) \cdot [(X-1)(\beta+1) + 1]$

$$\gamma = \frac{1 + \beta \cdot p}{1 - p} \Rightarrow S = \frac{1}{\gamma} = \frac{1-p}{1 + \beta \cdot p}$$

PIF - כל מחשב שמקבל את ההודעה (מאב) שולח לכל שכניו ומחכה לקבלת ההודעה בחזרה מכל שכניו, רק אז שולח את ההודעה בחזרה לאב. משתנים חדשים - pi (האב), $-ei(l)$ - הפרש בין מסי' ההודעות שנשלחו לשכן l לבין מסי' ההודעות שהתקבלו ממנו. **תכונות** - לא מתמודד עם אובדן מסגרות, בגרף קשיר המידע מגיע תוך זמן סופי, התפשטות מהירה ביותר, כשהצומות שהתחיל את הפרוטוקול מקבל פידבק מכל שכניו - סיימו.

PI - כל מחשב שמקבל הודעה שולח אותה לכל שכניו פרט לזה ממנו קיבל את ההודעה. משתנים בכל צומת - Gi (שכנים), $[nil]$ (צומת וירטואלי לתחילת הפרוטוקול), mi - דגל, האם צומת i נכנס לפרוטוקול, מאותחל ל-0. **תכונות** - רמה 3, בגרף קשיר המידע יגיע תוך זמן סופי, התפשטות מידע היא המהירה ביותר האפשרית, הצומת היוזם לא יודע מתי נגמר הפרוטוקול, יכול להיות מורץ רק פעם אחת, אותה הודעה עוברת.

הסתברות

$$P(A \cap B) = P(AB) \quad P(B - A) = P(B) - P(AB)$$

$$P(A \cup B) = P(A) + P(B) - P(AB) \quad P(A|B) = P(AB) / P(B)$$

יהי B מאורע כלשהו ויהיו A_i מאורעות המקיימים כי: א. המאורעות זרים בזוגות. ב. $\bigcup_{i=1}^{\infty} A_i = \Omega$. ג. $\forall i, P(A_i) > 0$.

$$P(A_i | B) = \frac{P(A_i B)}{P(B)} = \frac{P(B | A_i) P(A_i)}{\sum_{i=1}^n P(B | A_i) P(A_i)}$$

$$P(B) = \sum_{i=1}^n P(BA_i) = \sum_{i=1}^n P(B | A_i) P(A_i)$$

$P(n, \lambda, t) = \frac{(\lambda t)^n}{n!} \cdot e^{-\lambda t} \quad E = \lambda$	התפלגות פואסונית - n הגעות עד זמן t כאשר תוחלת מסי' ההגעות ליחידת זמן היא λ
$f(\lambda, t) = \lambda e^{-\lambda t} \quad E = 1/\lambda$ $F(\lambda, t) = \Pr[\text{next arrival at time} \leq t] = 1 - e^{-\lambda t}$	התפלגות מעריכית (אקספוננציאלית) - עם פרמטר λ . ההסתברות להפרש זמן t מאז ההגעה האחרונה. פונקציית ההתפלגות המצטברת \leftarrow
$\Pr[\text{next arrival at time} \leq t + \Delta t \mid \text{next arrival at time} \geq t] = \Pr[\text{next arrival at time} \leq \Delta t]$	תכונת חוסר-הזיכרון של ההתפלגות המעריכית

מספר המופעים ב- t יחידות זמן בקצב הגעה פואסוני λ הינו λt . עבור השורות הבאות, p הינו הסיכוי להצלחה.

$E(x) = p \cdot 1 + (1-p) \cdot 0 = p$	ברנולי	מקבל 1 בהתאם להצלחה בסיכוי p ו-0 בהתאם לכישלון בסיכוי $1-p$
$E(x) = \sum_{k=0}^n \binom{n}{k} p^k q^{n-k} \cdot k = np(p+q)^{n-1} = np$	בינומי	X מספר ההצלחות ב- n ניסויים $P(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$
$E(x) = \frac{1}{p}$	גיאומטרי	X מספר הניסויים הדרוש להצלחה ראשונה: $P(X = k) = (1-p)^{k-1} \cdot p$

שרשראות מרקוב - כמו דיאגרמת מצבים רגילה של תורת התורים, רק שפה כל מצב מתאר את אחד ממצבי המערכת והמעבר ביניהם מתבצע בקצב אחיד. על הקשתות ישנן **הסתברויות** המעבר (ולא קצב המעבר), כאשר סכום ההסתברויות של הקשתות היוצאות ממצב הוא 1. כדי לבנות את המרקוביאן נמקם מצב ונמקם את כל המצבים מהם ניתן להגיע ממצב זה, ועל הקשתות נרשום את ההסתברות להגיע ממצב למצב. בניית המשוואות: הסתברות המערכת להימצא במצב כלשהו שווה לסכום ההסתברויות להגיע לאותו המצב. בנוסף יש משוואה האומרת שסכום ההסתברויות של המצבים הינו 1.

Point Coordination Function (PCF)
Centralized mode לעומת **distributed mode (DCF)**
 • ה-**AP** משדר הודעות בקרה בקצב קבוע, וכך מודיע מילי שניות לשרת ומתי. חלק מהערוץ מוקצה ל-**polling** חלק מהערוץ מוקצה לתחנות שהזמינו גישה לערוץ.
 • אם לאחר K תשאלים לתחנה אין מה לשדר, **AP** יכול להוציא מהתשאלות לפרק זמן מסוים

SONET/SDH - טכנולוגיית **Physical layer** לרשתות אופטיות שנמצאות בערים (טווחים של עשרות ק"מ). שימוש עיקרי: **isochronous** (טלפון). טופולוגיית **טבעת** - טופולוגיה מינימאלית, עם שרידות של נפילת תחנה אחת. שידור מסגרות בקצב גבוה, מסגרות ("רכבות") באורך קבוע עם מסי' קבוע של סלוטים ("קרונות"). קובעים מראש איזה קרון מחבר בין אלו תחנות. באופן כללי הפס וחולק ל- $N(N-1)/2$ ערוצים לוגיים. N זהו מספר התחנות.
דרך זו גוררת חוסר יעילות: ההתפלגות של המידע לא ניתנת לחיזוי. **פיתרון:** ערוץ יחיד שישמש כל תחנה שרוצה לשלוח מידע.

Resilient Packet Ring (RPR) - תקן לשידור מידע **non-isochronous** ברשתות טבעת (כדוגמת הנייל).
 • כל מסגרת מכילה **src\dest MAC address**.
 • תחנה שמקבלת מסגרת בודקת האם היא מיועדת אליה. אם כן, מעתיקה את המסגרת אליה. אם לא, מעבירה הלאה (אלא אם כן היא שלחה אותה למשל ב-**broadcast**).
 • דרוש **פרוטוקול MAC** כבקרת גישה לערוץ למשל אם תחנה משדרת, ובמהלך השידור מקבלת מסגרת ששלחה.

buffer insertion ring
 חוצץ a - מסגרות מקומיות לשידור, חוצץ b - מסגרות שעוברות בטבעת. כש- b ייק נתן להתחיל לשדר מ- a , אם מגיעות עוד מסגרות - נוספים ל- b , כש- a יסיים, נשדרן. גודל a - מסגרת מקסימאלית שמותר לשדר. גודל b - תלוי בפרוטוקול **MAC** במקרה זה כמו a .
 ← ההשהייה הארוכה ביותר שמוסיפה תחנה למסגרת שווה לזמן שידור המסגרת הארוכה ביותר.

רוחב הפס הזמין: BW - הקצב בו ניתן להכניס מידע חדש לטבעת.
 קצב שידור תחנה הינו R ביטים לשנייה.
 d - מספר תחנות בהן כל הודעה עוברת **בממוצע**.
 N - מסי' התחנות, K - מסי' התחנות המשדרות.
באופן כללי: $BW = K \cdot R$
ניצולת = הקצב בו ניתן להכניס מידע חדש לטבעת מנומלל בקצב השידור.
 $K \cdot R$ **קצב הכנסת מידע חדש לטבעת**
 R **קצב שידור בטבעת**

בעצם הניצולת הינה N/d .
אידיאל: כל תחנה משדרת לתחנה העוקבת $d=1$
ממוצע: כל הודעה עוברת מרחק ממוצע $d=N/2$
 הניצולת: אידיאל N , ממוצע = 2.
בעניית הגינות: (למשל במקרה של שרת קבצים) אלגוריתם שיבטיח **fairness**: למשל **Meta Ring**. לכל תחנה מסי' מקסימאלית של הודעות לשדר רק לאחר שכולם שידרו את המקסימום יקבלו הקצאה חדשה של הודעות.

טבעת BIR דו כיוונית:
 תחנות לומדות את הטופולוגיה של הרשת, וכך נשלחות מסגרות במסלול הקצר ביותר.
ממוצע ניצולת: $d=N/4$, רוחב הפס הזמין בטבעת בשני הכיוונים הוא $8R$, $2 \cdot 4 \cdot R = 8R$.
 על כל אחת משתי הטבעות: $4R/R = 4$
 על שתי הטבעות ביחד: $8R/2R = 4$
הגנות מפני נפילה:

• **Steering mode:** תחנה שולחת הודעות ליעד במסלול שלא עובר בתחנה או בקו שנפלו.
 • **Wrapping mode:** עד שתחנת המקור עוברת ל-**steering**, תחנה הסמוכה לנפילה יכולה להעביר מידע מטבעת אחת לשנייה.
מנגנון עדיפויות: לאפשר לשדר לתוך הטבעת מסגרות בעדיפות גבוהה (**HP**) לפני מסגרות בעדיפות נמוכה (**LP**). במקום b יחיד יש שני חוצצים $b(h)$ ו- $b(l)$.
 • מסגרת בעדיפות נמוכה תשדר אם $b(h) > 1$
 • $b(l)$ ריקים, והאלגוריתם **fairness** מרשה.
 • מסגרת בעדיפות גבוהה תשדר כל זמן ש-**fairness** מאפשר, ו- $b(h) > 1$.
 גודל $b(h)$ - כגודל המסגרת הארוכה ביותר ב-**HP**.
 גודל $b(l)$ - גודל ככל האפשר - אם גולש משדרים ממנו.

MAC domain - שתי תחנות נמצאות על אותו *MAC domain* אם מסגרת *MAC* המשודרת ע"י האחת מתקבלת ע"י השנייה ללא התערבות של תחנות ביניים אחרות בשכבה הגבוהה משכבה 1. אם מחירים שני סגמנטים של *Ethernet bus* באמצעות *repeaters* (ברמת *phy*), הרי התחנות עדיין נמצאות באותו *MAC domain* - החיסרון: רק תחנה אחת יכולה לשדר בכל רגע, מקבלת את ה-*LAN available* לקצב השידור.

פיתרון: חיבור LAN באמצעות גשרים

- הגשר מבודד את שני ה-*MAC domains* זה מזה לא מעביר אותות פיזיים, אלא מסגרות שלמות.
- הגשר קורא מ-*LAN segment* לתוך חוצץ מקומי, וע"פ פרוטוקול ה-*MAC* מתמין לסגמנט השני וזו משדר.

Learning Algorithm

האלגוריתם הזה עובד כל עוד הטופולוגיה היא עץ. הגשר קורא כל מסגרת המשודרת על ה-*LAN* המחובר אליו, כדי ללמוד מי הן התחנות המחוברות לשם (ע"פ *src MAC addr*).

אם גשר מקבל מסגרת שתחתנה היעד לא מוכרת לו הוא מפיץ אותה לכל ה-*LAN* אליה מחובר.

בניית עץ פורש (אם יש מעגלים)

המטרה: לקבוע זהות הגשר של העץ. לקבוע גשר אחראי עבור כל *LAN*.

הנחות:

- לכל גשר זהות יחידית משלו, למשל כתובת *MAC* (שמגדירה גם את הגשרים ברשת זו).
- כל גשר מצמיד מזהה יחודי *port ID* לכל ה-*ports* היוצאים ממנו.
- Root Bridge** - הגשר שזהותו מינימאלית.
- Path cost** - ערך אותו מצמיד גשר לכל אחד מה-*ports* שלו (המחיר להגיע לשורש).
- Root Port** - ה-*port* שכל אחד בחר כדי להגיע ממנו לשורש.
- Root Path Cost** - המחיר להגיע לשורש דרך ה-*Root Port*.
- Designated Bridge** - הגשר בכל *LAN* שמחיר המעבר דרכו לשורש מינימאלי.
- גשרים (או *ports*) שאינם משתתפים בעץ אינם מתנתקים מהרשת - הם ימשיכו לקבל מסגרות, אבל יזרקו אותם.

הפרוטוקול: כל גשר שמתעורר ולא יודע על קיום שורש, מציע את עצמו כשורש ע"י משלוח *BPDU* (נוצר ע"י שכבה 2), גשר בחר *root port* - ה-*port* דרכו הגשר ה-*BPDU* עם זהות מינימאלית. גשר שמקבל *BPDU* מה-*root port* משדר *BPDU* על כל אחד מה-*LANs* שלו עבורם הוא *designated bridge*. *BPDU* מכיל את זהות הגשר השולח, ה-*port* שלו שמהווה *Designated port* באותו *LAN*, זהות הגשר שהשולח חושב שהוא השורש, מחיר המסלול שמוגבל לשורש דרך הגשר השולח.

התגברות על שינויי טופולוגיה

השורש מפיץ *designated* פעם בעץ הודעת *hello* וזה מופץ ע"י ה-*bridge* *designated* לכל ה-*LANs*, אם הודעה כזו מתקבלת - מתחיל אלגוריתם הבנייה מחדש. **Broadcast** - כשגשר מקבל כתובת *MAC* של 111...111 (48 פעם), הוא שולח אותה לכל הענפים, מלבד זה שדרכו התקבלה.

Wireless LAN (802.11)

- ה-*APs* מתפקדים כ-*Ethernet switches* המגשרים בין התחנות שלהם לבין עצמם, וכן בין התחנות שלהם ל-*BSS* אחרים.

תהליך ההצטרפות:

תחנה יכולה לשלוח הודעת בקרה *probe req* (בקשה מ-*AP* לענות), ה-*AP* יענה ב-*probe response* (להילופף התחנה מתמין ל-*probe frame* של *AP*). **authentication**: ע"י סיסמא. **association**: הצטרפות התחנה ל-*BSS*. ה-*APs* ששייכים לאותה רשת ומחוברים זה לזה מריצים אלגוריתם ליצירת עץ בוניים טבלת *bridging*. כל מסגרת ב- 802.11 מכילה *src/dets* וכאשר *AP* מקבל מסגרת כזו הוא יודע אם עליו לשדר אותה לתחנה מרומית או שעליו לשלוח אותה על העץ הפורש.

מעבר ל-Basic Service Set אחר:

ביצוע *authentication* מחדש מול ה-*AP* החדש. וביצוע *re-association*, דורש מה-*AP* החדש: - לסמן לעצמו שמעתה התחנה הזו נמא בתא שלו. - לשדר לכל הגשרים / *switches*... הודעה כאילו שודרה ע"י התחנה ע"י שיעדכו את טבלאות ה-*bridging* שלהם.

TCP segment: *src port num.*, *dest port num.*, *sequence num.*, *ack num.*, *header len.*, *reserved.*, *flags*(URG, ACK, PSH, RST, SYN, FIN), *window*, *TCP checksum*, *urgent pointer*, *options*, *data*.

כתובת *MAC* - 48 ביטים, כתובת *IP* - 32 ביטים. *IP header*, *TCP header* - 20 ביטים. *UDP header* - 8 ביטים. *Ethernet header* - 14 ביטים, *Ethernet trailer* - 4 ביטים.

האינטרנט ופרוטוקולי TCP/IP

רשת ה-IP:

בנויה מתתי רשתות המחוברות באמצעות התקנים (*Routers*) המבצעים ניתוב ברמה 3. ניתוב ללא *Router* מתבצע ברמה 2. כל התחנות באותה תת רשת (הנקראת *MAC domain*, *IP subnet*, *broadcast domain*, *extended LAN*) ישובות מעל סגמנט של *LAN* אחד, או מס' סגמנטים.

כתובת IP

- מורכבת מ-4 בתים
- מרחב הכתובות מחולק על 5 מחלקות
- הכתובת של ה-*A-C* (כתובות *unicast*) ואלו מחולקות לשני שדות: כתובות רשת וכתובת מחשב.
- A* - רשתות גדולות עד 2^{24} מחשבים, אבל רק 2^7 רשתות אפשריות.
- C* - רשתות קטנות עד 2^8 מחשבים, אבל עד 2^{21} רשתות אפשריות.

כתובות IP מיוחדות:

- src = network address, dest = host address*
- src: X dst: 111...111-direct broadcast address* בדרכי תורק, אלא ה-*netID* הוא של הרשת שלי
- src & dst: 111...1-limited broadcast address* לכל המחשבים ברשת זו
- src & dst: 000...000-this host* נשלח ממני

להתחבר לרשת ה-*Internet* יש לפנות לארגון *IANA* כדי לקבל כתובת רשת. כתובת הרשת נקבעת או ידנית או אוטומטית ע"י פרוטוקול *DHCP*.

Router & Bridges

שימוש רק ב-*bridges* לא אפשרי! חסרונות: - כל *bridge* צריך לשמור את מידע ניתוב לכל תחנה. ניתוב של עץ פורש אינו יעיל (שימוש לא יעיל ב-*Bandwidth*, שההייה ארוכה, יצירת מוקדים לגודש (למשל בשורש), שינוי גורר זמן רב בבניית עץ חדש). שימוש רק ב-*routers* לא אפשרי! כי כל מחשב בודד יהיה *subnet*.

יתרונות השילוב:

- הגדלת רשתות מביא להקטנת מספרן, ולהקטנת טבלאות הניתוב.
- ניצול יותר טוב של מרחב הכתובות של *IP*.
- פעולת *bridge* מהירה יותר משל *router*, יכול לעמוד בקצבי עבודה גדולים יותר.

פרוטוקול ARP - מיפוי מכתובת IP ל-MAC

ה-*src* שולח הודעת *ARP request* שמשמעותה של מי ה-*ip* הזה, לכתובת *broadcast* limited. היעד בלבד יענה את כתובתו. הפרוטוקול יושב מעל *Ethernet*, רמה 2.

Virtual LAN - הרחבה של *Ethernet*

ב-*switch* מוגדר לכל תחנה לאיזו רשת מקומית היא שייכת ע"י *IP subnet*. התחנה משדרת מסגרת *Ethernet* רגילה, וה-*switch* מרחיב את המסגרת ע"י הוספת *VLAN tag* שמכיל שני שדות:

- בית אחד מצייין לאיזה *VLAN* השולח שייך.
- יש שתי אפשרויות עבור בניית העץ הפורש:
 - לבנות עץ שונה לכל *VLAN* (הרחבה: 802.1s)
 - לבנות עץ משותף לכל ה-*VLANs*

פרוטוקול DHCP

תחנה חדשה המתחברת מחפשת *DHCP server* מקומי ע"י משלוח *ip(me)* אל *ip(broadcast)* ו-*MAC(client)* אל *MAC(broadcast)*. השרת עונה ב-*ip* שגשגה מ-*ip(server)* ל-*ip(broadcast)* ומ-*MAC(server)* ל-*MAC(broadcast)*. ה-*client* מבקש מאחד השרתים נתונים - כתובת *ip* לזמן קצוב, כתובת *default router*, כתובת שרת *DNS*.

שכבת IP - "best effort"

Package structure: *VERS.*, *HELEN.*, *SERVICE TYPE.*, *TOTAL LENGTH.*, *IDENTIFICATION.*, *FLAGS.*, *FRAGMENT-OFFSET.*, *TIME TO LIVE.*, *PROTOCOL.*, *UDP/TCP.*, *HEADER CHECKSUM.*, *SOURCE IP ADDRESS.*, *DESTINATION IP ADDRESS.*, *OPTION.*, *PADDING.*, *DATA*

TTL: מאותחת ל-255 וכל *router* מפחית ב-1 לפחות.

ניתוב ב-IP

הפונקציונאלית העיקרית של *IP* היא הניתוב ברשת (למחשבים רחוקים). לכל *router* טבלת ניתוב, בה יש *entry* עבור כל רשת, או קבוצת רשתות. בדרכי לכל תחנה מוגדר *default router*, אם לרשת מחוברים מס' *router*-ים יתכן שהדבר לא יעיל, פיתרון: הודעת *ICMP redirect*. בעיה בבנייה: ספירה לאינסוף בנפילות מסויימות פיתרון: *Split Horizon* דיווח על אינסוף - לא לדווח לשכן על מסלולים שעוברים דרך השכן. עדיין בעיה במעגלים שמכילים 3 תחנות ויותר.

מבנה האינטרנט

- אוסף של רשתות המחוברות ע"י נתבים.
- מבחינה מנהלית מחולק ל-*AS*, כאשר כל *AS* מנוהל ע"י ישות אחת, ובנוי ממספר רשתות.

פרוטוקולי ניתוב:

- RIP Routing Information Protocol** - עובד בשיטת *distance vector routing*
- BGP - Boarder Gateway Protocol * OSPF - Open Shortest Path First**
- distance vector routing**: עבור ניתוב בתוך *AS*
- אחת לפרק זמן שולח ה-*router* את המרחקים שלו, השכנים מעדכנים את הטבלה בהתאם (מימוש מבוזר של *Bellman Ford*)

שכבת התובלה

UDP - מוסף מעט פונקציונאליות ל-*IP*, כגון גילוי שגיאות (לא אפשרות לתקן או לבצע שידור חוזר), ביצוע מספר אפליקציות בו זמנית בעזרת *port-number* (לאפליקציות ידועות פורטים קבועים). צריך את *UDP* כי *TCP* מסובך יותר ולכן זמן העיבוד גדול יותר, ויש אפליקציות כמו שידור וידאו מעל *IP* שאינן צריכות שידורים חוזרים וידועות להתמודד עם אובדן חבילות.

TCP - מבטיח *end to end reliability* כלומר חבילות לא אובדות, לא מגיעות עם שגיאות, שמירה על סדר, וזאי שכולל מידע, מאפשר *multiplexing* של כמה אפליקציות מעל אותו *IP*.

Triple handshake - הוצה ליצור תקשורת *TCP* עם *A*: שולח הודעת בקרה *Syn*, *B* שולח *Syn ack*, *A* שולח *ack* ל-*B* שקיבל את *Syn*. *ack* שולח *A*: שולח *data*. בסוף *A* שולח הודעת *Fin* ל-*B* על סיום התקשורת.

הבדל בין TCP ל-ARQ:

- TCP* מסובך הרבה יותר, כי *TCP* יושב ברשת ו-*ARQ* יושב על קו פיזי.
- ARQ* מבצע רק *flow control* (=בקרת זרימה), *TCP* מבצע גם *congestion control* (=בקרת גודש)
- ARQ* מובטח שמירה על סדר יחסי ע"י השכבה מתחת, ב-*TCP* לא מובטח.
- הרבה יותר קל לחשב את *round trip delay* ב-*ARQ* כי השונות היא 0.
- ה-*TCP* צריך להיות פרוטוקול דינאמי יעיל לבניה וניתוק של הקשר, ב-*ARQ* הקשרים קבועים.
- TCP* מהווה שילוב של *GBN* (להודיע רק על הסגמנט הבא) ו-*Selective Repeat* (לקבל סגמנטים לא לפי הסדר).
- TFTP** - מעל *UDP*, לקוח פונה לפרטר 69, השרת מנחה לפרטר פניו ו-*ack* עם הסגמנט הבא לו הוא מוכן, לקוח שולח את הסגמנט המבוקש ומנחה לאישור. *Ack*. *Timeout* - ברמת התוכנה

ARQ - echo-request

כדי לדעת אם תחנה פעילה. *ARP* עובד רק עבור תחנות באותו *LAN*, *echo* עבור 2 תחנות כלשהן. אם כתובת ה-*MAC* לא ידועה עדיף להשתמש ב-*ARP* כיוון שבמקרה *echo* גם ככה יקרא ל-*ARP*.

traceroute - מאפשר לתחנה כלשהי לקבל את המסלול אל תחנת רשת אחרת (ה-*IP* שלה נמסר כפרמטר). הפקודה שולחת חבילת *IP* המכילה *echo-request* אל תחנת היעד. החבילה נשלחת עם *TTL=1* ולכן נזרקת ע"י הנתב הראשון במסלול, ותתקבל הודעה מהנתב שהודעה נזרקה, וכך הלאה נגדיל את *TTL* עד אשר נקבל הודעה מתחנת היעד.

TCP

מגלה שסגמנט אבד בעזרת 3 *duplicate ACKs*. אם אי אפשר לקבל 3 *ACKs* (חלון קטן מ-3 סגמנטים) אז משתמשים במנגנון *timeout*. **בקרת גודש (congestion control):** השולח נמצא באחד המצבים:

- slow start*: גודל החלון הוא 1, גדל ב-1 עם כל קבלת *ack* עד שמגיע ל-*slow start threshold* ועובר למצב השני
- congestion avoidance*: *congestion window* גדל ב-*1/cwnd* עם קבלת כל *ack*
- אם השולח מגלה שסגמנט אבד הוא מייחס זאת לגודש ברשת ומוריד את גודל החלון.
- אם הגילוי הוא בעקבות *timeout* גדול חלון השליחה יורד ל-1.
- slow start* משתנה ל-*cwnd/2* והשולח נכנס ל-*slow start*
- אם הגילוי הוא בעקבות 3 *duplicate ACKs* גדול חלון השליחה יורד ל-*slow start*. *cwnd/2* משתנה ל-*cwnd/2* והשולח נכנס ל-*congestion avoidance*

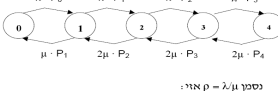
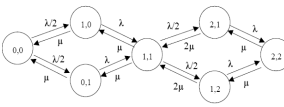
UDP segment: *src port num.*, *dest port num.*, *UDP message len.*, *UDP checksum*, *data*.

דוגמא 2
באיור 500 מציגות 2 סוגיות צילום. ידוע כי בשעת העומס מופעי הרשת הסטטיסטי המעניינים לצלם מתפלגים בעזרת התפלגות ע"י מומצע λ . קצב הצילום מתפלג אז הוא פואסוני עם מומצע λ (בכל מרווח).
סטטיסטיקה המסומנת באיור 500 מוקלת לפי האלטרנטיבה הבאה: אם מומצתים לצילום יותר מ-3 אנשים היאמונה ואלתבת לבית הסטודנט. אם לא, היאמונה נוספתת לתור הקצר יותר. אם שני התורים שרורים באורכם הסטטיסטיים שטילה ממוצע של 10 שניות ומסומנת ע"י הנוסחה לאיור 500 לרובנוקף, שני סטטיסטיים האחרים נותרו רציף שתייה העניי מבין קצב יותר מה שהיאמונה עלולתה בסטטיסטיים מנב עברת לתור השני.

- א. שרטוט דיאגרמת מצבים של המערכת שבה כל מצב משקף ציורף אפשרי של אורכי שני התורים.
- ב. חשב את ההסתברות P_i לכך שיהיו שני תורים / סטודנטים (לכל / אפשרי).
- ג. מוכח סטטיסטיים נומטיים במומוצע בתור בכל רגע נתון;
- ד. מרוז זמן ההמתנה הממוצע של סטודנטית בתור.

הערות: ש.ימוי כל שבתורת התורים האיבוקיים המשותרת ע"י השרת נחשב כשמום בתור.

פתרון: א. דיאגרמת מצבים (כגם בסתברויות המצבים הושתים):



נסמן $\lambda = \lambda_0 + \lambda_1 + \lambda_2 + \lambda_3$ אזי:

$$P_1 = \rho \cdot P_0$$

$$P_2 = 1/2\rho \cdot P_1$$

$$P_3 = 1/2\rho \cdot P_2$$

$$P_4 = 1/2\rho \cdot P_3$$

$$P_0 + P_1 + P_2 + P_3 + P_4 = 1$$

ג. חישוב מספר הסטודנטים הממוצעים במומוצע:

$$E(N) = 1 \cdot P_1 + 2 \cdot P_2 + 3 \cdot P_3 + 4 \cdot P_4$$

ד. חישוב השרתים הממוצעים מתבצע ע"י מטריס $E(N) = E(N)E(\lambda)$ ויש לשים לב כי מדובר ב- $E(\lambda)$ ולא ב- λ הנחון משתלה. במקרה שלנו $E(\lambda) = \lambda \cdot (P_0 + P_1 + P_2 + P_3)$ (במצב בו אין בסיסה לשרת).